



UNIVERSIDAD DE CASTILLA-LA MANCHA
ESCUELA SUPERIOR DE INGENIERÍA INFORMÁTICA

GRADO EN INGENIERÍA INFORMÁTICA
TECNOLOGÍA ESPECÍFICA DE
INGENIERÍA DEL SOFTWARE

TRABAJO FIN DE GRADO

Variación Dinámica del Comportamiento de las Aplicaciones en
Función de las Emociones de Usuario

José María García García

Julio de 2017





UNIVERSIDAD DE CASTILLA-LA MANCHA

ESCUELA SUPERIOR DE INGENIERÍA INFORMÁTICA

GRADO EN INGENIERÍA INFORMÁTICA

**TECNOLOGÍA ESPECÍFICA DE
INGENIERÍA DEL SOFTWARE**

TRABAJO FIN DE GRADO

**Variación Dinámica del Comportamiento de las Aplicaciones en
Función de las Emociones de Usuario**

Autor: José María García García

Director: Víctor Manuel Ruiz Penichet
María Dolores Lozano Pérez

Julio de 2017

Declaración de Autoría

Yo, JOSÉ MARÍA GARCÍA GARCÍA con DNI 71722265P, declaro que soy el único autor del trabajo fin de grado titulado *Variación dinámica del comportamiento de las aplicaciones en función de las emociones de usuario* y que el citado trabajo no infringe las leyes en vigor sobre propiedad intelectual y que todo el material no original contenido en dicho trabajo está apropiadamente atribuido a sus legítimos autores.

En ALBACETE, a 25 de enero de 2017.

Fdo: José María García García

Resumen

La tecnología se desarrolla a una velocidad vertiginosa. Dicen que si presentásemos la historia del planeta Tierra en un periodo de 24 horas, los humanos no aparecerían hasta el último minuto. Siguiendo esa analogía, dentro del último minuto los humanos pasaron de no tener tecnología informática a tener ciudades enteras interconectadas y ordenadores cuánticos en cuestión de microsegundos, quizás en menos.

En este punto de la historia, dado que es muy complejo hacer que la tecnología sea más potente, se están dedicando esfuerzos a que la tecnología sea más inteligente. Dado que la inteligencia lógico-matemática está muy lograda actualmente, se está intentando dotar a las máquinas de inteligencia emocional: hacer que una máquina detecte las emociones de sus usuarios y adapte su funcionamiento a dicho estado emocional. Es aquí donde entra en juego la Computación Afectiva.

El aspecto de la Computación Afectiva más interesante para este propósito es la detección de emociones. Expresión facial, voz, lenguaje corporal, texto escrito, dilatación de las pupilas, ritmo cardíaco: el cuerpo está revelando cómo se siente una persona en todo momento. Con las herramientas adecuadas, se pueden obtener datos sobre esa persona y calcular, de forma computacional, cómo se siente. No obstante, el uso de estas tecnologías se da en ámbitos muy reducidos y de una forma poco (o nada) automatizada: en la gran mayoría de los casos, los resultados se almacenan para ser contrastados sin llegar a ser usados por la aplicación que los detectó.

El trabajo desarrollado en este Trabajo de Fin de Grado pretende solventar esa deficiencia y dotar de esa sensibilidad emocional comentada con anterioridad a una aplicación concreta. El objetivo perseguido es desarrollar un prototipo que haga uso de herramientas de detección de emociones y utilice los resultados generados por éstas para modificar en tiempo real su comportamiento en base a las emociones detectadas en el usuario, mejorando así la experiencia de usuario, entre otras cosas.

Este prototipo, desarrollado bajo un marco que combina Scrum y Kanban, consiste en un juego para el aprendizaje de inglés para niños de primaria. En función de las emociones detectadas en el usuario, el juego incrementa o disminuirá su nivel de dificultad, velocidad y/o complejidad, para mantener la motivación y atención del usuario. La evaluación preliminar realizada con usuarios reales ha obtenido resultados muy positivos.

Agradecimientos

Agradecer a todos los miembros del grupo de investigación “Interactive Systems Everywhere” que me recibieran con los brazos abiertos, como si fuera un viejo amigo, así como el apoyo que me han brindado.

En especial, quisiera dar las gracias a mi tutor, Víctor, por ver en mí algo que yo a veces solo puedo intuir y ofrecerme la posibilidad de entrar en este mundo que es la investigación. También agradecer a mi tutora María su dedicación y volcado en este proyecto desde que este no era más que una idea comentada de pasada, que han contribuido a que me sienta como uno más.

Dedicatorias

Dedico este proyecto a los que han sido mi mayor apoyo durante estos cuatro años, a mis amigos, que han sabido ofrecerme empatía y comprensión durante todo este tiempo.

También quiero dedicarlo a mis padres, mis hermanas y a toda mi familia, estén donde estén, capaces de insuflarme ánimos con unas pocas palabras.

ÍNDICE

ÍNDICE DE FIGURAS	XV
ÍNDICE DE TABLAS	XVII
CAPÍTULO 1. INTRODUCCIÓN.....	19
1.1 Motivación	20
1.2 Objetivos	23
1.3 Estructura de la memoria	24
CAPÍTULO 2. CONCEPTOS RELACIONADOS Y ESTADO DEL ARTE.....	25
2.1 Computación Afectiva	26
2.1.1 Emoción a partir de la voz	28
2.1.2 Emoción a partir de expresiones faciales	33
2.1.3 Emoción a partir del lenguaje corporal	37
2.1.4 Emoción a partir de información fisiológica	38
2.1.5 Emoción a partir de texto	38
2.1.6 Sistemas multimodales	41
2.2 Serious games	42
2.3 Conclusiones	47
CAPÍTULO 3. PROPUESTA DE SOLUCIÓN.....	49
3.1 Metodología	50
3.1.1 Scrum.....	52
3.1.2 Kanban.....	58
3.2 Tecnologías utilizadas.....	59
3.2.1 Afectiva	59
3.2.2 Beyond Verbal	59
3.2.3 Detección basada en pulsación de teclas y errores	60
3.2.4 Phaser	60
3.2.5 Heroku	60
3.2.6 mLab.....	61
3.2.7 Aspectos legales	61
3.3 Herramientas de soporte al desarrollo del proyecto.....	62
3.4 Conclusiones	64
CAPÍTULO 4. DESARROLLO DEL SISTEMA.....	67
4.1 Introducción al desarrollo	68

4.2	Sprint 0. Estudio base sobre Computación Afectiva.....	72
4.3	Sprint 1. Preparación de primer prototipo.....	73
4.3.1	Planificación	73
4.3.2	Desarrollo	75
4.3.3	Revisión	77
4.3.4	Retrospectiva	78
4.4	Sprint 2. Diseño de niveles	79
4.4.1	Planificación	79
4.4.2	Desarrollo	80
4.4.3	Revisión	84
4.4.4	Retrospectiva	84
4.5	Sprint 3. Integración de tecnología afectiva en el prototipo	85
4.5.1	Planificación	85
4.5.2	Desarrollo	85
4.5.3	Revisión	87
4.5.4	Retrospectiva	87
4.6	Sprint 4. Implementación de afección en el prototipo	88
4.6.1	Planificación	88
4.6.2	Desarrollo	89
4.6.3	Revisión	93
4.6.4	Retrospectiva	94
4.7	Conclusiones	95
CAPÍTULO 5. DESCRIPCIÓN DE LA APLICACIÓN		97
5.1	Introducción	98
5.2	Requisitos y recomendaciones de uso.....	99
5.3	Estructura de la aplicación	100
5.3.1	Zona global	101
5.3.2	Zona usuarios.....	102
CAPÍTULO 6. EVALUACIÓN.....		107
6.1	Introducción	108
6.1.1	Descripción del producto	108
6.1.2	Objetivos de la evaluación.....	108
6.2	Método	109
6.2.1	Participantes.....	109
6.2.2	Contexto de uso del producto en la evaluación	110
6.2.3	Diseño del experimento	112
6.2.4	Métricas de usabilidad	113

6.2.5	Resultados.....	114
6.3	Conclusiones de la evaluación.....	117
CAPÍTULO 7. CONCLUSIONES Y PROPUESTAS.....		119
7.1	Conclusiones.....	120
7.2	Trabajo futuro y posibles ampliaciones	121
BIBLIOGRAFÍA		123
LIBROS Y ARTICULOS		123
ENLACES INTERNET		126
CONTENIDO DEL CD.....		131
ANEXO A.	INCREMENTOS DE SPRINTS	133
ANEXO B.	DOCUMENTOS DE LA EVALUACIÓN	151
ANEXO C.	FORMATO DE RESPUESTA DE LOS SERVICIOS DE DETECCIÓN 155	
ANEXO D.	DIAGRAMAS UML GENERADOS.....	161

ÍNDICE DE FIGURAS

Figura 1. Interacciones con el sistema.....	21
Figura 2. Funcionamiento de Aplicación	22
Figura 3. Flujo de trabajo de aprendizaje supervisado (Morgun, Web).....	27
Figura 4. Preprocesamiento de audio en Beyond Verbal (Beyond Verbal, Web).....	29
Figura 5. Flujo de tareas del marco de trabajo SSI	33
Figura 6. Puntos clave de la cara (Microsoft, Web)	34
Figura 7. Demo de Emotion API con foto de ejemplo	35
Figura 8. Representación de emociones en Synesketch (Synesketch, Web).....	41
Figura 9. Teach your monster to read.....	44
Figura 10. Interfaz de creación de monstruo	44
Figura 11. Juego de Typing	45
Figura 12. A Slower Speed of Light.....	45
Figura 13. Nevermind.....	46
Figura 14. Tablero Kanban.....	52
Figura 15. Proceso Scrum (Agile Methods, Web)	54
Figura 16. Tablero Kanban al comienzo de Sprint 1	64
Figura 17. Planificación temporal real	70
Figura 18. Proyecto base a comienzos de sprint 1 (2).....	76
Figura 19. Proyecto base a comienzos de sprint 1 (2).....	76
Figura 20. Sprint 2. Gráfica burn-down	85
Figura 21. Sprint 3. Gráfica burn-down	88
Figura 22. Sprint 4. Gráfica burn-down	94
Figura 23. Pantalla de bienvenida de prototipo (final)	101
Figura 24. Pantalla de ayuda de prototipo	102
Figura 25. Panel de control.....	103
Figura 26. Barra superior del juego.....	103
Figura 27. Plataformas en juego	104
Figura 28. Botón “Clic para empezar” brillando.....	105
Figura 29. Gráfica sobre datos estadísticos de los resultados de navegación	115
Figura 30. Gráfica sobre datos estadísticos de los resultados de juego.....	116
Figura 31. Sprint 1. Pantalla de bienvenida de prototipo	134
Figura 32. Sprint 1. Pantalla de inicio de sesión de prototipo.....	134
Figura 33. Sprint 1. Pantalla de juego de prototipo.....	135
Figura 34. Sprint 1. Pantalla con traductor en funcionamiento.....	135
Figura 35. Sprint 1. Pantalla de registro de usuario	136
Figura 36. Sprint 1. Pantalla de modificar usuario	136
Figura 37. Sprint 1. Pantalla de eliminar usuario	137
Figura 38. Sprint 2. Pantalla de juego de prototipo.....	138

Figura 39. Sprint 4. Pantalla de bienvenida de prototipo (1).....	139
Figura 40. Sprint 4. Pantalla de bienvenida de prototipo (2).....	139
Figura 41. Sprint 4. Pantalla de bienvenida de prototipo (3).....	140
Figura 42. Sprint 4. Aviso legal.....	140
Figura 43. Sprint 4. Pantalla de inicio de sesión de prototipo	141
Figura 44. Sprint 4. Pantalla de registro de usuario.....	141
Figura 45. Sprint 4. Pantalla de información de receta – Nivel 1.....	142
Figura 46. Sprint 4. Pantalla de juego – Nivel 1.....	142
Figura 47. Sprint 4. Pantalla de juego – Nivel 1.....	143
Figura 48. Sprint 4. Pantalla de juego – Nivel 1.....	143
Figura 49. Sprint 4. Fin de juego (perdido) - Nivel 1	144
Figura 50. Sprint 4. Fin de juego (ganado) - Nivel 1.....	144
Figura 51. Sprint 4. Fase de dictado – Comienzo.....	145
Figura 52. Sprint 4. Fase de dictado - Aplicación escuchando.....	145
Figura 53. Sprint 4. Fase de dictado - Frase correcta (animación).....	146
Figura 54. Sprint 4. Fase de dictado - Pronunciación incorrecta.....	146
Figura 55. Sprint 4. Fase de dictado - Frase incorrecta tres veces (animación)	147
Figura 56. Sprint 4. Pantalla de espera de resultados	147
Figura 57. Sprint 4. Resultados	148
Figura 58. Sprint 4. Pantalla de información de receta - Nivel 2	148
Figura 59. Sprint 4. Pantalla de modificar usuario	149
Figura 60. Sprint 4. Pantalla de eliminar usuario	149
Figura 61. Sprint 4. Pantalla de ajustes	150
Figura 62. Formato de resultados de Beyond Verbal	156
Figura 63. Formato de resultados de Vokaturi	157
Figura 64. Formato de resultados de Emotion API	158
Figura 65. Formato de resultados de Afectiva	159
Figura 66. Formato de resultados de Kairos.....	160
Figura 67. Diagrama de clases (modelo conceptual).....	162
Figura 68. Diagrama de clases definitivo	163
Figura 69. Diagrama de actividad (partida completa)	164
Figura 70. Diagrama de actividad (juego)	165
Figura 71. Diagrama de actividad (dictado)	166
Figura 72. Diagrama de estados (dificultad)	167
Figura 73. Diagrama de Componentes	168
Figura 74. Diagrama de despliegue	169
Figura 75. Diagrama de navegación	170

ÍNDICE DE TABLAS

Tabla 1. Estados de ánimo de Beyond Verbal.....	31
Tabla 2. Sprints realizados	70
Tabla 3. Sprint 0. Product Backlog	72
Tabla 4. Sprint 1. Product Backlog	74
Tabla 5. Sprint 1. Sprint Backlog	74
Tabla 6. Product Backlog inicial	78
Tabla 7. Sprint 2. Product Backlog	79
Tabla 8. Sprint 2. Sprint Backlog	80
Tabla 9. Valores aceptables para determinantes de dificultad	82
Tabla 10. Progresión de dificultad	83
Tabla 11. Sprint 3. Sprint Backlog	86
Tabla 12. Sprint 4. Product Backlog	89
Tabla 13. Sprint 4. Sprint backlog.....	89
Tabla 14. Comparativa de sprints: estimado/real	95
Tabla 15. Datos sobre los participantes de la evaluación.....	110
Tabla 16. Resultados de tareas de navegación	114
Tabla 17. Datos estadísticos sobre resultados de navegación	114
Tabla 18. Resultados de tareas de juego.....	115
Tabla 19. Datos estadísticos sobre resultados de juego.....	116
Tabla 20. Resultados de cuestionarios SUS	117

CAPÍTULO 1. INTRODUCCIÓN

En el siguiente capítulo se presentan las distintas motivaciones que han llevado a la realización de este trabajo, así como los objetivos que se persiguen durante el desarrollo del mismo. Se recoge también una explicación de la estructura y organización del presente documento.

1.1 Motivación

Durante los primeros años de la historia de la Informática, los esfuerzos de todas las personas implicadas en dicho campo estaban enfocados en construir máquinas *funcionales*: cuando se afrontaba la construcción de un computador, se buscaba dotarlo de una capacidad de cómputo lo mayor posible, sin dedicar tiempo a hacerlo usable o intuitivo. Cuando Alan Turing construyó la Bomba para romper los códigos de Enigma, no la hizo pensando en que una persona cualquiera pudiera ponerla en marcha sin mirarse un manual, sino en que tardase lo menos posible en descubrir el código correcto. No obstante, según se ha ido desarrollando la Informática en todos sus aspectos, un nuevo objetivo — muchas veces secundario — ha entrado en escena: hacer las cosas usables, intuitivas.

Dada la trayectoria que ha tomado el diseño de productos, parece que el siguiente paso natural es diseñar productos que se adapten a sus usuarios. ¿Y si un producto cambiase dinámicamente su aspecto y/o funcionamiento en función del estado anímico del usuario? Es aquí cuando entra en escena uno de los pilares del presente proyecto: la *Computación Afectiva*.

Definida por primera vez en 1995 por Rosalind W. Picard, la Computación Afectiva es toda aquella computación relacionada con, provocada por, o que influye en emociones (Picard, 1995). Este campo de la Informática, que no está lejos del clímax en la curva de Gartner sobre tecnologías emergentes (Gartner, Web), abarca todas aquellas relaciones persona-ordenador en las cuales están presentes las emociones, ya sea porque se detectan las emociones del usuario (rama con más peso de la Computación Afectiva) o porque el computador simula tenerlas. A pesar de la astronómica evolución de esta disciplina y de los innumerables trabajos y estudios que se han llevado — y se están llevando — a cabo, la totalidad de su potencial aún está desaprovechado. Al margen de los numerosos experimentos realizados en distintas disciplinas sobre Computación Afectiva, su aplicación real se limita casi exclusivamente a la *publicidad y evaluación de productos* (Afectiva, Web). De hecho, incluso la explotación que se hace en dichos campos es bastante superficial: haciendo uso de tecnologías de detección de emociones, se registran las emociones de los usuarios mientras ven un anuncio o prueban un producto y se utilizan esos resultados para ver si una campaña tiene el impacto previsto o si el uso de un producto o prototipo es satisfactorio.

En el presente trabajo se pretende dar un paso más allá respecto al uso que se está haciendo de estas tecnologías, esto es, construir una herramienta que, no solo detecte las emociones del usuario, sino que automáticamente consuma los resultados producidos como parte de su funcionamiento, ya sea para detectar problemas, situaciones de estrés, de distracciones, etc.

Como caso de estudio, se propone el desarrollo de una aplicación basada en la Web pensada para enseñar inglés a niños de primaria (7 a 11 años), aplicándose para dicho desarrollo conceptos relacionados con la Computación Afectiva, de modo que, además de reaccionar a la interacción que el usuario realiza de forma consciente (clics, pulsaciones de

teclas, arrastre de elementos), reaccione a la interacción inconsciente (expresión facial, postura del cuerpo, ratio de clics/segundo, conductancia de la piel, etc.) que no será otra sino la expresión de las emociones del usuario (Figura 1). Para este prototipo se ha elegido la cocina como temática central dada la existencia de un proyecto previo similar a este (centrado en aprendizaje para niños) dentro del equipo de investigación en el que está enmarcado este TFG: Master Chef (de la Guía et al., 2016).



Figura 1. Interacciones con el sistema

Dado que se va a construir un juego cuya función lúdica va a estar en segundo plano, siendo el aprendizaje de inglés el objetivo principal, este prototipo caerá bajo el conjunto de juegos que se denominan *serios* (“serious games”), esto es, juegos cuyo objetivo principal no es entretener (Djaouti, Alvarez & Jessel, 2011). No obstante, no se han utilizado guías ni directrices para el desarrollo de ese tipo de juegos en concreto, estando el desarrollo centrado en el aspecto afectivo.

En la Figura 2 podemos ver un esquema de cuál será la secuencia de funcionamiento de la Aplicación. A través de los periféricos que correspondan, el usuario proporcionará información de entrada al sistema (paso 1), la cual se utilizará para controlar la aplicación (clics, pulsaciones, arrastre de elementos) y para decidir cómo ha de variar su comportamiento en el tiempo (gestos faciales, tono y timbre de la voz, etc.). Ante esa entrada para controlar la aplicación, ésta reaccionará de una determinada forma (paso 2). A su vez, la información afectiva se utilizará para decidir cómo habrá de modificarse el comportamiento de la aplicación en sí (pasos 3 a 8), lo que a su vez influirá en cómo ha de reaccionar la misma en el paso 2 a las entradas del usuario en el paso 1.

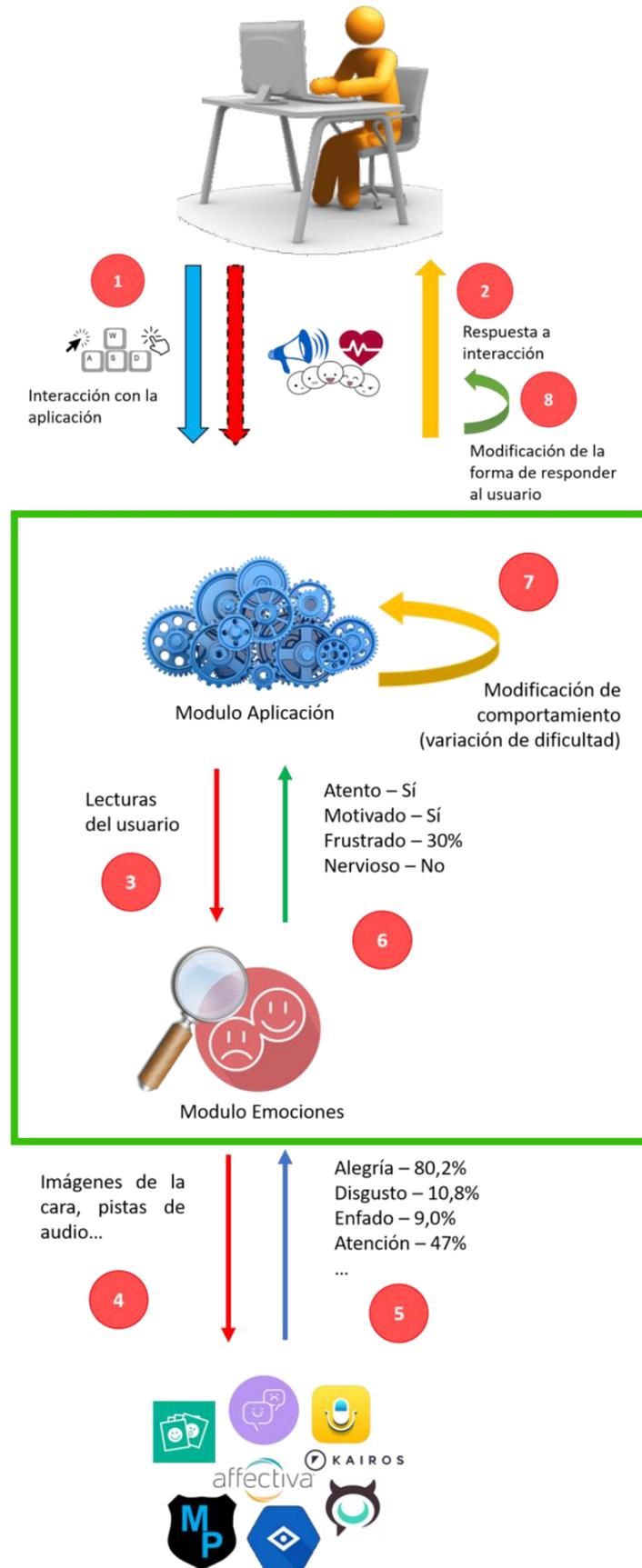


Figura 2. Funcionamiento de Aplicación

Durante los pasos 4 y 5, la aplicación se comunica con una serie de servicios externos para obtener información sobre el estado del usuario, información que luego se refinará y ajustará (teniendo en cuenta limitaciones de dichos servicios) para obtener una información más correcta y en nuestro contexto (paso 6).

Por último, los pasos 7 y 8 consisten en modificar el comportamiento del sistema y la forma de responder del mismo según las emociones obtenidas en el paso 6. Si se ha detectado una falta de atención, llamaremos la atención del usuario insertando un enemigo sorpresa o un objetivo que le suponga un pequeño desafío; si se detecta estrés, se disminuirá la velocidad o dificultad del nivel, etc.

Son muchos los canales de información que se pueden emplear para obtener información afectiva sobre un usuario, como podemos ver en (Santos, 2016). Se realizará un análisis de las circunstancias del sistema y de la arquitectura para decidir qué canales de información merece la pena explotar.

1.2 Objetivos

El objetivo principal de este trabajo es desarrollar un prototipo de sistema afectivo cuyo comportamiento varíe según las emociones que muestre el usuario al interactuar con el sistema, teniéndose como fin último conseguir la mejor experiencia de usuario posible.

Como caso de estudio, se propone que dicho prototipo sea una herramienta educativa dotada de sensibilidad emocional, esto es, de capacidad para detectar las emociones del usuario y reaccionar a ellas modificando su comportamiento en tiempo real. Dicha sensibilidad se aprovechará para, entre otras cosas, detectar si el usuario se frustra, en cuyo caso se ajustarán los parámetros de la aplicación para eliminar ese estado de frustración, o si el usuario está centrado y volcado en el uso del sistema, en cuyo caso se intentará mantener dicho estado, o si está aburrido, para aumentar la complejidad o el ritmo del juego para despertar su interés.

Este objetivo principal se desglosará en los siguientes subobjetivos.

- Estudiar las emociones y su influencia en el uso de las aplicaciones.
- Estudiar y probar tecnologías y mecanismos de detección de emociones.
- Profundizar conocimientos en la disciplina de la Computación Afectiva.
- Estudiar aplicaciones de naturaleza similar.
- Desarrollar un prototipo que demuestre cómo una aplicación podría variar dinámicamente en función de las emociones del usuario.
- Validar el prototipo desarrollado.

1.3 Estructura de la memoria

Este documento queda dividido en siete capítulos, incluido el actual. A continuación, se muestra una breve descripción acerca del contenido de cada capítulo:

- **CAPÍTULO 1. Introducción.** Este capítulo contiene las motivaciones que han llevado a la realización de este trabajo, así como los objetivos perseguidos durante el desarrollo del mismo.
- **CAPÍTULO 2. Conceptos relacionados y estado del arte.** Este capítulo crea una base de conocimiento sobre la que se construirá este trabajo. Se presentarán conceptos relacionados principalmente con la Computación Afectiva y los “serious games”. Dentro de la rama de la Computación Afectiva, este capítulo se centrará en la detección de emociones, explicándose los distintos canales a través de los cuales se puede obtener información afectiva y las tecnologías disponibles para detectar emociones a través de ese canal. Por último, se comentarán distintos proyectos que relacionen estos temas: Computación Afectiva aplicada a entornos educativos y/o de aprendizaje.
- **CAPÍTULO 3. Propuesta de solución.** En este capítulo se presenta el marco de trabajo que se va a seguir para el desarrollo del prototipo, así como una justificación de la elección de un marco sobre otro, junto con las tecnologías y herramientas que se van a utilizar.
- **CAPÍTULO 4. Desarrollo del sistema.** En este capítulo se describen detalladamente los cuatro sprints en los que ha quedado dividido el desarrollo de la aplicación. Cada sprint está dividido a su vez en cuatro apartados, a saber, planificación, desarrollo, revisión y retrospectiva, en los que se presentan, respectivamente, las tareas a realizar en ese sprint, una breve explicación sobre la ejecución de las mismas, un análisis del producto obtenido al final del sprint y un análisis del sprint en sí.
- **CAPÍTULO 5. Descripción de la aplicación.** En este capítulo se hace una descripción final del producto obtenido al acabar todos los sprints. Se detallan los aspectos más destacados de la aplicación: el hardware necesario para usarla, los controles de la misma, las distintas fases a las que se enfrentaría un usuario y un esquema global de toda la aplicación.
- **CAPÍTULO 6. Evaluación.** En este capítulo se detalla la información relativa a la evaluación de usabilidad que se llevó a cabo.
- **CAPÍTULO 7. Conclusiones y propuestas.** Por último, se hace una revisión del trabajo realizado, extrayéndose unas conclusiones relativas a los temas tratados a lo largo del trabajo, y se proponen una serie de mejoras del sistema que podrían realizarse en un futuro.

CAPÍTULO 2. CONCEPTOS RELACIONADOS Y ESTADO DEL ARTE

En este capítulo se presentarán los conceptos sobre los cuales se construirá este TFG. Se presentarán términos relacionados principalmente con la Computación Afectiva y los “serious games”. Dentro de la Computación Afectiva, este capítulo se centrará en la detección de emociones, así como en herramientas disponibles actualmente para llevarla a cabo.

Se expondrán también otras tecnologías y proyectos existentes relacionados con este trabajo, esto es, proyectos en los que se haya aplicado la Computación Afectiva en entornos educativos y/o en combinación con “serious games”.

2.1 Computación Afectiva

La Computación Afectiva, término introducido por primera vez en (Picard, 1995), se define como “*computación relacionada con, provocada por, o que influye en emociones*”. En otras palabras, cualquier forma de computación que tiene algo que ver con emociones de forma directa: las emociones no son algo separado de la interacción persona-ordenador, sino que forman parte de la misma. Por ejemplo, una página web que contuviese información sobre emociones o que contuviese una historia que provocase tristeza en el lector de la misma no se consideraría Computación Afectiva, mientras que una página que modificase su aspecto en base al estado anímico del usuario sí lo sería.

A pesar de que la definición de Computación Afectiva pueda resultar un poco vaga, en la práctica se traduce, en la gran mayoría de los casos, en detectar las emociones del usuario y en usar esas emociones con la finalidad de mejorar su experiencia de usuario al interactuar con el sistema. Es por ello que este trabajo se centra en la rama de la Computación Afectiva relativa a la *detección de emociones*.

Aunque la Computación Afectiva es una rama relativamente nueva, las emociones han sido un tema que lleva preocupando a filósofos y pensadores durante siglos. Su origen, su significado, su influencia, su transmisión: estos temas no son ninguna novedad en el panorama científico. En la actualidad, sabemos que las emociones se originan en el *sistema límbico*, una estructura del cerebro, situada debajo del córtex, a cargo de las emociones, la atención y la memoria (Picard, 1995). Al sentimiento en sí que siente una persona en un momento dado se le denomina *estado afectivo*, mientras que la comunicación de dicho estado al exterior tiene lugar a través de la *expresión de emociones*. Dicha expresión tiene lugar a través de *respuestas físicas* del cuerpo humano: lenguaje corporal, ritmo cardíaco, expresiones faciales, tono de la voz, etc. Es por esto que merece la pena destacar que las distintas tecnologías que se expondrán a continuación van a trabajar detectando *emociones*, intentando a través de estas dilucidar el *estado afectivo* de la persona. Mientras que los ordenadores no sean capaces de mapear con una precisión absoluta respuestas fisiológicas a estados afectivos, no se podrá decir que un computador detecta estados afectivos.

Dado que, por el momento, un computador solo puede ver las emociones que provoquen alguna clase de respuesta externa en el cuerpo o que la persona revele durante la interacción con otra entidad, la detección se reduce a los siguientes medios (Tao & Tan, 2005):

- Voz
- Expresiones fáciles
- Lenguaje corporal y movimientos del cuerpo
- Respuestas fisiológicas
- Texto

De manera adicional, se puede usar información combinada de estos canales para obtener una mejor tasa de acierto en la detección de emociones. Esto se puede conseguir a través de *sistemas multimodales*. En las siguientes subsecciones, se hará un análisis de las tecnologías de detección disponibles para extraer información afectiva de esos canales previos.

Aunque cada tipo de tecnología trabaja de una manera concreta, la mayoría de ellas comparte un núcleo común en lo que concierne a su funcionamiento. Esto se debe a que un detector de emociones como tal es, fundamentalmente, un clasificador automático. La creación de un clasificador automático implica recoger información, extraer las características que resulten verdaderamente importantes para el objetivo que se persigue y finalmente, entrenar el modelo para reconocer y clasificar ciertos patrones (Morgun, Web).

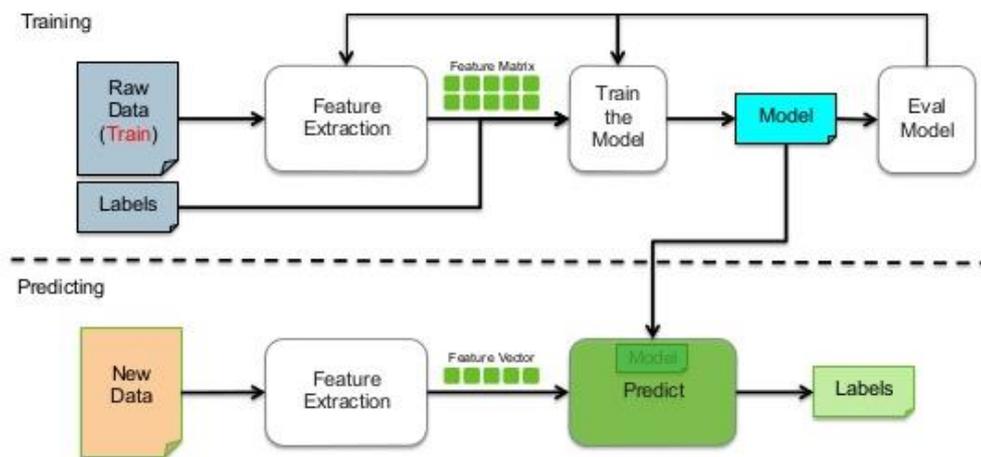


Figura 3. Flujo de trabajo de aprendizaje supervisado (Morgun, Web)

Posteriormente, se utilizará el modelo entrenado para clasificar nuevos datos. En la Figura 3 podemos ver todo el flujo de entrenamiento y posterior uso de un clasificador. Por ejemplo, supongamos que se quisiera construir un modelo que fuese capaz de identificar alegría y tristeza en expresiones faciales. Para entrenar este modelo, habría que suministrarle imágenes ya clasificadas: las imágenes que muestren gente sonriente deben ir clasificadas con la etiqueta “Alegría”, mientras que las que muestren a gente de expresión alicaída y ceño fruncido deben ir clasificadas con la etiqueta “Tristeza”¹. Después de llevar a cabo este entrenamiento, cuando el modelo recibiese una imagen de una persona sonriendo, este identificaría la emoción expresada como “Alegría”, mientras que ante una imagen de una persona con expresión alicaída devolvería “Tristeza” como resultado.

¹ En esta situación ficticia consideramos que solo existen esos dos tipos de expresiones, que solo están determinadas por la boca y el ceño de una persona. En la práctica, la implementación de un clasificador es una tarea muy compleja dada la gran cantidad de variaciones que existen al expresarse una misma emoción y de la gran cantidad de expresiones faciales que puede componer un rostro.

2.1.1 Emoción a partir de la voz

Uno de los canales de información que se puede explotar para intentar detectar el estado anímico de una persona es su voz. Como se comentó en la subsección anterior, cuando una persona está sintiendo una emoción (se encuentra en un estado afectivo concreto), ésta se manifiesta externamente a través de ciertos cambios físicos. La voz puede ser una de esas cosas afectadas por la expresión de una emoción. Por ejemplo, cuando una persona se enfada, el ritmo de su respiración se acelera y los músculos se tensan, afectando a la forma en la que vibran las cuerdas vocales y variando, en última instancia, las cualidades sonoras de la voz (Pfister, 2010).

En lo que respecta a las tecnologías que extraen información afectiva a través de la voz, resulta interesante reseñar el siguiente aspecto. Cuando una persona habla, se puede dividir la información que produce en dos canales distintos: el canal primario y el canal secundario (Casale, Russo, Scebba & Serrano, 2008). El canal primario es aquel asociado a la información sintáctico-semántica —lo que una persona está diciendo literalmente—, mientras que el canal secundario está asociado a información paralingüística (características acústicas de la voz, estado emocional y gestos que acompañen el discurso). Por ejemplo, una persona dice “¡qué divertido!” con un tono serio y carente de emoción. A través del primer canal obtenemos lo que la persona ha dicho, algo le parece divertido, mientras que a través del segundo podemos interpretar el significado de esas palabras: esa persona está mintiendo o siendo sarcástica.

Las tecnologías que se expondrán a continuación utilizan solo la información del segundo canal para detectar la emoción en una persona, dejando de lado la información del primer canal, conocida también como información vocal. Atendiendo a las características acústicas de la voz (haciéndose uso especialmente del tono y de los Coeficientes Cepstrales en las Frecuencias de Mel²), las siguientes tecnologías detectan la emoción reflejada en una pista de audio recibida:

- Beyond Verbal
- Vokaturi
- EmoVoice
- Good Vibrations

² Los Coeficientes Cepstrales en las Frecuencias de Mel (Mel Frequency Cepstral Coefficients) son unos coeficientes que se utilizan para representar señales digitales de audio, aplicándose una serie de operaciones sobre la señal original del sonido.

Beyond Verbal

Beyond Verbal (Beyond Verbal, Web) es una compañía con sede en Israel que lleva trabajando desde 2013 en un servicio accesible vía API que extrae, descifra y mide el estado anímico de una persona a partir de una pista de audio. El servicio de Beyond Verbal recibe una pista de audio, que habrá de contener la voz de una persona hablando, y devuelve información afectiva sobre dicha voz. Este servicio es de pago, aunque la empresa pone a disposición de sus clientes distintos paquetes mensuales para contratar, ampliando cada cual la cantidad de audio que se puede analizar cada mes. Además de esa versión de pago, Beyond Verbal ofrece una versión gratuita que permite escanear un volumen de audio de hasta 100 minutos al mes. La principal ventaja que ofrecen este tipo de servicios —basados en la nube— es que su integración en un proyecto no supone mucho más allá de realizar una petición HTTP, suponiendo claro que dicho proyecto esté pensado para funcionar con conexión a Internet. Las únicas restricciones que impone el servicio de Beyond Verbal son las siguientes:

- El audio enviado a la API ha de tener formato *WAV o PCM a 16 bits, con un solo canal*.
- Para generar un resultado, la API necesita al menos *13 segundos de habla sin pausa*. No obstante, no es necesario enviar un audio de estas características para que la API lo acepte, ya que los algoritmos de Beyond Verbal son capaces de recortar silencios y pausas de una pista de audio. Por regla general, un audio de entre 20 y 30 segundos de conversación normal tiene audio suficiente para obtener ese tiempo mínimo (Figura 4).

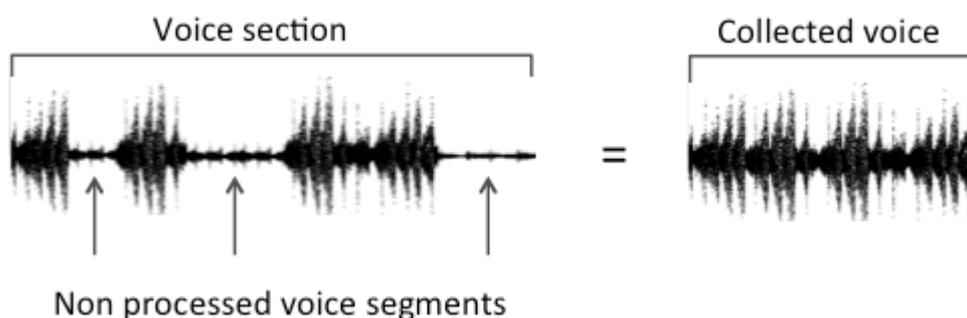


Figura 4. Preprocesamiento de audio en Beyond Verbal (Beyond Verbal, Web)

Ante una pista de audio que cumpla esos requisitos, Beyond Verbal nos devuelve los resultados en un objeto JSON que contiene los siguientes datos:

- **Humor.** Este indicador refleja el temperamento detectado en la voz analizada. El temperamento se expresa mediante dos medidas distintas: un valor que puede ser Bajo, Medio o Alto y un número perteneciente al rango [0, 100], estando los valores más cercanos a cero asociados a temperamentos bajos y los

más cercanos a cien, a temperamentos altos. Los temperamentos bajos se asocian a emociones agresivas (odio, ira, hostilidad, etc.) mientras que los temperamentos más bajos están asociados a emociones depresivas (tristeza, dolor, ansiedad, miedo, etc.). Por último, los temperamentos medios están asociados a emociones positivas/neutras. El valor al que prestaremos más atención será el número del rango [0, 100], ya que para valores cuyo temperamento no sea obvio, la clasificación categórica resulta demasiado absoluta. Por ejemplo, supongamos que la API devuelve en un momento dado una respuesta que para Humor contiene {categoría: Baja, número: 28}. Si solo se comprobase la categoría, se concluiría que la persona en cuestión está asustada/triste, mientras que viendo el número podemos apreciar que el número está casi al límite de corresponderse con un temperamento medio, es decir, neutral.

