



SCRUM FINAL

Grupo WHO

Alba Bautista Núñez
Natalia Rodríguez Caballero
Beñat Pérez de Arenaza Eizaguirre
Javier Amado Lázaró
Mario Calvarro Marines
Lucía Alonso Mozo

ÍNDICE

Introducción	4
Estructura y funcionamiento del equipo scrum	5
Historias de usuario	9
1.Como usuario quiero jugar al UNO	9
2.Como usuario quiero poder elegir el número de jugadores a participar	10
3.Como usuario quiero jugar con las cartas:	10
3.1 Normales:	10
3.2 Especiales:	10
4.Como usuario quiero que las cartas se comporten adecuadamente	11
5.Como usuario quiero conocer los nombres de los otros jugadores de la partida	11
6.Como usuario quiero poder guardar una partida en curso	12
7.Como usuario quiero poder cargar una partida que he guardado con anterioridad	12
8.Como usuario quiero poder introducir mediante teclado un comando y que se ejecute en la partida	12
9.Como usuario quiero poder jugar contra la máquina en diferentes dificultades	13
Fácil	13
Medio	13
Difícil	13
10.Como usuario quiero jugar una partida en red	15
Sprint Reviews	16
Sprint Review 1	17
Sprint Review 2	18
Sprint Review 3	20
Sprint Review 4	22
Sprint Review 5	24
Sprint Review 6	26
Sprint Review 7	28
Resumen Sprint Reviews	30
Conclusión final	37
Sprint Retrospectives	38
Sprint Retrospective 1	39
Sprint Retrospective 2	40
Sprint Retrospective 3	41
Sprint Retrospective 4	42
Sprint Retrospective 5	43
Sprint Retrospective 6	44
Sprint Retrospective 7	45
Resumen Sprint Retrospectives	46
Sprint plannings	49
Sprint Planning 1	50

Sprint Planning 2	51
Sprint Planning 3	52
Sprint Planning 4	53
Sprint Planning 5	54
Sprint Planning 6	55
Sprint Planning 7	56
Resumen Sprint Plannings	57
Product Backlog	61
Evolución del Product Backlog	63
Sprint 1	63
Sprint 2	64
Sprint 3	65
Sprint 4	66
Sprint 5	67
Sprint 6	69
Sprint 7	70
Sprint Backlog	71
Sprint Backlog 1	72
Sprint Backlog 2	73
Sprint Backlog 3	74
Sprint Backlog 4	75
Sprint Backlog 5	76
Sprint Backlog 6	77
Sprint Backlog 7	78
Trabajo realizado por cada componente del equipo	79
Mario Calvarro Marines:	79
Lucía Alonso Mozo:	79
Natalia Rodríguez Caballero:	79
Javier Amado Lázaro:	80
Alba Bautista Nuñez:	80
Beñat Perez de Arenaza Eizaguirre:	80
Incidencias	82

Introducción

En este documento vamos a presentar la documentación que hemos ido desarrollando durante todo el proyecto.

Lo dividiremos en varios bloques principales que serán:

- Estructura y funcionamiento del equipo Scrum
- Historias de usuario
- Sprint Reviews
- Sprint Retrospectives
- Sprint Plannings
- Product Backlog
- Sprint Backlog
- Trabajo realizado por cada miembro del grupo

En la mayoría de estos apartados, primero daremos una breve definición teórica de cada uno de ellos, y luego desarrollaremos la documentación que hemos realizado a lo largo de todos los Sprints del proyecto.

Estructura y funcionamiento del equipo scrum

Uno de los requisitos que se nos impusieron al principio del proyecto fue el de seguir Scrum.

Scrum es un método ágil para el desarrollo del software basado en tres pilares fundamentales:

- Transparencia: debe haber una alta visibilidad de las partes importantes del proyecto para todos aquellos que están involucrados en el desarrollo del mismo.
- Inspección: los usuarios de Scrum deben revisar los artefactos de Scrum para evitar que se produzcan cambios no deseados.
- Adaptación: en caso de detectar los cambios indeseados del apartado anterior, el proceso de desarrollo debe ajustarse.

Esta metodología divide los equipos en tres roles:

- **Product Owner**: debe ser la única persona responsable de gestionar el Product Backlog y responsable de maximizar el valor del producto del trabajo del Equipo de Desarrollo.
- **Scrum Master**: es el responsable de que Scrum se utilice correctamente. Se encarga de organizar y ayudar al equipo de desarrollo.
- **Equipo de Desarrollo**: está formado por un grupo de profesionales encargados de realizar el trabajo que compone el Incremento a entregar al final de cada Sprint.

En nuestro caso, hemos decidido repartir los roles de la siguiente manera:

- **Product Owner**: Mario Calvarro Marines
- **Scrum Master**: Lucía Alonso Mozo
- **Equipo de Desarrollo**: Alba Bautista Núñez, Natalia Rodríguez Caballero, Beñat Pérez de Arenaza Eizaguirre, Javier Amado Lázaro, Mario Calvarro Marines y Lucía Alonso Mozo.

Como se puede observar, el Product Owner y el Scrum Master pertenecen también al Equipo de Desarrollo.

Vamos a hacer un resumen de la manera en la que nos hemos **organizado** a lo largo de un Sprint:

Al comienzo del Sprint, se produce el Sprint Planning, que es la reunión que utilizamos para fijar los objetivos que queremos alcanzar en ese incremento (Todo esto se recogería en la documentación del Sprint Planning).