- **Valencia.** Este valor identifica si el habla analizada expresa sentimientos positivos o negativos. Al igual que el Humor, la Valencia se expresa mediante un valor de entre las categorías Negativo, Neutral y Positivo y un número perteneciente al rango [0, 100]. Los valores cercanos al cero significan que la voz analizada expresa emociones negativas (agresividad o tristeza), mientras que los valores cercanos al 100 significan que la voz analizada expresa emociones positivas (extroversión, amor, etc.). Los valores intermedios están asociados a la ausencia de emoción (neutralidad).
- **Excitación.** Este indicador expresa la energía reflejada por la voz analizada. Se identifica también mediante un valor de una categoría (Baja, Media, Alta) y un número del mismo rango comentado anteriormente. Valores bajos están asociados a tonos de voz que expresen aburrimiento o cansancio, mientras que los valores altos reflejan excitación (pasión comunicativa, nerviosismo, ira).
- **Estado de ánimo.** Este valor nos indica la emoción detectada en la muestra analizada usando una frase en lenguaje natural. Mientras que los indicadores anteriores son números y categorías, este elemento de la respuesta contendrá valores del tipo “Ansiedad”, “Soledad”, “Preocupación”, etc. La API de Beyond Verbal dispone de cuatro grupos de estados de ánimo, a saber, Grupo 7, Grupo 11, Grupo 21 y Grupo Compuesto, cada uno más grande y complejo que el anterior (con más estados de ánimo y más específicos). Mientras que los Grupos 7, 11 y 21 (los números indican las emociones que contiene cada grupo) expresan emociones mediante una única palabra, el Grupo Compuesto, que contiene 432 elementos, expresa las emociones de forma más compleja, usando frases enteras. En la Tabla 1 podemos apreciar estas diferencias; mientras que en los tres primeros grupos podemos encontrar etiquetas como “Felicidad”, “Preocupación” y “Admiración”, en el Grupo Compuesto tenemos valores como “Superación de emociones. Lucha interna o conflicto interpersonal”.

Tabla 1. Estados de ánimo de Beyond Verbal

Grupo	Emociones
Grupo 7	Enfadado; Tranquilo; Entusiasta; Frustrado; Alegre; Triste; Preocupado.
Grupo 11	Creativo, Apasionado; Crítico, Cínico; Defensivo, Ansioso; Amigable, Cálido; Hostil, Enfadado; Líder, Carismático; Solitario, Insatisfecho; Amor, Alegría; Triste, Apenado; Autocontrol, Práctico; Supremacía, Arrogante.
Grupo 21	Admiración; Ira; Ansiedad; Convencido; Creatividad; Disgusto; Dominante; ...
Compuesto	Superación de emociones. Lucha interna o conflicto interpersonal; Confianza y ambición para conseguir objetivos. Furia orientada a los objetivos. Fuerza para conseguir metas; ...

Las respuestas de la API³ contienen solo emociones de los Grupos 11 y Compuesto y para cada grupo se distingue entre emociones primaria y secundaria. La emoción primaria es la emoción detectada con más fuerza dentro de ese grupo, mientras que la secundaria se corresponda con la segunda emoción detectada con más fuerza. Así, en cada grupo se identifica la emoción más fuerte, junto con una segunda emoción que confirme la primera o que añada un matiz que la primera no termina de capturar.

La respuesta de la API contiene también información sobre la calidad del audio y la edad y el género del hablante, aunque estos datos no se utilizarán durante el flujo del prototipo. En la Figura 62 (Anexo C) podemos ver el formato en el que llegan las respuestas ante una petición válida a Beyond Verbal.

Vokaturi

Vokaturi es una empresa con sede en Ámsterdam fundada en 2016 que desarrolla software que pretende “reflejar el estado del arte en lo que respecta a la detección de emociones a partir de la voz” (Vokaturi, Web). Vokaturi ofrece su tecnología de detección en forma de librerías, concretamente para los lenguajes C y Python, de manera que los

³ De la versión utilizada en este proyecto (v3).

desarrolladores puedan integrar la detección de emociones a partir de la voz en sus proyectos. Para el uso de sus librerías, Vokaturi ofrece tres licencias:

- *OpenVokaturi*. Versión de código abierto de las librerías de Vokaturi, publicada con licencia GPL. La versión libre de la tecnología Vokaturi tiene una tasa de acierto en la detección del 66.5%
- *VokaturiPlus*. Versión no pública de las librerías de Vokaturi. Esta es la versión que han de usar aquellas personas que quieran comercializar la aplicación que use estas librerías. La tasa de aciertos asciende al 76.1%.
- *VokaturiPro*. Versión de Vokaturi hecha a medida para el cliente. Entre otras cosas, permite al cliente elegir que emociones quiere que detecte Vokaturi. También tiene una tasa de aciertos del 76.1%.

A diferencia de Beyond Verbal, Vokaturi se ejecuta en la máquina en la que se encuentran las librerías, lo que por un lado elimina la dependencia de conexión a Internet, pero al mismo tiempo significa que la clasificación no será tan potente como lo pueda ser la de un servicio que dedique clusters enteros para procesar una pista de audio.

Los resultados de Vokaturi utilizan las seis emociones básicas de Paul Ekman (Handel, Web) —salvo la sorpresa y el disgusto, y considerando también la neutralidad—, estando la presencia o ausencia de una emoción indicada mediante un valor relativo entre 0 y 1 que representa el peso que tiene esa emoción sobre el resto. La suma de todos esos índices da como resultado 1.

En la Figura 63 (Anexo C) se puede apreciar el formato de salida de una petición de análisis a Vokaturi. Si, por ejemplo, se detecta neutralidad con 0.7, significa que el resto de emociones solo podrán sumar como máximo 0.3.

EmoVoice

EmoVoice es un marco de trabajo para la detección de emociones en tiempo real a través de las características acústicas del habla humana, sin hacer uso de información vocal (Vogt, André & Bee, 2008) (Wagner, Lingenfelser & Andre, 2011). Desarrollado por el departamento de Multimedia Centrada en Humanos (Human-Centered Multimedia) de la Universidad de Augsburg (Universität, Web), ha sido recientemente integrado en el marco de trabajo para la Interpretación de Señales Sociales, SSI por sus siglas en inglés (Social Signal Interpretation).

El marco de trabajo SSI ofrece herramientas para registrar, analizar y reconocer el comportamiento humano en tiempo real. Mientras que las tecnologías comentadas previamente eran servicios ya implementados y listos para usar, SSI trabaja creando un modelo desde cero y entrenándolo para reconocer emociones. Este modelo puede ser

conectado a Internet posteriormente para convertirse en el servidor que reciba peticiones de análisis y devuelva resultados.

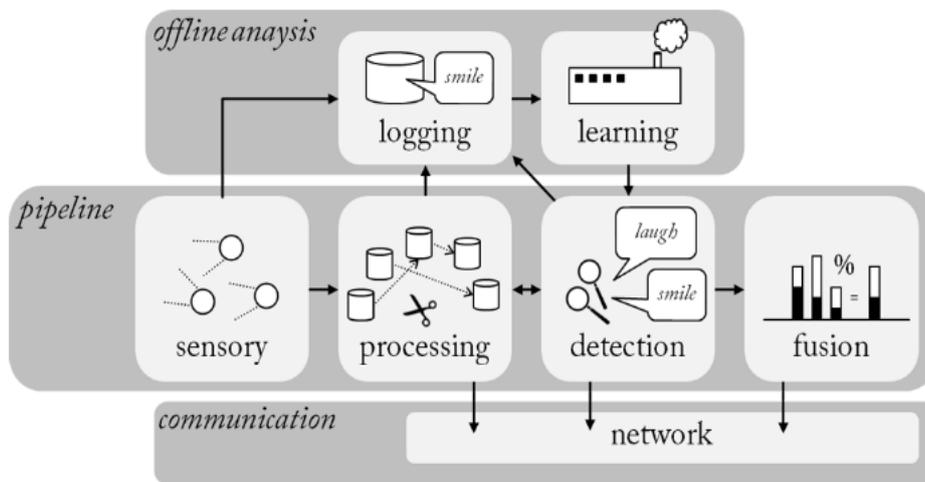


Figura 5. Flujo de tareas del marco de trabajo SSI

En la Figura 5 vemos las fases por las que pasaría un modelo implementado mediante los marcos de trabajo previos (tanto EmoVoice como SSI).

Un aspecto destacable de esta tecnología es que permite a los usuarios de la misma crear su propia base de datos de audio, lo que unido al hecho de que estos marcos son “patch-based” se traduce en una gran flexibilidad para el diseñador del proyecto que use SSI. Mediante el uso de XML, los desarrolladores pueden especificar cada uno de los elementos de cada punto del flujo de tareas, de manera que se obtiene un detector completamente personalizado.

Good Vibrations

Otra opción disponible en el mercado es la empresa Good Vibrations. Esta empresa ofrece un servicio de detección de emociones en forma de SDK, aunque no ofrece ningún tipo de versión de prueba en su página web y hasta finales de año no puede ofrecer información sobre su servicio debido a la participación de la empresa en un proyecto⁴.

2.1.2 Emoción a partir de expresiones faciales

Al igual que la voz, las expresiones faciales reflejan las emociones que una persona puede estar sintiendo, de una forma incluso más clara. Una voz es casi única para cada

⁴ Esto se descubrió gracias a un intercambio de correos electrónicos con el personal de contacto de Good Vibrations.

persona, lo que puede hacer necesario cierto ajuste a la hora de usar tecnologías de detección. Si una persona tiene un tono de voz bajo, puede que este sea interpretado como un tono de voz triste sin que esa persona esté realmente triste. Las expresiones faciales, por el contrario, son “casi universales” (Price, 2016). Durante más de un siglo, ya en tiempos de Darwin, se planteó la cuestión de si las emociones eran universales o no. El reputado psicólogo Paul Ekman llevó a cabo el siguiente experimento en Papúa Nueva Guinea: enseñó a los nativos de la zona imágenes de personas de occidente reflejando las emociones ira, alegría, tristeza y sorpresa, y les pidió que asociaran cada imagen a la emoción reflejada. La clasificación que realizaron los nativos fue correcta y con eso se aceptó la premisa de que las expresiones faciales tenían un componente biológico y eran universales. No obstante, recientemente se realizó otro experimento que parece apuntar a lo contrario, a pesar de que los profesionales del sector están de acuerdo en que, aunque importante, dicho test necesita realizarse de forma más exhaustiva y ampliada para considerar que la hipótesis de la universalidad ha quedado demostrada como falsa. Es por ello que, por ahora, se considera que las expresiones faciales son “universales”. Aun así, los servicios de detección de emociones que utilicen el rostro han de tener en cuenta este hecho.

Las tecnologías de detección de emociones que utilizan el rostro funcionan de forma similar a las que utilizan la voz: atendiendo a unas características concretas de la entrada, en este caso una imagen del rostro, se genera un resultado según la emoción detectada. En el caso de la cara, se analizan una serie de puntos clave del rostro (Figura 6), observándose su posición tanto absoluta como relativa, para determinar que expresión hay en ese rostro.

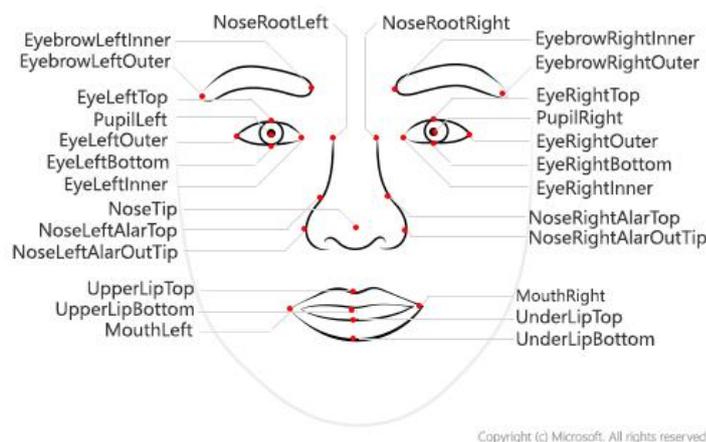


Figura 6. Puntos clave de la cara (Microsoft, Web)

En lo que respecta a este tipo de detección, se han analizado los siguientes servicios:

- Emotion API
- Affectiva
- nViso
- Kairos

Emotion API

Uno de los servicios ofrecidos dentro del paquete de servicios Microsoft Cognitive es Emotion API. Esta API recibe una imagen o un video e identifica las caras detectadas y la emoción expresada por cada una de ellas en la imagen o en cada uno de los fotogramas del video. La presencia de una emoción se denota con la confianza con la cual se detecta dicha emoción. Por ejemplo, en una imagen de una persona sonriendo, se detectará alegría con un alto nivel de confianza, mientras que la tristeza y el medio se detectarán con valores muy bajos. Las emociones que distingue Emotion API son las seis básicas de Paul Ekman (Handel, Web) así como desdén y neutralidad (ausencia de emoción). En (Emotion API, Web) se puede encontrar una demo disponible para probar la detección de emociones en imágenes, en la cual el usuario puede enviar una foto (o utilizar una de las que ofrece Microsoft a modo de ejemplo) y comprobar si la clasificación se ha realizado correctamente (Figura 7).



Figura 7. Demo de Emotion API con foto de ejemplo

Las puntuaciones que utiliza Emotion API para indicar la intensidad con la que se ha detectado cada emoción son similares a las que utiliza Vokaturi, en el sentido de que son valores relativos entre 0 y 1 y la suma de estos ha de ser igual a 1. En la Figura 64 podemos ver qué aspecto tiene la respuesta de una petición de detección de emociones en una imagen. Este servicio tiene una versión gratuita, aunque el volumen de imágenes mensuales que se pueden analizar con ésta es limitado. Dispone también de unas versiones de pago en las cuales se cobra por transacción, pero se limitan las transacciones (peticiones de detección de emociones) que se pueden hacer por segundo.

Afectiva

Otra opción en el terreno de la detección de emociones a partir de expresiones faciales es la empresa Afectiva, que ofrece, entre otras cosas, un servicio de detección que nos indica,

no solo las emociones detectadas, sino también la presencia o ausencia de determinadas microexpresiones. Haciendo uso del Sistema de Codificación de Acciones Faciales de Paul Ekman (Blog Afectiva, Web), FACS por sus siglas en inglés (Facial Action Coding System), Afectiva reconoce, además de la emoción detectada, si la persona tiene la boca abierta, si frunce el ceño, si aprieta los labios, etc. Esto puede suponer una gran ventaja a la hora de interpretar unos resultados, pues la información de las microexpresiones puede ayudar a confirmar o a desmentir la emoción detectada. El servicio de Afectiva se ofrece a través de una SDK para Java, Objective-C, C++, Unity, Javascript y Windows. Además, su servicio está disponible de forma gratuita siempre que sea con fines educativos o cuando la empresa que lo utilice genere menos de un millón de dólares al año.

Además de la información comentada, Afectiva puede reconocer ciertas características del aspecto de la persona (género, edad, etnia y uso de gafas), si la emoción detectada es positiva o no (mediante un número en el rango [-100, 100]), el compromiso del usuario con la actividad que esté realizando durante la monitorización y la atención que está prestando el usuario. En la Figura 65 podemos apreciar el formato que tiene una respuesta producida por la SDK de Afectiva.

nViso

La empresa nViso ofrece “la solución basada en inteligencia artificial más escalable, robusta y precisa para medir reacciones emocionales instantáneas en clientes tanto en entornos reales como virtuales” (nViso, Web). A través de un sistema similar al que usa Afectiva, nViso indica la emoción detectada usando las emociones básicas de Ekman junto con la emoción “neutral” (nViso, 2011).

Kairos

Kairos (Kairos, Web) es una empresa de inteligencia artificial especializada en reconocimiento facial que ofrece, entre otros, un servicio de detección de emociones. Los servicios de Kairos incluyen detección, identificación y verificación de rostros, detección de la edad, etnia, género, emociones expresadas, etc. Al igual que los servicios previos, Kairos ofrece sus productos vía API, aunque las licencias de pago ofrecen un SDK que no necesita de conexión a Internet.

Los servicios de Kairos, que pueden ser probados en su web a través de diversas demos, requieren como entrada una imagen o un video, y en sus resultados devuelven, para cada cara detectada, información sobre las emociones que está expresando (usando las seis básicas de Ekman), sobre la postura de la persona detectada (inclinación de la cabeza en los ejes X, Y y Z, si está prestando atención, etc.) y sobre la persona en sí (edad, género, gafas). El análisis de los videos consiste en realizar la detección de emociones en los fotogramas del video, analizándose fotogramas a intervalos de 0.033 segundos. En la Figura 66 podemos ver el aspecto de una respuesta de Kairos.

Los servicios de Kairos se ofrecen a través de varias licencias, siendo gratis una de ellas. Una desventaja notable de este servicio es que requiere subir el contenido que se quiere analizar a alguna plataforma en la nube. Si unimos esto al hecho de que hay que estar comprobando activamente cuándo se han generado los resultados (hay que hacer una petición POST para solicitar el análisis y una petición GET para conseguir los resultados, habiendo que repetir esta última hasta que estén listos), el tiempo se convierte en un inconveniente que impide, por ejemplo, simular un funcionamiento en tiempo real.

2.1.3 Emoción a partir del lenguaje corporal

Aunque no es un medio utilizado para comunicar información de forma activa y consciente, el cuerpo humano está emitiendo constantemente información afectiva. Zapatear con el pie en el suelo, mover la pierna, cruzarse de brazos, inclinar la cabeza, cambiar la posición todo el tiempo, etc. El lenguaje corporal de una persona revela cómo se siente de la misma forma que lo hace su voz y su cara (Mind Tools, Web). Mientras que esto no es un conocimiento nuevo, el uso del lenguaje del cuerpo humano para la detección de emociones es bastante novedoso y aun no existe un entendimiento claro relativo a cómo crear los sistemas que lean las emociones en una postura. La mayoría de estudios realizados en el campo de la detección de emociones (el 95% aproximadamente) se ha centrado en las expresiones faciales, ignorando el resto de canales a través de los cuales se revela información afectiva (Kleinsmith & Bianchi-Berthouze, 2013).

Aun así, existen diversas propuestas relativas a llevar a cabo este reconocimiento y usar los resultados para otros propósitos. La psicología experimental ha demostrado que, efectivamente, existe un vínculo entre ciertas emociones y ciertos movimientos (Piana, et al., 2014). Por ejemplo, una persona experimentando miedo apartará su cuerpo del punto en el que se encuentre ese generador de miedo, mientras que una persona experimentando alegría, sorpresa o ira acercará su cuerpo hacia el punto que esté provocando esas emociones.

Dado que no existen tecnologías disponibles para practicar este tipo de detección, no existe un consenso claro sobre los datos necesarios para realizarla correctamente. Normalmente, los experimentos realizados con este tipo de detección hacen uso de herramientas como SSI o tecnologías que detecten el cuerpo de un usuario como Kinect, dejando a los investigadores la creación de los modelos y esquemas que necesiten para su situación concreta. Dichos modelos suelen construirse, la mayoría de las veces, pensando en que analicen las distintas articulaciones del cuerpo, su posición relativa respecto al resto de articulaciones y el ángulo que forman con las extremidades que conectan (Tao & Tan, 2005), pero en última instancia, el funcionamiento del modelo y su forma de extrapolar las emociones dependerán de los investigadores que lo desarrollen y su caso particular.

2.1.4 Emoción a partir de información fisiológica

Como se comentó al comienzo de este capítulo, las emociones se generan en el sistema límbico y estas a su vez tienen una manifestación física externa: actividad eléctrica en los músculos y la piel (esta última conocida como respuesta galvánica de la piel o actividad electrodérmica), dilatación de pupilas, ritmo cardíaco y respiratorio, presión sanguínea, actividad eléctrica cerebral, etc. Estas manifestaciones pueden detectarse con las herramientas adecuadas.

No obstante, la información proveniente de estas señales directas es más difícil de clasificar, al menos con el enfoque basado en categorías usado habitualmente. Es por ello que cuando se trabaja con señales fisiológicas es mejor adoptar un enfoque dimensional (Picard, 2009) para llevar a cabo la clasificación de emociones. Mediante este enfoque, una emoción detectada ya no es simplemente “Alegría” o “Tristeza”, sino que queda definida como un estado a su vez determinado por varias dimensiones, siendo valencia y excitación las más empleadas. Es, en parte, debido a esto que este tipo de detección se reserva para investigación y estudios como, por ejemplo, relativos al autismo.

Aunque no existen servicios disponibles para leer emociones a partir de estos datos, es muy sencillo adquirir el hardware necesario para llevar a cabo las mediciones. Este suele ser uno de los principales motivos por los cuales la detección de emociones a través de este tipo de señales queda en un segundo plano, y es el hecho de que hay que cubrir de sensores al usuario. Incluso cuando el impacto de los sensores se puede minimizar, estos implican cargar al usuario con objetos extraños a este, lo que puede resultar incómodo o poco natural para los mismos. Mientras que actualmente no existe ningún servicio equivalente a los servicios anteriores para este tipo de señales, se están desarrollando tecnologías (Conner-Simons & Gordon, 2016) para poder medir datos fisiológicos de forma no invasiva, por lo que puede que el día de mañana la información fisiológica se analice igual que se analiza un rostro.

2.1.5 Emoción a partir de texto

Una de las cosas que se comentó al comienzo de este capítulo es que las emociones quedaban expuestas cuando (a) estas tenían una respuesta física externa asociada o (b) eran reveladas por el usuario durante la interacción con otra entidad. En otras palabras, la expresión de una emoción no se manifiesta solo en nuestro cuerpo sino también en la forma en la que interactuamos con el mundo. No obstante, pueden darse casos en los cuales esa interacción no disponga del componente visual que ofrece una interacción cara a cara. En un mundo como es el nuestro, dominado por las telecomunicaciones, las palabras se convierten en un poderoso aliado para descubrir cómo puede estar sintiéndose una persona. Solo para comienzos de junio de 2017 se habían generado más de 100,000 millones de “twits”, solo en ese año (Internet live stats, Web). Incluso cuando la detección de emociones en el texto tiene

que hacer frente a más obstáculos que las tecnologías previas (faltas de ortografía, metáforas, coloquialismos, etc.), sigue siendo una fuente de información afectiva a considerar. Dado que esta detección se realiza atendiendo a las palabras (cosa que se ignora en la detección basada en la voz), el proceso de analizar un texto tiene algunos pasos adicionales respecto al de analizar una cara o una voz. Aunque se sigue haciendo uso de un modelo que ha de ser entrenado, hay que “sanear” el texto antes de dárselo al modelo. Este saneamiento incluye procesos de división del texto en unidades más pequeñas, de simplificación del texto, de etiquetado de palabras, entre otros (Binali & Potdar, 2012). En los siguientes apartados se realizará una revisión de los siguientes servicios:

- Tone Analyzer
- Receptiviti
- BiText
- Synesketch

Tone Analyzer

Tone Analyzer es un servicio desarrollado en Bluemix que utiliza el poder de computación de IBM Watson para detectar emociones en un trozo de texto (Tone Analyzer, Web). Tone Analyzer se apoya en el campo de la *psicolingüística*, que estudia la relación entre comportamientos lingüísticos y teorías de la psicología. Esta tecnología, ofrecida en forma de API, analiza la relación entre el tono del texto y las características lingüísticas del mismo (Science Tone, Web). Los resultados de Tone Analyzer están divididos en tres categorías (Understand, Web).

- **Tono emocional.** Emociones detectadas en el texto; se utilizan las seis emociones básicas.
- **Tono social.** Se indican las tendencias sociales en el texto usando categorías extraídas del modelo de los cinco grandes, modelo que expresa la personalidad de una persona en función de cinco factores.
- **Tono del lenguaje.** Las métricas asociadas a este tono indican como se percibe la escritura en sí: el tono de la persona escritora es analítico, dubitativo, confiado, etc.

Aunque se ofrece una demo, es necesaria una membresía de Bluemix para usar esta tecnología plenamente.

Receptiviti

Receptiviti (Receptiviti, Web) es una empresa que lleva trabajando en el análisis de texto desde hace 20 años, siendo su aplicación de análisis LIWC (Linguistic Inquiry and Word Count) el núcleo de la misma. Dicha aplicación podría equipararse a un diccionario,

en el cual cada palabra está etiquetada en ciertas categorías según el tipo de palabra y su significado afectivo. Por ejemplo, la palabra “sonreír” podría etiquetarse con las palabras “verbo”, “alegría”, “emoción positiva”, etc. El análisis de un texto podría resumirse entonces como el mapeo de cada palabra dentro del diccionario.

Entre los servicios ofrecidos por esta empresa tenemos el análisis de emociones que, a través de peticiones a su API, permite analizar la emoción expresada por un texto. La única restricción impuesta por este servicio es un mínimo de palabras en cada texto a analizar.

Este tipo de servicios suelen utilizarse en campañas para conocer la impresión que tiene el público de una empresa sobre la misma, porque lo que suelen incluir herramientas para analizar publicaciones en redes sociales y sitios de reseñas y opinión. En el caso de Receptiviti, una membresía con su servicio permite analizar “twits” en base a “hashtags” y/o palabras claves para conocer la opinión de los usuarios acerca de un tema.

BiText

Otra opción que sigue los pasos de Receptiviti (centrada en marketing y satisfacción del cliente) es BiText, que nos ofrece un servicio de análisis de texto a través de una API que combina técnicas de aprendizaje automático con técnicas de análisis lingüístico profundo (Blog BiText, Web) para detectar los sentimientos reflejados en un texto. El servicio de BiText, que ofrece al mismo tiempo análisis de emociones, extracción de conceptos y entidades y categorización del texto, destaca por reconocer un número considerable de idiomas, a saber, inglés, catalán, alemán, francés, italiano, portugués y español. BiText divide el texto recibido en frases y analiza el tema y las emociones reflejados por cada uno.

Synesketch

Synesketch (Synesketch, Web) es una herramienta de código abierto para la detección de emociones en texto, destacando por las representaciones artísticas que utiliza para reflejar los resultados: además de la forma habitual consistente en puntuaciones entre 0 y 1, Synesketch dibuja una imagen que representa la emoción detectada haciendo uso de física de partículas y colisiones.

En la Figura 8 podemos ver, de arriba a abajo, la representación de disgusto, ira y tristeza. De izquierda a derecha, representaciones de menor a mayor intensidad de cada emoción.

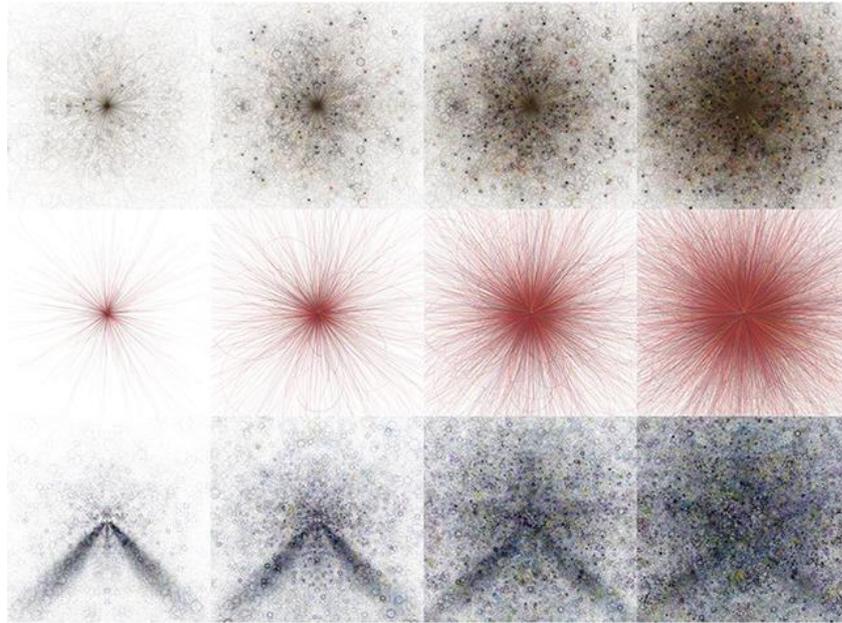


Figura 8. Representación de emociones en Synesketch (Synesketch, Web)

2.1.6 Sistemas multimodales

Un inconveniente que presentan todas las tecnologías previas, y que es casi inevitable, es la imprecisión de las mismas. Las mejores máquinas de detección de emociones que existen, las personas, son capaces de evaluar el estado afectivo de una persona en cuestión de milisegundos con un margen de error mínimo, y esto es posible porque una persona es un *detector multimodal* (Tao & Tan, 2005): combinando información relativa a una persona, un observador humano es capaz de precisar el estado afectivo de la misma con mucha más precisión que si usase solo información de la cara o del cuerpo. Aunque un sistema multimodal informático no puede aún emular a una persona (que no solo tiene en cuenta información relativa al exterior de la persona observada, sino también información sobre su historia y contexto), sigue siendo lo mejor a lo que se puede aspirar en cuestión de detección de emociones.

Un *sistema multimodal* es aquel que combina información afectiva de varias entradas (cara, voz, postura, palabras escritas, fallos cometidos durante la escritura, etc.) para generar un resultado. Está demostrado que la precisión que se obtiene combinando resultados es mucho mayor que si medimos las emociones de forma individual. Es importante destacar que un sistema no es multimodal cuando recibe múltiples entradas y hace una mera media de resultados, sino cuando combina efectivamente los valores leídos, analizando cada valor en el contexto de los demás. Por ejemplo, un sistema modal que combinase información textual y facial para detectar emociones, al detectar una cara seria junto con el mensaje “que

divertido jaja” devolvería como resultado “sarcasmo/falta de interés”, mientras que un sistema no multimodal, que se limitase a “unir” resultados devolvería “alegre/neutral”.

A su vez, siempre existe la opción de enriquecer la detección, tanto de forma complementaria como añadiendo un componente más al sistema multimodal, usando métodos más “caseros” como, por ejemplo, detectar los fallos cometidos al escribir o al interactuar con una interfaz. En (Santos, 2016) podemos encontrar una recopilación de estudios realizados sobre detección de emociones, explorándose medios más allá de los “básicos”.

Aunque no existe actualmente ningún servicio como Afectiva o Beyond Verbal que ofrezca un sistema de detección multimodal (salvo quizás el marco de trabajo SSI que se puede preparar para leer información a través de varios canales de información), la riqueza de herramientas en este sector ofrece una muy buena base para el desarrollo de los mismos.

2.2 Serious games

Es un hecho consumado que existe una relación directa entre la cognición, el acto de conocer, y las emociones. Las emociones juegan un papel clave en la toma de decisiones, la memorización de información y en la capacidad de atención de una persona (Wilkinson, 2013). Por ejemplo, una persona motivada (ya sea por una emoción positiva o negativa) en la realización de una tarea tiene una mayor voluntad para desempeñarla, memorizar información sobre ella, recuperar más tarde esa información, etc. En términos de almacenamiento de información, aquella que esté asociada a una emoción de gran valencia (tanto positiva como negativa) se almacenará y recuperará con mucha más facilidad. Al mismo tiempo, esta relación entre cognición y emoción puede ser un lastre: según la ley de Yerkes-Dodson, una gran excitación puede afectar negativamente al rendimiento de una persona, debido a los efectos negativos del estrés sobre la memoria y la atención.

Otro hecho conocido es que los juegos, sobre todo los digitales, tienen a provocar respuestas emocionales en los jugadores, tanto por sus componentes lúdicos como por sus componentes narrativos (la historia del juego). En los juegos actuales, en los cuales el jugador puede personalizar su avatar —su representante en el mundo virtual—, este vínculo emocional se robustece, ya que el avatar no es un mero icono en una pantalla, sino un elemento personalizado, en el cual el jugador ha volcado su tiempo y sus gustos.

Ya incluso antes de que estos fenómenos fuesen expresados con rigor científico, eran hechos observables: los niños que se divertían en el proceso de hacer algo recordaban eso mucho mejor. De este afán de facilitar el aprendizaje aprovechando el “engagement” provocado por los juegos nacen los “serious games”.

Las primeras definiciones formales de estos juegos serios aparecen en los años setenta, hablándose de juegos llevados a cabo en clase para facilitar el aprendizaje de ciertos temas. Con el desarrollo del campo relativo a los videojuegos y los medios informáticos, empezaron a aparecer videojuegos cuyo principal propósito no era ofrecer entretenimiento sino enseñar algo. Así, una definición de juego serio actual aceptada, es “cualquier pieza de software que mezcle propósitos que no sean relativos al entretenimiento con estructuras de videojuegos”. En otras palabras, videojuegos cuyo objetivo principal no sea entretener (Djaouti, Alvarez & Jessel, 2011).

Aun cuando esta información no es precisamente una novedad y las ideas expresadas en los párrafos anteriores han estado aplicándose de alguna manera o de otra, no ha sido hasta recientemente que ha empezado a aplicarse en colegios y demás centros educativos de manera más formal y sistemática (Deleurme, 2017). La *neuroeducación* (disciplina que utiliza este conocimiento del cerebro para mejorar el aprendizaje) está empezando a consolidarse, en combinación con la presencia de juegos serios y técnicas de gamificación, en el ámbito de la educación.

Ahora bien, que un videojuego se desarrolle con fines de entretenimiento no significa que no se pueda usar para un fin “serio”, tanto en su forma original o tras alterarlo mediante modificadores (mods). Se puede decir que, en última instancia, es el educador que haga uso del juego el que determinará si un juego es “serio” (se utiliza con un objetivo que no es el de entretener) o no.

Algunos ejemplos de juegos serios podrían ser los que siguen.

Teach your monster to read

Teach your monster to read (Teach your monster, Web) es un juego serio disponible para varias plataformas pensado para enseñar a leer en inglés a niños pequeños (Figura 9). El juego cubre desde letras y sonidos hasta la lectura de frases completas, y está basado en minijuegos que el usuario ha de completar haciendo de uso de su avatar, un monstruo completamente personalizado por el usuario (Figura 10).



Figura 9. Teach your monster to read

Galardonado con múltiples premios, este juego está disponible de forma completamente gratuita en su versión para ordenador, facilitándose la accesibilidad del mismo a profesores que quieran usarlo en sus clases.



Figura 10. Interfaz de creación de monstruo

Typing

Typing (Typing, Web) es una colección de juegos serios pensados para enseñar mecanografía a personas de todas las edades, aunque enfocado a un público más juvenil. En este caso no tenemos una historia formada por niveles y minijuegos interconectados, sino varios minijuegos independientes, teniendo cada uno de ellos unos niveles de dificultad que podemos elegir con libertad.



Figura 11. Juego de Typing

En la Figura 11 podemos ver un ejemplo de los juegos de *Typing*. En este juego, el jugador, cuyo avatar es el niño que hay en el centro, ha de escribir caracteres y palabras para derrotar a los zombis que intentarán llegar hasta él. Para derrotar a un zombi, el jugador ha de colocarse en el carril en el que se encuentre el zombi y apuntar hacia él. Una vez en ese carril, el jugador simplemente tiene que escribir la palabra asociada al zombi para derrotarlo.

A Slower Speef of Lighth

A Slower Speed of Light (Slower Speed, Web) es un juego desarrollado por el MIT Game Lab pensado para enseñar aspectos relacionados con la teoría de la relatividad y la velocidad de la luz (Figura 12). Mediante un sistema de recolección de objetos, el jugador se mueve por el plano a distintas velocidades, viendo cómo cambia la escena según la velocidad.



Figura 12. A Slower Speed of Light

No podemos ignorar el hecho de que este desarrollo de videojuegos y neuroaprendizaje ha tenido lugar a la par que el nacimiento y progreso de la Computación Afectiva. A pesar

de ser algo que aún no ha despegado, la unión de Computación Afectiva y videojuegos ya existe. Aunque no son muchos, ya existen juegos que utilizan las emociones de los jugadores para crear una experiencia más inmersiva y/o personalizada. Un buen ejemplo de esto es el juego *Nevermind* (Nevermind, Web). En *Nevermind* (Figura 13) ocupamos el lugar de un médico que es capaz de introducirse en las mentes de sus pacientes traumatizados haciendo uso de una tecnología futurista. El objetivo del juego consiste en resolver puzzles dentro de la psique de un paciente, al mismo tiempo que se recuperan recuerdos sobre el trauma en sí para poder tratarlo. Si bien *Nevermind* puede jugarse como un juego normal (sin recibir y utilizar información afectiva), de forma opcional puede jugarse haciendo uso de una cámara web y/o un sensor que lea el pulso del jugador, con el fin de utilizar esa información afectiva para generar una experiencia de juego mucho más intensa.



Figura 13. Nevermind

En este caso, la generación de una mejor experiencia viene dado por una modificación del propio juego, que se hace más terrorífico cuando más alterado esté el usuario.

Esta afectividad, aunque levemente presente en el campo de los videojuegos normales (aquellos cuyo único fin es lúdico), está menos presente aun en los juegos serios, cuando debería haberse implantado primero en dicho campo dada la importancia de las emociones en el campo de la cognición. No obstante, ya existen proyectos que pretenden unir en una misma herramienta las ventajas de los videojuegos en el aprendizaje y la monitorización que permiten las herramientas de detección de emociones, tal y como se puede ver en el proyecto realizado en (Kostoulas et al., 2012).

Este Trabajo de Fin de Grado pretende realizar un esfuerzo similar y ofrecer una herramienta para facilitar el aprendizaje de vocabulario que a su vez utilice herramientas de detección de emociones para mantener la atención y el “engagement” del usuario todo lo que se pueda, garantizándose así una mejor experiencia de usuario.

2.3 Conclusiones

Merece la pena destacar la detección de emociones basada en la voz y en el rostro, puesto que son los principales medios usados para conocer el estado afectivo de una persona actualmente. Incluso en la sociedad en la que vivimos, dominada por las telecomunicaciones, la interacción entre un sistema y un usuario suele ser “en persona”. También es interesante tener en mente que los mejores resultados se obtienen cuando se combina información afectiva de distintos medios. Puesto que el prototipo a desarrollar necesita ser capaz de detectar las emociones de los usuarios, habrá que decidir qué sistemas de detección vamos a integrar en el mismo. Esta información, junto a una explicación de las metodologías que se van a utilizar, puede encontrarse en el capítulo siguiente.

Respecto a la naturaleza del juego, dado que su componente lúdico tendrá un papel secundario, éste podrá definirse como un juego serio, por lo que a lo largo de este capítulo se ha realizado una revisión de la influencia de las emociones en el aprendizaje y sobre cómo se ha aplicado ese componente lúdico de los juegos en entornos educativos.

CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

En este capítulo se expondrán de manera detallada los medios que se han utilizado para llevar a cabo la implementación del prototipo comentado anteriormente. Se presentará la metodología de trabajo utilizada, una combinación de Scrum y Kanban, así como sus prácticas, elementos y artefactos. Se comentará también qué otras alternativas a éstas existen y por qué se han elegido éstas sobre las demás. Del mismo modo, se detallarán qué tecnologías tanto hardware como software se han utilizado para desarrollar el producto.

Se comentarán también qué tecnologías se han utilizado para dar soporte al proyecto de desarrollo en sí.

3.1 Metodología

A lo largo del grado, se han conocido diversas metodologías de trabajo: metodología RUP (Rational Unified Process), metodología Iconix, metodología Scrum, metodología Kanban, metodología Extreme Programming (XP), metodología Lean, etc. Todas estas metodologías tienen ventajas y desventajas, presentando cada una prácticas y artefactos distintos. No obstante, podemos agruparlas en dos tipos:

- **Metodologías dirigidas por planes (plan-driven).** En estas metodologías se dispone de toda la información sobre el producto que se va a construir desde un principio. No se afronta la implementación del producto hasta que no se tiene toda la información reunida. Más adecuadas para proyectos grandes y largos. Se busca minimizar la incertidumbre y maximizar el control. Reflejan una forma de trabajar más tradicional (Learning Tree, Web).
- **Metodologías dirigidas por cambios (change-driven).** En estas metodologías la prioridad es entregar valor al cliente. Se basan en hacer entregas periódicas de valor al mismo, de manera que se va reduciendo la incertidumbre del proyecto gradualmente. Más adecuadas para proyectos cortos y pequeños. Debido a esa forma de trabajar mediante incrementos, resulta un enfoque muy flexible. Las metodologías ágiles están incluidas en esta categoría (Learning Tree, Web).

Dadas estas características y diferencias, lo primero que había que decidir era que tipo de metodología se adaptaba mejor a las circunstancias del proyecto. El presente proyecto presenta las siguientes características.

- **Equipo pequeño.** El equipo de trabajo está formado únicamente por el alumno y los tutores del mismo.
- **Proyecto de corta duración.** Incluso aunque este trabajo se convierta en la base de una futura línea de tesis, en principio su desarrollo abarcará solo un periodo de aproximadamente un cuatrimestre.
- **Inestabilidad tecnológica.** Aunque cabe la posibilidad de que en un futuro las herramientas de detección de emociones sean de fabricación propia, concretamente para la elaboración de este prototipo existe una dependencia tecnológica con las empresas que nos ofrecen sus detectores. Si bien la aplicación se construirá para ser capaz de funcionar sin el reconocimiento de emociones, el hecho de que dejasen de estar disponibles los detectores o su uso se encareciese afectaría negativamente al desarrollo.
- **Requisitos no estables.** Dada la gran actividad que está teniendo lugar en el campo de la Computación Afectiva, no sería extraño que el día de mañana apareciese una nueva tecnología para llevar a cabo la detección de emociones o un nuevo problema a considerar que no hubiese aparecido antes, por ejemplo.

Estas situaciones obligarían a modificar el rumbo del proyecto con cierta urgencia.

- **Enmarcado en un grupo de investigación.** El grupo de investigación establece una forma de trabajar según la cual se producen reuniones semanales para controlar el progreso del trabajo, así como para analizar el rumbo que ha de seguir el desarrollo del mismo.

Dado los altos niveles de riesgo e incertidumbre que rodean este proyecto, queda en evidencia que para llevar a cabo el desarrollo de este trabajo se necesita la flexibilidad que aporta una metodología ágil. Una vez acordado esto, hemos de determinar qué metodología ágil ha de regir dicho desarrollo. A la hora de tomar esta decisión, se consideraron aquellas metodologías vistas a lo largo del grado: Scrum, Extreme Programming (XP), Lean, Kanban, Crystal y Desarrollo guiado por Características (FDD). Por sencillez y debido a un mayor conocimiento de algunas de esas metodologías, las alternativas se redujeron a Scrum, XP y Kanban. A su vez, a la hora de decidir en base a estas tres, se tuvieron en cuenta los siguientes aspectos.

- **Mayor práctica con Scrum.** A lo largo de la intensificación en Ingeniería del Software se estudiaron y/o se trabajó con las metodologías previamente comentadas. No obstante, dada la predominancia de Scrum en el mundo laboral, la aplicación práctica ha girado siempre en torno a metodologías ágiles, concretamente Scrum.
- **Kanban muy adaptativo.** Kanban es una metodología ágil de la que podemos destacar ser poco prescriptiva. En esencia, Kanban solo impone una restricción: dado un flujo de trabajo con distintas etapas, ha de limitarse la cantidad de trabajo que puede haber en desarrollo en cada una de esas etapas. En la Figura 14 podemos ver un tablero Kanban, en el cual se reflejan las tareas que hay en cada etapa. A la cantidad de tareas que puede haber como máximo en una etapa se la denomina WIP (“Work in Progress”, Trabajo en Proceso). Aunque esta limitación resulta muy útil para evitar cuellos de botella, la metodología en sí resultaba demasiado flexible para adaptarse sola
- **Scrum más extendido.** Según el Scrum Report de 2016 realizado por Scrum Alliance (Scrum Alliance, 2016), la metodología Scrum era utilizada por el 89% de los participantes en el estudio (más de dos mil personas de entre 76 países), seguida de Kanban con un 43% y de metodologías híbridas con un 23%.

Dadas esas consideraciones, se decidió que Scrum regiría el desarrollo del presente trabajo. No obstante, la limitación de la cantidad de trabajo que puede haber en progreso en cada etapa resultaba una característica muy interesante como para ignorarla. Por ello, se ha utilizado una *metodología basada en Scrum haciendo uso también de la limitación de trabajo y de los tableros que impone Kanban.*

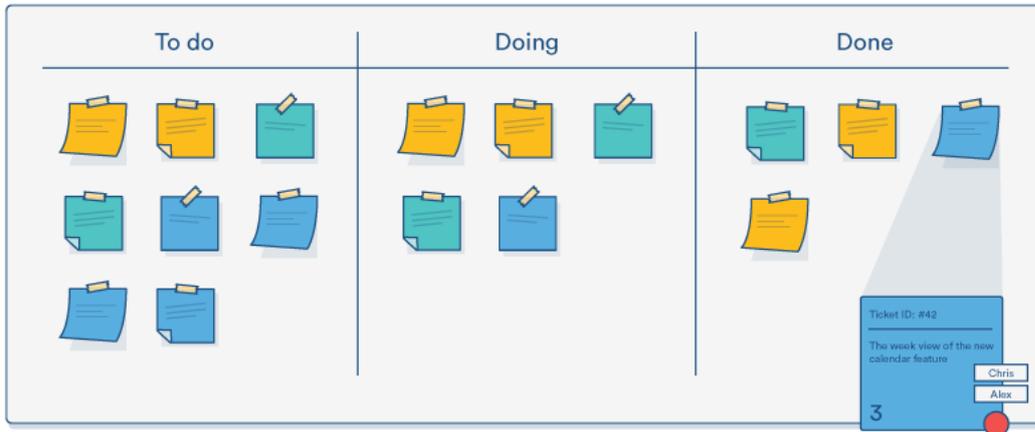


Figura 14. Tablero Kanban

3.1.1 Scrum

Scrum es una metodología que forma parte del movimiento “Ágil” (Scrum Methodology, Web). Tras observarse que los paradigmas clásicos de desarrollo de proyectos equivalían al fracaso en la mayoría de los casos, se hizo patente que era necesario un cambio. En 2001, varios pioneros con esta mentalidad de cambio se reunieron en Utah para tratar este tema. De dicha reunión nació el Manifiesto Ágil (Agile, Web), una declaración de cuatro valores y doce principios en los cuales se basan las metodologías ágiles. A continuación, se realiza un análisis de cada uno de esos cuatro valores (Smartsheet, Web).

- **Individuos e interacciones sobre procesos y herramientas.** Un error que se comete en las metodologías más tradicionales es valorar sobremanera los procesos y las herramientas que se utilizan. Procesos y herramientas son elementos que han de apoyar el desarrollo: estos han de trabajar para los desarrolladores, no al revés. Es por ello que el primer principio da más valor a los individuos, que son los que trabajarán en satisfacer las necesidades de los clientes. De igual manera, si el desarrollo queda dirigido por los individuos, la comunicación entre los mismos será más fluida y natural, mientras que si la comunicación está dictada por el proceso, esta será más artificial.
- **Software funcional sobre documentación extensa.** En metodologías tradicionales, la elaboración de documentación era un paso tedioso que tomaba una gran cantidad de tiempo. En las metodologías ágiles no se elimina la documentación, puesto que es un elemento útil y necesario, sino que se realiza de forma paralela al trabajo de desarrollo de manera que proporciona a los desarrolladores la información que necesitan sin detenerlos en minucias.

- **Colaboración con el cliente sobre negociación de contratos.** En enfoques más clásicos, el cliente acordaba con el director del proyecto cómo quería que fuese el producto final, especificando con gran detalle todos los aspectos de este producto, y sellándose estas características con un contrato. Todos estos acuerdos se realizaban antes de empezar el trabajo, de manera que era más fácil que se dieran casos en los que, incluso haciendo exactamente lo que el cliente pidió, no se satisficieran sus necesidades. El Manifiesto Ágil aboga por un cliente que participe en el proceso de desarrollo, ya sea involucrándose en el desarrollo de forma diaria o participando en pruebas y demos de forma regular.
- **Responder al cambio sobre seguir un plan.** En lugar de dedicar una gran cantidad de tiempo a elaborar planes extensos, detallados y llenos de interdependencias, se establecen periodos de trabajo breves en los cuales se van detallando los objetivos más prioritarios, que van cambiando según avanza el proyecto.

Descripción general

Scrum se define en su guía oficial (Schwaber & Sutherland, 2016) como un marco de trabajo que permite afrontar problemas complejos, entregando productos con el mayor valor posible de forma productiva y creativa. Aunque lleva usándose desde los noventa para llevar a cabo el desarrollo de productos, Scrum no es un proceso, sino un marco de trabajo en el cual se pueden usar otros procesos y/o técnicas.

El marco de trabajo Scrum está formado por los equipos Scrum y sus relativos roles, artefactos, eventos y reglas. Cada componente de este marco tiene una función concreta y es necesario para implementar dicho marco correctamente. Las reglas de Scrum establecen las relaciones y los mecanismos de interacción entre estos componentes.

En la Figura 15 podemos un esquema general del marco de trabajo Scrum. En los siguientes apartados se irán explicando las distintas partes de dicho esquema.

Teoría de Scrum

Scrum está fundamentado en el Control Empírico de Procesos o Empirismo (Schwaber & Sutherland, 2016). El Empirismo es una corriente que afirma que el conocimiento proviene de la experiencia y de la toma de decisiones en función de lo que ya se sabe. En base a esto, Scrum adopta un enfoque iterativo e incremental para aumentar la predictibilidad y el control sobre los riesgos.

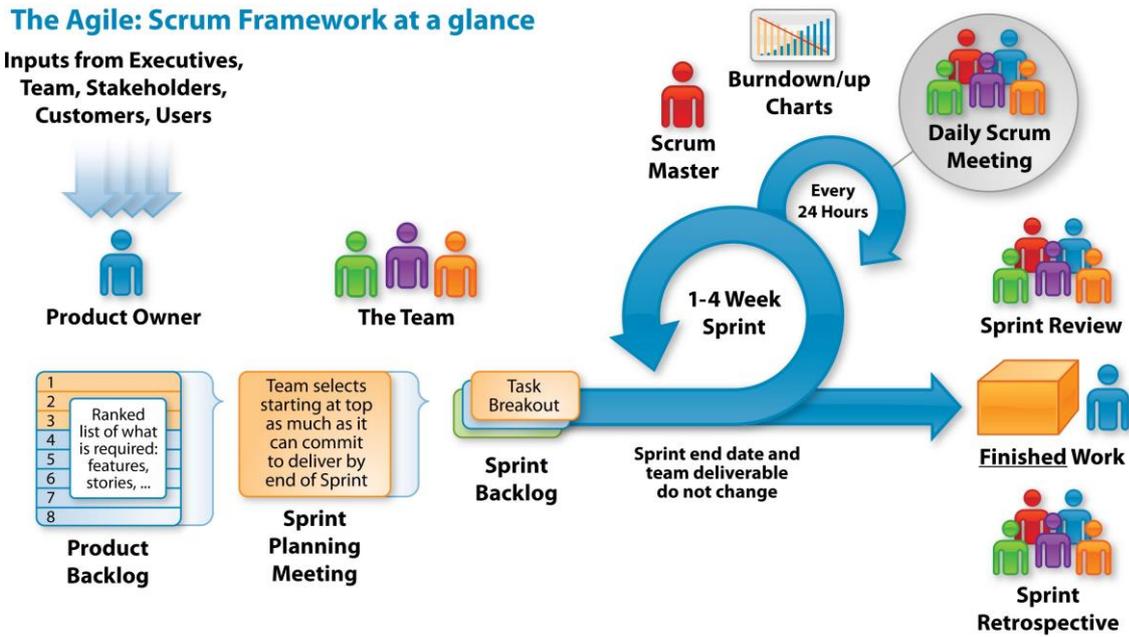


Figura 15. Proceso Scrum (Agile Methods, Web)

A su vez, el Control Empírico de Procesos se sustenta sobre tres pilares.

- **Transparencia.** Los aspectos importantes del proceso han de ser visibles para los responsables de los resultados de los mismos. La transparencia también alude a que los implicados en el proceso han de hablar el “mismo idioma”, siendo necesario definir ciertos aspectos de forma estándar para que sean entendidos de la misma forma por todos.
- **Inspección.** Los artefactos generados, así como el propio proceso, han de ser revisados con cierta frecuencia para detectar desviaciones indeseadas, procurando que esa frecuencia no sea tal como para entorpecer el trabajo en sí.
- **Adaptación.** Si tras una de esas inspecciones se determina que el proceso se está desviando de su estado óptimo o que el producto final no será aceptable, han de hacerse los ajustes pertinentes para corregir esa deficiencia. Dichos ajustes han de realizarse cuanto antes para que la desviación provocada sea mínima.

Para llevar a cabo la inspección y la adaptación, Scrum prescribe una serie de reuniones para controlar el progreso del desarrollo y el proceso en sí. En las siguientes secciones se explicarán estos distintos mecanismos de inspección y adaptación, así como los distintos roles y artefactos que se utilizan en Scrum.

Personas implicadas (roles)

Un equipo Scrum (Schwaber & Sutherland, 2016) está formado por un propietario (Product Owner), un equipo de desarrollo y un maestro de scrum (Scrum Master).

- **Propietario o Product Owner (PO).** El propietario es el encargado de maximizar el valor del producto y del trabajo del equipo de desarrollo. Es la persona encargada de gestionar la pila de trabajo del producto (Product Backlog), lo que incluye, entre otras cosas, determinar los elementos de dicha pila, ordenarlos por valor y/o prioridad de implementación, asegurar que los elementos de dicha pila son claros, visibles y entendidos por todos, etc. El propietario ha de ser una única persona, aunque esta puede representar a un comité, y sus decisiones han de ser respetadas.
- **Equipo de desarrollo o Team.** El equipo de desarrollo consiste en un grupo de profesionales que se encargan de entregar un producto terminado al final de cada iteración. Los equipos de desarrollo son autogestionados y de funcionalidad cruzada, no distinguen roles de forma interna y no se subdividen.
- **Maestro de Scrum o Scrum Master (SM).** El maestro de Scrum es el encargado de supervisar el proceso en sí. Se asegura de que el equipo se ciñe a las directrices de Scrum.

Artefactos

Los artefactos de Scrum representan el trabajo realizado, aportan transparencia y oportunidades de inspección y adaptación.

Pila de trabajo o Product Backlog (PB)

La pila de trabajo es una lista ordenada de todos los aspectos que puede ser necesario que el producto tenga. Es la única fuente de requisitos del producto y es responsabilidad del PO mantenerla accesible, actualizada y ordenada.

La pila de trabajo es un ente dinámico que existe mientras existe el producto. Al inicio del desarrollo solo contendrá aquellos requisitos más claros y/o necesarios, mientras que según avance el desarrollo del producto evolucionará también la pila de trabajo, identificándose cada vez las características que hagan el producto más útil, más apropiado para su propósito, etc. Cuando el producto se lance al mercado, el uso del mismo hará surgir más requisitos, de manera que la pila seguirá creciendo.

Los requisitos de la pila de trabajo suelen expresarse en forma de **historias de usuario** (Mountain Goat, Web), esto es, una breve descripción de una característica del producto

expuesta desde el punto de vista de la persona que quiere esa nueva característica, como un usuario final o un cliente. Una historia de usuario tiene el formato “Yo, como (tipo de usuario), quiero (un objetivo) de manera que (justificación)” o “Yo, como (tipo de usuario) quiero (capacidad o tarea) para (objetivo)”. Cada historia de usuario suele ir acompañada de otros valores como su prioridad, un indicador de su valor, una estimación del tiempo para implementar la característica expresada, etc. Una historia puede dividirse en historias más pequeñas, a la vez que varias pequeñas pueden unirse en una más grande.

Para estimar lo que va a costar implementar una característica se suelen usar medidas relativas (Cohn, 2005), como los puntos de historia (Story Points), que expresan el peso de una tarea respecto a otra, evitando así problemas que puedan surgir al intentar estimar usando valores absolutos independientes. Para realizar la estimación en sí, se suele recurrir a técnicas como el Planning Poker, en el cual, los miembros del equipo hacen sus “apuestas” sobre cuánto creen que costará implementar una nueva característica. En base a estas apuestas, se decide qué estimación se asignará a esa característica. Para hacer esas apuestas, se suelen utilizar secuencias de números como la de Fibonacci, que permiten alcanzar un consenso más rápido⁵.

Pila de sprint o Sprint Backlog (SB)

La pila del sprint está formada por los elementos de la pila de trabajo que se han seleccionado para ser implementados durante un sprint. Esta pila se define a comienzos del sprint y evoluciona a lo largo del mismo, según el equipo va aprendiendo cosas nuevas y descubriendo qué hay que hacer para cumplir ciertos objetivos.

Incremento

El incremento está formado por las historias que se han implementado a lo largo de un sprint. Al acabar un sprint, el incremento debe ser completamente funcional y estar listo para liberarse.

Gráfica de trabajo restante o Sprint burn-down

Una gráfica burn-down es un tipo de gráfica utilizada durante los sprints para visualizar fácilmente el trabajo que queda, cómo se ha desarrollado el trabajo a lo largo del sprint, etc. Suele reflejar días en el eje de las x y puntos de historia (o la unidad que utilice un equipo) en el eje y, de manera que la gráfica va bajando (“down”) hasta llegar a cero según el trabajo se completando (quemando, “burn”).

⁵ Es más fácil decidir si una tarea tiene un coste 5 u 8 que decidir si tiene un coste de 5, 6, 7 u 8.

Reuniones o eventos

Scrum prescribe una serie de eventos y reuniones para crear regularidad y eliminar la necesidad de reuniones no definidas en Scrum. Estos eventos determinan intervalos de tiempo concretos: para cada uno se define el tiempo máximo que puede durar. Se consideran una oportunidad para incrementar la transparencia y la inspección, por lo que la no inclusión de alguna de ellas equivale a la pérdida de una oportunidad para revisar y adaptar el trabajo (Schwaber & Sutherland, 2016).

Sprint o iteración

Los sprints o iteraciones son el corazón de Scrum: un periodo de tiempo de aproximadamente un mes durante el cual ha de generarse un nuevo Incremento, esto es, un producto usable y listo para entregar. Los sprints son contenedores del resto de reuniones y eventos. Durante un sprint no se pueden hacer cambios que pongan en peligro el objetivo del mismo y tampoco pueden modificarse los objetivos de calidad. Cada sprint ha de considerarse como un proyecto único de un mes (como máximo) de duración. De establecerse una duración más larga, podrían aparecer cambios en especificaciones, riesgos, etc.

Un sprint puede ser cancelado durante su ejecución por el PO, debido a cambios en la dirección de la empresa o en la situación del mercado, aunque esto inusual dada la brevedad de los sprints.

Planificación del sprint o Sprint planning

La reunión de planificación del sprint es una reunión que puede durar como máximo ocho horas para sprints de un mes, en la cual se elige qué trabajo se va a realizar en ese sprint. En colaboración con el PO, el equipo decide qué funcionalidad se va a implementar durante ese sprint y cómo. Esta reunión se divide a su vez en dos reuniones de un máximo de cuatro horas cada una. La primera reunión es entre el PO y el equipo, durante la cual el PO prioriza los elementos de la pila de trabajo y el equipo elige aquellos que se compromete a implementar durante el sprint, quedando definida la pila del sprint (Sprint Backlog). En la segunda reunión, el equipo decide cómo va a afrontar la implementación de dicha funcionalidad, dividiéndose cada requisito en las tareas necesarias para completarlo, estimando la duración de cada tarea y asignándose a los miembros del equipo.

Reunión diaria o Daily Scrum

La reunión diaria es una reunión de una duración máxima de 15 minutos en la cual el equipo analiza en qué tareas se están trabajando, en qué se trabajó el día anterior y posibles impedimentos que hayan surgido o puedan surgir. Estas reuniones permiten controlar el progreso del trabajo y ver la tendencia del mismo.

Revisión del sprint o Sprint Review

La reunión de revisión del sprint es una reunión informal de un máximo de cuatro horas (para sprints de un mes) en la que participan el equipo de desarrollo, el propietario del producto y los interesados en el producto. Durante esta reunión, se analiza el incremento obtenido al final de un sprint, incluyendo este análisis la comprobación de lo que se ha hecho y de lo que no, estudiar problemas que hayan surgido, considerar cambios en la pila de trabajo, etc. El resultado de esta reunión es una pila de trabajo revisada para el siguiente sprint.

Retrospectiva del sprint o Sprint Retrospective

Esta reunión, que tiene lugar tras la reunión de revisión pero antes de la planificación del siguiente sprint, es una reunión de un máximo de tres horas (para sprints de un mes) en la que el maestro de Scrum analiza, junto con el equipo de desarrollo, el trabajo realizado en sí. Se discute cómo se ha seguido Scrum a lo largo del sprint y se estudian posibles mejoras de la aplicación del proceso en sí.

3.1.2 Kanban

Kanban es un marco de trabajo utilizado por grupos de desarrollo de software ágiles creado por la empresa Toyota en 1950. Este marco de trabajo está definido únicamente por tres principios (VersionOne, Web):

- **Hacer visible el flujo de trabajo.** Facilitar la visibilidad de todo el flujo de trabajo de un grupo puede ayudar enormemente, debido a que permite al equipo tener una idea global acerca de cómo avanza el desarrollo. Para este fin se utilizan los tableros Kanban (Figura 14).
- **Limitar la cantidad de trabajo.** Si se establecen límites acerca de la cantidad de trabajo que puede haber en progreso en todo momento, se equilibra el flujo de trabajo. Esos límites también contribuyen a eliminar los cuellos de botella. El límite puede ser global (por ejemplo, en cada columna del tablero Kanban puede haber como máximo 6 tareas) o local, y haber uno distinto para cada columna del tablero.
- **Favorece el desarrollo del trabajo.** Al estar basado en la entrega continua, cuando se acaba una tarea, se continúa con la siguiente que haya pendiente.

Dado que Kanban es una metodología más flexible y adaptativa (no impone roles ni periodos de tiempo concretos, ni artefactos), suele utilizarse en **combinación** con otras

metodologías, ya que Kanban respeta las restricciones de la metodología que se utilizase previamente.

3.2 Tecnologías utilizadas

En los siguientes apartados se presentan las tecnologías que se han utilizado durante el proyecto, tanto las herramientas usadas para gestionar el proyecto en sí como las herramientas utilizadas durante la implementación. Dado que las tecnologías de detección de emociones ya se han detallado en el capítulo anterior, aquí solo se hará una mención a las mismas.

3.2.1 Afectiva

Afectiva es un proyecto que tuvo comienzo en el MIT y que terminó convirtiéndose en una empresa entera. Dicha empresa ofrece diversos servicios al público, siendo la detección de emociones uno de ellos. Para este trabajo haremos uso de su SDK para Javascript, que nos permitirá leer video directamente de una cámara conectada al equipo donde se ejecute la aplicación y analizará las emociones expresadas por las caras detectadas en los fotogramas de dicho video.

Para utilizar la tecnología de detección de emociones basada en el rostro solo será necesaria una conexión a Internet y una cámara conectada al ordenador.

3.2.2 Beyond Verbal

La empresa Beyond Verbal ofrece un servicio de detección de emociones vía API basado en la voz. Dicho servicio recibe una pista de audio en formato WAV o PCM y analiza los distintos aspectos afectivos de dicha voz.

Para generar las pistas de audio que suministraremos al sistema de Beyond Verbal será necesario un micrófono que nos permita capturar la voz del usuario, junto con una conexión a Internet para realizar las peticiones.

3.2.3 Detección basada en pulsación de teclas y errores

Por último, se va a llevar a cabo un registro de las teclas pulsadas por el usuario, los errores cometidos y la velocidad de pulsación de ciertas teclas para analizar el estrés que pueda estar sintiendo el usuario, y tener en cuenta esta información, junto con la información producida por los servicios anteriores, en la modificación del comportamiento de la aplicación.

Está demostrado que el estrés y la frustración provocada por errores y fallos no solo producen descontento personal, debido al hecho de haber fallado, sino que rompen la concentración del usuario y ralentizan el aprendizaje (Lazar, Jones, Hackley & Shneiderman, 2005). Analizando la tasa de fallos, así como la velocidad de pulsación de las teclas, tenemos un agregado más a la información producida por los servicios de Afectiva y Beyond Verbal.

3.2.4 Phaser

Phaser es una librería gratuita de código abierto para la elaboración de juegos de navegador haciendo uso de Canvas y tecnología WebGL (Phaser, Web). Debido a que Phaser está pensado para juegos que se ejecutan en el propio navegador, un juego construido con esta tecnología puede ejecutarse en múltiples plataformas.

Phaser nos permite gestionar el movimiento de elementos, la animación de personajes, la entrada de distintos periféricos, el sonido del juego, la física aplicada a cada uno de los elementos del juego, etc. En su página web se ofrecen una serie de ejemplos de cada una de las funciones que ofrece, de manera que sin demasiado conocimiento de Javascript (lenguaje en que se programan los juegos de Phaser) puede empezarse con un proyecto base con cierta funcionalidad incluida.

Phaser se empezó a utilizar en una asignatura previa del grado, durante la cual se desarrolló el juego que se ha utilizado como base en este proyecto, por lo que se ha mantenido para este trabajo.

3.2.5 Heroku

Heroku es una Plataforma ofrecida como Servicio (Platform-as-a-Service) que permite realizar el despliegado de aplicaciones de forma rápida y sencilla (Heroku, Web). Aunque ofrece diversos paquetes de servicios, hay también una opción gratuita que permite llevar a cabo el despliegado de una aplicación en una url del tipo “http://nombre_de_aplicacion.herokuapp.com”. Incluso en la versión gratuita, Heroku ofrece

una gran cantidad de opciones y métricas para controlar la ejecución de la aplicación. También permite vincular Heroku con un repositorio de GitHub, de manera que si el código de la aplicación está en GitHub (como es el caso), se puede automatizar el despliegado para que tenga lugar cada vez que ocurre un “push” del código.

Heroku soporta el despliegado de varios tipos de tecnologías, estando entre ellas NodeJS, la tecnología con la que está hecha el lado del servidor de la aplicación. La configuración de una aplicación para su despliegado se limita a determinar con qué comandos se pone en marcha el servidor y qué otras tecnologías utiliza NodeJS.

3.2.6 mLab

mLab (mLab, Web) es una Base de datos ofrecida como Servicio (Database-as-a-Service) basada en MongoDB, una base de datos NoSQL que nos ofrece la funcionalidad de una base de datos tradicional unida a la flexibilidad que aporta el hecho de ser NoSQL. Las “tablas” de mLab son colecciones de **objetos** (en el sentido de objeto de Javascript, una colección de propiedades), de manera que en una misma “tabla” pueden convivir objetos con distintas propiedades sin que esto suponga ningún problema. Esta flexibilidad es una gran ventaja sobre todo en fases iniciales, cuando puede que surjan cambios que impliquen cambiar la forma en la que se insertan nuevos objetos, sin afectar esto a la recuperación de objetos previos a dicha modificación. Al igual que Phaser y Heroku, esta fue la tecnología adoptada en su momento y se ha mantenido por comodidad. mLab ofrece 500 megabytes de almacenamiento de forma gratuita, más que suficiente para el desarrollo del prototipo.

3.2.7 Aspectos legales

Según la Agencia Española de Protección de Datos, un dato de carácter personal es *“cualquier información numérica, alfabética, gráfica, fotográfica, acústica o de cualquier otro tipo concerniente a personas físicas identificadas o identificables”* (Agencia, Web). Dado que las tecnologías de Afectiva y Beyond Verbal utilizan imágenes de la cara, así como la voz del usuario del juego, hay que considerar las implicaciones legales que eso pueda tener.

En primer lugar, la política de privacidad de Afectiva establece que, si bien Afectiva si recopila información con vistas a mejorar su servicio, esta información es completamente anónima, únicamente relativa a la detección de emociones realizada, y no a las imágenes sobre las que se ha hecho la detección (Discuss Afectiva, Web). Ahora bien, incluso aunque Afectiva no manipule datos de carácter personal, la aplicación si va a realizar un tratamiento de los mismos. El tratamiento de datos está definido en la Ley Orgánica de Protección de Datos (en adelante LOPD) como “operaciones y procedimientos técnicos de carácter

automatizado o no, que permitan la recogida, grabación, conservación, elaboración, modificación, bloqueo y cancelación, así como las cesiones de datos que resulten de comunicaciones, consultas, interconexiones y transferencias” (BOE, Web). Aunque los datos extraídos de la cámara web no se almacenan ni alteran de ninguna manera, si se realiza un tratamiento con ellos, que consiste en enviarlos a Afectiva a través de su detector.

Aun cuando se da este tratamiento, el artículo 2 de la LOPD estipula que dicha ley solo es aplicable “a los datos de carácter personal registrados en soporte físico” y puesto que la aplicación no registra esos datos de carácter personal, para hacer uso de la tecnología de Afectiva no hay que tener mayor consideración que la de avisar al usuario de cómo funciona la aplicación. Además, el apartado 1.c del artículo 2 de la ley anterior establece que la LOPD no es aplicable cuando el tratamiento se hace fuera de la Unión Europea y los medios situados en España son solo para el tránsito de dichos datos. Dado que Afectiva está radicada en Estados Unidos y los medios que hay en España se limitan a enviar los datos, se cumple esta excepción.

En el caso de Beyond Verbal, esta empresa sí almacena las pistas de audio que recibe para trabajar en mejorar en su servicio, aunque la aplicación se mantiene el amparo de la excepción comentada anteriormente (los medios situados en España son únicamente para transmisión). No obstante, lo hace de forma similar a Afectiva, esto es, no almacenan pistas de audio que contengan datos de carácter personal. Beyond Verbal almacena las pistas de audio que no contengan datos de carácter personal y las utiliza para entrenar sus sistemas de clasificación. Pasado un tiempo, esas pistas de audio son borradas. Si a la hora de usar las pistas se descubre que contienen información de carácter personal, se eliminan inmediatamente⁶. Además, la pista de audio grabada solo contendrá la voz del usuario leyendo unas frases en inglés.

Respecto a la detección que se hace de las teclas, solo se miden errores y excesiva velocidad de pulsación de las teclas de control, por lo que no se registra ningún tipo de dato de carácter personal.

3.3 Herramientas de soporte al desarrollo del proyecto

GitHub

GitHub es una plataforma pensada para desarrollar software de forma colaborativa que utiliza el control de versiones Git para gestionar los cambios realizados sobre el código (GitHub, Web). GitHub permite llevar un seguimiento de los fallos y problemas que han

⁶ Esta información ha sido extraída de una serie de correos electrónicos intercambiados con el personal de contacto de Beyond Verbal, ya que esa información relativa al uso del audio guardado no está disponible en su página web.

aparecido, crear una página colaborativa con información asociada al proyecto, llevar un registro de quién ha hecho qué y cuándo, dividir el desarrollo en ramas, de manera que se pueden estar desarrollando cosas distintas sobre una misma base sin conflictos, etc. Cada proyecto en Github es un repositorio, siendo estos públicos, aunque se pueden hacer privados mediante el pago de una cuota.

Pese a que existen otras opciones para gestionar los cambios del software (algunas como Bitbucket ofrecen repositorios privados de forma gratuita), el proyecto base sobre el que se partió se encontraba ya configurado en GitHub, por lo que se continuó usando ese repositorio.

Kunagi

Kunagi es una herramienta web gratuita para la gestión de proyectos basada en Scrum (Kunagi, Web). Al estar pensada para una gestión basada en Scrum, dispone de los mecanismos propios de Scrum (Product y Sprint Backlog, gráficas burn-down, gestión de riesgos, impedimentos) a la vez que otras características para facilitar la gestión de proyectos como, por ejemplo, una pizarra interactiva para la gestión de las tareas (que recuerda mucho a un tablero Kanban), una herramienta para llevar a cabo el Planning Poker, una sección para la comunicación de los miembros del proyecto, etc.

Aunque existen otras herramientas para llevar a cabo la gestión del proyecto, como Visual Studio Team Services, se optó por hacer uso de Kunagi debido a la experiencia previa que se tenía por haberlo trabajado en asignaturas pasadas y a que el servidor a utilizar ya estaba configurado y se tenían credenciales del mismo. Incluso cuando dicho servidor cayó, se volvió a levantar uno local también de Kunagi.

StarUML

StarUML es una herramienta para el modelado UML que ofrece soporte para diagramas de clases, de desplegado, de casos de uso, de secuencia, de comunicación, de estado, de actividad, entre otros. Si bien no es una herramienta gratuita, se ofrece una versión de evaluación sin tiempo límite con toda la funcionalidad disponible (StarUML, Web).

Trello

Trello es una herramienta gratuita pensada para la gestión de proyectos haciendo uso de tableros, listas y tarjetas (Trello, Web). Las tarjetas pueden agruparse mediante etiquetas, contener archivos, asignarse a colaboradores, etc.

Se hará uso de Trello para visualizar y gestionar los tableros Kanban. Aunque Kunagi ya ofrece una pizarra basada en Kanban (tareas repartidas en columnas de “Pendiente”, “En

progreso”, “Acabado”), Trello es más accesible a la hora de compartir los tableros con otros implicados en el proyecto. Cuando hubo que hacer uso de un servidor local de Kunagi, las tareas solo eran accesibles para los tutores del trabajo a través de Trello. En la Figura 16 podemos ver el aspecto que tenía el tablero Kanban asociado al TFG al empezar el sprint 1.

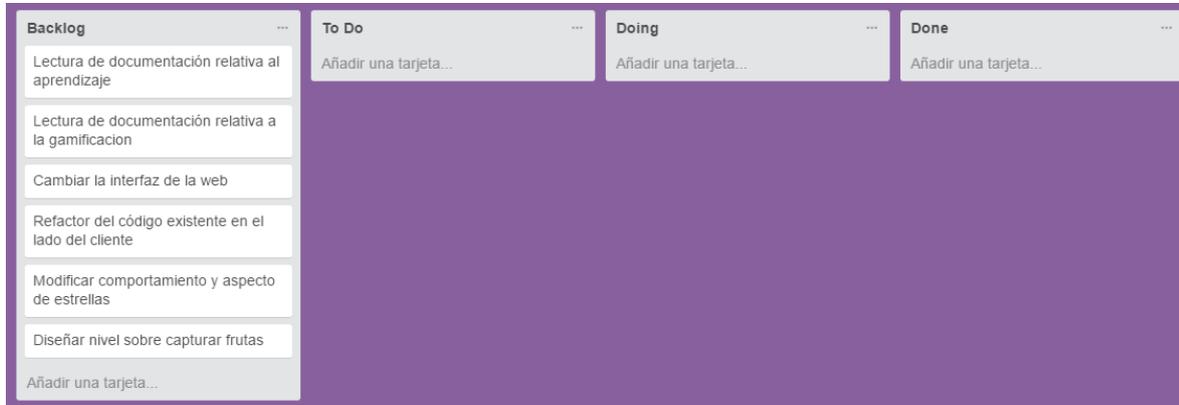


Figura 16. Tablero Kanban al comienzo de Sprint 1

Visual Studio 2015

Poderosísimo entorno de desarrollo perteneciente a Microsoft, en este proyecto se utilizará para modelar aquellos aspectos más complejos del sistema que StarUML no soporte, como los diagramas de actividad con actividades compuestas.

Visual Studio Code

Visual Studio Code es un editor de código que da soporte a tareas de desarrollo como la búsqueda de errores, la ejecución de código y el control de versiones (Visual Studio Code, Web). Es una aplicación de código abierto mantenida por Microsoft y aunque en un principio es un editor de código, su gran mercado de extensiones permiten usarlo como se utilizaría un IDE completo, pero con la ligereza de un editor de código.

Las tareas de desarrollo y control de versiones se llevarán a cabo con este editor.

3.4 Conclusiones

En este capítulo se ha presentado la metodología por la que se va a regir el desarrollo del prototipo, una combinación de Scrum y Kanban, que a su vez será adaptada a la situación del proyecto y el contexto de trabajo.

Asimismo, se ha dejado constancia de las tecnologías que van a utilizarse durante el proyecto: desde las tecnologías de detección de emociones elegidas y/o implementadas hasta

las herramientas usadas para gestionar dicho proceso de desarrollo, pasando por las tecnologías sobre las que se construirá la aplicación. En el siguiente capítulo se verá cómo éstas son aplicadas durante el desarrollo del proyecto.

CAPÍTULO 4. DESARROLLO DEL SISTEMA

En este capítulo se detallará el proceso a través del cual se ha desarrollado la aplicación del TFG. El proceso de desarrollo, realizado bajo el marco de trabajo de Scrum y haciendo uso de tableros Kanban y de la limitación del trabajo que puede haber en desarrollo en un momento dado, se ha alargado durante cuatro sprints, existiendo una fase de exploración previa (Sprint 0) y una fase de evaluación final posterior. En el siguiente documento veremos cómo la aplicación fue creciendo y desarrollándose desde el punto de partida inicial.

Dado que se ha utilizado Kunagi, que da la opción de generar informes con los datos de un sprint, parte de esa información se ha utilizado en este capítulo, encontrándose los informes completos dentro del CD que se entregó junto con este TFG.

4.1 Introducción al desarrollo

En este primer apartado se presentan las adaptaciones que se han realizado sobre la metodología de trabajo empleada, así como los stakeholders⁷ del proyecto y la estructura de los siguientes apartados, donde se detalla la evolución del mismo.

Adaptaciones de las metodologías

Dado que el presente proyecto ha sido desarrollado por un equipo de una única persona durante el transcurso de un cuatrimestre, se han realizado algunas adaptaciones sobre las metodologías aplicadas.

- **Se prescinde de las reuniones diarias de Scrum.** Puesto que el equipo consta de una única persona, no es necesario hacer estas reuniones.
- **Se produce un solapamiento de los roles impuestos por Scrum:** tanto el rol de Scrum Master (SM para futuras referencias) como el de Equipo serán desempeñados por el autor de este trabajo, mientras que el rol de Product Owner (PO para futuras referencias) será desempeñado conjuntamente por el autor y los directores de este proyecto.
- **Se elimina la restricción de continuidad de los sprints.** Para poder compaginar mejor la elaboración del trabajo con las asignaturas del autor, se permite retrasar el comienzo de un sprint de manera que no tenga que empezar el mismo día que acaba el anterior.
- **La limitación de trabajo en progreso (WIP para futuras referencias)** se aplica solo a las tareas que estén en la fase de desarrollo (“Doing”). La cantidad de tareas que puede haber en desarrollo como máximo será la mitad de tareas que hubiera inicialmente en el Sprint Backlog (SB para futuras referencias). Si hay tareas que se pueden clasificar en distintas categorías, se distinguirá una sublimitación dentro de esa categoría, estando determinada esa sublimitación por la proporción de tareas que suponga esa categoría sobre el total de tareas. Esta cantidad queda determinada por la ecuación de vemos a continuación.

$$WIP_{\text{categoría } i} = \frac{NT_{\text{categoría } i}}{NT_{SB}} * WIP_{\text{global}}$$

Siendo NT_{SB} el número de tareas que tiene inicialmente el SB y $NT_{\text{categoría } i}$ el número de tareas de esa categoría que había inicialmente en el SB.

⁷ Persona interesada en el proyecto.

Si solo hay una categoría, el WIP de esa categoría coincidirá con el WIP global, estando éste determinado por la ecuación presentada a continuación. Si, por ejemplo, un sprint está compuesto únicamente por tareas de implementación y desarrollo, y hay 10 tareas en el SB, entonces el WIP global será de 5. Si además de esas 10 tareas de implementación, tenemos 8 de documentación, entonces el WIP de las tareas de implementación será de 5 y el de las tareas de documentación será de 4.

$$WIP_{global} = \frac{NT_{SB}}{2}$$

A la hora de determinar el WIP, no se ha tenido en cuenta la duración de las tareas, debido a que los tiempos de duración son muy similares.

Stakeholders identificados

Al decirse la temática y la dirección que iba a tomar el proyecto, se identificaron los principales stakeholders del proyecto.

- **Cliente.** El rol del cliente es desempeñado por los directores del proyecto. Su colaboración en el proyecto es vital, ya que en colaboración con ellos se establecían requisitos, se acordaba el rumbo que debía seguir el proyecto, etc.
- **Estudiantes de primaria.** Los usuarios finales de la aplicación serán niños en edad de primaria entre los 7 y los 11 años. Dado su interés en explorar otras formas de aprendizaje aparte de la memorización y la repetición, se han mostrado muy participativos en las fases de evaluación. Su opinión será muy importante para determinar ciertos aspectos de la interacción con la aplicación.
- **Docentes.** Aunque el prototipo desarrollado no estará cerrado a un usuario particular que quiera registrarse y probar la aplicación, en la mayoría de los casos tendremos una figura docente que registrará a los niños con los que vaya a usar la aplicación y que los guiará durante el juego. Es por ello que los docentes también tienen interés en la aplicación, así como en las herramientas que ofrezca para gestionar a los usuarios a su cargo, analizar sus resultados, etc.

Planificación del proyecto

En la Tabla 2 podemos ver los sprints que se han llevado a cabo para el desarrollo de la aplicación.

Tabla 2. Sprints realizados

Id Sprint	Descripción
0	Estudio base sobre Computación Afectiva
1	Preparación de primer prototipo
2	Diseño de niveles
3	Integración de tecnología afectiva en el prototipo
4	Implementación de afección en el prototipo

El cronograma final de este proyecto queda recogido en la Figura 17, en la cual se han reflejado la evolución real del proyecto en el tiempo. Al acabarse el último sprint, se empezaron a desarrollar labores de evaluación y documentación, existiendo un intervalo entre el último sprint y éstas debido a entregas y exámenes de otras asignaturas.

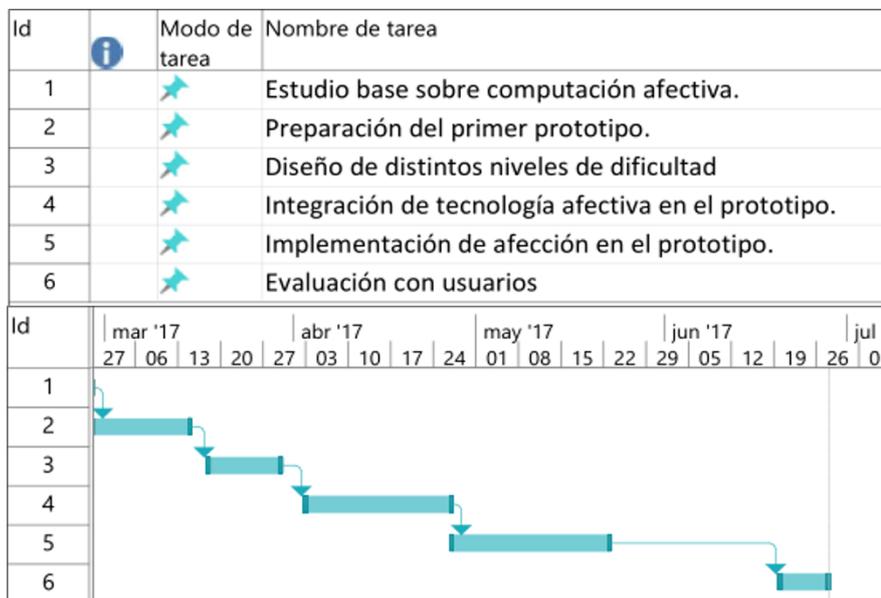


Figura 17. Planificación temporal real

Estructura de la documentación del presente capítulo

Los siguientes apartados contienen los detalles acerca de cada sprint realizado. A excepción del sprint 0, que no es un “sprint” como tal, estos apartados tienen las siguientes secciones.

Planificación

En este apartado se detalla el objetivo del sprint, su duración y se comentan decisiones previas que se tomaron relativas a lo que debía producirse al final del sprint. Las dos reuniones previas al sprint que impone Scrum se fusionan en una sola, dado el tamaño del

equipo de desarrollo. En esta primera fase resultan de vital importancia los siguientes artefactos.

- **Product Backlog (PB para futuras referencias).** En este apartado se indicarán cuáles han sido las modificaciones y/o adiciones realizadas sobre el PB. Cada historia llevará una estimación en Story Points que se habrá realizado con la técnica del Planning Poker. La prioridad de las historias estará marcada por su orden en el PB, de manera que las primeras serán las de mayor prioridad.
- **Sprint Backlog (SB para futuras referencias).** Se indicarán las historias que se van a implementar en ese sprint, junto con las tareas asociadas a cada historia y una estimación de la duración de cada tarea.

Desarrollo

En este apartado se expone el trabajo realizado durante el sprint correspondiente, así como los artefactos que hayan podido generarse y los impedimentos que hayan podido surgir.

Revisión

Al final del sprint se realiza la revisión del incremento con el PO, quedándose recogido en este apartado una pequeña acta de dicha reunión.

Retrospectiva

Por último, se realiza un análisis de la gráfica burn-down resultante de este sprint, para ver como se ha desarrollado el sprint, posibles aspectos a mejorar, etc.

4.2 Sprint 0. Estudio base sobre Computación Afectiva

Aunque no se trabajó bajo el marco de Scrum hasta el sprint 1, previamente se estuvo estudiando el campo de la Computación Afectiva, se realizó una investigación base sobre aplicaciones que variasen su comportamiento dinámicamente, etc. Durante esta “sprint” se sentaron las bases del presente proyecto

Esta investigación se realizó de forma completamente libre, sin ningún marco de trabajo. No obstante, se ha utilizado una notación similar a la del PB para expresar las historias que habrían compuesto este sprint. En la Tabla 3 podemos ver el aspecto que habría tenido el PB de esta investigación inicial si se hubiese seguido Scrum.

Tabla 3. Sprint 0. Product Backlog

ID	Nombre	Descripción
SA	Creación de base de conocimiento acerca de la Computación Afectiva.	Como cliente quiero saber con cierto nivel de detalle en qué consiste la Computación Afectiva con la finalidad de saber qué aspectos pueden resultar más interesantes de investigar y explotar.
SB	Creación de base de conocimiento acerca de las tecnologías enmarcadas en la Computación Afectiva.	Como cliente quiero saber qué tecnologías ofrece el mercado de la Computación Afectiva, con la finalidad de saber de qué herramientas dispondría en un proyecto de este campo.
SC	Creación de un primer prototipo del juego.	Como cliente quiero que mi aplicación varíe su comportamiento en base a las emociones de los usuarios para que la experiencia de usuario sea óptima.

La realización de la historia SA consistió en realizar una exploración acerca del campo de la Computación Afectiva: cuándo surgió, quién o quiénes lo desarrollaron, cómo es la Computación Afectiva en la actualidad, qué proyectos existen relacionados con este campo, dónde se ha aplicado con más interés, qué aspectos de la disciplina resultan más novedosos y/o más explotables, etc. De la realización de esta historia surgió la mayor parte de la bibliografía de que se dispone actualmente.

En segundo lugar, la realización de la historia SB implicó una exploración de los distintos servicios que se ofrecían en este campo. Dado que la Computación Afectiva gira en torno a la detección de las emociones (la mayoría de proyectos y estudios relacionados con la Computación Afectiva consisten en utilizar las emociones detectadas para estudiar el impacto que un estímulo tiene en un sujeto concreto, ya sea en un usuario probando un producto o una persona con trastorno del espectro autista reaccionando a una situación), ha surgido un gran número de empresas centradas en ofrecer, exclusivamente, servicios de

detección, siendo predominantes las tecnologías de detección basadas en la expresión facial y en texto escrito. La realización de esta historia consistió en investigar acerca de los tipos de detectores de emociones que existían, qué tipo de información se podría utilizar para extraer información afectiva, cómo se integrarían dichas tecnologías en un sistema, etc. Fruto de esta tarea surgió la información que puede encontrarse en el apartado 2.1 de este documento. A lo largo de este “sprint” se mantuvieron reuniones —más o menos periódicas— con el que luego sería el PO del proyecto en el marco de Scrum para analizar la información recogida y estudiar los hallazgos realizados.

Por último, se utilizaron las herramientas comentadas en la sección 3.2 (Phaser, Heroku y mLab) para desarrollar un primer prototipo del juego. Cuando se elaboró este primer prototipo aún se estaban barajando distintos temas, por lo que el tema de esta primera versión quedó en manos del autor, que desarrolló un juego de plataformas con un planteamiento muy clásico en lo que a temática respecta. En este primer prototipo no se habrían integrado tecnologías de detección, puesto que aún se estaba analizando y decidiéndose cuales eran las más interesantes para las circunstancias.

Al cierre de este “sprint”, se había obtenido una buena base de conocimiento acerca de la Computación Afectiva y las tecnologías de desarrollo web empleadas durante la elaboración del prototipo. Para que el desarrollo fuera más rico, y siguiendo el espíritu de un proyecto desarrollado previamente (de la Guía et al., 2016) en el seno del equipo de investigación en el que se va a desarrollar este TFG, se optó por seguir desarrollando este primer prototipo para que no fuese solo un juego con un fin lúdico, sino que aportase algo más a sus usuarios. Así, se acordó que, a la par que se dotaba de afectividad al prototipo, se modificaría para convertirlo en una aplicación que pudiera ser usada por niños en edad de educación primaria para aprender inglés. Se acordó que la temática del juego sería la cocina y se dejó para el sprint 1 la elaboración de un primer prototipo de este juego educativo. De esta manera, el prototipo inicial constituye el proyecto base de este trabajo.

4.3 Sprint 1. Preparación de primer prototipo

4.3.1 Planificación

Objetivo. Recopilar información sobre gamificación y motivación de estudiantes. Adaptar el proyecto base para crear un primer prototipo de la aplicación acordada.

Duración. 16 días (27/02/2017 - 14/03/2017)

En la primera reunión de planificación del sprint celebrada entre el PO y el equipo, se estableció el objetivo inicial del proyecto. Como primerísimo paso, era necesario crear una base de conocimiento relativa a las nuevas variables que se habían introducido en la ecuación: la posibilidad de utilizar juegos y/o mecanismos propios de juegos para facilitar un proceso de aprendizaje. Antes de decidir cómo introducir todo esto en el prototipo, era necesario entender y conocer dichos elementos, para poder estudiar su unión en la aplicación. De esta reunión surgió la primera versión del PB, que podemos ver en la Tabla 4.

Tabla 4. Sprint 1. Product Backlog

ID	Nombre	Descripción	SP
S1	Creación de base conceptual sobre gamificación y aprendizaje.	Como cliente quiero que mi aplicación se fundamente en técnicas de gamificación y aprendizaje con la finalidad de que su aprovechamiento por parte del usuario sea máximo.	13
S2	Modificación del proyecto base.	Como cliente, quiero que la aplicación base pueda usarse como herramienta para aprender expresiones en inglés usando la cocina y la comida como temática, con la finalidad de atraer al público más joven.	5
Total			18

Las historias de la Tabla 4 se descompusieron en las tareas que pueden verse en la Tabla 5, cuya completitud nos permitirá determinar una primera versión completa del PB.

Tabla 5. Sprint 1. Sprint Backlog

Historia	Tarea	Horas	+/-
Creación de base conceptual sobre gamificación y aprendizaje.	Lectura de documentación relativa a la gamificación.	6	0
	Lectura de documentación relativa al aprendizaje.	6	-1
Modificación del proyecto base.	Cambiar la interfaz de la web para que refleje su contenido y finalidad educativa.	5	0
	“Refactoring” del código existente en el lado del cliente.	7	-2
	Modificar comportamiento y aspecto de estrellas.	2	0
	Diseñar nivel sobre capturar frutas.	3	0
Total		29	-3

4.3.2 Desarrollo

En lo que respecta a la creación de la base de conocimiento inicial, se empezó investigando los aspectos claves de la gamificación, puesto que fue de las primeras cosas que se comentaron cuando se habló de desarrollar una aplicación con finalidad didáctica. La gamificación se define como la aplicación de mecanismos propios de juegos en entornos que no son de juegos (Huotari & Hamari, 2012). La gamificación se ha aplicado en muchos campos durante los últimos años y se ha vuelto un mecanismo muy popular, ya que los usuarios se muestran mucho más dispuestos a usar un sistema cuando este ofrece, por ejemplo, recompensas tras alcanzar ciertos hitos.

Ahora bien, dado que la gamificación implica aplicar mecanismos de juegos a elementos que no lo son, no es algo que pudiéramos aplicar al prototipo. Dado que el proyecto base ya es un juego, no podríamos gamificarlo, salvo que eliminásemos el juego y se implementase otra aplicación en su lugar que no fuese un juego, a la cual si podríamos aplicar técnicas de gamificación. Se tomó la decisión de mantener el juego, salvo que se modificaría para que pudiera usarse para aprender inglés. Esta decisión hizo que el objetivo de desarrollo virase, de manera que en lugar de gamificar el prototipo, se desarrollaría para convertirlo en un juego serio (“serious game”), estando un juego serio definido como un juego cuyo objetivo principal no es entretener (Djaouti, Alvarez & Jessel, 2011).

En lo que respecta a la educación, se analizaron sobre todo noticias e informes acerca del fracaso escolar, de cómo los alumnos pierden la motivación, etc. Además, se hizo una búsqueda acerca de investigaciones y noticias que combinaran aprendizaje y Computación Afectiva, siendo (Santos, 2016) una de las más importantes, debido a las investigaciones que a su vez se recogen en dicho artículo. Destacaron también las charlas de Ken Robinson durante las TED Talks Education (TED, Web).

En lo que respecta al desarrollo como tal, el primer paso fue eliminar funcionalidad que no iba a utilizarse, para facilitar a su vez las modificaciones pertinentes. Para el comienzo de este sprint, el juego del proyecto base presentaba el aspecto que puede verse en la Figura 18. En este punto, el juego consistía en alcanzar la franja azul que se ve en la parte superior del juego, esquivando al mismo tiempo los meteoritos, ya que estos quitaban vidas.

A su vez, el juego se presentaba como un juego de rescate en el que había que rescatar a Kairi, una amiga del protagonista (Figura 19). Así, se abordó la tarea de modificar el aspecto gráfico de la interfaz, de manera que comunicase que el prototipo era un juego educativo con un objetivo distinto al actual.

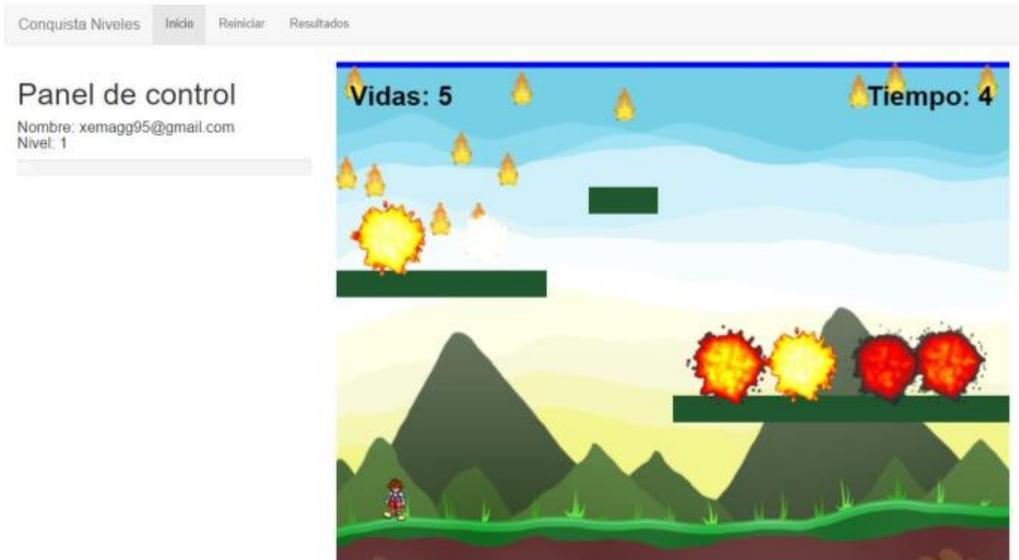


Figura 18. Proyecto base a comienzos de sprint 1 (2)

Durante la reunión de planificación, se habló de crear un prototipo de un primer nivel en el que cayesen frutas u otros ingredientes (para aprovechar parte de la funcionalidad que presentaban las estrellas) y que el usuario tuviese que coger. Aunque se consideraron varias opciones sobre cómo implementar ese nivel prototípico, se optó por elegir esta opción para un primer nivel de prueba. Se cambiaron las estrellas por plátanos y manzanas y se determinó que el usuario tenía que coger cierta cantidad de cada fruta para completar ese nivel. Se añadió también un arcaico reconocedor de voz para estudiar la posibilidad de combinar ese juego con el uso de la voz, y también se implementó un pequeño ejemplo de “traductor”, para estudiar la posibilidad de utilizar textos en inglés con el fin de dar instrucciones, en cuyo caso los usuarios podrían traducir las instrucciones colocando el cursor encima de la frase que corresponda.

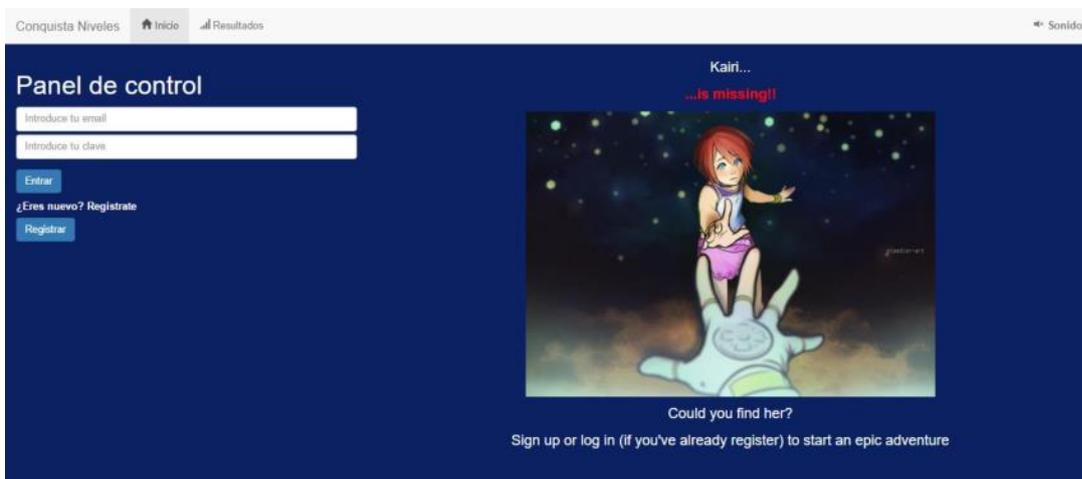


Figura 19. Proyecto base a comienzos de sprint 1 (2)

El sprint transcurrió sin mayor inconveniente, pero durante los últimos días del mismo, el servidor remoto Kunagi que se estaba utilizando para el proyecto sufrió un problema que obligó a un reinicio absoluto, lo que ocasionó la pérdida del informe de este sprint, así como de la gráfica burn-down del mismo.

4.3.3 Revisión

Durante la revisión del prototipo con el PO, se repasaron los cambios que se habían realizado sobre la interfaz, como se puede apreciar en las figuras Figura 31, Figura 32, Figura 33 y Figura 34 (véase Anexo A, apartado A.1). De forma temporal, se dio al prototipo el nombre de “Aprende Jugando” y se optó por dejar en segundo plano el aspecto estético del mismo, aunque se acordó que habría que modificarlo y se propusieron guías de diseño como Material Design (Material, Web). Se elaboró un también un diagrama de clases para reflejar el modelo conceptual y sobre el que discutir ciertos aspectos de la aplicación (Anexo D, Figura 67).

En cuestión de funcionalidad, se acordó que se mantendría el reconocedor de voz y que se crearían niveles y/o tareas que forzasen al usuario a hablar, de manera que pudieran utilizarse servicios de detección de emociones en la voz. Se acordó también que se introducirían los detectores de emociones basados en expresiones faciales, ya que se pueden insertar en el prototipo sin necesidad de pedir al usuario que haga nada concreto para detectar su rostro. Se analizó también la opción de añadir un detector más, ya que está probado que cuando se combinan resultados afectivos (como en un sistema multimodal) la tasa de acierto de la detección es mucho más alta, pero esto se dejó para más adelante. Los resultados de los detectores habrán de usarse para evaluar la experiencia de juego del usuario y modificar la dificultad del juego en consecuencia: si se ha detectado estrés y/o frustración, la dificultad habrá de disminuir, mientras que, si se ha detectado aburrimiento y/o una gran neutralidad, ésta deberá incrementarse.

En lo que respecta al juego, se decidió que los niveles se plantearían en forma de recetas. Cada nivel sería una receta, lo que implicaría recoger ciertos ingredientes, y para complicar la recogida de los mismos, deberían caer otros elementos que no fuesen válidos y que supusieran una penalización. Además, se comentó que los ingredientes que habría que coger debían ponerse más cerca del juego en sí, de manera que el usuario no tuviera que desviar tanto los ojos para saber qué ingredientes tiene que coger (Figura 33), cuántos necesita de cada uno, etc. Se decide también prescindir de la traducción de instrucciones, y mantenerlas en español de forma inicial, aunque se considera para un trabajo futuro la posibilidad de implementar la aplicación para que ofrezca varios idiomas.

De esta reunión, y teniendo en cuenta las consideraciones previas, surgió lo que podemos considerar un primer PB (Tabla 6).

Tabla 6. Product Backlog inicial

ID	Nombre	Descripción
S3	Diseño de distintos niveles de dificultad.	Como cliente quiero que el juego invite al usuario jugar con él, existiendo distintos niveles de dificultad, cada cual más complejo, para favorecer el “engagement” de los usuarios.
S4	Integración de prototipo con tecnología de detección afectiva.	Como cliente, quiero que la aplicación sea capaz de estimar el estado emocional del usuario.
S5	Desarrollo de lógica afectiva.	Como cliente, quiero que el juego sea capaz de deducir un estado emocional en base a los resultados de los servicios de detección.
S6	Implementación de prototipo afectivo.	Como cliente, quiero que el juego varíe su comportamiento y dificultad según las emociones del jugador.

4.3.4 Retrospectiva

A pesar de que se trataba del primer sprint realizado en el marco que constituye este TFG, la experiencia previa de Scrum adquirida durante la asignatura de Procesos de Ingeniería del Software estaba reciente, por lo que tanto las estimaciones como la regularidad del trabajo fueron fáciles de conseguir.

Gracias a que este sprint coincidió con el comienzo del segundo cuatrimestre, la carga de trabajo fue muy llevadera. Solo hubo dos estimaciones que hubo que modificar, relativas a la lectura de documentación relativa al aprendizaje y a la refactorización en el lado del cliente. En lo que respecta al primero, al poco tiempo de empezar a investigar se descubrió que, si bien el campo del aprendizaje y neuroaprendizaje es muy amplio, existía una serie de conceptos básicos que se adquirieron más rápido de lo esperado. Respecto a la refactorización, esta llevó también menos tiempo del estimado debido a que parte de la misma consistió en eliminar código, lo que a su vez redujo la tarea de recodificar ciertas cosas.

Debido a la caída sufrida por el servidor de Kunagi que se estaba utilizando, no se conserva información del primer sprint, lo que implica que no se tiene acceso a la gráfica burn-down. A modo de tarea de emergencia, se aprovecha el tiempo entre este sprint y el siguiente para preparar un servidor local de Kunagi.

4.4 Sprint 2. Diseño de niveles

4.4.1 Planificación

Objetivo. Llevar a cabo un diseño consistente de los distintos niveles de dificultad que puede presentar el prototipo, estando determinada la variación de los mismos por el estado anímico del usuario.

Duración. 12 días (18/03/2017 – 29/03/2017)

Durante la reunión de planificación del sprint, el PO estuvo de acuerdo en que, si bien las historias más importantes eran aquellas relativas a implementación del prototipo afectivo y al desarrollo de la lógica afectiva, era preferible dejarlas para el final, dándose prioridad, por el momento, al diseño del sistema de niveles de dificultad y, posteriormente, a la selección e integración de detectores de emociones en el proyecto. En la Tabla 7 podemos ver el aspecto del PB al comienzo de este sprint, ordenado por prioridad de las historias.

Tabla 7. Sprint 2. Product Backlog

ID	Nombre	Descripción	SP
S3	Diseño de distintos niveles de dificultad.	Como cliente quiero que el juego invite al usuario jugar con él, existiendo distintos niveles de dificultad, cada cual más complejo, para favorecer el “engagement” de los usuarios.	13
S4	Integración de prototipo con tecnología de detección afectiva.	Como cliente, quiero que la aplicación sea capaz de estimar el estado emocional del usuario.	21
S5	Desarrollo de lógica afectiva.	Como cliente, quiero que el juego sea capaz de deducir un estado emocional en base a los resultados de los servicios de detección.	8
S6	Implementación de prototipo afectivo.	Como cliente, quiero que el juego varíe su comportamiento y dificultad según las emociones del jugador.	13
Total			55

De esta manera, el SB de este sprint está formado únicamente por la historia, que a su vez queda dividida cuatro tareas, tal y como podemos ver en la Tabla 8.

Tabla 8. Sprint 2. Sprint Backlog

Historia	Tarea	Horas	+/-
Diseño de distintos niveles de dificultad.	Modelado de la estructura que soportará el sistema de niveles.	2	0
	Implementación del sistema de niveles.	3	0
	Integración del sistema de niveles con el prototipo base.	4	0
	Selección de conjunto de información afectiva que empleará la aplicación.	6	-1
Total		15	-1

4.4.2 Desarrollo

Tras varios planteamientos y posibilidades, se estableció un vocabulario común en lo que respecta a los niveles y a la dificultad de los mismos para facilitar la comunicación entre los implicados en el proyecto. Así, quedan definidos los términos:

- **Nivel.** Se entiende por nivel al conjunto de actividades que lleva asociada una receta. En este momento esas actividades implican solo la recogida de ingredientes, pero se añadirán más pasos en un futuro para aprovechar de alguna manera la emoción que revela una voz.
- **Receta.** Conjunto de ingredientes que hay que recoger durante un nivel. La secuencia de recetas (el orden en el que se juegan) es fijo, y está determinado por el orden de las recetas en el fichero JSON donde se almacena la información para la construcción de niveles.
- **Juego.** Aunque a veces se puede usar el término juego como una generalización para la aplicación, este término está reservado para la parte del nivel que consiste en la recolección de ingredientes.
- **Dificultad.** Valor numérico que determina los valores que habrán de tomar las variables que influyen en la dificultad del juego.

Respecto a cómo modelar e implementar la dificultad, hubo que tomar ciertas decisiones. La primera fue respecto a cuándo modificarla. Para este propósito se consideraron dos enfoques:

- **Modificar la dificultad durante el propio juego.** Leer las emociones mientras el usuario se encuentra en la fase de juego, y variar la dificultad en base a las emociones leídas, modificando en tiempo real aspectos como la velocidad de

los ingredientes, el número de ingredientes que caen, la velocidad del personaje, etc.

- **Modificar la dificultad entre niveles.** Leer las emociones mientras el usuario se encuentra en la fase de juego y analizar los resultados cuando termine. En base a las emociones leídas, modificar la dificultad del siguiente nivel.

Se optó por adoptar el segundo enfoque, ya que el primero podría provocar mayores frustraciones si el juego se sobreajustase: si la dificultad oscila demasiado en un mismo nivel, esto podría resultar contraproducente en el objetivo de mantener la atención del usuario.

En segundo lugar, había que decidir qué aspectos se considera que determinan la dificultad. En lo que se refiere a esta decisión, se consideraron también dos enfoques:

- **Dificultad en base a características de la receta.** Se pueden definir varios criterios en base a las características de una receta (número de ingredientes, número de plataformas, cantidad de ingredientes de cada tipo) para determinar la dificultad de la misma. A su vez, habría que definir varios niveles de dificultad (Ej., Fácil, Normal y Difícil), de manera que al acabar un nivel se determinase cual debía ser la dificultad de la siguiente receta, eligiéndose al azar una receta que tuviese ese nivel de dificultad.
- **Dificultad en base a características del propio juego.** A la hora de construir el juego en sí, hay que determinar el valor de ciertas características del juego, como la gravedad de los elementos que caen, la velocidad a la que se mueve el usuario, el tipo de elemento que se crea cuando se hace aparecer un nuevo ingrediente, el tamaño de las plataformas, etc. Según este enfoque, la dificultad se definiría como un estado, un conjunto con valores que tienen que tomar esas variables para esa dificultad dada. Así, podríamos definir tantos niveles de dificultad como conjuntos tuviésemos, lo que aportaría más riqueza al juego que un mero sistema de categorías.

Finalmente se optó por adoptar el segundo enfoque y se determinó que las variables a tener cuenta para determinar la dificultad serían las siguientes.

- **Gravedad con la que son atraídos los ingredientes.** A mayor gravedad, más rápido caerán y más difícil será llegar hasta un ingrediente.
- **Proporción de ingredientes no válidos frente a ingredientes válidos.** Al mismo tiempo que caen ingredientes de la receta, estarán cayendo ingredientes que no forman parte de la misma, que llevarán asociada una penalización en caso de ser recogidos. El número de distintos ingredientes no válidos se calcula dinámicamente cuando se accede a los datos de una receta, momento en que se establece la proporción de ingredientes que no serán válidos. Así, si una receta

tiene cuatro ingredientes y la proporción de ingredientes no válidos está establecida en $\frac{1}{2}$ (50%), entonces habrá dos ingredientes que no serán válidos. Cuanto mayor sea esa proporción, más difícil resultará el juego.

- **Probabilidad de aparición de ingredientes válidos.** Cada vez que un ingrediente entra en contacto con una plataforma o con el usuario, se crea otro en un punto aleatorio a lo largo del ancho de la ventana del juego. Cuando tiene lugar esta creación, se genera un número aleatorio. Si este es menor o igual de un valor establecido como umbral, se genera un ingrediente válido, mientras que, si es mayor, se genera un ingrediente no válido. Cuando mayor sea ese umbral, mayor será la probabilidad de que el siguiente ingrediente que se cree sea válido y, por tanto, más sencillo resultará el juego.

De esta manera, la dificultad estará determinada por la tupla (*PRB*, *PRP*, *G*), siendo *G* el valor de la gravedad, *PRP* la proporción de ingredientes no válidos frente a válidos y *PRB* el valor de la probabilidad de aparición de un ingrediente válido⁸. Gracias a este enfoque, podemos tener tantas tuplas como queramos, siendo mayor el conjunto de posibles tuplas cuantos más valores se acepten para esas variables. Para hacer el problema más sencillo, se ha definido que cada una de ellas puede tomar tres valores, correspondientes al nivel fácil, normal y difícil respectivamente de cada variable (Tabla 9).

Tabla 9. Valores aceptables para determinantes de dificultad

Variable\Dificultad	Difícil	Normal	Fácil
Gravedad.	200	125	50
Proporción de ingredientes no válidos respecto a ingredientes válidos.	1/1	1/2	1/3
Probabilidad de aparición de ingrediente válido.	0.5	0.75	0.9

Dado que tenemos tres variables, y cada una puede tomar a su vez tres valores distintos, esto nos deja con un conjunto de 27 tuplas, esto es, **27 niveles de dificultad**. La progresión de la dificultad se ha definido como un reloj binario. Siendo 0 el valor “Fácil” de una variable, 1 el valor “Normal” y 2 el valor “Difícil”, la dificultad progresa tal y como se ve en la Tabla 10. En la Figura 72 se ha modelado un diagrama de estados del juego centrado en la dificultad del mismo. Cada posible tupla se representa con un estado en el que se puede encontrar la dificultad del juego. Si los resultados afectivos recabados del usuario revelan que se está aburriendo o que no está prestando atención, se pasará a un estado de dificultad superior,

⁸ Estas abreviaturas (*G*, *PRB*, *PRP*) se utilizarán más adelante

mientras que, si el usuario se está estresando o se está poniendo nervioso, se pasará a un estado de dificultad inferior.

Tabla 10. Progresión de dificultad

Nivel de dificultad	Tupla (reloj binario)	Tupla (valores reales)
0	(0,0,0)	(0.9, 1/3, 50)
1	(0,0,1)	(0.9, 1/3, 125)
2	(0,0,2)	(0.9, 1/3, 200)
3	(0,1,0)	(0.9, 1/2, 50)
...
12	(1,1,0)	(0.75, 1/2, 50)
13	(1,1,1)	(0.75, 1/2, 125)
...
26	(2,2,2)	(0.5, 1/1, 200)

A la hora de implementar la variación de la dificultad, se optó por codificarla como un entero que podría tomar valores entre 0 y 26, haciéndose uso de la operación módulo para calcular el valor (en términos del reloj binario) que tiene que tomar cada variable, que a su vez se traduce al valor real de las mismas. En las siguientes ecuaciones podemos ver las operaciones realizadas para calcular el valor de reloj “binario”, siendo *mod* la operación modulo (devuelve el resto de operar dos números) y siendo la división entera.

$$G = \text{dificultad} \text{ mod } 3$$

$$\text{PRP} = \left(\frac{\text{dificultad}}{3} \right) \text{ mod } 3$$

$$\text{PRB} = \left(\frac{\text{dificultad}}{9} \right) \text{ mod } 3$$

En lo que respecta a la selección de información afectiva que se utilizará para decidir cómo variar la dificultad, se tomó la decisión de utilizar datos afectivos extraídos a partir de expresiones fáciles, a través de la voz del usuario y en base a los errores que cometiese a la hora de controlar el juego (el personaje del juego es controlado con las flechas del teclado o con las teclas AWS/D), así como la velocidad de pulsación de las teclas para llevar a cabo dicho control.

4.4.3 Revisión

Durante esta revisión se analizaron las consideraciones y propuestas realizadas respecto a la forma de modelar e implementar la dificultad, que fueron muy bien acogidas. Se decidió de forma definitiva que se utilizaría información afectiva extraída del rostro, de la voz y de la tasa de fallos y velocidad de pulsación de las teclas. Para la detección de emociones basada en la voz se propone la posibilidad de implementar una fase de lectura y/o dictado tras la fase de juego, en la cual los usuarios tendrían que leer ciertas frases, en relación con la receta que corresponda, para poder generar una cantidad de información acústica tal que el servicio de detección que sea no tenga problemas para producir un resultado.

Tras un análisis de las tecnologías exploradas durante el sprint 0, se decidió adoptar Afectiva como detector de emociones basado en expresiones faciales y Beyond Verbal como detector de emociones basado en la voz. Respecto a la pulsación de teclas, se propone implementar un controlador del teclado que lleve un registro de las teclas pulsadas, compruebe cuales de estas son válidas (tienen una función asociada en el juego) y mida la velocidad de pulsación de las mismas.

Se revisaron también las modificaciones realizadas al prototipo (Figura 38, Anexo A.2) y se acordó que (a) los ingredientes que tendrían que recoger los usuarios habrían de indicarse otra manera, puesto que usando la propia imagen del ingrediente se revela qué ingrediente es y que (b) las cruces habrían de cambiarse por alguna imagen de connotaciones menos negativas.

4.4.4 Retrospectiva

Aunque las fechas eran similares respecto al sprint anterior, la carga de trabajo empezaba a dejarse notar. No obstante, esto no impidió seguir un desarrollo del trabajo cercano al ideal. Las estimaciones mejoraron también respecto al anterior sprint, siendo en ese caso la sobrestimación de tres horas y habiéndose reducido a una en este (Figura 20).

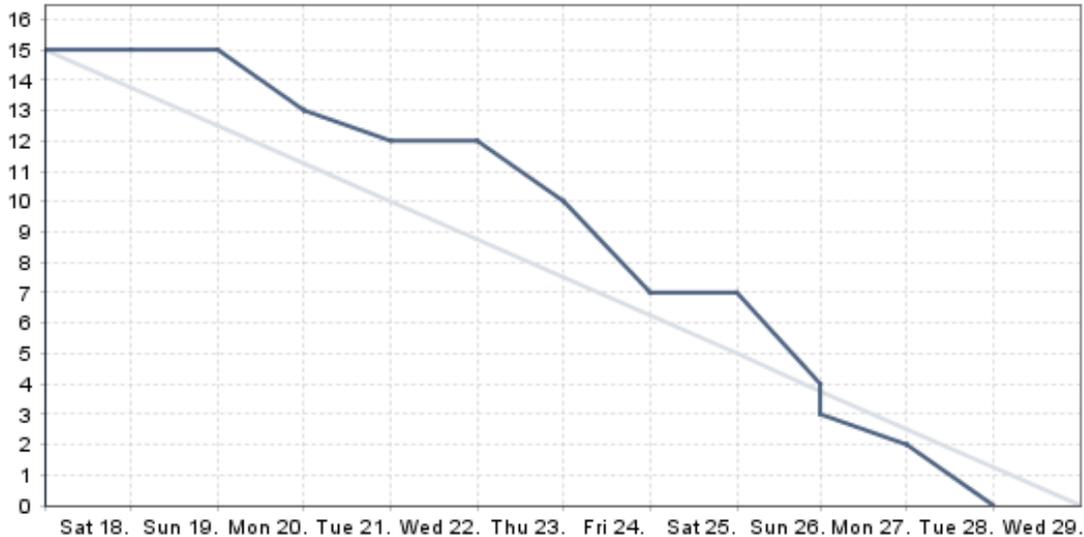


Figura 20. Sprint 2. Gráfica burn-down

4.5 Sprint 3. Integración de tecnología afectiva en el prototipo

4.5.1 Planificación

Objetivo. Integrar en el prototipo las distintas tecnologías de detección de emociones elegidas en el sprint anterior, así como el modelado de la estructura que soportará la tecnología afectiva dentro del mismo.