Una vez decididos los objetivos del Sprint, redactamos y añadimos al Product Backlog las nuevas historias de usuario propuestas para este Sprint.

A continuación, se crea el Sprint Backlog, que se trata básicamente del reparto de tareas. Siempre dividimos las tareas en varios bloques, es decir, algunos integrantes del grupo se encargan del código, otros de la creación de diagramas UML y otros de la documentación.

Una vez hecho esto, entonces comienzan las tareas del Sprint. Durante las dos semanas que dura el Sprint, cada miembro del grupo es responsable de las tareas que se le han adjudicado.

Para mantener una comunicación fluida en todo momento hemos utilizado las Daily Scrums, que son reuniones diarias en las que cada integrante del grupo explica qué progreso ha realizado en sus tareas en ese día. Como es difícil que todos los miembros del grupo puedan coincidir a la misma hora para hacer una reunión, también hemos usado la herramienta Whatsapp, en la que tenemos un grupo donde cada miembro del equipo comenta lo que considera oportuno acerca del progreso en sus tareas, si necesita algún tipo de ayuda al realizar alguna de sus tareas, si no entiende algo...

Una vez que llegamos al final del Sprint, primero realizamos el Sprint Retrospective. Se trata de una reunión en la que el equipo hace una autoevaluación del Sprint, enfocándose sobre todo en aspectos como la organización del equipo durante esas semanas, la comunicación entre los integrantes del grupo, las mejoras a realizar de cara a Sprints futuros...

A continuación realizamos lo que se llama Sprint Review, que es una reunión en la que se exponen los avances que se han hecho a lo largo de todo el Sprint. En esta reunión vemos si se han alcanzado los objetivos fijados al comienzo del Sprint.

Todo lo comentado tanto en el Sprint Retrospective como en el Sprint Review vendrá recogido en la documentación del Sprint.

Una vez mostrado cómo han sido nuestros Sprints en cuanto a organización, vamos a reflejar los **problemas** que nos han surgido:

- Desconocimiento de la metodología Scrum: sobre todo durante el primer Sprint, nos requirió mucho tiempo entender cómo funcionaba la metodología Scrum (para qué servían cada una de las reuniones (Sprint Planning, Sprint Review...), cuál eran las funciones del Product Owner y del Scrum Master...).
- Desconocimiento de la herramienta GitHub: al igual que nos pasó con la metodología Scrum, nos ocurrió lo mismo con GitHub. Al principio nos resultó difícil adaptarnos a esta herramienta para poder explotar al máximo todas sus funcionalidades.
- Falta de coordinación y comunicación: debemos reconocer que en algunas etapas del proyecto nos ha faltado algo de coordinación y comunicación a la hora de realizar algunas tareas. Esto ha producido que, en algunas ocasiones, hayamos tenido que repetir la tarea ya que en un principio no estaba bien hecha debido a estos factores que hemos comentado.

Gracias a la gran autocrítica que ha tenido el grupo durante todo el proyecto hemos conseguido solucionar estos problemas.

En el caso del desconocimiento de Scrum y Git, nos hemos informado para hacer un correcto uso de ambas.

Además, hemos convocado reuniones de manera más recurrente para mejorar la comunicación y la coordinación del grupo. A su vez, hemos sido cada vez más pacientes y calmados a la hora de realizar algunas tareas, cosa que en algunos momentos nos ha faltado por el simple hecho de querer cumplir todos los objetivos programados para el Sprint.

Al igual que hemos reflejado los principales problemas que hemos tenido durante el proyecto, vamos a desarrollar un par de aspectos que creemos que nos han servido de gran ayuda a lo largo de todo el proyecto.

En primer lugar, creemos que desde un primer momento, hemos sabido distribuir las tareas de la mejor manera posible, asignando a cada integrante del equipo la actividad que más potenciaba sus habilidades.

Por otro lado, hemos calculado los incrementos u objetivos de cada Sprint de una manera conservadora y segura. Es decir, en ningún Sprint nos hemos propuesto una cantidad de objetivos muy grande que no podríamos haber cumplido. Gracias a esto, en todos los Sprints hemos conseguido alcanzar todos los objetivos propuestos.

Por último, y a modo de conclusión final, cabe destacar el gran esfuerzo que han realizado cada uno de los integrantes de este grupo durante todo el proyecto, así como su buena actitud y predisposición para realizar cualquier tarea propuesta.

Las **herramientas** que hemos usado para la realización del proyecto han sido:

- Whatsapp: creamos un grupo de Whatsapp al comienzo del proyecto e incluimos a todos los integrantes del grupo en él. Nos ha servido para que cada miembro del grupo pueda informar de los avances que va consiguiendo a lo largo del Sprint, pueda pedir ayuda o consultar alguna duda con el resto.
- GitHub: aunque como ya hemos comentado anteriormente, al comienzo del proyecto nos costó aprender su correcto funcionamiento y uso, esta herramienta nos ha ayudado mucho a la hora de coordinarnos, ya que todos los miembros del grupo tenían acceso a cualquier tarea que estuviese haciendo otro integrante del grupo. Además, en caso de producirse algún problema con el código, podíamos recuperar una versión anterior del mismo y volver a empezar a trabajar desde ahí.
- Discord: hemos usado esta aplicación para concertar y realizar reuniones. También nos ha facilitado el trabajo en grupos.
- Google Docs: mediante esta herramienta hemos realizado la documentación de la entrega final. Nos ha servido de gran ayuda, ya que permite que varias personas puedan escribir en el mismo documento a la vez.
- Hojas de cálculo de Google: nos ha servido para crear las tablas que organizaban las tareas que debía hacer cada integrante y el estado de las mismas. Estas tablas vienen recogidas en el Sprint Backlog.
- Modelio: mediante esta aplicación hemos realizado nuestros diagramas de secuencias y de clases (diagramas UML).
- Eclipse: hemos desarrollado el código en esta aplicación.

Historias de usuario

1.Como usuario quiero jugar al UNO

Una partida consiste en una sucesión de turnos en los que cada jugador (entre 2 y 4) deberá hacer una de estas acciones:

Lanzar una carta, la cual deberá coincidir en color o en número con la que se encuentra situada en el centro del tablero, o bien, esta podrá ser una de las cartas especiales de color negro (cambio de color o +4). Robar una carta, únicamente en caso de no tener posibilidad de lanzar una carta de su mano, el jugador robará del montón de cartas central hasta que tenga posibilidad de echar.

Durante cualquier momento durante la partida, el jugador tendrá la posibilidad de salir de esta, bien guardándola, o perdiendo los datos registrados. El funcionamiento y las reglas del juego son las clásicas. La finalidad del juego es llegar a quedarse sin cartas en la mano, teniendo que pulsar el botón “uno” cuando solo dispongas de una carta, en caso de pasar de turno sin pulsarlo, se robará un total de 6 cartas, para volver a la situación inicial como castigo.

- Estimación: Duración del proyecto.
- Priorización: Debe tenerlo.
- Criterios de aceptación: El juego funciona correctamente.

2.Como usuario quiero poder elegir el número de jugadores a participar

Se muestra al usuario un menú, donde podrá elegir el número de usuarios que participarán en el juego durante la partida, entre 2 y 4 jugadores.

- Estimación: Se estima que en realizar esta historia de usuario se tardará 1 semana.
- Priorización: Debe tenerlo.
- Criterios de aceptación: El usuario puede elegir el número de jugadores dentro de un margen.

3.Como usuario quiero jugar con las cartas:

3.1 Normales:

Las cartas normales serán números del 0 al 9, con sus respectivos colores (amarillo, azul, rojo, verde). Habrá dos cartas con el mismo número y color.

- Estimación: Se estima que en realizar esta historia se tardará 1 semana.
- Priorización: Debe tenerlo.
- Criterios de aceptación: Cada carta tiene su color, número.

3.2 Especiales:

Las cartas especiales están formadas por: cartas con color: prohibido, cambio de sentido, +2; cartas sin color: +4, cambio de color.

- Estimación: Se estima que en realizar esta historia se tardará 1 semana.
- Priorización: Debería tenerlo.
- Criterios de aceptación: Cada carta tiene su color (si tiene), y funcionalidad

4.Como usuario quiero que las cartas se comporten adecuadamente

Al tirar una carta esta debe desaparecer de mis cartas, añadirse al montón conjunto, y ocurrir el efecto correspondiente.

- Estimación: Se estima que en realizar esta historia de usuario se tardará 2 semanas.
- Priorización: Debe tenerlo.
- Criterios de aceptación: La carta desaparezca realmente, no solo visualmente. Además, que se aprecie que el montón de cartas individuales ha disminuido, y el conjunto ha aumentado, es decir, el efecto sea el correcto.

5.Como usuario quiero conocer los nombres de los otros jugadores de la partida

Se mostrará al usuario los nombres del resto de los usuarios que están participando en su misma partida .

- Estimación: Se estima que en realizar esta historia se emplee 1 semana.
- Priorización: Debería tenerlo.
- Criterios de aceptación: Los usuarios podrán ver el nombre del resto de participantes de la partida.

6.Como usuario quiero poder guardar una partida en curso

Durante toda la partida existe la opción de SAVE GAME, con esta se podrá guardar la partida que se está jugando en ese momento. Todas las partidas se almacenan individualmente en un fichero que es generado al pulsar el botón descrito anteriormente, y posteriormente todas ellas se almacenan de manera más resumida en un fichero general.

- Estimación: Se estima que en realizar esta historia se emplearán 2 semanas.
- Priorización: Debería tenerlo.
- Criterios de aceptación: El usuario podrá guardar la partida que está jugando en el momento en el que él desee.

7.Como usuario quiero poder cargar una partida que he guardado con anterioridad

El usuario podrá cargar y así reanudar una partida que ha guardado con anterioridad. Todas las partidas se encuentran guardadas en ficheros, de los cuales se extraen los datos para dejar esta misma exactamente por el punto en el que se abandonó.

- Estimación: Se estima que en realizar esta tarea se emplearán 2 semanas.
- Priorización: Debería tenerlo.
- Criterios de aceptación: Se podrá cargar y por lo tanto reanudar una partida que ha sido previamente guardada.

8.Como usuario quiero poder introducir mediante teclado un comando y que se ejecute en la partida

El usuario introduce un comando por teclado, que se ejecutará inmediatamente en la partida. Tras pasar a una interfaz gráfica todo el juego, esta historia de usuario queda obsoleta, por eso decidimos **eliminarla**.