Duración. 24 días (03/04/2017 – 26/04/2017)

Durante esta reunión se acordó que se seguiría con lo comentado anteriormente, de manera que el orden que presentaban las tareas en el sprint anterior se mantiene. Dado que el PB no ha cambiado (salvo que ya no contiene la historia que se realizó en el sprint anterior), no se muestra la tabla asociada. Para este sprint decide abordarse la realización de la historia S4, quedando ésta dividida en las tareas que se aprecian en la Tabla 11.

4.5.2 Desarrollo

La labor de documentación que se realizó estaba pensada para avanzar el desarrollo del presente documento, así como para dejar constancia del funcionamiento de los servicios de

detección de emociones, con miras también a facilitar la comunicación con terceras personas, que solo necesitarían leer esta documentación para entender, al menos de forma superficial, cómo funcionan dichos servicios.

Tabla 11. Sprint 3. Sprint Backlog

Historia	Tarea	Horas	+/-
Integración de prototipo con tecnología de detección afectiva.	Documentación sobre el funcionamiento de las tecnologías de detección emociones basadas en expresiones faciales.	5	0
	Documentación sobre el funcionamiento de las tecnologías de detección emociones basadas en la voz.	5	0
	Modelado de la estructura que soportará el conjunto de información y funcionalidad afectiva.	3	0
	Análisis de tecnología de grabación de audio en el prototipo.	4	0
	Integración de tecnología de grabación de audio en el prototipo.	3	0
	Integración de tecnología de Beyond Verbal en el prototipo.	2	0
	Integración de tecnología de Afectiva en el prototipo.	2	0
	Implementación de módulo de control de pulsación de teclas	3	0
Total		27	0

Respecto al modelado de la estructura que soportaría la información y funcionalidad afectiva, se decidió que la información afectiva se almacenaría con los datos de la partida que se ha jugado. En el caso de Afectiva, se guardarán todos los datos, mientras que en el caso de Beyond Verbal se guardarán solo los valores medios de Humor, Valencia y Excitación, así como las emociones primarias de los grupos Grupo 11 y Composite.

Aun cuando no se lleguen a utilizar todos los datos para decidir cómo debe variar la dificultad, puede que en un futuro se proponga elaborar un módulo estadístico que permita analizar el progreso de los usuarios, en cuyo caso necesitaríamos tener guardada la información afectiva producida durante cada partida.

La integración de las tecnologías de detección, tanto de Afectiva como de Beyond Verbal, resultó bastante sencilla, pues ofrecían ejemplos de código en sus respectivas páginas web en los que estaban implementadas todas las funciones necesarias para llevar a cabo las solicitudes de detección de emociones. Esta facilidad para realizar la integración permitió dedicar más tiempo a la lectura de la documentación asociada a dichos ejemplos, lo que a su

vez generó un conocimiento más sólido acerca de cómo funcionaban los sistemas. Por último, se implementó el controlador del teclado que se preparó para reaccionar ante los eventos de pulsación y liberación de teclas. Cuando tienen lugar estos eventos, se calcula el tiempo que ha habido entre la pulsación de la tecla y su liberación y se comprueba si la tecla pulsada era válida (se entiende por tecla válida aquella que tiene una función asociada en el juego).

Respecto a la detección de emociones basada en la voz, el servicio de Beyond Verbal presentó un inconveniente: al contrario que Afectiva, cuyo servicio puede trabajar en directo usando imágenes capturadas directamente de una cámara web, el servicio de Beyond Verbal solo analiza flujos de audio que se envíen en forma de archivo. Debido a esto, hubo que recabar información acerca de cómo capturar audio desde el navegador para generar un fichero. Tras revisar varias opciones, se encontró un proyecto disponible para uso que permite grabar audio desde el navegador y generar un fichero para su descarga (Audio, Web). Se hicieron ligeras modificaciones en el mismo para que a la generación del fichero, en lugar de descargarse al ordenador, se enviase a Beyond Verbal. A pesar de que existía la posibilidad de que Beyond Verbal rechazase esos datos por no ser un fichero del formato adecuado (WAV o PCM), la API aceptó estos ficheros “puros” y respondió correctamente a las peticiones realizadas.

4.5.3 Revisión

Durante la revisión de este incremento, el equipo enseñó al PO la funcionalidad añadida al prototipo, siendo ésta muy bien recibida. Dado que ya se disponía de las tecnologías integradas en el prototipo y listas para usarse, se decidió que en el siguiente sprint se terminaría de implementar la afectividad del prototipo. Se abrió el debate sobre si el uso de la aplicación podía tener asociadas implicaciones morales, debido a que se va a monitorizar un aspecto íntimo del usuario del prototipo. En base a esto, y siguiendo el ejemplo de investigaciones previas realizadas en el campo de la Computación Afectiva (Picard, 2009), se decidió que el uso de los detectores de emociones sería opcional y la activación y desactivación de los mismos quedaría en manos del usuario. Así mismo, se acordó que debería reflejarse la información legal relativa al uso de las tecnologías de detección.

4.5.4 Retrospectiva

Debido a que la ejecución del sprint se solapó en parte con las vacaciones de Semana Santa, así como con entregas y trabajos de otras asignaturas, fue necesario ampliar la duración del sprint una semana más, lo que dio lugar a la gráfica que podemos ver en la Figura 21. El sprint inicialmente acababa el día 19, día en el cual acaba la línea que señala el progreso ideal del trabajo, pero dado que se amplió durante el propio sprint la estimación no se recalculó.

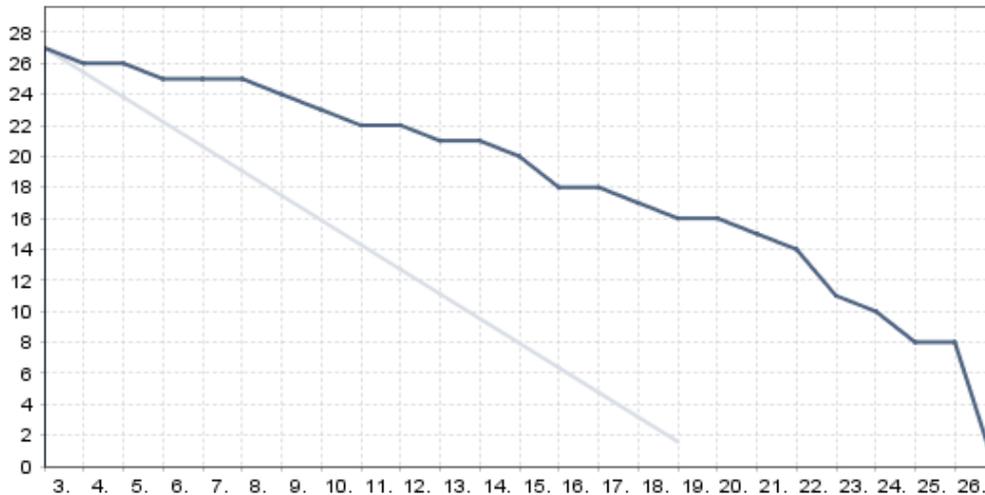


Figura 21. Sprint 3. Gráfica burn-down

En lo que se refiere a estimaciones, una vez más éstas resultaron bastante precisas. Las únicas tareas que estaban algo sobrestimadas eran las de integración de tecnologías, pero dado que se aprovechó el tiempo ganado para recabar documentación sobre las mismas, al final la cantidad de tiempo trabajado coincidió con el estimado.

4.6 Sprint 4. Implementación de afectación en el prototipo

4.6.1 Planificación

Objetivo. Utilizar las tecnologías integradas durante el sprint anterior para dotar de afectividad al prototipo. Esto implica implementar la lógica de registro y análisis de las emociones recogidas, así como la toma de decisiones.

Duración. 24 días (03/05/2017 – 26/05/2017)

Durante este sprint se decidió extraer las dos últimas historias contenidas en el PB (Tabla 12) y llevarlas al SB. Se hizo hincapié en los detalles reseñados durante la revisión del sprint anterior (reflejar información legal, que los usuarios puedan apagar y encender la detección) y se acordó también que el aspecto estético de la interfaz debía modificarse para este sprint.

Tabla 12. Sprint 4. Product Backlog

ID	Nombre	Descripción	SP
S5	Desarrollo de lógica afectiva.	Como cliente, quiero que el juego sea capaz de deducir un estado emocional en base a los resultados de los servicios de detección.	8
S6	Implementación de prototipo afectivo.	Como cliente, quiero que el juego varíe su comportamiento y dificultad según las emociones del jugador.	13
Total			21

En la Tabla 13 podemos ver el desglose de las tareas cuya implementación se afrontó en este sprint. Dado que la tarea acerca de los niveles de dictado implicaba llevar a cabo modificaciones en la interfaz de la aplicación, las modificaciones relacionadas con el aspecto estético quedan también enmarcadas bajo esa tarea.

4.6.2 Desarrollo

Como primeros pasos del sprint, se afrontaron las tareas relativas al envío de datos afectivos al servidor, a su escritura en la base de datos y a la recuperación de los datos de la misma.

Tabla 13. Sprint 4. Sprint backlog

Historia	Tarea	Horas	+/-
Desarrollo de lógica afectiva	Envío de los datos afectivos al servidor.	3	0
	Registro de los datos afectivos en la base de datos.	3	0
	Recuperación de los datos afectivos desde la base de datos.	3	0
	Implementación de la lógica afectiva.	8	-1
Implementación de prototipo afectivo	Implementación de niveles de dictado	7	0
	Implementación de variabilidad en los niveles.	6	0
Total		30	-1

Dado que los datos afectivos están asociados a la partida durante la cual se obtuvieron estos, la realización de esas tareas implica modificar la clase encargada de almacenar los resultados. Del mismo modo, llevar a cabo la implementación definitiva de la dificultad implica modificar la clase que almacena los datos de los usuarios. Además, recordar la decisión del usuario sobre qué tipo de detección acepta que se realice durante el juego implica

hacer persistente dicha decisión, lo que supone otra modificación más a la clase anterior. Estos cambios a su vez han de ser extendidos a la base de datos.

En la Figura 68 podemos ver el diagrama de clases de la aplicación, sobre cuyas clases quedan soportados los datos de los usuarios y de las partidas que han jugado. Se dispone de una clase *Juego* que contiene la información sobre los niveles y los usuarios, delegando el control de las partidas sobre una clase “cuidadora”⁹, *GestorPartidas*. Esta clase mantiene una lista de objetos, conteniendo cada objeto el id del usuario cuyas partidas está guardando y una lista de dichas partidas. En la base de datos se contiene una colección de objetos *Usuario* y otra colección que contiene los datos de *GestorPartidas*: objetos con el id del usuario correspondiente (lo que equivaldría a una clave foránea en una base de datos relacional) y las partidas asociados a los mismos. Se crea una nueva partida cada vez que el usuario pasa por el nivel 1 del juego, teniendo por identificador un valor obtenido a partir del momento temporal de creación.

Al mismo tiempo que se hicieron las modificaciones pertinentes en la parte del código que escribía información en la base de datos, se modificó el código que vuelca la base de datos en nuestro servidor cuando éste se inicia.

Respecto a las frases, se tomó como base el ejemplo de reconocedor de voz que se utilizó en los primeros sprints. Haciendo uso de la interfaz de Web Speech API para el reconocimiento de voz (SpeechRecognition, Web), se detectará la voz del usuario y se analizará si esta coincide con aquello que éste tuviera que leer. El comportamiento del reconocedor de voz quedó encapsulado en una clase, *_SpeechRecognition*, en la cual se controla el inicio, detención y reinicio de la detección, el procesamiento del texto reconocido en la voz, el establecimiento de las gramáticas (palabras que el reconocedor tiene que reconocer), etc.

Para no ser muy exigentes con los usuarios, y puesto que el objetivo en este prototipo no es garantizar una pronunciación perfecta, sino más bien hacerlos hablar con cierta fluidez durante un tiempo, se determinó que se consideraría que una frase había sido leída correctamente cuando la tasa de coincidencia entre la frase reconocida y la frase original fuese de, al menos, el 60%. En otras palabras, si el 60% de las palabras de la frase a leer se encuentran en la frase reconocida, se aceptará dicha pronunciación como válida. Las frases se almacenarán en el servidor, junto con los datos de una receta, y serán traídos al cliente junto con los datos usados para construir el juego. Cuando se traen las frases de un nuevo nivel, se refrescan las gramáticas del reconocedor, para prepararlo para el ejercicio de dictado. Se estudiaron distintas opciones acerca de cuándo hacer este ejercicio de habla y se optó por dejarlo para después del juego en sí. Cuando el jugador terminase la recolección de

⁹ La estructura Juego – GestorPartidas – Partida se corresponde con el patrón de diseño “Memento”. Esto procede de una implementación antigua que se realizó pensando en la posibilidad futura de que los usuarios pudieran repetir niveles para mejorar sus tiempos, siendo la partida el “Memento” y el GestorPartidas la clase “Catetaker”.

ingredientes, se mostraría una lista de los ingredientes que ha recogido —mostrando cada icono asociado a su nombre—, así como las instrucciones de la siguiente fase.

La fase de dictado consiste en presentar al usuario la lista de frases que tiene que leer y un botón para activar el reconocedor. Cuando el usuario esté listo, podrá hacer clic en el botón, que cambiará su color y su texto para dejar claro que se ha activado el reconocedor, y recitar la frase que corresponda. Si la pronunciación ha sido correcta, se pasará a la siguiente frase automáticamente, mientras que, si no lo ha sido, se indicará esto al usuario y se le otorgará otra oportunidad. Si la lectura de una frase se hace de forma incorrecta otras dos veces, la aplicación pasará a la siguiente frase, así hasta llegar a la última. Al empezar la primera frase, se pondrá en marcha el grabador de audio, que no se parará hasta acabar todas las frases. Cuando esto pase, se generará el archivo de audio capturado durante la pronunciación y se enviará a Beyond Verbal a través de su API. Dado que Beyond Verbal es con diferencia la tecnología más lenta, esto introducirá una latencia que retrasará la continuación al siguiente nivel (si permitiésemos al usuario continuar cuando aún no se han procesado los datos de Beyond Verbal, el juego no podría considerar la información afectiva y no se modificaría correctamente la dificultad). Para suavizar esa espera, se ha introducido una imagen animada tras la fase de dictado que se reproduce mientras se muestran unas frases para indicar que la aplicación está procesando y trabajando. Cuando se tienen los resultados de Beyond Verbal, se muestran las puntuaciones que el usuario ha tenido durante esa partida y se envía la información afectiva al servidor, donde se analizará y se decidirá cómo habrá de variar la dificultad en el siguiente nivel. Llegados a este punto, el usuario puede seguir jugando o abandonar la partida, quedando así establecido el flujo de navegación de la aplicación (Figura 69).

Durante las modificaciones gráficas que implicaron realizar la tarea del dictado se añadieron los apartados de “Ajustes”, donde los usuarios podrán activar y desactivar los diversos sistemas de detección, y “Aviso legal”, donde se explica en qué consiste la aplicación, las tecnologías que usa y se ofrecen a las páginas sobre privacidad de cada servicio. En lo que al juego respecta, las equis que se utilizaban para indicar las vidas y/o comodines que quedaban se sustituyeron por corazones, y en la lista de ingredientes que se coloca sobre la ventana del juego, los iconos de los ingredientes fueron sustituidos por el nombre de los mismos en inglés, ya que de la otra manera se revelaba qué ingredientes había que coger sin que hiciera falta conocer su nombre. Se introdujo también una pequeña sección de información antes de cada receta, a modo de introducción, y en la pantalla inicial del juego se añadieron algunas capturas del juego para que los usuarios pudieran intuir en qué consiste la aplicación. Haciendo uso de la herramienta de color de (Material, Web) se seleccionó un conjunto de colores legible para la aplicación, que se bautizó de forma definitiva como **emoCook**.

Por último, se llevó a cabo la implementación final de la dificultad y la lógica afectiva. En lo que respecta a la lógica afectiva, se decidió que, por el momento, *para determinar la dificultad que ha de tener un nivel solo se utilizará la información afectiva generada durante*

el nivel anterior. Se estudió también la posibilidad de considerar datos históricos, y tener en cuenta, no solo la información afectiva recabada durante el nivel anterior sino la información afectiva producida por el usuario en partidas previas del nivel que viene a continuación, pero se prefirió dejar de lado esta opción (aunque se guarda para trabajo futuro). La lógica afectiva, que se ejecuta cuando llegan los resultados afectivos —al completarse un nivel de juego y dictado en el cliente—, está constituida por un bloque de código que analiza diversos indicadores de cada tipo de información afectiva (cara, voz y teclas) para decidir cómo ha de variar la dificultad. Dicho bloque analiza los indicadores que se detallan a continuación para determinar si durante el nivel anterior se han detectado emociones negativas, como frustración y estrés, en cuya la dificultad disminuirá. Si, por el contrario, no se ha detectado ninguna emoción negativa, la dificultad aumentará.

- **Afectiva.** Tras una serie de pruebas experimentales, se concluyó que las partes de la cara que más se implicaban durante un juego eran la boca, la frente y la nariz, de manera que los indicadores (Figura 65) que se estudian son los que indican que el **ceño** está fruncido, que el **labio superior** está levantado, que una de las **comisuras** de la boca está levantada y que la **nariz** está arrugada. Dado que estos indicadores son los que están relacionados con sentimientos de **disgusto, tristeza e ira**, se estudia también el valor de estas emociones, así como la **valencia** de la emoción detectada.
- **Pulsación de teclas.** En este caso, se analiza el número de **errores** cometidos (en un caso normal, estos deben de ser ínfimos puesto que solo existen seis válidas que están muy juntas en el teclado) y el número de veces que se ha detectado una **excesiva pulsación** en cada una de las teclas válidas.
- **Beyond Verbal.** En este caso se atenderá sobre todo a la valencia y a la excitación detectadas, revisándose también los estados de ánimo (Grupo 11 y Composite) para ver si alguno contiene ciertas palabras clave como “ansiedad” o “aburrimiento”.

Para decidir cómo variar la dificultad, la lógica afectiva estudia los indicadores de cada grupo de información (cara, voz y teclas). Si el valor de un indicador supera el de un umbral establecido, se contabiliza que ha habido un “hallazgo” de estrés. Si el número de hallazgos supera un determinado valor, se determina que, en base a ese detector, hay que disminuir la dificultad. Si por el contrario no se alcanza ese umbral, el veredicto del detector es que hay que incrementar la dificultad. Tras analizar la información de los tres detectores, se comparan los veredictos para decidir cuánto ha de variar ésta. Para hacer más notable el incremento o disminución de la dificultad, se ha tomó la decisión de variarla en varios niveles cada vez, debido a que, de hacerse de uno en uno, habría casos en los que no se notaría el cambio (cuando solo variase el valor de la gravedad) mientras que habría otros en los que se notaría muchísimo (cuando cambiase la probabilidad de aparición de ingredientes válidos). Así, la dificultad disminuirá en tres unidades cuando los detectores coincidan en que hay que bajarla, en dos cuando solo decidan bajarla algunos de los detectores y se incrementará en dos en

cualquier otro caso. Esta última cláusula se ha dejado así para que incluso si el usuario ha deshabilitado la detección de emociones, esta vaya incrementándose poco a poco, disminuyéndose entonces solo cuando el usuario pierda todas sus vidas en un nivel, de manera que incluso los usuarios sin el hardware necesario pudieran experimentar esa variabilidad.

Casi al final de este sprint, surgió un problema que impidió utilizar la información de Beyond Verbal en el bloque de la lógica afectiva: durante la última actualización de la API, el **servicio de conversión** de la misma, que permitía enviar pistas de audio con una frecuencia y un número de canales distintos a los indicados en la web¹⁰, fue **eliminado** del tipo de licencia que se estaba utilizando, pasando a formar parte de uno de los privilegios que ofrecía la siguiente licencia. Aunque la tecnología estaba ya integrada en la aplicación, no podían solicitarse análisis a la misma, razón por la cual esta no se utiliza durante la lógica afectiva. Dado el estado del proyecto, se prepararon diagramas de componentes (Figura 73) y de despliegue (Figura 74) para mostrar al PO cómo había quedado la aplicación.

4.6.3 Revisión

Durante esta reunión, se empezó revisando las mejoras gráficas que se había realizado en la aplicación (Anexo A.3), que tuvieron una muy buena acogida, dado lo descuidado de este aspecto en sprints anteriores. Aunque sigue habiendo detalles mejorables, sobre todo al tratarse de niños que se distraen con facilidad, se decidió que por el momento no se iban a dedicar más esfuerzos a modificarla. Respecto a las frases, se comentó la posibilidad de resaltar el botón de alguna manera para recordar al usuario que había que pulsarlo antes de hablar. También se consideró adecuado añadir las reglas y los controles del juego en la propia página, disimulados de alguna forma que no sobrecargase al usuario. El PO se mostró conforme con las decisiones tomadas respecto a la variación de la dificultad y se consideró que no merecía la pena en este punto invertir tiempo en encontrar otro sistema sustituto basado en la voz o en implementar un conversor propio, por lo que se decidió seguir adelante sin el reconocimiento de emociones basado en la voz.

Llegados a este punto, se consideró que el siguiente paso era realizar una evaluación con usuarios. No obstante, dadas las fechas de este momento, reunir a todos los usuarios podía ser un problema: si se hacía la evaluación en ese momento, los usuarios potenciales estaban aún terminando las clases, mientras que, si se esperaba algunas semanas más, el problema podía venir dado por actividades extraescolares y/o viajes de vacaciones. Esto, unido a las propias responsabilidades como estudiante del autor, dificultaría la realización de

¹⁰ Beyond Verbal indica en su página que el audio enviado ha de ser un archivo WAV o PCM, con una frecuencia de 8000Hz y un solo canal.

un sprint como tal, además de que las correcciones que el proyecto necesitaba eran mínimas y no daban como para formar una historia.

Debido a este cúmulo de circunstancias, se decidió que este sería el último sprint realizado, quedando la aplicación desplegada en (emoCook, Web). Tanto en este sprint como los anteriores, la aplicación a una batería de pruebas que permitía comprobar la robustez y corrección de la aplicación. Estas pruebas pueden encontrarse en el CD adjunto con este trabajo.

4.6.4 Retrospectiva

El desarrollo del trabajo de este sprint puede observarse en la Figura 22. Este sprint ha resultado muy estimulante a nivel personal debido a la gran evolución que ha sufrido el prototipo. Aun cuando la nueva funcionalidad añadida ha sido bastante reducida, la mera mejora de los gráficos, así como la asignación del nombre, ha dado un aspecto completamente distinto al prototipo. Aunque las horas de ciertas tareas estaban ligeramente sobrestimadas, a su vez esto permitió la corrección de los aspectos gráficos, de manera que la única estimación errónea se debe al problema que surgió con Beyond Verbal, lo que supuso una menor inversión de tiempo en la implementación de la lógica afectiva.

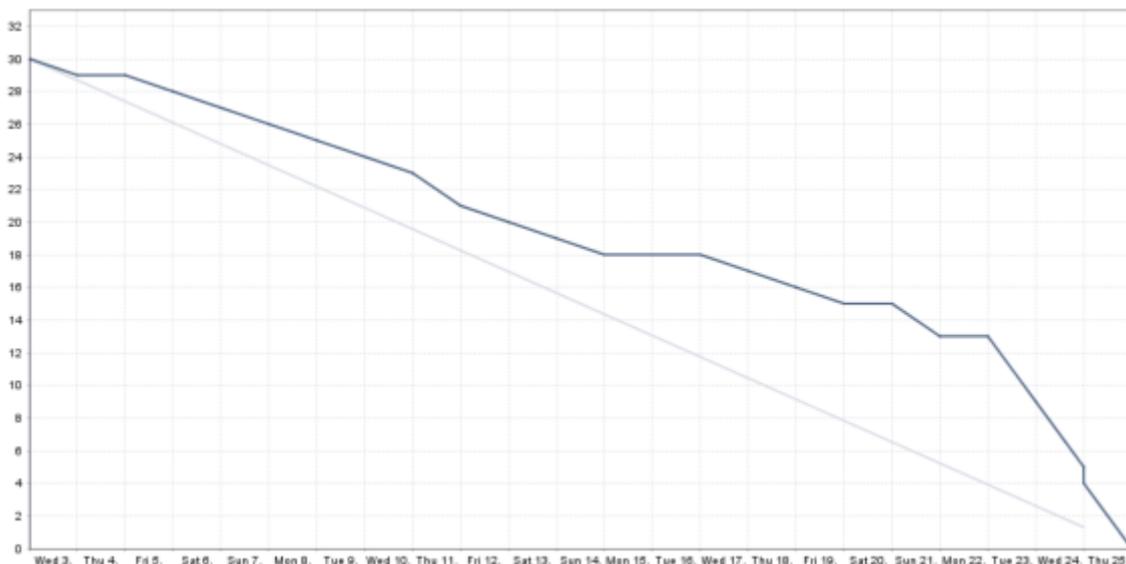


Figura 22. Sprint 4. Gráfica burn-down¹¹

¹¹ La estimación termina el día 24 debido a un error inicial al establecer las fechas del sprint en Kunagi, puesto que se planificó que el sprint terminase el día 25.

Ha resultado gratificante ver cómo la planificación realizada en el anteproyecto se ha seguido casi exactamente como se pensó. En la Tabla 14 podemos ver una comparativa entre los sprints que se estimaron cuando se redactó el anteproyecto de este TFG y los sprints que se han realizado realmente. La diferencia más notable es que se han hecho dos sprints menos, dado que dos sprints se fundieron en uno solo y que la evaluación no se va a realizar bajo el marco Scrum. No obstante, si así se hiciera, vemos que es muy probable que se cumpliera el plazo estimado meses atrás.

Tabla 14. Comparativa de sprints: estimado/real

Sprints estimados	Fecha de fin	Sprints realizados	Fecha de fin
Estudio base sobre Computación Afectiva.	26/02/2017	Estudio base sobre Computación Afectiva.	26/02/2017
Preparación del primer prototipo.	14/03/2017	Preparación del primer prototipo.	14/03/2017
Diseño de distintos niveles de dificultad.	30/03/2017	Diseño de distintos niveles de dificultad.	29/03/2017
Integración de tecnología afectiva en el prototipo.	19/04/2017	Integración de tecnología afectiva en el prototipo.	26/04/2017
Implementación de lógica de decisión.	09/05/2017	Implementación de afección en el prototipo	26/05/2017
Implementación de afección en el prototipo.	30/05/2017		
Evaluación del prototipo.	12/06/2017	<i>Evaluación del prototipo</i>	<i>Aun en plazo.</i>

4.7 Conclusiones

En este capítulo se ha detallado el proceso de desarrollo de una aplicación haciendo uso de los marcos de trabajo Scrum y Kanban. Estos fueron adaptados al contexto del proyecto, aprovechando su flexibilidad para satisfacer las necesidades del trabajo, cumpliéndose la estimación temporal realizada meses atrás.

A lo largo de estos sprints se ha trabajado con tecnologías de detección de emociones, uno de los campos de la Computación Afectiva que más atención ha estado recibiendo de un tiempo a esta parte. Durante el desarrollo hubo que afrontar diversos problemas, tanto intrínsecos de la tecnología como derivados de la naturaleza de los productos (son productos comerciales sujetos a cambios en el mercado, en la política de la empresa que lo desarrolla, etc.), pero tanto su análisis como su resolución han aportado un gran valor tanto a la aplicación como al autor del proyecto a título personal.

En el siguiente capítulo se realizará un análisis detallado del producto final. Dado que la apariencia del producto final coincide con la apariencia del producto al finalizar el último sprint, la mayor parte de las imágenes se encuentran en el Anexo A, apartado 3.

CAPÍTULO 5. DESCRIPCIÓN DE LA APLICACIÓN

En este capítulo se hará una descripción completa del prototipo desarrollado. Se analizará el hardware necesario para utilizarlo al completo, el contenido de sus distintas pantallas, los distintos mecanismos de detección del mismo, etc. Las imágenes referidas en este apartado pueden encontrarse en el Anexo A, apartado A.3.

5.1 Introducción

La propuesta inicial del presente trabajo consistía en estudiar cómo podía modificarse el comportamiento de una aplicación en tiempo real para que ésta se adaptase al estado emocional de los usuarios de la misma. Para ello, se abordaría el desarrollo de un prototipo que integrase tecnologías del campo de la Computación Afectiva capaces de detectar las emociones de una persona. Haciendo uso de estas tecnologías, se estudiaría el estado afectivo de un usuario para adaptar el funcionamiento del prototipo y así mejorar la experiencia de usuario al interactuar con el mismo.

Como primera propuesta, se decidió que dicho prototipo consistiría en un juego, de manera que la variación en función de las emociones tendría como objetivo ofrecer una buena experiencia de juego a los jugadores, manteniendo a los jugadores “comprometidos” (“engaged”) e inmersos en el juego.

Para que el alcance del prototipo llegase a un mayor número de plataformas, y con vistas a aprovechar un proyecto previo que podría constituir la base del prototipo, se decidió construir un juego para navegador, de manera que este fuera mucho más accesible que si se implementase una aplicación nativa para una plataforma concreta. Para enriquecer dicho juego, se optó por darle un objetivo que fuera más allá del mero entretenimiento, decidiéndose que se construiría una aplicación que pudiera ser usada con un objetivo didáctico en este caso. Así, se ha construido un juego que gira en torno al aprendizaje de vocabulario, estando éste relacionado con la cocina y la comida, concretamente la elaboración de recetas y la recolección de ingredientes.

emoCook, nombre con el que se ha bautizado el prototipo, hace uso de las imágenes del rostro del usuario, capturadas a través de una cámara conectada al dispositivo en que se use, y la voz del usuario, capturada a través de un micrófono conectado al dispositivo en que se use, para determinar su estado afectivo. De forma adicional, se lleva un registro de las teclas que pulsa el usuario al controlar la aplicación con vistas a comprobar la velocidad a la que las pulsa y los errores que comete (entendiendo por error pulsar una tecla que no lleva asociada ninguna función), usándose esta información adicional como un indicador afectivo más con el que detectar estrés.

En los siguientes sub-apartados, se detallarán los requisitos óptimos para usar la aplicación con toda su funcionalidad, así como los elementos de la misma, los componentes que la conforman, la estructura de los niveles del juego, etc.

5.2 Requisitos y recomendaciones de uso

Dado que el elemento más pesado para el dispositivo es el juego implementado en Phaser, y este es relativamente sencillo (la propia librería Phaser está pensada para ser ligera), no se necesita un equipo de gran poder de computación para ejecutar la aplicación.

- 1 GB RAM
- Procesador Dual Core 2 GHz
- Navegador Google Chrome
- Conexión a Internet
- Cámara web de tres megapíxeles (mínimo)
- Micrófono

Aunque el prototipo se encuentra desplegado y accesible desde (emoCook, Web), no todos los navegadores lo soportan igual. La librería Phaser ha funcionado sin problemas en los navegadores en lo que ha sido probada la aplicación (Internet Explorer, Edge, Mozilla Firefox, Google Chrome, Opera y Vivaldi), creando el juego y permitiendo jugar sin complicaciones. No obstante, al llegar a la parte de pronunciación, solo Chrome y Opera han sido capaces de cargar las frases —aunque el reconocedor de voz no detectaba nada en Opera—, mientras que en el resto de navegadores la aplicación se quedaba en blanco. Dado que el objetivo era construir un prototipo, se ha dejado como trabajo futuro la corrección de este problema.

Al margen de eso, existen algunas consideraciones para tener en cuenta respecto al entorno en el que se pruebe la aplicación para garantizar el correcto funcionamiento de la misma. Cabe mencionar que, aunque la aplicación se ha desarrollado pensando en que fuera accesible al mayor número de plataformas posibles, el prototipo se ha desarrollado pensando en un entorno de escritorio, no estando preparada para usarse en dispositivos portátiles. Algunas de las siguientes consideraciones se hacen teniendo dicho entorno de escritorio en mente.

- **Iluminación**
 - Dado que la cámara va a capturar la cara del usuario, ha de haber una buena iluminación de la misma para que el rostro sea detectado correctamente. Esto implica colocar la cámara de espaldas a los focos de luz y apuntando a la cara del usuario, de manera que el foco ilumine la cara del usuario, debiendo evitarse los focos de luz que iluminan con mucha fuerza desde arriba y que pueden provocar sombras en la cara que entorpezcan la detección.
- **Distribución y entorno**
 - Para ofrecer la mejor imagen del rostro posible, es preferible que la **cámara** se coloque frente al usuario centrada en su cara, evitando a ser

posible ángulos desde arriba, desde abajo o desde los lados. Un buen punto para colocar la cámara (asumiendo que no es una cámara integrada) es a unos 30-35 centímetros de la cara, a la altura de la barbilla. Si la cámara es integrada y no se puede cambiar su altura, habría de utilizarse una silla lo suficientemente alta (o una mesa más baja) como para alcanzar la posición descrita anteriormente.

- La cara ha de mantenerse despejada. Si bien el detector de Afectiva es capaz de detectar la cara incluso cuando esta no está completamente visible (el usuario lleva gafas, el usuario apoya la cabeza sobre el puño o sobre la palma abierta, etc.), ciertos gestos que se realizan inconscientemente la mayoría de las veces (pasarse la mano por pelo, taparse la boca) obstaculizarían la vista de la cámara e impedirían detectar el rostro.
- Aunque las pruebas con el reconocedor de voz y Beyond Verbal se han realizado en entornos que solían tener ruido de fondo (como música u otras personas hablando a poca distancia), un ambiente **libre de ruidos** es lo más óptimo para realizar la fase de dictado. Se recomienda asimismo no hablar excesivamente cerca del micrófono (la distancia entre la cara y la cámara también es apropiada para esto).

5.3 Estructura de la aplicación

En la Figura 75 podemos ver el diagrama de navegación de la aplicación. Visible a cualquier usuario queda la información de la página, así como el aviso legal de la misma, estando también accesible el formulario para iniciar sesión y para registrarse. Una vez que el usuario inicia sesión, se le descubren nuevas opciones (para modificar su usuario, eliminarlo y cambiar sus preferencias respecto a la detección de emociones), así como información sobre el nivel en el que se encuentra. Estando en la pantalla dicha información se puede acceder al nivel de juego, cuya finalización lleva a la fase de dictado, que a su vez lleva a la sección de resultados.

En este apartado analizaremos los elementos de cada una de las pantallas, estando todas recogidas en el Anexo A, apartado 3. Las únicas diferencias entre la versión que reflejan esas imágenes y la versión final se reseñarán aquí.

5.3.1 Zona global

En la zona global nos encontramos aquellas pantallas accesibles a todos los usuarios, tanto los que ya se han registrado como aquellos que no. Lo primero que se encuentra un usuario al acceder a la aplicación (emoCook, Web) es una sección inicial con información sobre la aplicación (Figura 23)¹². Esta sección inicial contiene otras dos pantallas que el usuario puede ver si se desplaza hacia abajo en la página. En dichas pantallas se da una muestra sobre la temática del juego, mostrándose capturas de las distintas fases del juego (Figura 40), y se presenta su naturaleza afectiva, puesto que este es el aspecto más novedoso del prototipo (Figura 41).



Figura 23. Pantalla de bienvenida de prototipo (final)

Esta pantalla nos da acceso también a las siguientes secciones.

Aviso legal.

En esta pantalla se explica en que consiste el juego y se hace una advertencia de las tecnologías que se van a utilizar, haciéndose hincapié en que se usarán el rostro y la voz del usuario, indicándose también dónde puede encontrarse la política de privacidad de los servicios empleados (Figura 42). Este apartado se ha colocado con miras a ser leído por los tutores legales de los usuarios y/o los docentes al cargo de los mismos (el público objetivo de la aplicación se encuentra en edad de educación primaria), aunque se ha utilizado un lenguaje más cercano para que resulte entendible lo que se quiere transmitir para cualquier lector.

¹² En la versión final del juego se añadió un botón de ayuda en la esquina inferior izquierda, con el cual se accede a una ventana en la que se muestran los controles del juego y las reglas de la fase de dictado. Este botón se mantiene en su posición durante todo el flujo de navegación y es accesible en cualquier momento.

Inicio de sesión.

Haciendo clic en el botón “Acepto el reto” se descubre un formulario para iniciar sesión, en el cual se pide al usuario su nombre de usuario (o correo electrónico asociado a su nombre de usuario) y su contraseña (Figura 43). Si el usuario no está registrado, puede hacerlo a través del formulario que se carga al pulsar en el botón “Registrarse” (Figura 44). Si el registro se realiza correctamente, el usuario recibirá un correo electrónico en la cuenta que usó al registrarse, conteniendo un enlace para confirmar el registro. Una vez hecho esto, el usuario podrá iniciar sesión, accediendo así a la zona que solo queda visible a los usuarios.

Ayuda

Para evitar sobrecargar otras pantallas con información que el usuario puede necesitar más de una vez a lo largo del juego, se añadió un botón de “Ayuda” que permanece en la esquina inferior izquierda en todo momento, revelando una pantalla con los controles del juego y las reglas del dictado cuando es pulsado (Figura 24).

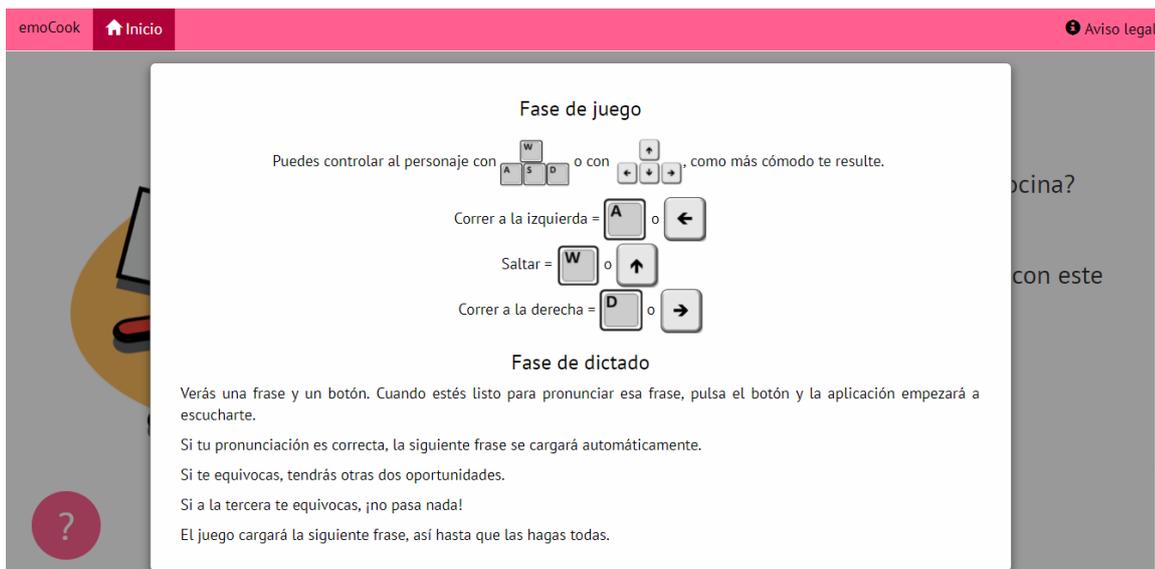


Figura 24. Pantalla de ayuda de prototipo

5.3.2 Zona usuarios

Cuando un usuario inicia sesión correctamente, es llevado a la pantalla de información de la receta (Figura 45) que corresponda: si es el primer inicio de sesión, se mostrará la información de la primera receta, mientras que, de no serlo, se mostraría la información de la receta que vaya a continuación de la última receta que se jugó. En las pantallas de la zona de usuarios es fácil apreciar que la pantalla está dividida en dos partes: en la parte derecha

se presenta la información, los formularios, el juego, las frases y los resultados, mientras que la parte izquierda supone un “panel de control”, que nos permite saber en qué nivel estamos, cuál es nuestro usuario y cuanto nos queda para terminar el juego, así como terminar la sesión en cualquier momento (Figura 25). A partir de este punto podemos acceder a las siguientes pantallas.