9. Como usuario quiero poder jugar contra la máquina en diferentes dificultades

Tras elegir el número de jugadores de la partida, el usuario podrá elegir para cada uno de ellos si quiere que sea una persona humana, jugando igual que él, o un bot. De este último tipo de jugador, se podrá elegir el nivel de dificultad que tendrá. Las tres dificultades son:

Fácil

En este modo, la máquina lanzará la primera carta que pueda de su lista. En caso de ser una carta de las encargadas de los cambios de color (*Add4Card* o *ChangeColorCard*), se seleccionará de manera aleatoria el color a elegir, sin tener en cuenta el resto de cartas que tiene en su mano. Este bot no será inteligente, y se le podrá ganar con facilidad.

Medio

Para cada turno de juego creamos una lista con todas las cartas de su actual lista que puedan ser lanzadas en ese momento, y contamos el número de cartas de cada color, así, cuando el bot lance una carta, elija entre el montón de las disponibles, la del color más repetido.

En el caso de que solo haya dos jugadores, el bot intentará lanzar un prohibido (*ForbiddenCard*), y así repetir turno.

El orden de prioridad de las cartas a lanzar entre las cartas del montón disponibles es:

1. *Add4Card*
2. *Add2Card*
3. *ForbiddenCard*
4. *ChangeCard*
5. *SimpleCard*
6. *ChangeColorCard*

Difícil

La estrategia de juego para este bot tiene un cambio crucial con respecto a la anterior, que es que la máquina, sabrá no sólo el número de cartas del siguiente jugador sino también todas y cada una de ellas. De esta manera resultará casi imposible al humano ganar, ya que el bot echará en todo momento la carta que menos convenga al usuario.

Si se trata de solo dos jugadores, como en la estrategia media, se intentará lanzar un prohibido para que el usuario repita turno. Luego, comprobaremos si para alguna de nuestras cartas, el otro jugador no puede tirar, obligándoles a robar. En el caso de que el otro contrincante pueda lanzar con cualquier carta que el bot escoja, este se

encargará de que sea la que peor le convenga: intentará cambiar al color del cual posea menos cartas o si está a punto de ganar lanzará el *Add4Card*.

El nuevo orden de prioridades de las cartas es:

1. *Add4Card*
2. *ChangeColorCard*
3. *ForbiddenCard*
4. *Add2Card*
5. *ChangeCard*
6. *SimpleCard*

La partida sin embargo sigue su curso normal y con las mismas reglas ya definidas anteriormente.

- Estimación: Se estima que en realizar esta tarea se emplearán 2 semanas.
- Priorización: Debería tenerlo.
- Criterios de aceptación: Se podrá jugar contra un bot al que previamente se le ha seleccionado su dificultad de juego.

10. Como usuario quiero jugar una partida en red

Esta historia de usuario es creada e implementada ante la demanda de esta funcionalidad.

El funcionamiento que siguió la conexión en red ya ha sido definido, pero de manera resumida se aplicó el patrón cliente-servidor aprovechando las características de los sockets de java.

La comunicación entre el cliente y el servidor es bidireccional, el cliente recibe del servidor los cambios que se van aplicando sobre el juego mientras que en el otro sentido los clientes indican al nodo compartido cuál es la acción que ha llevado a cabo un usuario. Las transferencias de datos han sido realizadas a través de objetos serializables para facilitar la codificación.

- Estimación: Se estima que en realizar esta tarea se emplearán 2 semanas.
- Priorización: Debería tenerlo.
- Criterios de aceptación: Se podrá jugar contra otros usuarios sin estar juntos, mediante conexión a la red.

Sprint Reviews



El **Sprint Review** es la reunión que se realiza al acabar el Sprint para evaluar el incremento producido durante ese Sprint.

Hemos decidido dividir esta parte por Sprints. En cada uno de ellos, realizaremos una breve introducción con los **objetivos** que habíamos marcado en el Sprint Planning, y a continuación de esto desarrollaremos las **tareas alcanzadas** durante este tiempo, así como el **trabajo añadido** que no estaba planificado.

Finalizaremos este apartado con una **tabla** en la que se mostrará las historias que han pasado por el **Product Backlog** y ya han sido finalizadas.

Acabaremos esta sección con un breve resumen de cada Sprint Review, en el que se podrá observar la evolución que hemos tenido a lo largo del proyecto mediante un gráfico de barras.

Sprint Review 1

Objetivos propuestos:

Durante este primer Sprint, nuestro principal objetivo era documentarnos bien y poder entender la metodología Scrum para poder sacarle el máximo partido a cada una de sus funcionalidades.

Habíamos decidido que íbamos a intentar crear las 4 primeras historias de usuario:

- Como usuario quiero jugar al UNO
- Como usuario quiero poder elegir el número de jugadores a participar
- Como usuario quiero jugar con las cartas (NORMALES Y ESPECIALES)
- Como usuario quiero que las cartas se comporten adecuadamente

Todos estos objetivos vienen detallados en la sección [Sprint Planning 1](#).

¿Qué se ha conseguido hacer?

Hemos logrado crear una versión básica del 'UNO' en la que se imprima la partida por consola. Para ello, hemos conseguido crear estas 4 historias de usuario:

ID	HISTORIA DE USUARIO
#1	Como usuario quiero jugar al UNO
#2	Como usuario quiero poder elegir el número de jugadores a participar
#3	Como usuario quiero jugar con las cartas (NORMALES y ESPECIALES)
#4	Como usuario quiero que las cartas se comporten adecuadamente

Las historias de usuario al detalle se encuentran en el apartado [Historias de usuario](#).

Asimismo, hemos conseguido entender mejor la metodología Scrum y la aplicación GitHub, algo que seguramente nos ayudará a la hora de organizarnos y que nos permitirá trabajar con más soltura.

La distribución de tareas que se ha llevado a cabo en el equipo durante este Sprint se puede ver en el apartado [Sprint Backlog 1](#).

Sprint Review 2

Objetivos propuestos:

Hacemos un breve resumen acerca de los objetivos que teníamos para este Sprint, entre los que se encuentran:

- Realización de numerosas pruebas acerca del código anterior
- Permitir que se muestre el nombre de los jugadores por pantalla
- Guardar una partida en curso
- Elaboración de diagramas UML

Todos estos objetivos vienen detallados en la sección [Sprint Planning 2](#).

¿Qué se ha conseguido hacer?

Los avances del Product Backlog que hemos conseguido realizar durante este Sprint son:

ID	HISTORIA DE USUARIO
#5	Como usuario quiero conocer los nombres de los otros jugadores de la partida
#6	Como usuario quiero poder guardar una partida en curso

Las historias de usuario al detalle se encuentran en el apartado [Historias de usuario](#).

Asimismo, hemos realizado pruebas con respecto a la funcionalidad del código que habíamos realizado en el Sprint anterior. Esto nos ha permitido asegurarnos de que funcionalidades como finalizar una partida, el comportamiento que tienen tanto las cartas especiales como las normales... van correctamente (las comprobaciones que íbamos a realizar fueron descritas más en detalle en el [Sprint Planning 2](#)).

A su vez, hemos comenzado a realizar los diagramas UML, tanto de secuencias como de clases, para poder mostrar gráficamente la estructura del código de nuestro proyecto.

Hemos comenzado a realizar algún test JUnit para poder comprobar el código. Hemos creado varios test de la lógica del juego para comprobar que las clases que hemos creado funcionan correctamente. Entre estas encontramos las clases de los tipos de cartas *Add2Card*, *Add4Card*, *ChangeCard*, *ForbiddenCard*, además de otras clases que hemos creado para la funcionalidad del juego como son *CardList*, *PlayerList*, *Player* y *Game*.

Durante este Sprint también se han realizado cambios en el código relacionado con algunas historias de usuario del anterior Sprint con el fin de mejorar el funcionamiento del juego. Este hecho viene detallado en el apartado "Diseño y Evolución de las Historias de usuario" que está en el otro documento de la entrega.

La distribución de tareas que se ha llevado a cabo en el equipo durante este Sprint se puede ver en el apartado [Sprint Backlog 2](#).

Trabajo añadido durante este Sprint

En la última sesión de laboratorio correspondiente a este Sprint, creamos un documento dónde desarrollamos la refactorización del código que habíamos hecho en estas dos semanas, debido a los cambios realizados en el código del proyecto gracias a las numerosas pruebas realizadas para comprobar su correcto funcionamiento.

Cabe destacar que durante este Sprint hemos realizado modificaciones con respecto al Product Backlog. En concreto se trata de la historia de usuario 2 (Como usuario quiero poder elegir el número de jugadores a participar). Nos hemos dado cuenta que aporta poco al proyecto, por lo que hemos decidido unirla a la historia de usuario 5 que hemos desarrollado en este Sprint (Como usuario quiero conocer los nombres de los otros jugadores de la partida).

Historias de usuario desarrolladas en el proyecto

ID	HISTORIA DE USUARIO
#1	Como usuario quiero jugar al UNO
#2	Como usuario quiero poder elegir el número de jugadores a participar (FUSIONADA CON LA 5)
#3	Como usuario quiero jugar con las cartas (NORMALES Y ESPECIALES)
#4	Como usuario quiero que las cartas se comporten adecuadamente
#5	Como usuario quiero conocer los nombres de los otros jugadores de la partida
#6	Como usuario quiero poder guardar una partida en curso

Como se puede comprobar, hemos añadido las historias 5 y 6, y hemos modificado la 2, que aparece fusionada con la 5.

Sprint Review 3

Objetivos propuestos:

Durante este Sprint, nos propusimos los siguientes objetivos:

- Realización de diagramas UML
- Cargar una partida
- Crear una serie de comandos en el juego
- Creación de test JUnit

Todos estos objetivos vienen detallados en la sección [Sprint Planning 3](#).

¿Qué se ha conseguido hacer?

Con respecto al Product Backlog, hemos añadido las siguientes historias de usuario

ID	HISTORIA DE USUARIO
#7	Como usuario quiero poder cargar una partida que he guardado con anterioridad
#8	Como usuario quiero poder introducir mediante teclado un comando y que se ejecute en la partida

Principalmente este avance se centra en la creación de comandos para mejorar la fluidez del juego y la adición de una nueva opcionalidad que te permita cargar una partida anteriormente guardada.

Las historias de usuario al detalle se encuentran en el apartado [Historias de usuario](#).

A su vez, hemos seguido realizando test de JUnit, que nos ayudan a encontrar posibles errores que surjan en el código. Además de refinar los test realizados en el previo sprint, hemos hecho uso de la nueva implementación de cargar partidas guardadas con anterioridad para realizar simplificar los test, lo cual ha reducido notablemente la complejidad y longitud de los mismos.

Cabe destacar que gran parte de este Sprint la hemos dedicado a la creación de diagramas de secuencias y de clases, que nos ayudan a tener una visión estructural del código que llevamos hecho hasta el momento.

Aunque se entrará más en detalle en el apartado “Diseño y Evolución de las Historias de usuario” que está en el otro documento de la entrega, hay que recalcar que durante estas dos semanas también se han hecho cambios en las historias de usuario anteriores para mejorar el funcionamiento del código.