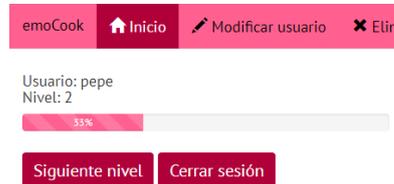


Figura 25. Panel de control

Juego

Pulsando el botón “Siguiete nivel” empieza la partida (Figura 46). El personaje del juego puede controlarse con las flechas del teclado (*Flecha izquierda* ←, moverse a la izquierda, *Flecha arriba* ↑, saltar, *Flecha derecha* →, moverse a la derecha) o con las teclas AWD (A, moverse a la izquierda, W saltar, D moverse a la derecha) y el objetivo es recoger los ingredientes necesarios para la receta cuya información se ha mostrado previamente. Justo encima del juego podemos ver los ingredientes que tenemos que recoger, mostrándose para cada ingrediente su nombre en inglés en lugar de su icono (cosa habitual en los videojuegos), forzando así al usuario a estudiar los ingredientes que caen del cielo para descubrir a qué ingrediente se refiere cada palabra. En la Figura 26 podemos ver un ejemplo de esta barra durante el primer nivel. En este caso, el usuario tiene que recoger manzanas y plátanos, al mismo tiempo que evita los ingredientes que no sean esos. Si el usuario no supiese cual es cual, tendría que ir probando, costándole cada error una vida. El número de corazones del lado derecho de la figura representa los errores que le quedan al usuario; si estos se acaban, se indicará esto al usuario y se le dará la opción de intentarlo de nuevo (Figura 49).



Figura 26. Barra superior del juego

Dado que la construcción de los niveles tiene cierta dinamicidad, los ingredientes que se considerarán “no válidos” en un nivel se deciden cada vez que se va a jugar un nivel, por

lo que, si se repite el mismo nivel varias veces, cada vez se establecerá como “no válido” un ingrediente distinto. En la Figura 46 se puede ver una captura del nivel uno, en el que hay que recoger manzanas y plátanos. En este caso solo hay un ingrediente no válido y es la leche. En la Figura 47 vemos el nivel después de haber recargado la página y empezado otra vez; ahora el ingrediente no válido es la sandía. Si repetimos el paso anterior, llegamos a lo que se ve en la Figura 48, en la cual el ingrediente no válido era la zanahoria¹³.

En lo que respecta a la dificultad, este valor no es visible en ningún momento para el usuario, sino que se refleja en el comportamiento del juego. La dificultad se establece en el nivel 13¹⁴ para todos los usuarios al registrarse y al iniciar sesión (la dificultad se reinicia a 13 con cada inicio de sesión), disminuyendo cuando se detecte estrés o se pierda y subiendo en caso contrario.

Por último, los niveles pueden presentar plataformas que los ingredientes no pueden atravesar, de modo que pueden suponer un refugio para el usuario, que también puede colocarse sobre ellas para llegar a sitios más altos (Figura 27).



Figura 27. Plataformas en juego

Si el usuario reúne los ingredientes necesarios, pasa a la pantalla previa a la fase de dictado (Figura 50), en la que se muestran los contadores de los ingredientes, asociados esta vez cada contador con el icono del ingrediente que corresponda. Si se pulsa el botón “Estoy listo”, cargarán las frases asociadas a la receta correspondiente y se iniciará la fase de dictado.

Dictado

Durante esta fase, se presentan al usuario una lista con frases en inglés, estando las frases de cada nivel relacionadas con la receta cuyos ingredientes se han recogido (Figura 51). Durante esta fase, el usuario ha de leer la frase que se encuentra en pantalla. Para empezar a leer una frase, el usuario debe pulsar el botón “Clic para empezar”, momento en el que éste cambiará de aspecto (tal y como vemos en la Figura 52) y se iniciará el reconocimiento de voz. Para la versión final se añadió una aureola que brilla alrededor del botón “Clic para empezar” durante un segundo cada vez que se pasa a una frase nueva, con

¹³ La decisión sobre cuáles serán los ingredientes no válidos es aleatoria. La secuencia de ingredientes no válidos comentada (primero leche, después sandía y por último zanahoria) fue aleatoria.

¹⁴ El nivel 13 equivale a que los tres valores que determinan la dificultad tengan su valor “Normal”, y que la tupla que expresa la dificultad sea (1, 1, 1).

el objetivo de recordar al usuario que tiene que pulsar el botón antes de empezar a hablar **para todas las frases**.



Figura 28. Botón “Clic para empezar” brillando

Si la pronunciación de una frase es correcta, esta se colorea en verde y se inicia la transición para pasar a la siguiente frase: el botón se desliza hacia arriba barriendo la frase, de manera que cuando vuelve a su posición se carga la siguiente frase. En la Figura 53 podemos ver el cambio de frase en plena transición tras una pronunciación correcta.

Si la pronunciación es incorrecta, se reseña este detalle y se señala la frase en rojo (Figura 54). A partir de ahora, el usuario tiene dos intentos más para pronunciar la frase correctamente. Si falla una tercera vez, la aplicación pasa automáticamente a la siguiente frase (Figura 55), para evitar bloquear al usuario y no seguir haciendo hincapié en su fallo, además de que incluso usuarios de la misma edad pueden tener niveles de inglés muy diferentes. Tras hacer la última frase, se pasa a la pantalla de resultados.

Resultados

Como ya se comentó durante el sprint correspondiente, el procesamiento de la pista de audio grabada durante la fase de dictado introduciría una latencia considerable en la aplicación: hasta que no se tenga el resultado de Beyond Verbal no se pueden traer los resultados ni determinar cómo ha de variar la dificultad del siguiente nivel. Aunque la tecnología de Beyond Verbal no se utiliza debido al problema surgido, se ha añadido una latencia manualmente para simular su funcionamiento. Para simular esa espera, se ha añadido una imagen animada de tres círculos saltando de forma regular, imitando una pantalla de carga, al mismo tiempo que se van sucediendo frases para dar a entender que la aplicación está procesando información y analizando resultados (Figura 56). Esta latencia manual es de tres segundos. Tras esta espera, se muestra una tabla con los resultados que el usuario ha conseguido durante la partida que esté jugando (Figura 57). Si el usuario decide continuar, se pasará a la pantalla de información de la siguiente receta (Figura 58).

Modificar usuario

La aplicación dispone de una pantalla a través de la cual el usuario puede modificar el correo electrónico a su usuario, su nombre de usuario y su contraseña (Figura 59).

Eliminar usuario

Al igual que se pueden crear y modificar usuarios, la aplicación ofrece una pantalla para eliminarlos (Figura 60).

Ajustes

A través de esta pantalla, los usuarios tienen la opción de habilitar y deshabilitar los distintos detectores de emociones que se utilizan en la aplicación (Figura 61).

CAPÍTULO 6. EVALUACIÓN

Tras el desarrollo de la aplicación se ha realizado una evaluación con usuarios. Para este TFG se ha llevado a cabo una evaluación de la calidad en uso, con el objetivo de comprobar si los usuarios finales son capaces de usar la aplicación. Queda para trabajos futuros realizar una evaluación para comprobar la efectividad de la aplicación como “serious game” en comparación con otros métodos de aprendizaje más tradicionales. Dado que la aplicación está pensada para ser usada por niños, la evaluación se ha realizado con usuarios de entre 7 y 11 años de edad. Como referente para la evaluación se ha tomado la ISO/IEC 25062, según la cual también se ha estructurado este capítulo. Se ha hecho también uso de un cuestionario SUS (System Usability Scale) levemente adaptado, para evaluar la satisfacción de los usuarios.

6.1 Introducción

En este apartado se presentará la descripción del producto que se va a evaluar, así como los objetivos de la evaluación que se va a realizar.

6.1.1 Descripción del producto

La aplicación **emoCook** es un prototipo que se ha construido a lo largo de 5 sprints en la que se han integrado diversas tecnologías de detección de emociones (véase junto con tecnologías para el desarrollo de videojuegos que se ejecutan en navegador. El resultado final ha sido un mini juego que utiliza las emociones detectadas en el usuario a través de las tecnologías de detección anteriormente mencionadas para variar el comportamiento del juego, de manera que la experiencia de juego sea mucho más rica.

El desarrollo del juego contenido en el prototipo nació bajo la idea de hacer un juego que pudiera ser utilizado para aprender inglés, y debido a la existencia de un proyecto previo relacionado dentro del grupo de investigación llamado Master Chef (de la Guía et al., 2016), se optó por hacer que la temática del juego girase en torno a aspectos de la cocina, concretamente de la elaboración de recetas y la recolección de ingredientes.

6.1.2 Objetivos de la evaluación

Dado que el objetivo inicial era analizar la posibilidad de dotar de afectividad a una aplicación, para que ésta se adapte al estado afectivo del usuario, se ha llevado a cabo una evaluación de la calidad en uso, reservándose para trabajos futuros la evaluación de la efectividad de la aplicación como juego serio. La evaluación se ha centrado en la activación y desactivación de los servicios de detección y en la realización completa de los niveles. Dicha evaluación comprueba los siguientes aspectos:

- **Facilidad de uso.** Tanto usuarios que han sido previamente informados como usuarios que ven la interfaz por primera vez son capaces de moverse por la aplicación, cambiar parámetros y completar los distintos niveles.
- **Funcionalidad.** Los usuarios no echan ningún aspecto en falta durante el uso de la aplicación, y a cada paso que dan, son conscientes de que la aplicación está reaccionando correctamente.
- **Utilidad.** Los usuarios consideran preferible aprender vocabulario mediante el uso de la aplicación a hacerlo mediante la elaboración de ejercicios escritos y la memorización de listas.

6.2 Método

En el siguiente apartado se detallarán el proceso que se siguió para llevar a cabo la evaluación, incluyéndose información sobre los participantes, el contexto en el que se realizó la evaluación (no solo el lugar físico en el que se realizó sino también los dispositivos que se utilizaron para ello), las tareas que los usuarios tuvieron que llevar a cabo, los pasos del proceso de evaluación y las métricas que se utilizaron.

6.2.1 Participantes

La evaluación se realizó con usuarios que se encontraban entre los 7 y los 11 años, de manera que la aplicación fuese probada por usuarios que forman parte del público objetivo de la misma. En la Tabla 15 pueden verse los datos relativos a los usuarios. Para cada usuario se ha recogido su edad, su nivel de conocimiento en Informática, su experiencia con videojuegos y el tipo de plataforma en la que suele jugar en caso de tener experiencia.

Para cada usuario nos resultaba interesante conocer principalmente su experiencia y desenvolvura con videojuegos, así como la plataforma en la que estaban más acostumbrados a jugar, ya que incluso usuarios acostumbrados a jugar con cierta frecuencia pueden tener problemas al manejar según que tipo de interfaces y controles. Por ejemplo, aunque tengamos a dos usuarios que juegan a videojuegos una hora todos los días, si uno solo juega en ordenador y el otro solo juega a la Play Station 3, si se les pone a jugar en otra plataforma, ambos tendrían problemas para adaptarse a los controles.

Dado que el objetivo de la evaluación es medir la calidad en uso del prototipo y no el nivel de inglés del usuario, se les permitirá solicitar asistencia relativa al vocabulario en inglés que pueda presentar el juego (sobre todo en la sección del dictado) sin que esto se registre como asistencia propiamente dicha.

Es muy probable que los futuros usuarios de la aplicación reciban primero una clase en la que se analice el vocabulario que posteriormente se verá en la aplicación. También cabe la posibilidad de que los usuarios de menor edad tengan más dificultades con la pronunciación y que sus ejercicios de dictado se adapten o sean más flexibles a la hora de detectar palabras.

Tabla 15. Datos sobre los participantes de la evaluación

Participante	Edad	Conocimientos informática	Experiencia con videojuegos	Plataforma habitual de juegos
1	9	Normal	Alta	Consola portátil Móvil/Tablet
2	9	Normal	Alta	Consola fija
3	9	Normal	Alta	Móvil/Tablet
4	11	Normal	Alta	Consola fija Ordenador Móvil/Tablet
5	7	Normal	Alta	Consola fija Móvil/Tablet
6	11	Normal	Alta	Consola fija Ordenador Móvil/Tablet
7	11	Normal	Normal	Consola fija Ordenador Móvil/Tablet
8	11	Normal	Alta	Ordenador

6.2.2 Contexto de uso del producto en la evaluación

Tareas

Descripción de tareas

Las tareas que los usuarios debían desempeñar estaban explicadas en la hoja que se entregó a los usuarios, que podemos encontrar en el Anexo B.1. Estas tareas consistían en navegar por la aplicación y completar un nivel (juego y dictado), así como en activar y desactivar la detección de los diversos detectores.

Justificación de la elección de tareas

La razón de elegir las tareas comentadas es la de verificar que, no solo los usuarios saben navegar por el sistema y detectan e interpretan la retroalimentación que devuelve la página durante esa navegación, sino que también saben completar niveles enteros. Dado que los niños están más que habituados a este tipo de videojuegos (juegos de plataformas), lo que se quería comprobar sobre todo era su reacción al encontrarse con ejercicios de dictado, puesto que la pronunciación es un aspecto que queda en segundo plano a la hora de enseñar idiomas a niños pequeños.

Origen de las tareas

Las tareas elegidas son las que debería afrontar cualquier usuario de la aplicación que la utilizase por primera vez y que quisiera jugar varios niveles.

Información proporcionada a los jugadores

La información entregada a los usuarios puede encontrarse en la hoja de instrucciones que se entregó a los usuarios antes de realizar la prueba (Anexo B.1). En dicha hoja se les indica los objetivos que tienen que conseguir, sin especificarse el cómo, y las interacciones que pueden hacer.

Criterio de cumplimiento

Se considera que las tareas de navegación se han completado cuando un usuario es capaz de visitar los distintos apartados del juego, así como llevar a cabo la creación, modificación y eliminación de un usuario creado por ellos mismo. Se considera que un nivel ha sido completado cuando el usuario alcanza la pantalla en la que se muestran sus resultados del juego, tras completar el juego en sí y la fase de pronunciación.

Lugar de evaluación

Las pruebas se realizaron con el usuario sentado en una silla frente a la mesa en la que se colocó el ordenador, de manera que la cámara web integrada del mismo quedase a la altura de la cara. El usuario se colocó de manera que la luz de la habitación le iluminase la cara sin crear sombras que pudieran ocasionar dificultades a los detectores faciales. El evaluador está presente durante la evaluación para supervisar al usuario, analizar su comportamiento y realizar asistencias cuando se precise.

Entorno computacional de la evaluación

La prueba se realizó desde un portátil con sistema operativo Windows 10 con procesador Intel Core i7-6500U a 2.50 GHz y 8 GB de memoria RAM. Para la detección de emociones de la cara se utilizaron la cámara y el micrófono integrados en el propio portátil.

Herramientas del administrador de la evaluación

El evaluador utilizará un cronómetro para medir el tiempo que tardan en realizarse las operaciones, así como un cuestionario SUS adaptado que los usuarios rellenaron al final de la prueba. La adaptación de dicho cuestionario consistió en cambiar la escala numérica del 1 al 5 por una escala de emoticonos que mostraban distintas emociones, desde una cara enfadada, pasando por una cara seria, hasta una cara sonriente.

6.2.3 Diseño del experimento

El evaluador recibió a los participantes en el experimento y les explicó con más detalle en que consistía el experimento, el objetivo de este TFG y los aspectos que se querían probar.

Los participantes habían sido informados levemente cuando se habló a sus padres, madres y/o tutores legales acerca del proceso de evaluación y se les pidió permiso para realizar el experimento. En el Anexo B.3 podemos encontrar la autorización que estos tuvieron que firmar para autorizar el experimento, en la que permitían a su hijo o hija participar en la evaluación.

Se les habló de cómo funcionaban a grandes rasgos las tecnologías de detección que iban a leer sus voces y sus rostros y se les enseñó en funcionamiento las demos que ofrecen tanto Afectiva como Beyond Verbal.

Dado que la aplicación estaría pensada para ser usada por un docente que previamente explicaría a los usuarios el funcionamiento de la aplicación, se les habló de las distintas secciones que conformaban el juego, así como de las reglas tanto del juego como del dictado. Se dejó claro desde el primer momento que el objetivo no era evaluar el nivel de inglés del participante, de manera que no debían preocuparse por fallar las fases de dictado o por tener un vocabulario innecesario. Pensando precisamente en esto, se les dio la opción de pedir la asistencia del evaluador de forma ilimitada para cuestiones de vocabulario, sin que esto se registre como asistencia debido a problemas de usabilidad.

Tras estas explicaciones, se daba al usuario la hoja de tareas y se le daba algunos minutos para leerla y ver que entendían el vocabulario que se usaba. Tras esta lectura, el usuario empezaba su prueba, siguiendo las tareas indicadas y solicitando el auxilio del evaluador cuando tuviese problemas, registrándose solo las peticiones de auxilio relacionadas con la usabilidad. Acabada la prueba, los usuarios rellenaban un cuestionario SUS adaptado, previamente explicado por el evaluador.

Instrucciones de las tareas de los participantes

Las tareas que los usuarios tenían que realizar están detalladas en el Anexo B.1, aunque estas se pueden resumir en dos categorías:

- **Navegación por la aplicación.** Estas tareas se realizan de forma paralela a la finalización de un nivel del juego. Durante la navegación por la aplicación se hizo hincapié en que los usuarios leyesen la información mostrada para comprobar que transmitía correctamente lo que el equipo de desarrollo pretendía reflejar.

- **Realización de nivel completo.** Se indicó a los usuarios que realizaran un nivel al completo, esto es, recoger los ingredientes (primera fase, juego) y pronunciar las frases correctamente (segunda fase, dictado).

6.2.4 Métricas de usabilidad

Efectividad

- **Tasa de completitud**
 - **Tasa de completitud desasistida.** Porcentaje de tareas que el usuario ha completado sin necesidad de ayuda.
 - **Tasa de completitud asistida.** Porcentaje de tareas que el usuario ha completado con ayuda del evaluador.
- **Errores.** Se produce un error cuando usuario no consigue completar una tarea o tiene que repetir una parte de la tarea varias veces para conseguirla.
- **Asistencias.** Se considera que se ha producido una asistencia cuando el evaluador tiene que ayudar al usuario para poder continuar o para terminar una tarea.

Eficiencia

- **Tiempo de tarea.** Tiempo en minutos y segundos que el usuario ha tardado en hacer la prueba entera

Satisfacción

Para medir la satisfacción se hizo uso de un cuestionario SUS (System Usability Scale) que se entregó a los usuarios para que rellenasen tras realizar la prueba (SUS, Web). El cuestionario SUS está formado por 10 preguntas relacionadas con diversos aspectos de la experiencia de usuario que se ha tenido, valorándose cada aspecto con una escala de 5 valores, indicando los valores bajos que no se está de acuerdo con el aspecto considerado y los valores altos que si se está de acuerdo. Asignando a cada valor de cada aspecto un número y aplicando unas operaciones matemáticas, podemos expresar la usabilidad del sistema mediante un número.

Dado que los cuestionarios iban a ser rellenados por niños, en la escala para responder a las preguntas se cambiaron los valores numéricos del 1 al 5 por emoticonos que expresaban desde enfado hasta alegría, pasando por una expresión de indiferencia. Algunas de las preguntas fueron también ligeramente reescritas para que resultasen más naturales.

6.2.5 Resultados

Siguiendo la ISO/IEC 25062, en este apartado se realiza un análisis de los datos obtenidos. En primer lugar, se reflejan los datos obtenidos durante las distintas evaluaciones, distinguiéndose los datos recogidos durante las tareas de navegación de los recogidos durante el propio juego. Para cada grupo de datos se realiza un análisis estadístico, calculándose valores medios, mínimos y máximos. Se presentan también los resultados recogidos a través de los cuestionarios SUS, y por último se comentan diversos problemas a considerar que se observaron durante las evaluaciones.

Resultados de desempeño

En la Tabla 16 se presentan los datos relativos a las tareas de navegación. En general, todos los usuarios tuvieron tasas de completitud desasistidas altas, si bien es cierto que algunos tardaron más tiempo en entender para que servían los distintos apartados por los que navegaban. En la Tabla 17 se muestra el resumen estadístico de estos datos, mientras que en la Figura 29 podemos ver una representación gráfica de los mismos.

Tabla 16. Resultados de tareas de navegación

Participante	Tasa de completitud desasistida	Tasa de completitud asistida	Tiempo	Errores	Asistencia
1	80	20	5.50	0	1
2	100	0	8.44	1	0
3	80	20	8.26	0	1
4	80	20	8.26	1	1
5	100	0	9.18	0	0
6	100	0	5.48	0	0
7	100	0	5.30	1	0
8	80	20	5.39	1	1

Tabla 17. Datos estadísticos sobre resultados de navegación

Valores estadísticos	Tasa de completitud desasistida	Tasa de completitud asistida	Tiempo	Errores	Asistencia
Media	90	10	6.98	0.50	0.50
Min	80	0	5.30	0	0
Max	100	20	9.18	1	1

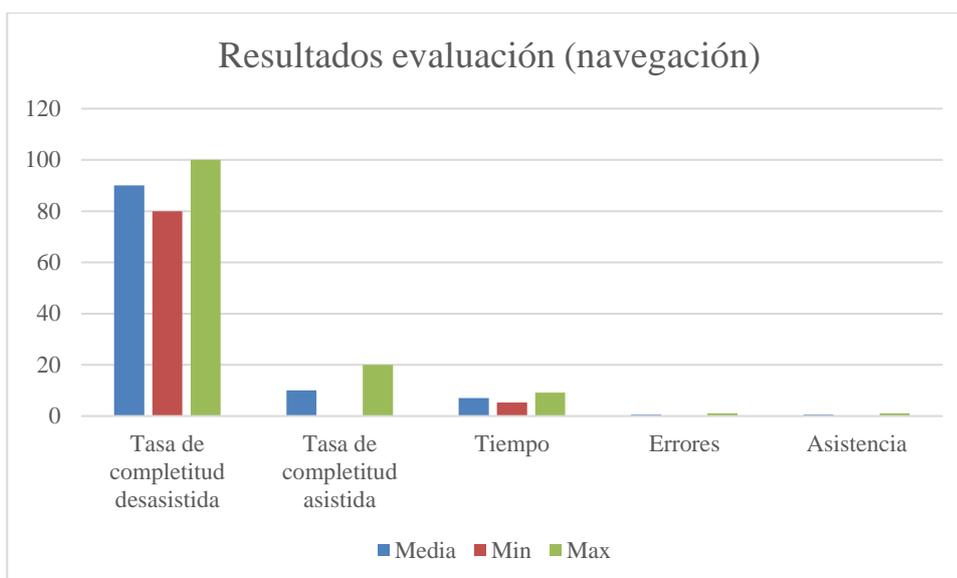


Figura 29. Gráfica sobre datos estadísticos de los resultados de navegación

Resultados de tareas de juego

En lo que respecta al juego y al dictado, aunque la mayoría supo terminar el juego sin problemas y en el primer intento, es cierto que muchos se vieron abrumados cuando este comenzó, incluso después de haber visto las capturas en la pantalla inicial que mostraban como era el juego (Figura 40).

Tabla 18. Resultados de tareas de juego

Participante	Tasa de completitud desasistida	Tasa de completitud asistida	Tiempo	Errores	Asistencia
1	80	20	5.00	2	1
2	80	20	4.90	1	1
3	80	20	7.14	0	1
4	100	0	5.31	1	0
5	60	40	6.99	1	2
6	100	0	3.99	0	0
7	100	0	3.79	0	0
8	60	40	4.8	1	2

En esta parte de la evaluación, tal y como podemos ver en la Tabla 18, la tasa de completitud desasistida es algo menor que en las tareas de navegación, siendo muchas de estas asistencias preguntas acerca de los controles de la aplicación. Algunos cometieron errores intentando usar otras teclas para controlar el juego y solo uno se acordó del botón de ayuda para consultar los controles.

En la fase del dictado, el brillo del botón para empezar resultó ser de ayuda, pero aun así hubo asistencias sobre qué era lo que había que hacer en esa fase, así como errores debido a que los usuarios olvidaban pulsar el botón. Al igual que antes, se presentan los valores estadísticos asociados a los datos de esta fase (Tabla 19) junto con la representación gráfica de los mismos (Figura 30).

Tabla 19. Datos estadísticos sobre resultados de juego

Valores estadísticos	Tasa de completitud desasistida	Tasa de completitud asistida	Tiempo	Errores	Asistencia
Media	82.50	17.5	5.24	0.75	0.71
Min	60	0	3.79	0	0
Max	100	40	7.14	2	2

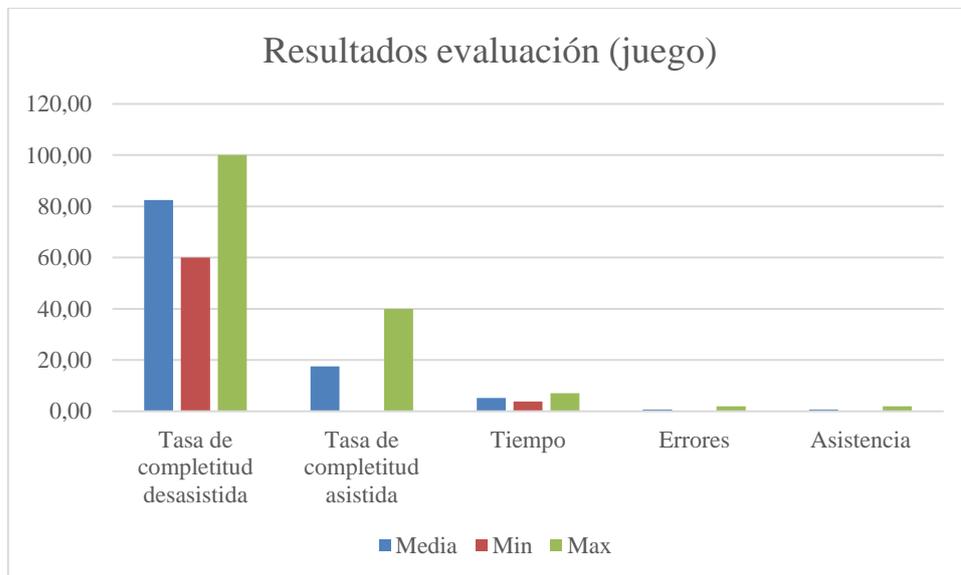


Figura 30. Gráfica sobre datos estadísticos de los resultados de juego

Resultados de satisfacción

Por último, se suministró un cuestionario SUS a los evaluados para que puntuasen la experiencia de usuario. En la Tabla 20 podemos ver la puntuación SUS que dio cada uno, así como la media de los resultados.

Tabla 20. Resultados de cuestionarios SUS

Participante	Puntuación SUS
1	77.5
2	60
3	72.5
4	80
5	85
6	95
7	95
8	80
Media	80.625

6.3 Conclusiones de la evaluación

Es interesante tener en cuenta los siguientes detalles que se observaron en la evaluación.

- Aunque en la primera pantalla de la aplicación se muestran capturas del juego y antes de cada nivel se habla de la receta que se va a cocinar, los usuarios se veían algo desconcertados cuando empezaban el nivel (aun cuando todos han afirmado jugar a videojuegos con frecuencia y los controles del juego son los clásicos). Pese a ser inusual, dado que el punto de aparición de cada ingrediente es aleatorio, pueden darse casos en los que un usuario gane o pierda en apenas unos segundos sin haber tomado consciencia de que ha empezado el juego. Esto plantea la necesidad de establecer un nivel cero a modo de tutorial, o de presentar una captura del juego completo con los controles y las reglas del juego, o incluso que el juego se “ponga en marcha” de forma gradual.
- En la primera pantalla hay una captura de la fase de dictado en la que se puede ver una frase y un botón para empezar a pronunciar. Algunos usuarios se confundieron y al pasar a la tarea de iniciar sesión hicieron clic en esa foto. Esto pone en evidencia que cuando un usuario lee el resto de la página olvida el botón inicial que se encontró, por lo que sería necesario modificar esa captura o hacer más clara la función del botón “Acepto el reto”.
- Al rellenar el formulario de inicio de sesión, un par de usuarios pulsó el botón de “Registrar”, justo debajo de “Iniciar sesión”, en lugar de este último. Se pone en evidencia que la opción de registrarse tiene que ser más discreta, de manera que, en el cuadro de iniciar sesión, solo se encuentre claramente visible el botón “Iniciar sesión”.
- Algunos usuarios confundían el apartado “Ajustes” y el botón de “Ayuda”.

- Durante la fase de dictado, tras haberse leído las reglas del mismo, algunas personas pulsaban el botón para empezar el dictado y no leían las frases, quedándose a la espera de que pasase algo. Sería necesario reforzar la idea de que ese apartado implica que hablen a lo largo de los distintos puntos informativos de la aplicación, o añadir alguna frase a modo de aviso que recuerde, tras pulsar el botón de empezar el dictado, que el usuario tiene que hablar.
- En el dictado hay que pulsar el botón “Clic para empezar” para cada frase. Se hizo que el botón brillase cada vez que se pasa a una nueva frase para recordar esto. No obstante, varios de los usuarios empezaban a hablar sin pulsar el botón cuando la aplicación pasaba a la siguiente frase. Podría utilizarse un aviso como el comentado anteriormente.

CAPÍTULO 7. CONCLUSIONES Y PROPUESTAS

Por último, en este capítulo se presentan las conclusiones extraídas de la elaboración de este trabajo, así como una revisión de los conocimientos adquiridos durante el desarrollo del mismo. Dado que el autor del TFG pretende seguir en el equipo de investigación en el que se comenzó el desarrollo, se incluyen posibles mejoras que podrían hacerse al proyecto próximamente.

7.1 Conclusiones

Este proyecto se inició con el objetivo de desarrollar una aplicación que fuese capaz de detectar las emociones de sus usuarios para poder modificar su comportamiento en tiempo real en base a los mismos, ofreciendo así una experiencia de usuario mucho más rica que si se hubiera planteado una aplicación estática. Para satisfacer este objetivo, se planteó la posibilidad de desarrollar un juego para navegadores web, lo que lo haría más accesible a múltiples plataformas, que utilizase las emociones detectadas en el usuario —leídas a través de la expresión facial, la voz y el teclado— para, en base a estas, modificar el comportamiento del juego, alterando su complejidad, velocidad y/u objetivos, con el fin de ofrecer una mejor experiencia de juego. Si se detecta que el usuario se estresa, el juego se hará más sencillo, mientras que si se detecta aburrimiento, éste se volverá más complejo.

Se aceptó esta propuesta, pero para enriquecer dicha aplicación, se decidió dotarla de un fin educativo. Siguiendo el planteamiento de un proyecto previo (de la Guía et al., 2016) del equipo de investigación en el que se enmarcaba este trabajo, se eligió como tema principal del juego la cocina, concretamente la elaboración de recetas y la recolección de ingredientes. Se desarrollaría la aplicación para ser un juego para aprender inglés para niños en edad de primaria, un juego que utilizaría las emociones de los usuarios para decidir cómo variar la dificultad del mismo, haciéndose más sencillo para aquellos usuarios que se pusieran nerviosos o se estresasen y más desafiante para los usuarios que se aburriesen. Este objetivo ha sido finalmente satisfecho, habiéndose realizado pruebas de usabilidad con usuarios que han dado buenos resultados.

Este trabajo está enmarcado en un campo nuevo que aún está emergiendo, la Computación Afectiva, pero que aun así bulle de actividad. Este hecho se ha manifestado repetidas veces a lo largo del proyecto: durante el sprint 0, Afectiva aún era un software que se prestaba bajo licencia de cuarenta y cinco días, mientras que durante el sprint 4, la tecnología que se había estado utilizando para analizar la voz actualizó su API y restringió el acceso a ciertas funcionalidades. Dada esta actividad principalmente comercial, unido al hecho de que los más interesados en este campo son empresas de desarrollo de detectores de emociones, son pocos los estudios que realmente estudian o utilizan las emociones para aplicarlas a un problema, a la demostración de una hipótesis, etc. Es debido a esto que el trabajo aquí realizado supone una aportación a la disciplina de la Computación Afectiva, como una nueva propuesta para combinar la Computación Afectiva con el aprendizaje.

Merece la pena destacar que este proyecto resultó cautivador desde el primer momento debido a su relación con la psicología y la medicina, campos de estudios en los que el autor está fervientemente interesado. Del mismo modo, el desarrollo futuro de este proyecto y otros de la misma temática implicará ahondar en otros aspectos de la detección de emociones como la clasificación automática, un campo que además tiene mucho futuro a nivel profesional.

Respecto al desarrollo propiamente dicho, este trabajo ha permitido al autor seguir trabajando con tecnologías web, tecnologías en las que el autor está interesado debido al alcance que pueden tener. Por último, pero no por ello menos importante, se pudo afrontar el desarrollo de forma completamente autónoma haciendo uso de una metodología que hasta este trabajo solo se había utilizado en el aula, Scrum, lo que ha permitido afianzar conocimientos sobre la misma que no se podrían haber afianzado de forma teórica.

En resumen, ha resultado de lo más gratificante ver cómo han aflorado todas las cosas aprendidas a lo largo de la intensificación (y el grado) durante el transcurso del trabajo. Además, la realización de este trabajo ha permitido adentrarse en disciplinas que seguirían siendo desconocidas para el autor si no fuera por este proyecto, lo que a su vez aporta valor adicional al autor de cara al mundo laboral.

Por último, destacar que fruto de parte del trabajo realizado en este TFG se ha publicado un artículo en un congreso del ámbito de la Interacción Persona-Ordenador, Interacción 2017, que lleva por título “Emotion Detection: A Technology review” (García-García, Penichet & Lozano, 2017).

7.2 Trabajo futuro y posibles ampliaciones

Durante el transcurso de este proyecto, han ido apareciendo numerosas ideas centradas en la problemática de la Computación Afectiva, aunque se mantuvieron fuera de los artefactos de Scrum dado que eran ideas para un futuro a largo plazo. Aquí se presentan algunas de estas propuestas, ordenadas de menor a mayor dificultad.

Mejora de la interacción

Atendiendo a los errores y asistencias producidas durante la evaluación, resultaría interesante corregirla para que estos se eviten, al mismo tiempo que se hacen las modificaciones pertinentes para que la aplicación funcione en un mayor número de navegadores y en un mayor número de dispositivos.

Módulo para docentes

Dado que la aplicación está pensada para ser usada por niños como un medio distinto a los habituales para aprender nuevo vocabulario, podría resultar interesante para un docente a cargo de dichos niños disponer de una herramienta que le permitiera hacer un seguimiento de los mismos, analizar su progreso y los estados afectivos presentados durante una partida, comparar los resultados de todos ellos, etc.

Inclusión de nuevos detectores

Dado que se dispone de una buena base conceptual del campo de la detección de emociones, sería interesante incluir nuevos tipos de detectores de emociones en el juego. Aprovechando que el público objetivo está conformado por niños, se podría recurrir a sistemas de detección que impliquen el uso de hardware adicional y disfrazar dicho hardware como un componente del juego. De esta manera, una pulsera para leer la actividad electrodérmica de la piel y detectar estados de estrés y alerta puede convertirse, con un poco de trabajo artístico y un sensor NFC, en el Brazaletes del Poder que se utiliza para desbloquear niveles, coleccionables, etc.

Implementación de detectores propios

En este proyecto no ha existido una preocupación excesiva respecto a los detectores que se pudieran utilizar, dado que el objetivo era construir un prototipo y analizar como variaría el funcionamiento del mismo. No obstante, el hecho de tener que depositar la responsabilidad de la detección de emociones en servicios externos supone un gran riesgo, sobre todo cuando la mayoría son de empresas privadas. Con miras a eliminar esta dependencia, sería interesante estudiar la posibilidad de implementar nuestros propios detectores de emociones y desplegarlos en un servidor privado (o incluso público, si queremos que estén disponible para más investigadores).

BIBLIOGRAFÍA

LIBROS Y ARTICULOS

- (Binali & Potdar, 2012) Haji Binali and Vidyasagar Potdar. (2012). *Emotion detection state of the art*. En *Proceedings of the CUBE International Information Technology Conference (CUBE '12)*. ACM, New York, NY, USA, 501-507. DOI: <http://dx.doi.org/10.1145/2381716.2381812>
- (Casale, Russo, Scebba & Serrano, 2008) S. Casale, A. Russo, G. Scebba and S. Serrano. (2008). *Speech Emotion Classification Using Machine Learning Algorithms*. En *IEEE International Conference on Semantic Computing*. Santa Clara, CA, 158-165. DOI: 10.1109/ICSC.2008.43
- (Cohn, 2005) Cohn, Mike. (2005). *Agile Estimating and Planning*. Stoughton, Massachusetts: Prentice Hall.
- (Conner-Simons & Gordon, 2016) A. Conner-Simons and R. Gordon. (2016). *Detecting emotions with wireless signals*. Recuperado de <http://news.mit.edu/2016/detecting-emotions-with-wireless-signals-0920>
- (Deleurme, 2017) Deleurme, O. (1 de mayo de 2017). *Me emociono, luego aprendo*. El País. Recuperado de http://economia.elpais.com/economia/2017/04/26/actualidad/1493238722_987075.html
- (Djaouti, Alvarez & Jessel, 2011) Djaouti, D., Alvarez, J., Jessel, J.-P. (2011). *Classifying Serious Games: The G/P/S Model*. In: *Handbook of Research on Improving Learning and Motivation through Educational Games: Multidisciplinary Approaches*. pp. 118-136. IGI Global, Hershey
- (García-García, Penichet & Lozano, 2017) García-García, J. M., Penichet, V. M. R. & Lozano, M. D. (2017). *Emotion Detection: A Technology review*. En *Interacción '17*, 2017, Cancún, México.
- (Gilleade, Dix & Allanson, 2005) Gilleade, K., Dix, A., & Allanson, J. (2005). *Affective videogames and modes of affective gaming: assist me, challenge me, emote me*.