La distribución de tareas que se ha llevado a cabo en el equipo durante este Sprint se puede ver en el apartado [Sprint Backlog 3](#).

Trabajo añadido durante este Sprint

Durante este Sprint teníamos dudas acerca de cómo adaptar nuestro proyecto al Modelo-Vista-Controlador. En una de las sesiones de laboratorio, el profesor nos ha ayudado a comprender mejor lo que es el M-V-C, así que hemos decidido comenzar a realizar algunos cambios en nuestro código para que se adapte mejor a este modelo.

Además hemos diseñado un prototipo en el que nos basaremos posteriormente para crear la interfaz gráfica.

Historias de usuario desarrolladas en el proyecto

ID	HISTORIA DE USUARIO
#1	Como usuario quiero jugar al UNO
#2	Como usuario quiero poder elegir el número de jugadores a participar (FUSIONADA CON LA 5)
#3	Como usuario quiero jugar con las cartas (NORMALES Y ESPECIALES)
#4	Como usuario quiero que las cartas se comporten adecuadamente
#5	Como usuario quiero conocer los nombres de los otros jugadores de la partida
#6	Como usuario quiero poder guardar una partida en curso
#7	Como usuario quiero poder cargar una partida que he guardado con anterioridad
#8	Como usuario quiero poder introducir mediante teclado un comando y que se ejecute en la partida

Se puede observar que durante este Sprint hemos añadido las historias 7 y 8.

Sprint Review 4

Objetivos propuestos:

Nos habíamos propuesto los siguientes objetivos para estas dos semanas:

- Comenzar a crear la interfaz gráfica
- Elaboración de test JUnit
- Creación de diagramas UML
- Adaptación del código al Modelo-Vista-Controlador

Todos estos objetivos vienen detallados en la sección [Sprint Planning 4](#).

¿Qué se ha conseguido hacer?

Durante este Sprint hemos realizado cambios en las anteriores historias de usuario para mejorar la eficiencia del código, así como su funcionalidad. Estos cambios aparecen descritos con más detalle en el apartado “Diseño y Evolución de las Historias de usuario” que está en el otro documento de la entrega.

Sin embargo, no hemos realizado ningún incremento en lo que al Product Backlog se refiere, es decir, no hemos añadido ninguna historia de usuario.

Hemos conseguido adaptar nuestro código al Modelo-Vista-Controlador, y hemos continuado realizando test de JUnit para comprobar el código. En cuanto a estos últimos, al igual que ha sucedido con el resto del código, simplemente hemos refinado algunos test, pero en ningún caso hemos añadido más.

Cabe destacar que la mayor parte del tiempo la hemos dedicado a la creación de la interfaz gráfica. Al principio del Sprint habíamos fijado como objetivo comenzar a hacerla, y hemos conseguido acabarla prácticamente entera.

Lamentablemente no hemos podido realizar los diagramas UML, ya que hemos preferido esperarnos a acabar la interfaz gráfica para comenzar a realizar los diagramas de clases y secuencias.

La distribución de tareas que se ha llevado a cabo en el equipo durante este Sprint se puede ver en el apartado [Sprint Backlog 4](#).

Historias de usuario desarrolladas en el proyecto

ID	HISTORIA DE USUARIO
#1	Como usuario quiero jugar al UNO
#2	Como usuario quiero poder elegir el número de jugadores a participar (FUSIONADA CON LA 5)
#3	Como usuario quiero jugar con las cartas (NORMALES Y ESPECIALES)
#4	Como usuario quiero que las cartas se comporten adecuadamente
#5	Como usuario quiero conocer los nombres de los otros jugadores de la partida
#6	Como usuario quiero poder guardar una partida en curso
#7	Como usuario quiero poder cargar una partida que he guardado con anterioridad
#8	Como usuario quiero poder introducir mediante teclado un comando y que se ejecute en la partida

Como se ha descrito anteriormente, durante este Sprint no hemos añadido ninguna historia de usuario.

Sprint Review 5

Objetivos propuestos:

Los objetivos que habíamos planteado para este Sprint eran:

- Creación de las estrategias del juego
- Elaboración de diagramas UML
- Creación de test JUnit

Todos estos objetivos vienen detallados en la sección [Sprint Planning 5](#).

¿Qué se ha conseguido hacer?

Respecto al Product Backlog, hemos añadido la siguiente historia de usuario:

ID	HISTORIA DE USUARIO
#9	Como usuario quiero poder jugar contra la máquina en diferentes dificultades
	NIVEL FÁCIL
	NIVEL MEDIO
	NIVEL DIFÍCIL

Principalmente se basa en la creación de una nueva funcionalidad que permite al usuario elegir el nivel de la máquina contra la que va a jugar.

Las historias de usuario al detalle se encuentran en el apartado [Historias de usuario](#).

Durante este Sprint también hemos realizado cambios en las historias de usuario anteriores. Estos cambios aparecen descritos con más detalle en el apartado “Diseño y Evolución de las Historias de usuario” que está en el otro documento de la entrega, o en la [Evolución del Product Backlog](#) en este mismo documento.

Hemos realizado diagramas de secuencias y de clases, así como test JUnit para probar el código. En cuanto a estos últimos, con el objetivo de verificar la correcta funcionalidad de las nuevas estrategias, hemos creado varios test por cada nivel de dificultad. Con esto nos hemos cerciorado de que a priori las estrategias se comportan según lo esperado.

La distribución de tareas que se ha llevado a cabo en el equipo durante este Sprint se puede ver en el apartado [Sprint Backlog 5](#).

Trabajo añadido durante este Sprint

Hemos repasado y corregido algunas historias de usuario que no estaban bien desarrolladas.

A su vez, cabe recordar que en el Sprint 2 habíamos fusionado la historia de usuario 2 (Como usuario quiero poder elegir el número de jugadores a participar) con la historia 5 (Como usuario quiero conocer los nombres de los otros jugadores de la partida). En este Sprint hemos llegado al acuerdo de eliminar las historias 2 y 5, ya que entendemos que comparten funcionalidad con la historia 9 (Como usuario quiero poder jugar contra la máquina en diferentes dificultades). En esta historia ya tenemos la opción de elegir el número de jugadores, el tipo (bot o humano), y el nombre.

Algo parecido nos ha pasado con la historia 8 (Como usuario quiero poder introducir mediante teclado un comando y que se ejecute en la partida). Nos hemos dado cuenta de que la historia de usuario no aporta nada sustancial al proyecto, por lo que la hemos eliminado.

Historias de usuario desarrolladas en el proyecto

ID	HISTORIA DE USUARIO
#1	Como usuario quiero jugar al UNO
#2	Como usuario quiero poder elegir el número de jugadores a participar (FUSIONADA CON LA 9) (ELIMINADA)
#3	Como usuario quiero jugar con las cartas (NORMALES Y ESPECIALES)
#4	Como usuario quiero que las cartas se comporten adecuadamente
#5	Como usuario quiero conocer los nombres de los otros jugadores de la partida (FUSIONADA CON LA 9) (ELIMINADA)
#6	Como usuario quiero poder guardar una partida en curso
#7	Como usuario quiero poder cargar una partida que he guardado con anterioridad
#8	Como usuario quiero poder introducir mediante teclado un comando y que se ejecute en la partida (ELIMINADA)
#9	Como usuario quiero poder jugar contra la máquina en diferentes dificultades

Sprint Review 6

Objetivos propuestos:

Durante este Sprint habíamos fijado los siguientes objetivos:

- Creación de test JUnit
- Elaboración de diagramas UML
- Confección del código necesario para poder jugar una partida en la red

Todos estos objetivos vienen detallados en la sección [Sprint Planning 6](#).

¿Qué se ha conseguido hacer?

Con respecto al Product Backlog, hemos logrado crear el código necesario para desarrollar la siguiente historia de usuario:

ID	HISTORIA DE USUARIO
#10	Como usuario quiero poder jugar una partida en la red

Esta nueva funcionalidad se basa principalmente en la posibilidad de jugar una partida en la red, es decir online.

Las historias de usuario al detalle se encuentran en el apartado [Historias de usuario](#).

Además, hemos seguido haciendo cambios en el código de las historias de usuario anteriores para mejorar la funcionalidad del juego. Estos cambios aparecen descritos con más detalle en el apartado “Diseño y Evolución de las Historias de usuario” que está en el otro documento de la entrega.

A su vez, hemos seguido realizando los diagramas de secuencias y de clases que van ligados al código de esta nueva historia de usuario.

Hemos realizado test de JUnit para probar el código. En este caso, hemos añadido más test para verificar la funcionalidad esperada de las estrategias, llegando a tener 4 por cada tipo de estrategia. No obstante, al no haber añadido mucha más funcionalidad no hemos creado más test.

Hemos decidido comenzar a realizar la documentación necesaria para la entrega final. De esta manera hemos logrado adelantar trabajo y así poder ir un poco más tranquilos a los últimos Sprints.

Hemos seguido redactando el documento relacionado con Scrum, y hemos comenzado el documento relacionado con el Diseño UML.

La distribución de tareas que se ha llevado a cabo en el equipo durante este Sprint se puede ver en el apartado [Sprint Backlog 6](#).

Historias de usuario desarrolladas en el proyecto

ID	HISTORIA DE USUARIO
#1	Como usuario quiero jugar al UNO
#2	Como usuario quiero poder elegir el número de jugadores a participar (FUSIONADA CON LA 9) (ELIMINADA)
#3	Como usuario quiero jugar con las cartas (NORMALES Y ESPECIALES)
#4	Como usuario quiero que las cartas se comporten adecuadamente
#5	Como usuario quiero conocer los nombres de los otros jugadores de la partida (FUSIONADA CON LA 9) (ELIMINADA)
#6	Como usuario quiero poder guardar una partida en curso
#7	Como usuario quiero poder cargar una partida que he guardado con anterioridad
#8	Como usuario quiero poder introducir mediante teclado un comando y que se ejecute en la partida (ELIMINADA)
#9	Como usuario quiero poder jugar contra la máquina en diferentes dificultades
#10	Como usuario quiero poder jugar una partida en la red

Durante este Sprint hemos añadido la historia de usuario 10.

Sprint Review 7

Objetivos propuestos:

Durante este Sprint habíamos fijado los siguientes objetivos:

- Acabar de redactar documento Scrum
- Acabar de redactar documento Diseño UML
- Revisión del código y de los test JUnit

Todos estos objetivos vienen detallados en la sección [Sprint Planning 7](#).

¿Qué se ha conseguido hacer?

En cuanto al Product Backlog, tal y como habíamos planificado en el Sprint Planning, no ha sido necesario añadir ninguna historia de usuario nueva.