- (de la Guía et al., 2016) de la Guía, E., López, V., Orozco-Barbosa, L., Brea, V. M., Penichet, V. M. & Lozano, M. (2016). *Introducing IoT and Wearable Technologies into Task-Based Language Learning for Young Children*. En *IEEE Transactions on Learning Technologies*, vol. 9, no. 4, Octubre-Diciembre 2016.
- (Huotari & Hamari, 2012) Huotari, K. & Hamari, J. (2012). *Defining Gamification – A Service Marketing Perspective*. En *MindTrek 2012*. Octubre 3-5. Tampere, Finlandia.
- (ISO/IEC 25062, 2006) ISO/IEC 25062. (2006). *SQuaRE - Common Industry Format (CIF) for usability test reports*. ISO
- (Kleinsmith & Bianchi-Berthouze, 2013) A. Kleinsmith and N. Bianchi-Berthouze. *Affective Body Expression Perception and Recognition: A Survey*. In *IEEE Transactions on Affective Computing*, vol. 4, no. 1. Jan.-March 2013. 15-33. DOI: 10.1109/T-AFFC.2012.16
- (Kostoulas et al., 2012) T. Kostoulas, I. Mporas, O. Kocsis, T. Ganchev, N. Katsaounos, J. J. Santamaria, S. Jimenez-Murcia, F. Fernandez-Aranda, and N. Fakotakis- (2012). *Affective speech interface in serious games for supporting therapy of mental disorders*. *Expert Syst. Appl.*, vol. 39, no. 12, pp. 11072–11079. DOI: 10.1016/j.eswa.2012.03.067
- (Lazar, Jones, Hackley & Shneiderman, 2005) Lazar J., Jones A., Hackley M. & Shneiderman B. (2005). *Severity and impact of computer user frustration: A comparison of student and workplace users*. En *Interacting with Computers*, Volumen 18. Págs 187-207. DOI: 10.1016/j.intcom.2005.06.001
- (nViso, 2011) nViso. (2011). *Say to my face! Applying facial imaging to understand consumer emotional response*. En *AMSRS Conference*. Recuperado de http://www.nviso.ch/assets/pdf/nViso_AMSRS_2011_3D_Facial_Imaging_Best_Presentation_Award.pdf
- (Pfister, 2010) Pfister, Tomas. (2010). *Emotion Detection from Speech*. University of Cambridge Part II dissertation.
- (Piana, et al., 2014) S. Piana, A. Staglianò, F. Odone, A. Verri and A. Camurri. 2014. *Real-time Automatic Emotion Recognition from Body Gestures*. arXiv:1402.5047

- (Picard, 1995) Picard, Rosalind W. (1995). *Affective Computing. Tech. Rep. 321*, MIT Media Lab
- (Picard, 2009) Picard, Rosalind W. (2009). *Future affective technology for autism*. En *Philosophical Transactions of The Royal Society B Biological Sciences*. Págs 3575–3584.
- (Price, 2016) Price, M. (17 de octubre de 2016). *Facial expressions—including fear—may not be as universal as we thought*. Science. Recuperado de <http://www.sciencemag.org/news/2016/10/facial-expressions-including-fear-may-not-be-universal-we-thought>. DOI: 10.1126/science.aal0271
- (Santos, 2016) Santos, O. C. (2016). *Emotions and personality in adaptive e-learning systems: an affective computing perspective*. En *Emotions and Personality in Personalized Services* (pp. 263-285). Springer International Publishing.
- (Schwaber & Sutherland, 2016) Schwaber, K. & Shuterland, J. (2016). *The Scrum Guide*. Recuperado de: <http://www.scrumguides.org/docs/scrumguide/v2016/2016-Scrum-Guide-US.pdf>
- (Scrum Alliance, 2016) Scrum Alliance. (2016). *2016 State of Scrum report: How the world is successfully applying the most popular Agile approach to projects*. Recuperado de: https://www.scrumalliance.org/scrum/media/ScrumAllianceMedia/Files%20and%20PDFs/State%20of%20Scrum/StateofScrum_2016_FINAL.pdf?aliId=240941886
- (Tao & Tan, 2005) Tao, Jianhua & Tan, Tieniu. (2005). *Affective computing: a review*. En *Proceedings of the First international conference on Affective Computing and Intelligent Interaction (ACII'05)*, Jianhua Tao, Tieniu Tan, and Rosalind W. Picard (Eds.). Springer-Verlag, Berlin, Heidelberg, 981-995. DOI=http://dx.doi.org/10.1007/11573548_125
- (Vogt, André & Bee, 2008) Vogt T., André E., & Bee N. (2008). *EmoVoice — A Framework for Online Recognition of Emotions from Voice*. In: André E., Dybkjær L., Minker W., Neumann H., Pieraccini R., Weber M. (eds) *Perception in Multimodal Dialogue Systems*. PIT 2008. Lecture Notes in Computer Science, vol 5078. Springer, Berlin, Heidelberg. DOI: 10.1007/978-3-540-69369-7_21

- (Wagner, Lingenfelter & Andre, 2011) J. Wagner, F. Lingenfelter, and E. Andre. (2011). *The Social Signal Interpretation Framework (SSI) for Real Time Signal Processing and Recognitions*. In *Proceedings of INTERSPEECH 2011*. Florence, Italy, 2011.
- (Wilkinson, 2013) Wilkinson, P. (2013). *Affective educational games: Utilizing emotions in game-based learning*. En *2013 5th International Conference on Games and Virtual Worlds for Serious Applications (VS-GAMES)*, Poole, 2013, pp.1-8. doi: 10.1109/VS-GAMES.2013.6624219

ENLACES INTERNET

- (Afectiva, Web) Afectiva. Emotion Recognition Software and Analysis. <http://www.afectiva.com/> (marzo, 2017).
- (Agencia, Web) Agencia Española de Protección de Datos. *Glosario de términos*. https://www.agpd.es/portalwebAGPD/canalresponsable/inscripcion_fic_heros/preguntas_frecuentes/glosario/index-ides-idphp.php (mayo, 2017)
- (Agile, Web) Agile Manifesto. <http://agilemanifesto.org/> (mayo, 2017)
- (Agile Methods, Web) Agile Methods of Software Development. <https://eternalsunshineoftheismind.wordpress.com/2013/02/04/agile-methods-of-software-development-2/>
- (Audio, Web) Audio Recorder. <https://webaudiodemos.appspot.com/AudioRecorder/index.html> (marzo, 2017)
- (Beyond Verbal, Web) Beyond Verbal – the emotions analytics company. <http://www.beyondverbal.com/> (mayo, 2017)
- (Blog Afectiva, Web) The Emotion Behind Facial Expressions. <http://blog.afectiva.com/the-emotion-behind-facial-expressions> (mayo, 2017)
- (Blog BiText, Web) Machine Learning & Deep Linguistic Analysis in Text Analytics <https://blog.bitext.com/machine-learning-deep-linguistic-analysis-in-text-analytics> (mayo, 2017)

- (Boe, Web) BOE. Ley Orgánica 15/1999, de 13 de diciembre, de Protección de Datos de Carácter Personal. <https://www.boe.es/buscar/act.php?id=BOE-A-1999-23750> (mayo, 2017)
- (Discuss Afectiva, Web) The Emotion SDK & Privacy. <http://discuss.affectiva.com/t/the-emotion-sdk-privacy/54> (mayo, 2017)
- (emoCook, Web) emoCook. <https://emocook.herokuapp.com/>
- (Emotion API, Web) Emotion API: Detección de expresiones emocionales. <https://azure.microsoft.com/es-es/services/cognitive-services/emotion/> (mayo, 2017)
- (Gartner, Web) Gartner's 2016 Hype Cycle for Emerging Technologies Identifies Three Key Trends That Organizations Must Track to Gain Competitive Advantage. <http://www.gartner.com/newsroom/id/3412017> (marzo, 2017)
- (GitHub, Web) GitHub. <https://github.com/> (marzo, 2017)
- (Handel, Web) Steven Handel. Classification of Emotions. <http://www.theemotionmachine.com/classification-of-emotions/> (mayo, 2017)
- (Heroku, Web) Heroku. <https://www.heroku.com/> (junio, 2017)
- (Internet live stats, Web) Twitter Usage Statistics. <http://www.internetlivestats.com/twitter-statistics/> (junio, 2017)
- (Kairos, Web) Face Recognition, Emotion Analysis & Demographics. <https://www.kairos.com> (mayo, 2017)
- (Kunagi, Web) Kunagi. Agile project management based on Scrum. <http://kunagi.org/> (mayo, 2017)
- (Learning Tree, Web) Is Your Project Plan-Driven or Change-Driven? <http://blog.learningtree.com/is-your-project-plan-driven-or-change-driven/> (marzo, 2017)

- (Material, Web) Material Design. <https://material.io/> (marzo, 2017)
- (Microsoft, Web) How to Detect Faces in Image. <https://docs.microsoft.com/es-es/azure/cognitive-services/Face/Face-API-How-to-Topics/HowtoDetectFacesinImage> (mayo, 2017)
- (Mind Tools, Web) Body Language Understanding Non-Verbal Communication. https://www.mindtools.com/pages/article/Body_Language.htm (mayo, 2017)
- (mLab, Web) mLab. <https://mlab.com/> (junio, 2017)
- (Morgun, Web) Ivan Morgun. Types of machine learning algorithms. <http://en.proft.me/2015/12/24/types-machine-learning-algorithms/> (mayo, 2017)
- (Mountain Goat, Web) Mountain Goat Software. User stories. <https://www.mountaingoatsoftware.com/agile/user-stories> (junio, 2017)
- (Nevermind, Web) Nevermind. <http://nevermindgame.com/> (mayo, 2017)
- (Neuroscience, Web) Emotional Ads Work Best. <http://www.neurosciencemarketing.com/blog/articles/emotional-ads-work-best.htm> (marzo, 2017)
- (nViso, Web) Artificial Intelligence Emotion Recognition Software. <http://nviso.ch/> (mayo, 2017)
- (Phaser, Web) Phaser. A fast, fun and free open source HTML5 game framework. <http://phaser.io/> (junio, 2017)
- (Receptiviti, Web) Receptiviti. <https://labs.receptiviti.com/> (mayo, 2017)
- (Science Tone, Web) Science behind the service – Tone Analyzer. <https://www.ibm.com/watson/developercloud/doc/tone-analyzer/science.html> (mayo, 2017)

- (Scrum Methodology, Web) Scrum Methodology. Learn Scrum. <http://scrummethodology.com/>
- (Slower Speed, Web) A Slower Speed of Light. <http://gamelab.mit.edu/games/a-slower-speed-of-light/> (mayo, 2017)
- (Smartsheet, Web) Smartsheet. Comprehensive Guide to the Agile Manifesto. <https://www.smartsheet.com/comprehensive-guide-values-principles-agile-manifesto> (mayo, 2017)
- (SpeechRecognition, Web) Mozilla Developer Network. SpeechRecognition. <https://developer.mozilla.org/en-US/docs/Web/API/SpeechRecognition> (mayo, 2017)
- (StarUML, Web) StarUML. <http://staruml.io/> (mayo, 2017)
- (SUS, Web) System Usability Scale (SUS). <https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html> (junio, 2017)
- (Synesketch, Web) Synesketch. Free Open-Source Textual Emotion Recognition and Visualization. <http://krcadinac.com/synesketech/> (mayo, 2017)
- (Teach your monster, Web) Teach your monster to read. <https://www.teachyourmonstertoread.com/> (mayo, 2017)
- (Ted, Web) TED. (Abril 2013). How to escape education's death valley. [Archivo de video]. Recuperado de: https://www.ted.com/talks/ken_robinson_how_to_escape_education_s_death_valley (mayo, 2017)
- (Tone Analyzer, Web) Tone Analyzer. <https://tone-analyzer-demo.mybluemix.net/> (mayo, 2017)
- (Trello, Web) Trello. <https://trello.com/> (mayo, 2017)
- (Typing, Web) Typing. <https://www.typing.com/> (mayo, 2017)

- (Understand, Web) Understand your tone score – Tone Analyzer. <https://www.ibm.com/watson/developercloud/doc/tone-analyzer/understand-tone.html> (mayo, 2017)
- (Universität, Web) Universität Augsburg University. *EmoVoice - Real-time emotion recognition from speech.* <https://www.informatik.uni-augsburg.de/en/chairs/hcm/projects/tools/emovoice> (mayo, 2017)
- (VersionOne, Web) VersionOne. *What is Kanban?* <https://www.versionone.com/what-is-kanban/> (junio, 2017)
- (Visual Studio Code, Web) Visual Studio Code. *Code editing. Redefined.* <https://code.visualstudio.com/> (mayo, 2017)
- (Vokaturi, Web) Vokaturi. <https://vokaturi.com/> (mayo, 2017)

CONTENIDO DEL CD

En el contenido del CD que acompaña a la memoria podemos encontrar los siguientes recursos:

- Memoria del trabajo en los formatos PDF, DOCX y DOC dentro del directorio Memoria.
- Código fuente del trabajo dentro del directorio Código.
- Pruebas pasadas a la API REST dentro del directorio Código/emoCook.zip/test
- Libros y artículos a los que se ha hecho referencia durante la memoria y que se han utilizado como bibliografía. Los cuales podemos encontrar en el directorio Bibliografía/Libros y artículos.
- Páginas Web que han servido de bibliografía. Las podemos encontrar dentro del directorio Bibliografía/Enlaces Web.
- Informes generados por Kunagi al acabar cada sprint en Informes kunagi.

ANEXO A. INCREMENTOS DE SPRINTS

En este apartado se pueden encontrar capturas que reflejan el estado del prototipo desarrollado a lo largo de los distintos sprints. En cada subapartado se muestran solo las pantallas que hayan cambiado respecto a la iteración anterior, para evitar la redundancia de las imágenes.

A.1 Incremento del Sprint 1

Pantalla inicial del prototipo (Figura 31).

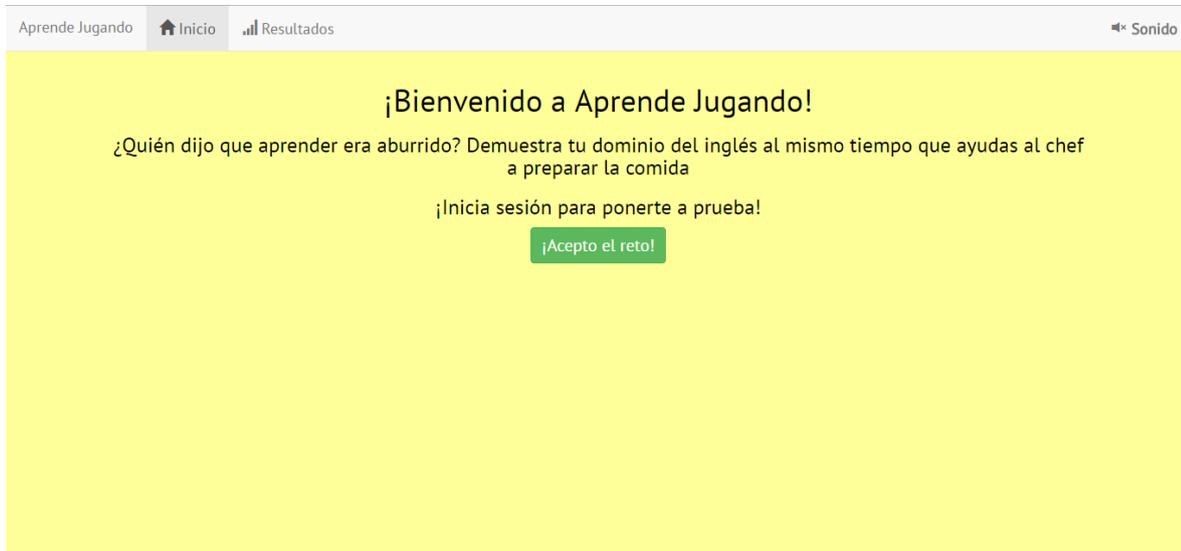


Figura 31. Sprint 1. Pantalla de bienvenida de prototipo

Pantalla de inicio de sesión del prototipo (Figura 32).

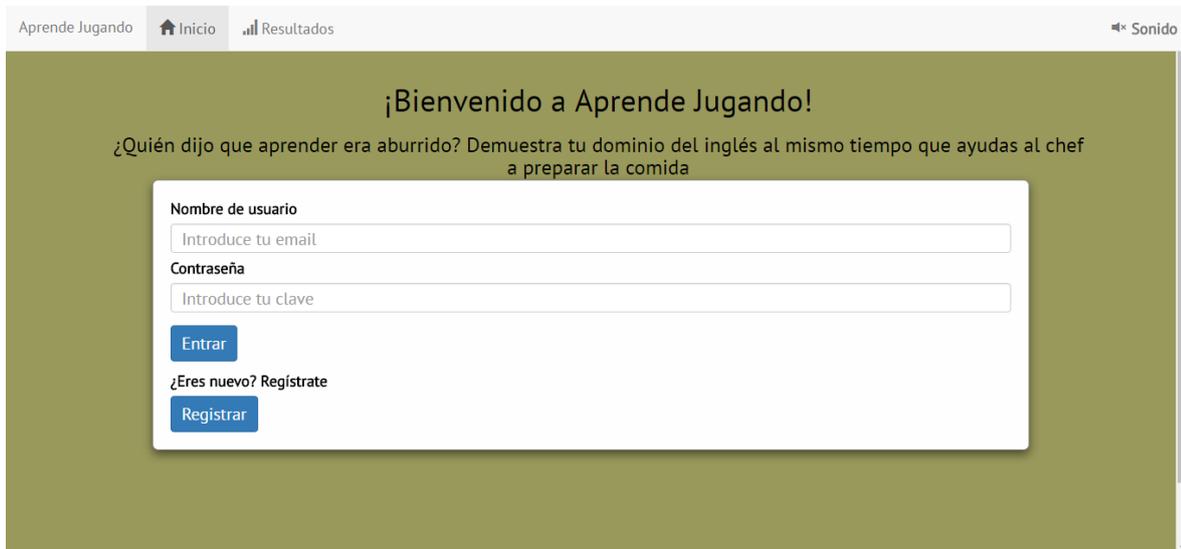


Figura 32. Sprint 1. Pantalla de inicio de sesión de prototipo

Pantalla de juego del prototipo (Figura 33).

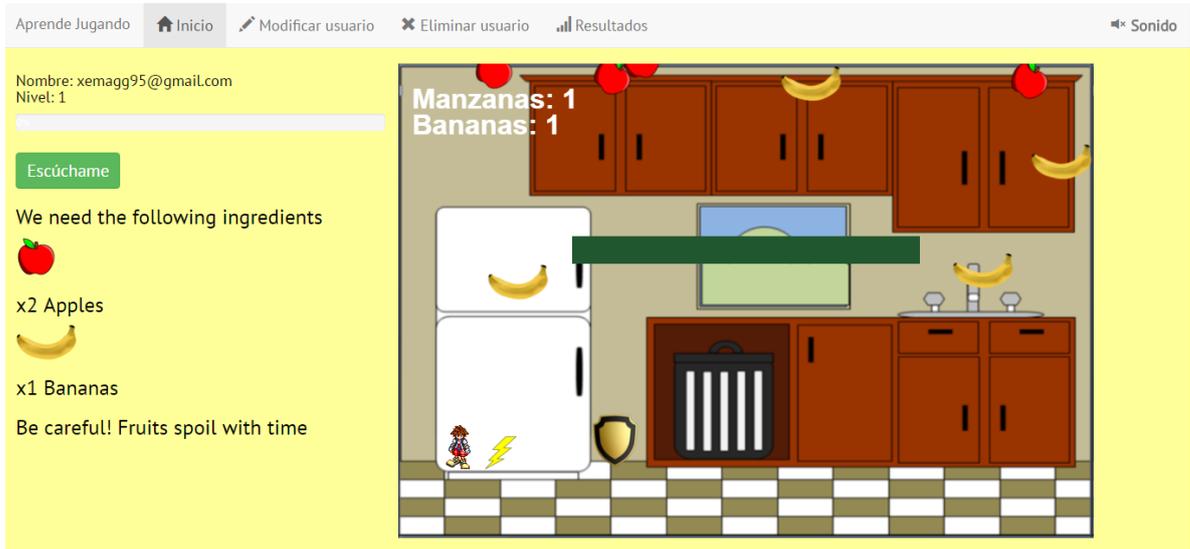


Figura 33. Sprint 1. Pantalla de juego de prototipo

Pantalla con traductor en funcionamiento (Figura 34).



Figura 34. Sprint 1. Pantalla con traductor en funcionamiento

Pantalla de registro de usuario (Figura 35).

Aprende Jugando Inicio Resultados Sonido

Nombre de usuario (único)

Correo electrónico (a esta dirección se mandará el correo de confirmación de la cuenta)

Contraseña

Repite la contraseña

Regístrate

Figura 35. Sprint 1. Pantalla de registro de usuario

Pantalla de modificar usuario (Figura 36).

Aprende Jugando Inicio Modificar usuario Eliminar usuario Resultados Sonido

Nombre: xemagg95@gmail.com
Nivel: 1

Escúchame

We need the following ingredients

x1 Apples

x1 Bananas

Be careful! Fruits spoil with time

Nombre de usuario (único)

Correo electrónico

Contraseña

Repite la contraseña

Guardar cambios

Figura 36. Sprint 1. Pantalla de modificar usuario

Pantalla de eliminar usuario (Figura 37).

Aprende Jugando Inicio Modificar usuario Eliminar usuario Resultados Sonido

Nombre: xemagg95@gmail.com
Nivel: 1

Escúchame

We need the following ingredients

🍏
x0 Apples

🍌
x1 Bananas

Be careful! Fruits spoil with time

Confirma tus credenciales. Vas a eliminar tus datos

Nombre de usuario (único)

Correo electrónico

Contraseña

Eliminar credenciales

Figura 37. Sprint 1. Pantalla de eliminar usuario

A.2 Incremento del Sprint 2

Pantalla de juego de prototipo (Figura 38).

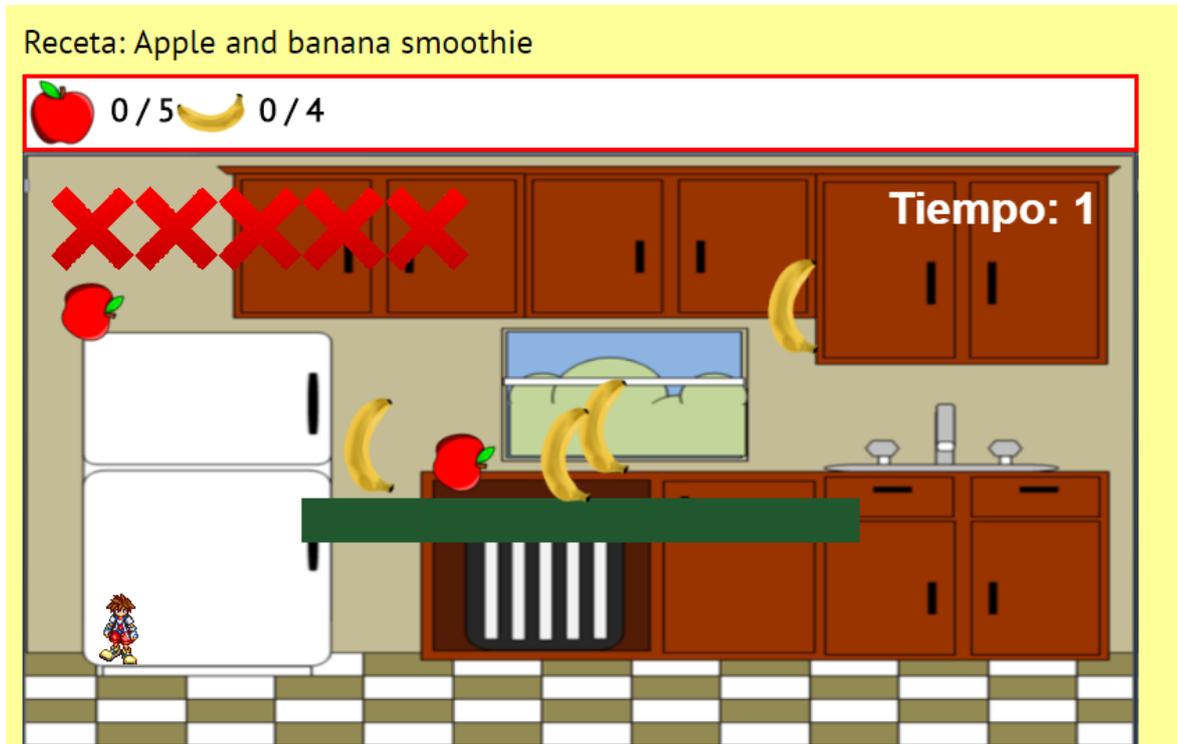


Figura 38. Sprint 2. Pantalla de juego de prototipo

A.3 Incremento del Sprint 4

Pantalla de bienvenida del prototipo: inicio (Figura 39).



Figura 39. Sprint 4. Pantalla de bienvenida de prototipo (1)

Pantalla de bienvenida de prototipo: explicación y capturas (Figura 40).



Figura 40. Sprint 4. Pantalla de bienvenida de prototipo (2)

Pantalla de bienvenida de prototipo: afectividad (Figura 41).



Figura 41. Sprint 4. Pantalla de bienvenida de prototipo (3)

Aviso legal (Figura 42).

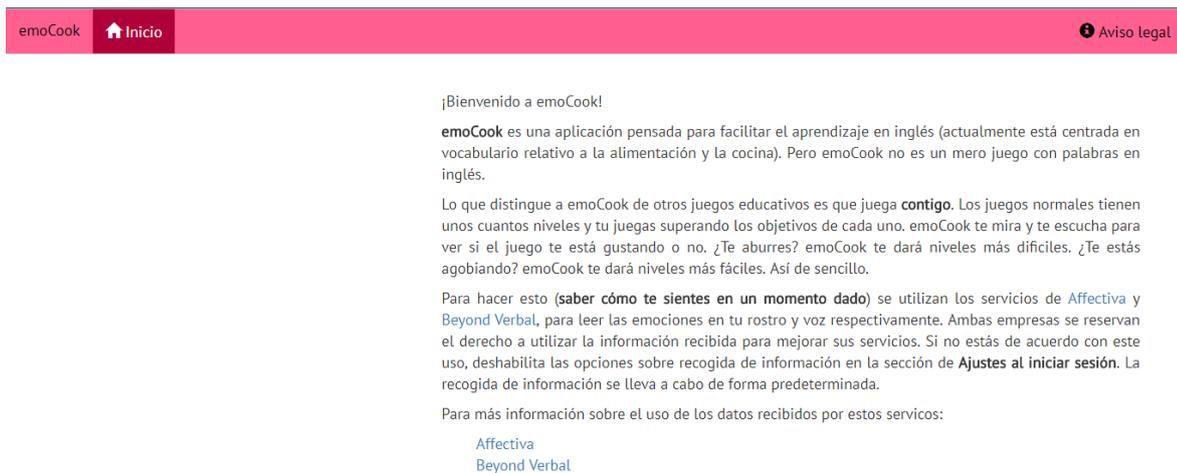
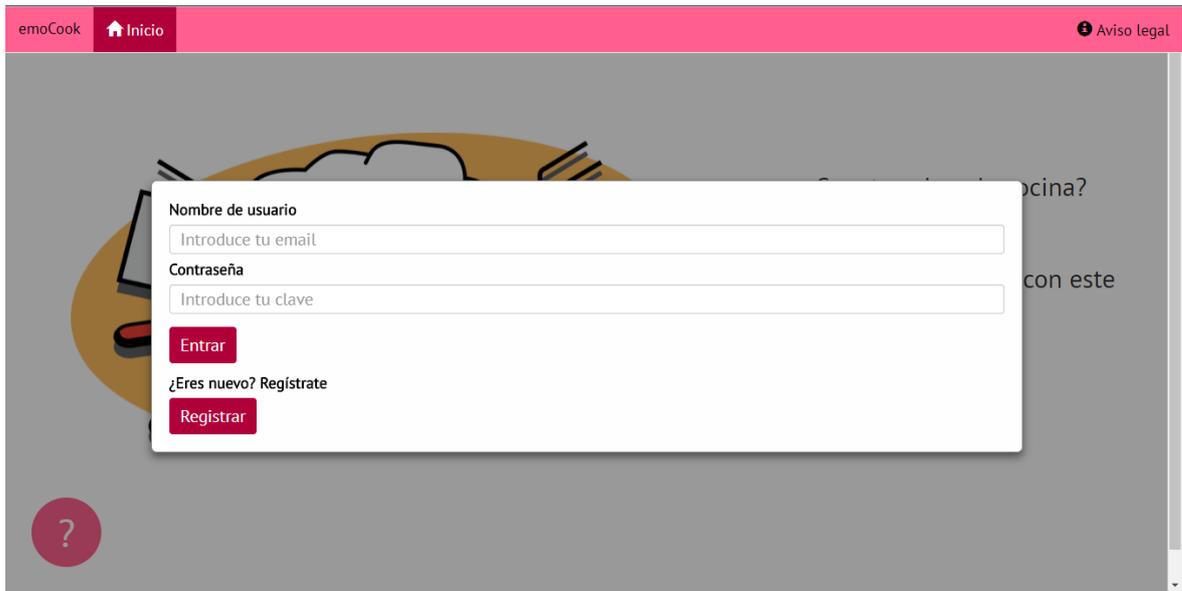


Figura 42. Sprint 4. Aviso legal

Pantalla de inicio de sesión (Figura 43).



emoCook Inicio Aviso legal

Nombre de usuario
Introduce tu email

Contraseña
Introduce tu clave

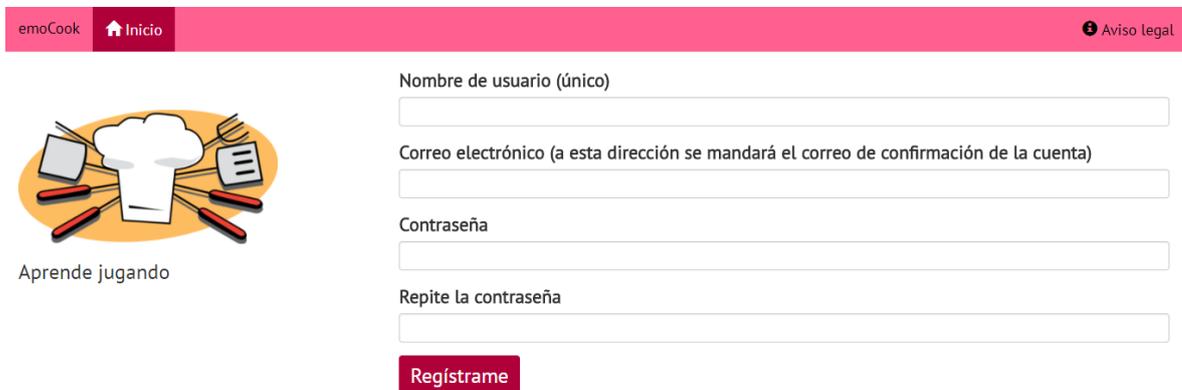
Entrar

¿Eres nuevo? Regístrate

Registrar

Figura 43. Sprint 4. Pantalla de inicio de sesión de prototipo

Pantalla de registro de usuarios (Figura 44).



emoCook Inicio Aviso legal

Aprende jugando

Nombre de usuario (único)

Correo electrónico (a esta dirección se mandará el correo de confirmación de la cuenta)

Contraseña

Repite la contraseña

Regístrate

Figura 44. Sprint 4. Pantalla de registro de usuario

Información de receta: nivel 1 (Figura 45).

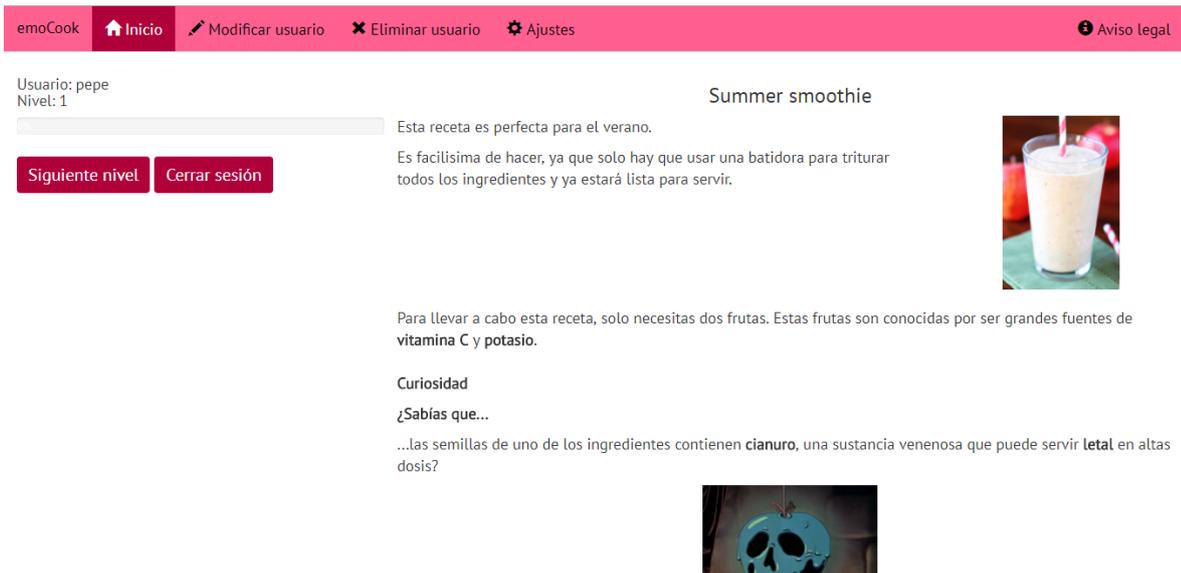


Figura 45. Sprint 4. Pantalla de información de receta – Nivel 1

Pantalla de juego del nivel 1

Se muestran distintas capturas para reflejar la dinamicidad de la selección de ingredientes no válidos. En la Figura 46, dicho ingrediente es leche. En la Figura 47, sandía, y en la Figura 48, zanahoria.



Figura 46. Sprint 4. Pantalla de juego – Nivel 1



Figura 47. Sprint 4. Pantalla de juego – Nivel 1



Figura 48. Sprint 4. Pantalla de juego – Nivel 1

Pantalla de fin de juego

Tras perder todas las vidas (Figura 49).



Figura 49. Sprint 4. Fin de juego (perdido) - Nivel 1

Tras recoger todos los ingredientes (Figura 50).

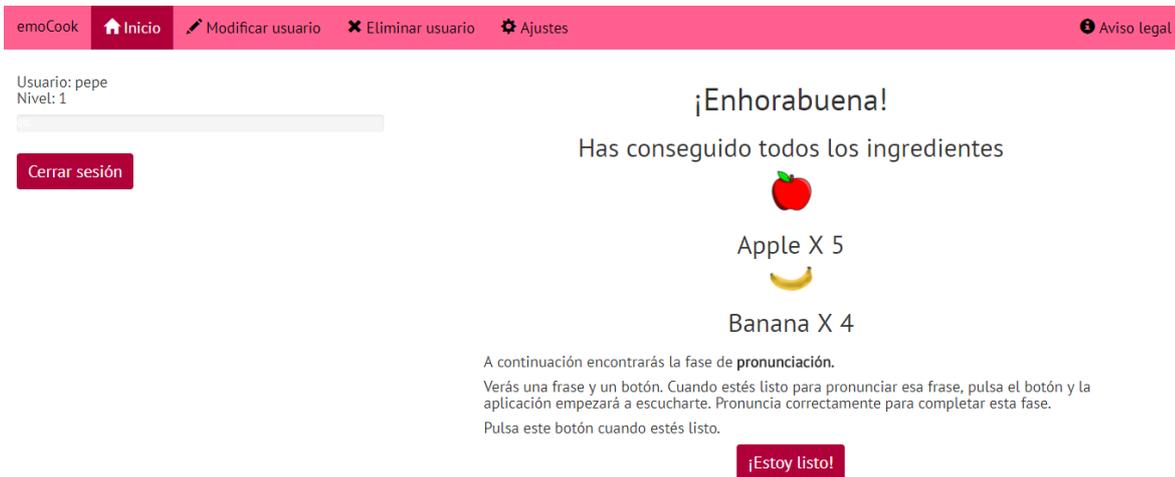


Figura 50. Sprint 4. Fin de juego (ganado) - Nivel 1

Fase de dictado: comienzo (Figura 51).



Figura 51. Sprint 4. Fase de dictado – Comienzo

Fase de dictado: aplicación escuchando al usuario (Figura 52).



Figura 52. Sprint 4. Fase de dictado - Aplicación escuchando

Fase de dictado: pronunciación correcta

En la Figura 53 podemos ver una imagen de la aplicación durante la animación que carga la siguiente frase tras haber pronunciado la frase actual correctamente.



Figura 53. Sprint 4. Fase de dictado - Frase correcta (animación)

Fase de dictado: primera pronunciación incorrecta (Figura 54).



Figura 54. Sprint 4. Fase de dictado - Pronunciación incorrecta

Fase de dictado: tercera pronunciación incorrecta

En la Figura 55 podemos ver una imagen de la aplicación durante la animación que carga la siguiente frase tras haber pronunciado la frase actual incorrectamente por tercera vez.



Figura 55. Sprint 4. Fase de dictado - Frase incorrecta tres veces (animación)

Pantalla de espera de resultados (Figura 56)



Figura 56. Sprint 4. Pantalla de espera de resultados

Resultados (Figura 57).

emoCook Inicio Modificar usuario Eliminar usuario Ajustes Aviso legal

Usuario: pepe
Nivel: 1

¡Enhorabuena!

Resultados

Nombre	Nivel	Tiempo
pepe	1	10

Siguiente receta

Figura 57. Sprint 4. Resultados

Información de receta: nivel 2 (Figura 58).

emoCook Inicio Modificar usuario Eliminar usuario Ajustes Aviso legal

Usuario: pepe
Nivel: 2

33%

Siguiente nivel Cerrar sesión

Yummy Ice-Cream

No hay nada mejor para refrescarse en verano que un buen helado.

Usando un poco de leche e imaginación, se pueden hacer helados de cualquier sabor: fresa, chocolate, vainilla, pistacho, chicle, yogur, nata, tiramisú, naranja, menta con chocolate, galleta, etc.



¡Reune los ingredientes para preparar un delicioso helado!

Figura 58. Sprint 4. Pantalla de información de receta - Nivel 2

Pantalla de modificar usuario (Figura 59).

emoCook Inicio Modificar usuario Eliminar usuario Ajustes Aviso legal

Usuario: pepe
Nivel: 2
33%

Siguiente nivel Cerrar sesión

Nombre de usuario (único)
pepe

Correo electrónico
pepe17tfg@mailinator.com

Contraseña

Repite la contraseña

Guardar cambios

Figura 59. Sprint 4. Pantalla de modificar usuario

Pantalla de eliminar usuario (Figura 60)

emoCook Inicio Modificar usuario Eliminar usuario Ajustes Aviso legal

Usuario: pepe
Nivel: 2
33%

Siguiente nivel Cerrar sesión

Confirma tus credenciales. Vas a eliminar tus datos

Correo electrónico

Contraseña

Eliminar credenciales

Figura 60. Sprint 4. Pantalla de eliminar usuario

Pantalla de ajustes (Figura 61).

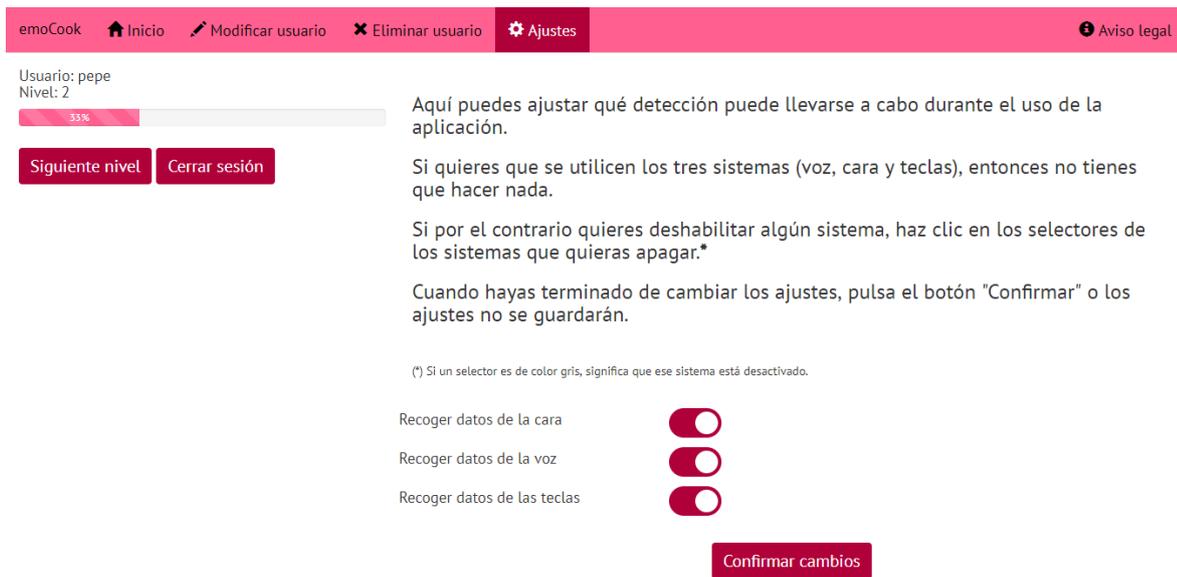


Figura 61. Sprint 4. Pantalla de ajustes

ANEXO B. DOCUMENTOS DE LA EVALUACIÓN

En este anexo se recogen los documentos que se utilizaron durante el proceso de evaluación según la ISO/IEC 25062. Estos documentos consisten en una hoja de instrucciones que se entregó a los usuarios, en la que se explicaban las tareas que estos tenían que hacer, y el cuestionario SUS que se les pidió que rellenasen después para evaluar la satisfacción. Además, dado que estos eran menores de edad, debían ser autorizados por sus padres, madres y/o tutores legales.

B.1 Tareas de la evaluación

En esta evaluación se pretende evaluar la usabilidad de la aplicación emoCook. La página web ya estará cargada al momento de empezar la evaluación, de manera que el usuario puede empezar directamente con la tarea 1. La evaluación consistirá en realizar dos tareas:

- **Navegación.** El usuario tendrá que moverse por la aplicación, accediendo a los distintos apartados de la página web y comprobando que entiende lo que se hace y/o se dice en ellos.
- **Jugabilidad.** El usuario jugará un nivel completo de la aplicación, para comprobar si la dificultad del juego es adecuada, si los controles son intuitivos, etc.

Dichas tareas se realizarán de forma paralela, y consisten en ejecutar los siguientes pasos.

1. Navega por la página. ¿En que consiste el juego?
2. Inicia sesión con este usuario
Usuario: **eval2** Contraseña: **eval2**
3. Accede a los distintos apartados de la barra superior, incluido el aviso legal. ¿Entiendes para qué es cada apartado?
4. Accede a los “Ajustes” y desactiva la detección de emociones en la voz.
5. Vuelve a la pantalla de información de la receta.
6. Empieza el primer nivel. ¿Sabes lo que tienes que hacer? Si se te acaban las vidas, vuelve a intentar el nivel.
7. Tras acabarlo, inicia la fase de dictado. Pulsa el botón de Ayuda (símbolo “?”) si necesitas consultar las reglas.
8. Durante el dictado, pronuncia una frase de forma incorrecta.
9. Cuando llegues a los resultados, pasa al siguiente nivel.
10. Tras leer la información de la siguiente receta, cierra sesión.

Ha concluido la evaluación, muchas gracias por participar.

B.2 Cuestionario SUS

Por favor, rellena la siguiente tabla en base a la aplicación que acabas de utilizar. Marca con una X tu respuesta a cada afirmación. Si estás de acuerdo con lo que se dice en una fila, marca con una X la columna de una cara sonriente. Si no estás de acuerdo, marca con una X alguna de las caras enfadadas. Si en alguna fila no estás seguro/a, marca el valor del centro. Si no te decides sobre dónde hacer la X, hazla en el primer sitio que hayas pensado.

	Escala de usabilidad (SUS)				
					
1. Creo que me gustaría utilizar esta aplicación con frecuencia.					
2. Creo que la aplicación es muy compleja.					
3. Creo que la aplicación es fácil de usar.					
4. Creo que no sabría usar la aplicación sin ayuda de un experto.					
5. Creo que las partes de la aplicación funcionan bien juntas.					
6. Creo que algunas partes de la aplicación quedan muy raras juntas.					
7. Creo que se tardaría muy poco en aprender a usar la aplicación.					
8. Creo que la aplicación es incómoda de usar.					
9. Me sentí muy a gusto usando la aplicación.					
10. Necesitaría aprender muchas cosas para poder usar la aplicación.					

Evaluación del cuestionario.

1	2	3	4	5	6	7	8	9	10	Total
- 1	5 - _	_ - 1	5 - _	_ - 1	5 - _	_ - 1	5 - _	_ - 1	5 - _	

Puntuación total: Total x 2.5 = _____

B.3 Autorización de la evaluación

Evaluación de emoCook

Evaluador: José María García García

Proyecto: Variación dinámica del comportamiento de las aplicaciones en función de las emociones del usuario

Yo, D/D^a _____,

Mayor de edad, con DNI _____ autorizo a

_____ como padre/madre

y/o tutor legal a participar en la evaluación del proyecto arriba mencionado.

Fecha: ____ / ____ / _____

Fdo.

ANEXO C. FORMATO DE RESPUESTA DE LOS SERVICIOS DE DETECCIÓN

En este Anexo se presenta el formato que tenían las respuestas de los servicios de detección de emociones. El funcionamiento de estos en la mayoría de los casos consistía en hacer una petición HTTP a la API del servicio, devolviendo esta un objeto JSON a modo de respuesta, en el que se indica si la petición era correcta, si se ha podido generar un resultado, etc.

C.1 Beyond Verbal

Junto con otros campos propios de la petición HTTP como tal, los resultados como tales devueltos por Beyond Verbal son los que pueden verse en la Figura 62.

```

Analysis: {
  Arousal: {
    Group: "low" – "medium" – "high",
    Value: 0 - 100
  }.
  Temper: {
    //Igual que Arousal
  },
  Valence: {
    //Igual que Arousal
  },
  AudioQuality: 0 – 100,
  Gender: "male" – "female" – "unkown",
  Mood: {
    Composite: {
      Primary: "...",
      Secondary: "..."
    },
    Group1: {
      Primary: "...",
      Secondary: "..."
    }
  }
}
    
```

Figura 62. Formato de resultados de Beyond Verbal

C.2 Vokaturi

El formato de los resultados que devuelve Vokaturi, cuyas peticiones se realizan directamente a la propia SDK, pueden verse en la Figura 63.

```
Analysis: {  
  Neutrality: 0 – 1,  
  Happiness: 0 – 1,  
  Sadness: 0 – 1,  
  Anger: 0 – 1,  
  Fear: 0 – 1  
}
```

Figura 63. Formato de resultados de Vokaturi

C.3 Emotion API

Los resultados de Emotion API consisten en una lista con tantos elementos como caras se hayan detectado, detallándose para cada una sus coordenadas dentro de la imagen y las emociones leídas en ella (Figura 64). Al igual que Vokaturi, los resultados de las emociones se expresan con números entre 0 y 1, siendo la suma de todos los valores igual a 1.

```
Analysis: [{  
  FaceRectangle: {  
    Top: "...",  
    Left: "...",  
    Width: "...",  
    Height: "..."  
  },  
  Scores: {  
    Anger: 0 - 1,  
    Contempt: 0 - 1,  
    Disgust: 0 - 1,  
    Fear: 0 - 1,  
    Happiness: 0 - 1,  
    Neutral: 0 - 1,  
    Sadness: 0 - 1,  
    Surprise: 0 - 1  
  }  
},  
{ ... }  
]
```

Figura 64. Formato de resultados de Emotion API

C.4 Afectiva

```

Analysis: [{
  Timestamp: 1,
  Appearance: {
    Gender: "Male" – "Female" – "Unknown"
    Glasses: "Yes" – "No"
    Age: "18 – 24", "25 – 34" ...
    Ethnicity: "..."
  }, Emotion: {
    Joy: 0 – 100,
    Sadness: 0 – 100,
    Disgust: 0 – 100,
    Contempt: 0 – 100,
    Anger: 0 – 100,
    Fear: 0 – 100,
    Surprise: 0 – 100,
    Valence: -100 – 100,
    Engagement: 0 – 100
  }, Expressions: {
    Smile: 0 – 100,
    Inner Brow Raise: 0 – 100,
    Brow Raise: 0 – 100,
    Brow Furrow: 0 – 100,
    Nose Wrinkle: 0 – 100,
    Upper Lip Raise: 0 – 100,
    Lip Corner Depressor: 0 – 100,
    Chin Raise: 0 – 100,
    Lip Pucker: 0 – 100,
    Lip Press: 0 – 100,
    Lip Suck: 0 – 100,
    Mouth Open: 0 – 100,
    Smirk: 0 – 100,
    Eye Closure: 0 – 100,
    Attention: 0 – 100,
    Lid Tighten: 0 – 100,
    Jaw Drop: 0 – 100,
    Dimpler: 0 – 100,
    Eye Widen: 0 – 100,
    Cheek Raise: 0 – 100,
    Lips Stretch: 0 – 100
  }
}]

```

Figura 65. Formato de resultados de Afectiva

C.5 Kairos

```

Analysis: [{
  Time: t,
  People: [{
    Person_id: ...,
    Demographics: {
      Age_group: "...",
      Gender: "...",
    }, Appearance: {
      Glasses: "Yes" – "No"
    }, Emotions: {
      Anger: 0 – 100,
      Disgust: 0 – 100,
      Fear: 0 – 100,
      Joy: 0 – 100,
      Sadness: 0 – 100,
      Surprise: 0 – 100
    }, Tracking: {
      Glances: "...",
      Dwell: "...",
      Attention: 0 – 100,
      Blink: "Yes" – "No"
    }, Pose: {
      Yaw: "...",
      Pitch: "...",
      Roll: "...",
    }, Face: {
      X: "...",
      Y: "...",
      Width: "...",
      Height: "...",
    }, Distance: "...",
  ]
},
{
  Time: t + 0.033,
  People: [...]}
]

```

Figura 66. Formato de resultados de Kairos

ANEXO D. DIAGRAMAS UML GENERADOS

En este anexo se recogen los artefactos UML que se han generado durante el desarrollo del prototipo. En este anexo pueden encontrarse los siguientes artefactos:

- Diagramas de clases.
- Diagrama de actividad. Se presenta el flujo de tareas que siguen los usuarios de la aplicación.
- Diagrama de estados. Se ha utilizado un diagrama de estados para modelar la dificultad del juego.
- Diagrama de paquetes.
- Diagrama de despliegue.
- Diagrama de navegación.

D.1 Diagramas de clases

Durante las fases iniciales de desarrollo se elaboró un diagrama de clases para reflejar el modelo conceptual de la aplicación (Figura 67).

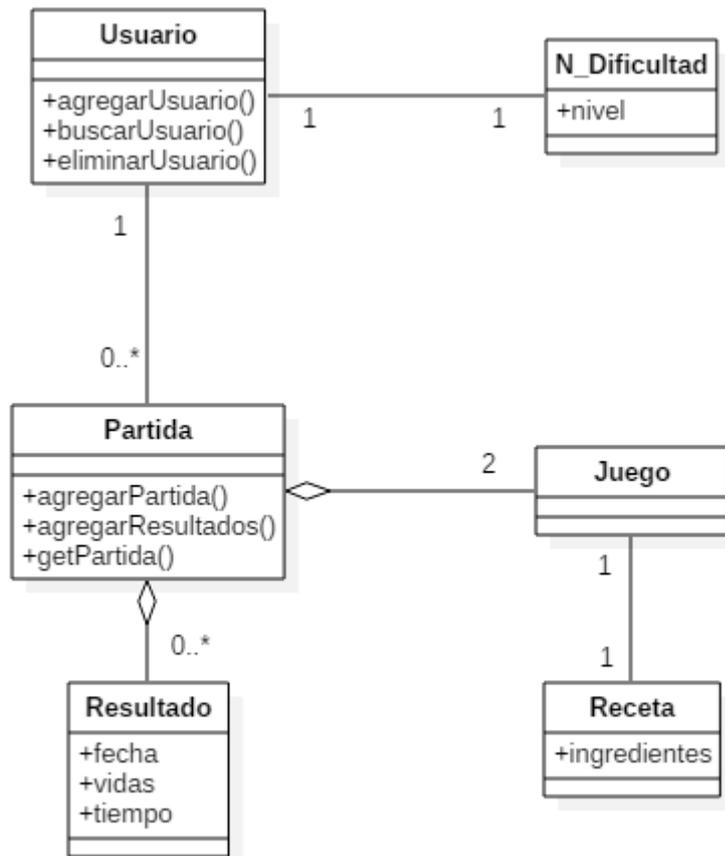


Figura 67. Diagrama de clases (modelo conceptual)

Posteriormente se elaboró un diagrama de clases para reflejar la implementación de la aplicación (Figura 68).

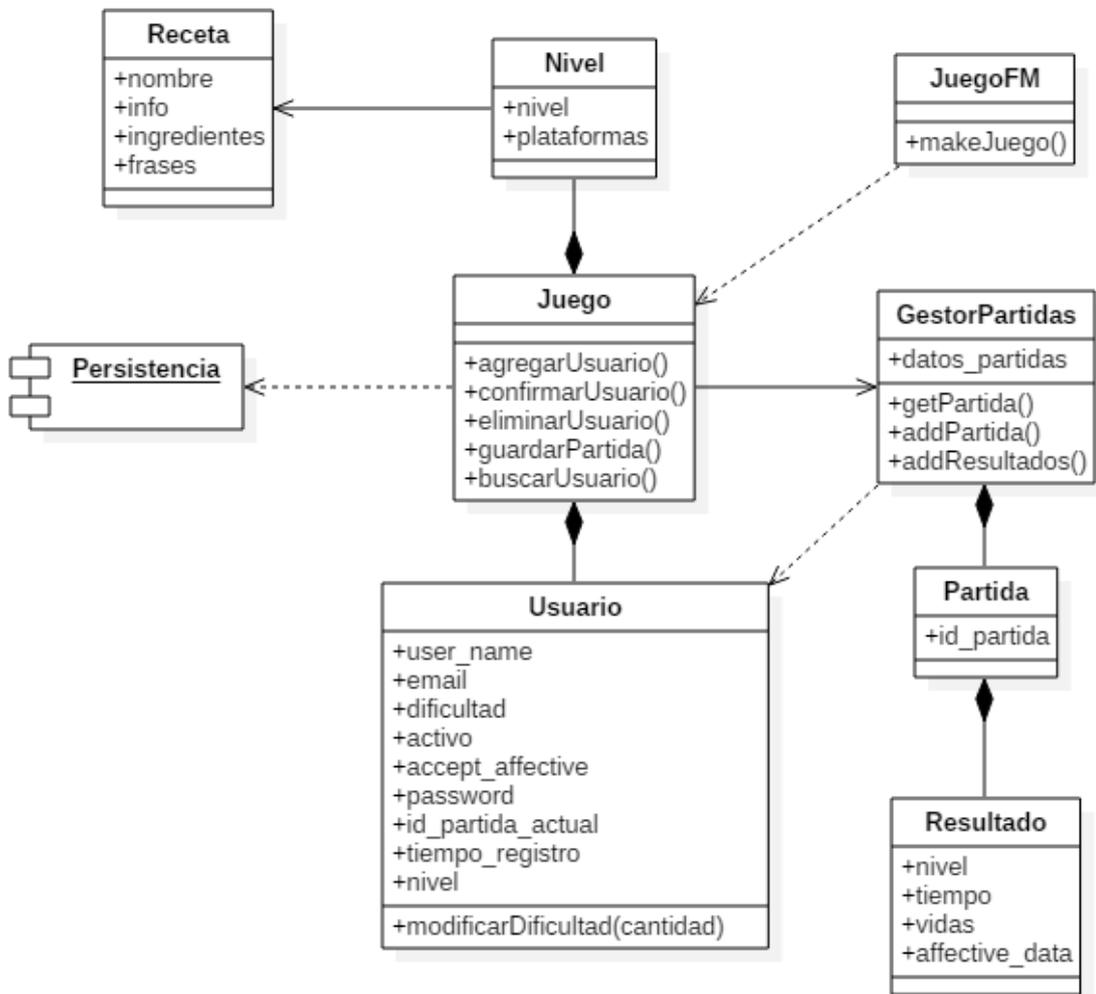


Figura 68. Diagrama de clases definitivo

D.2 Diagramas de actividad

El flujo de navegación¹⁵ de la aplicación se inicia cuando el usuario abre la aplicación. Para empezar a usarla, el usuario ha de iniciar sesión, usando sus credenciales para autenticarse. Si introduce sus credenciales correctamente, se muestran los datos del jugador (nivel, nombre, progreso en el juego) y la información de la receta que venga a continuación. En caso contrario, se mantiene al usuario en el formulario de inicio de sesión, indicando que las credenciales que ha usado son incorrectas. En la Figura 69 se modela el comportamiento de una partida completa.

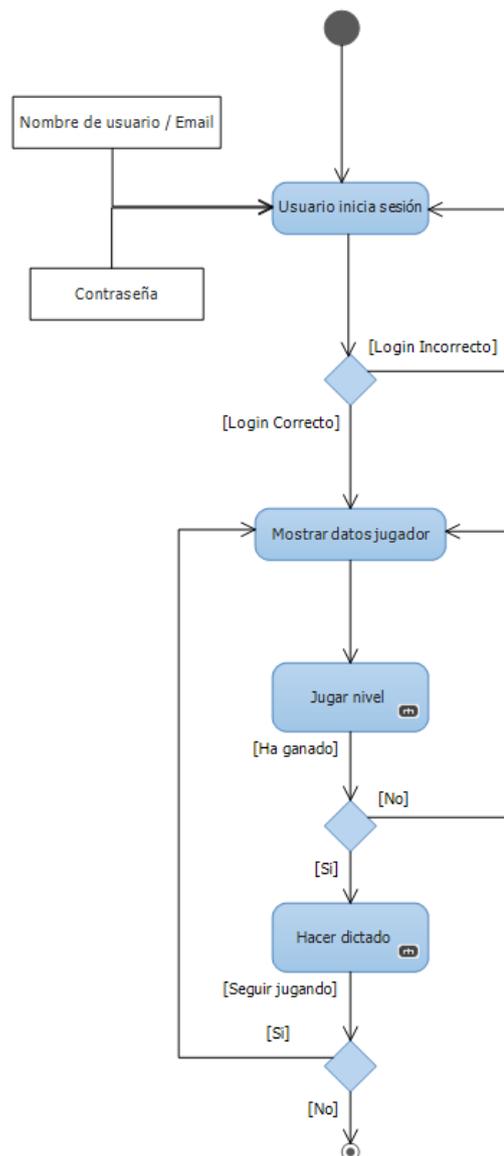


Figura 69. Diagrama de actividad (partida completa)

¹⁵ Solo se ha representado el flujo de actividades de jugar al juego, dejándose de lado las actividades de creación, modificación y eliminación de usuarios.

Tras haber iniciado sesión, el siguiente paso que daría el usuario sería comenzar a jugar el nivel en el que se encuentre. Las actividades realizadas dentro de dicho nivel quedan reflejadas en el diagrama de la Figura 70. El usuario controlará al personaje del juego haciendo uso de las flechas del teclado o de las letras AWD, recogiendo ingredientes válidos y evitando los no válidos. Si alcanza el objetivo del nivel (recoger todos los ingredientes) o pierde todas las vidas, el juego termina. En el caso de haber perdido todas las vidas, se vuelve a la pantalla con información sobre la receta, mientras que si se recogen todos los ingredientes se pasa a la fase de dictado (Figura 69). A su vez, el dictado tiene asociado el flujo de actividades que se detalla en la Figura 71.

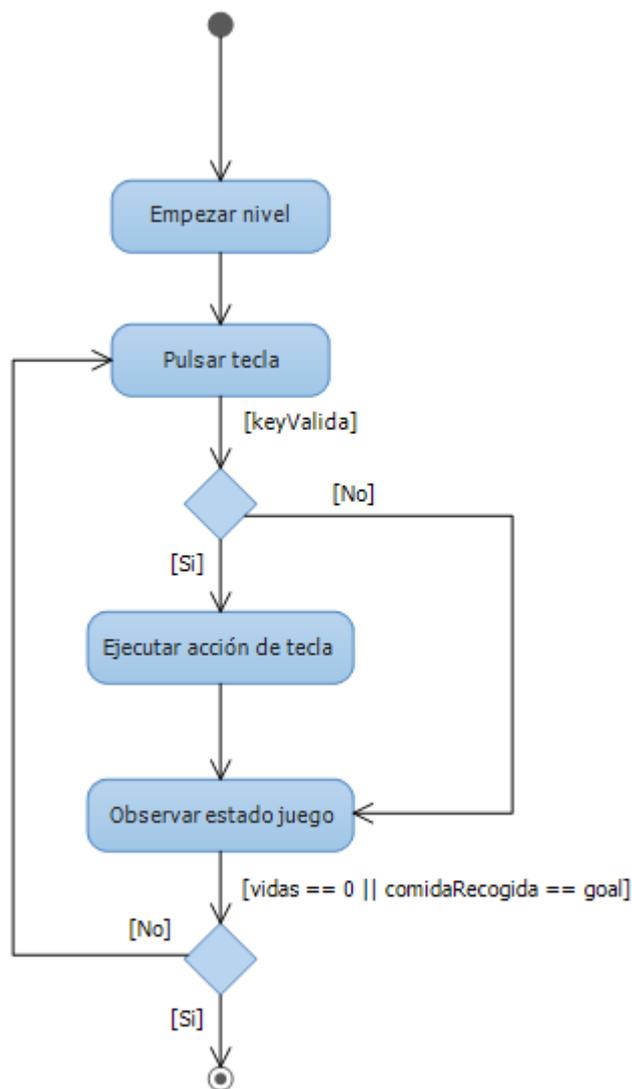


Figura 70. Diagrama de actividad (juego)

Cuando empieza el dictado, se muestra una frase al usuario junto con un botón para iniciar el dictado (Figura 51). Cuando el usuario pulsa el botón, el reconocedor de voz se activa y empieza a escuchar al usuario. Si el usuario pronuncia la frase correctamente, o la pronuncia incorrectamente tres veces, se pasará a la siguiente frase.

Esta secuencia de fases se repite hasta que se llega a la última frase, momento en el cual se nos muestran los resultados y se nos lleva hasta la pantalla de información de la siguiente receta.

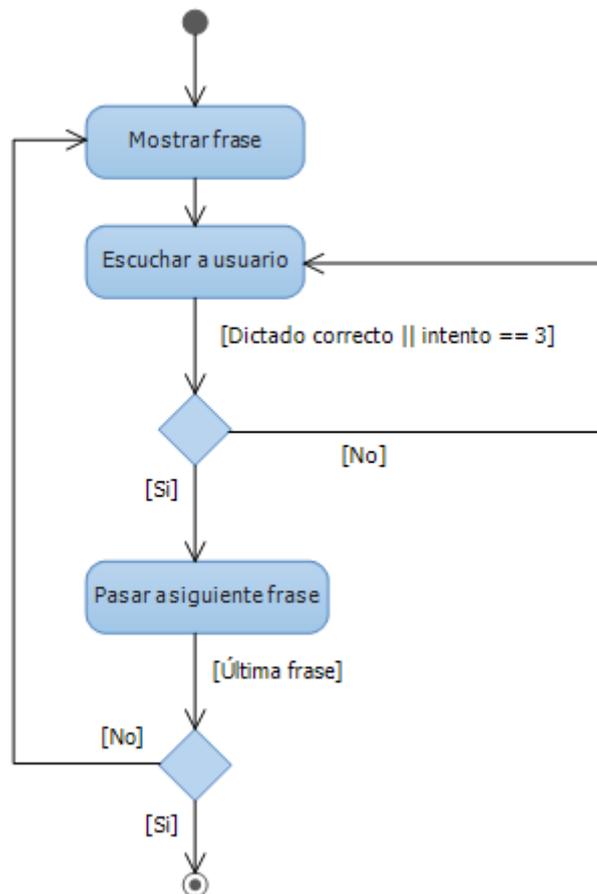


Figura 71. Diagrama de actividad (dictado)

D.3 Diagrama de estados

Tal y como se comentó durante el sprint correspondiente, la dificultad queda definida por una tupla de tres valores (X, Y, Z), pudiendo tomar cada otros tres valores, lo que nos deja con un conjunto de 27 tuplas, esto es, 27 niveles de dificultad. Cuando se detectan indicios de que el usuario se está aburriendo, se pasa al nivel de dificultad siguiente (más difícil), mientras que si se detecta que el usuario se está estresando pasamos al nivel anterior (más fácil). De esta manera, la dificultad queda modelada tal y como se refleja en el diagrama de estados de la Figura 72.

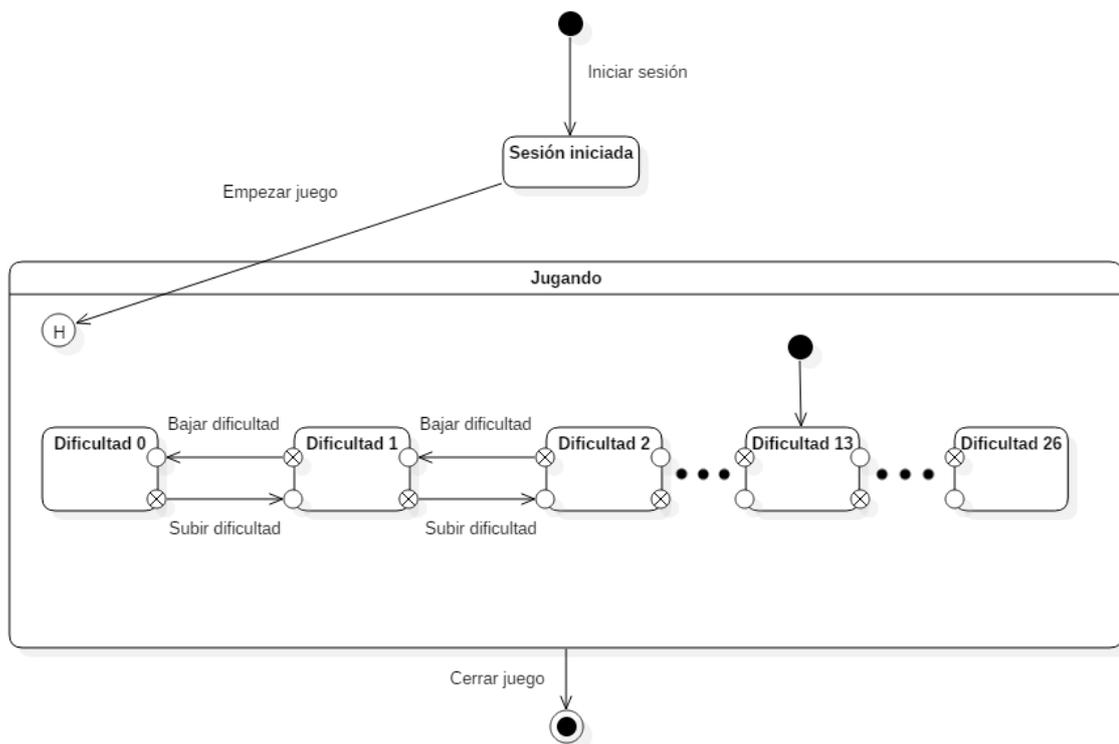


Figura 72. Diagrama de estados (dificultad)

D.4 Diagrama de Componentes

En este diagrama (Figura 73) se reflejan los paquetes en los que está dividida la aplicación:

- **Ciente.** Este paquete contiene todos los recursos y librerías relacionados con el lado del cliente de la aplicación. Esto abarca todos los aspectos de la interfaz, así como las librerías necesarias para llevar a cabo la detección de emociones en cara y voz, las librerías para grabar el sonido del navegador, las librerías de Phaser para construir el juego, etc.
- **Servidor.** El paquete del servidor contiene los ficheros del servidor en sí, así como las librerías para mantener los datos del servidor y para comunicarse con la base de datos.

Se han reflejado también los paquetes que representan los servicios de Afectiva, Beyond Verbal y mLab.

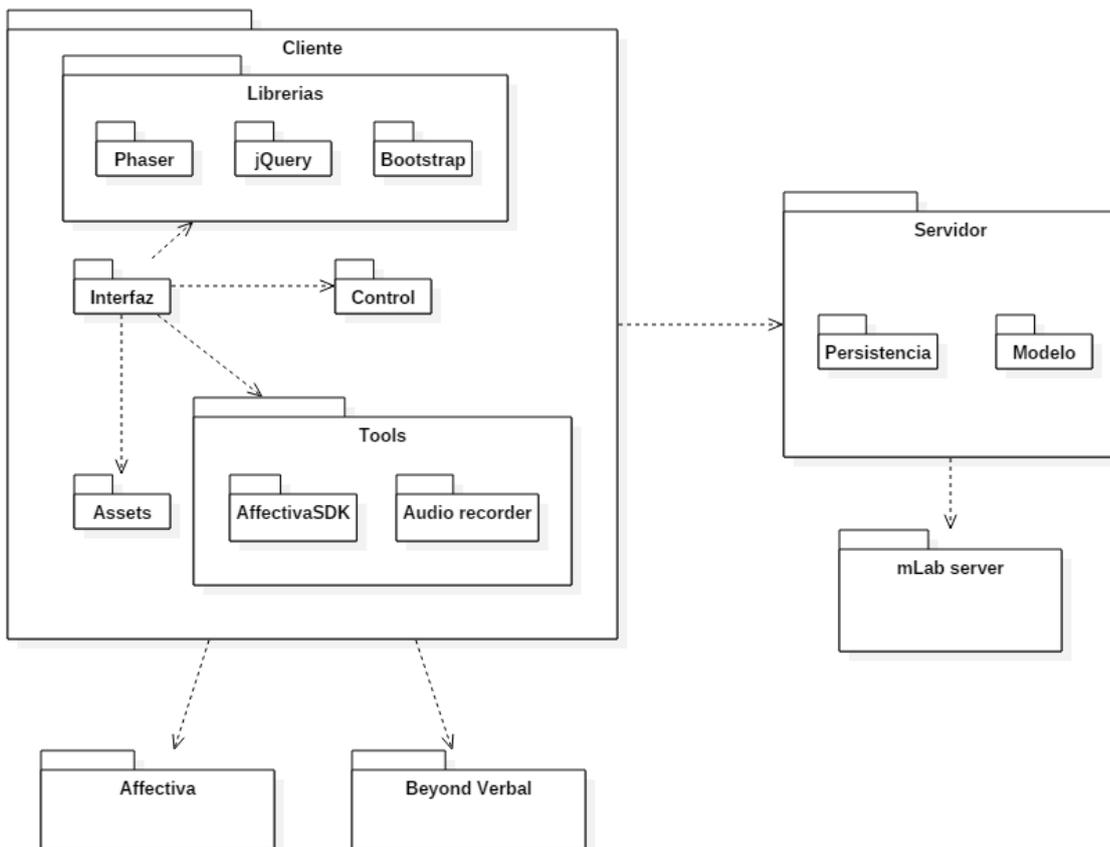


Figura 73. Diagrama de Componentes

D.5 Diagrama de despliegue

En el diagrama de despliegue de la Figura 74 se reflejan los elementos que intervienen durante la ejecución de la aplicación, así como la comunicación entre los mismos. Cabe destacar los siguientes aspectos.

- La información de los jugadores, los niveles, las partidas y las puntuaciones se almacena en el modelo, mientras que se vuelca en la base de datos a través de “Persistencia” para poder recuperar la información cuando el servidor se actualiza o se reinicia.
- Aunque existen más artefactos en el Cliente, el artefacto Proxy es el encargado de coordinarlos, así como de comunicarse con el servidor.

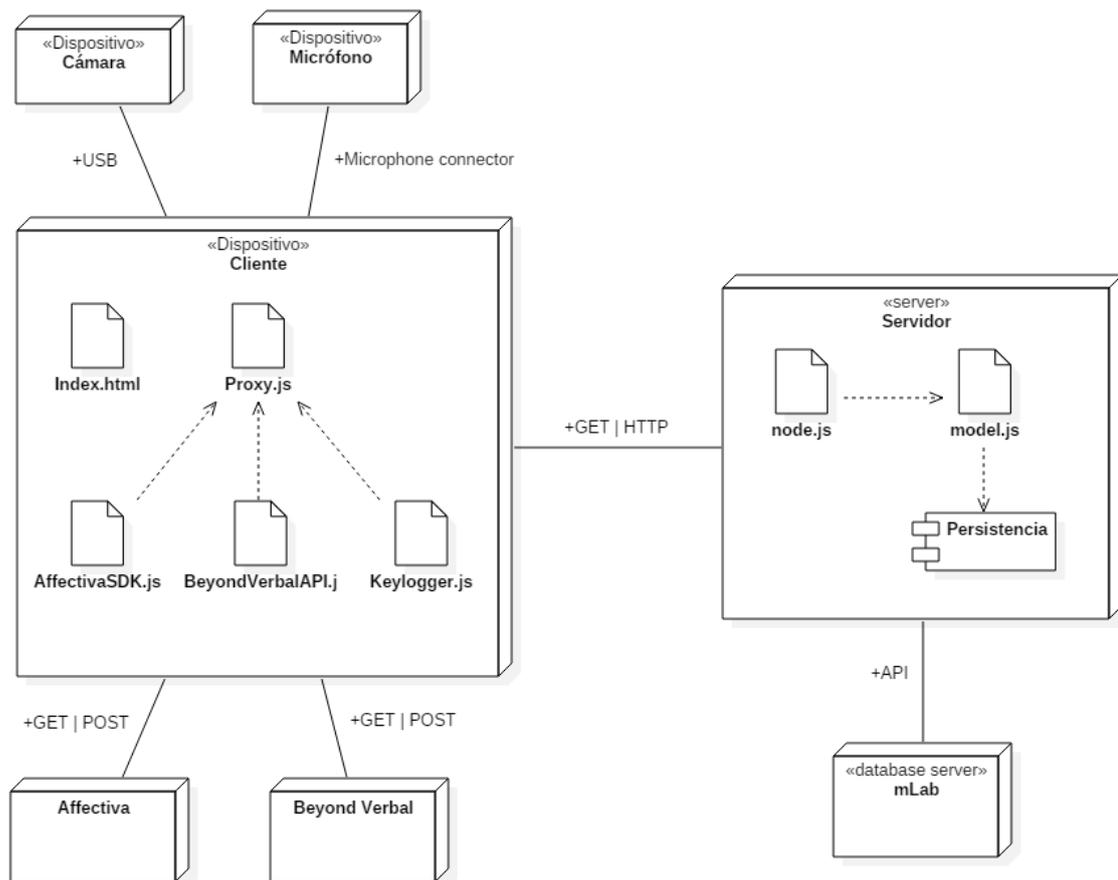


Figura 74. Diagrama de despliegue

D.6 Diagrama de navegación

En la Figura 75 podemos apreciar el diagrama de navegación de la aplicación. Los distintos sectores de la misma se han separado por colores, mostrándose en amarillo la zona pública que pueden ver todos los usuarios, y en verde la zona a la que solo pueden acceder los usuarios registrados mediante el inicio de sesión. Esto deja visibles otras opciones que antes estaban ocultas.

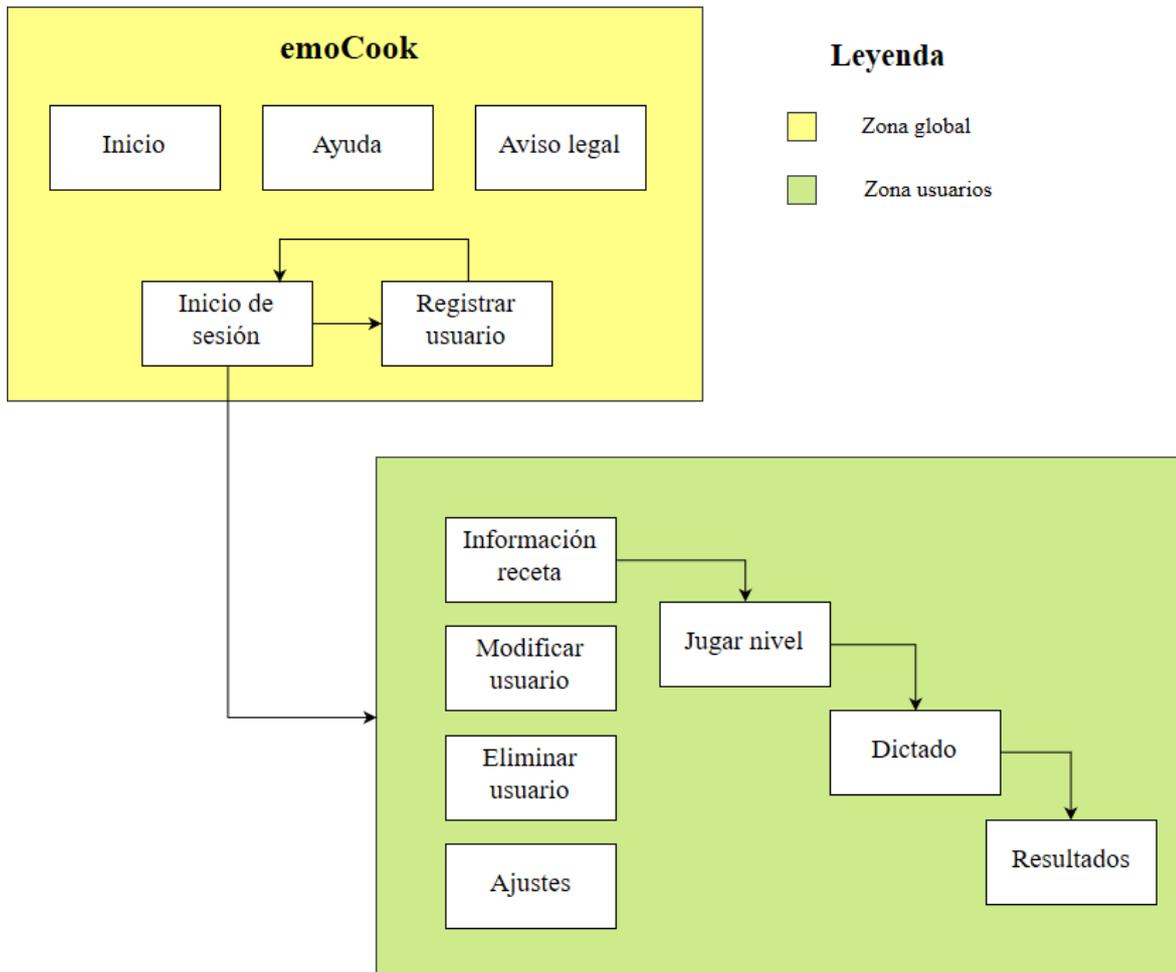


Figura 75. Diagrama de navegación