Cabe recordar que las historias de usuario están explicadas detalladamente en el apartado [Historias de usuario](#).

Aunque no hayamos añadido ninguna historia nueva, hemos realizado algunos cambios con respecto al código de algunas de las anteriores para mejorar la jugabilidad y arreglar algunos bugs. Además hemos refinado alguna de las pantallas de la interfaz gráfica para mejorar la calidad de las mismas.

Estos cambios aparecen descritos con más detalle en el apartado “Diseño y Evolución de las Historias de usuario” que está en el otro documento de la entrega.

Durante este Sprint hemos realizado numerosas pruebas para probar el correcto funcionamiento del código y de los JUnit. Además, hemos comentado detalladamente ambas partes.

Cabe destacar que nuestra principal tarea durante estas dos semanas ha sido acabar de redactar la documentación tanto del documento Diseño UML como del de Scrum.

En resumen, durante este Sprint hemos conseguido completar todas las tareas necesarias para acabar el proyecto de la mejor manera posible.

La distribución de tareas que se ha llevado a cabo en el equipo durante este Sprint se puede ver en el apartado [Sprint Backlog 7](#).

Historias de usuario desarrolladas en el proyecto

ID	HISTORIA DE USUARIO
#1	Como usuario quiero jugar al UNO
#2	Como usuario quiero poder elegir el número de jugadores a participar (FUSIONADA CON LA 9) (ELIMINADA)
#3	Como usuario quiero jugar con las cartas (NORMALES Y ESPECIALES)
#4	Como usuario quiero que las cartas se comporten adecuadamente
#5	Como usuario quiero conocer los nombres de los otros jugadores de la partida (FUSIONADA CON LA 9) (ELIMINADA)
#6	Como usuario quiero poder guardar una partida en curso
#7	Como usuario quiero poder cargar una partida que he guardado con anterioridad
#8	Como usuario quiero poder introducir mediante teclado un comando y que se ejecute en la partida (ELIMINADA)
#9	Como usuario quiero poder jugar contra la máquina en diferentes dificultades
#10	Como usuario quiero poder jugar una partida en la red

Durante este Sprint no hemos añadido ninguna historia de usuario.

Resumen Sprint Reviews

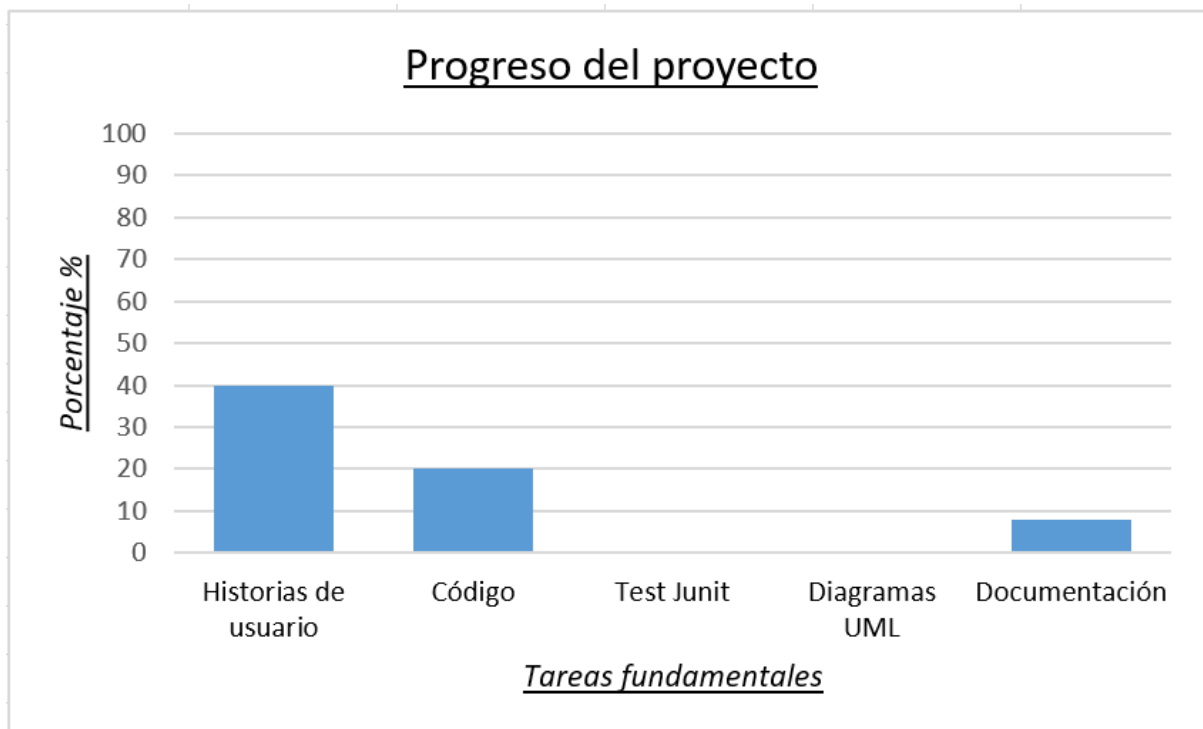
Vamos a realizar un breve resumen de cada Sprint Review, acompañado de un gráfico en el que mostraremos el progreso que hemos tenido en cada uno de los pilares fundamentales del proyecto, que han sido:

- Creación de historias de usuario
- Desarrollo de código
- Creación de test Junit para probar el código
- Elaboración de diagramas UML para ofrecer una visión estructural de nuestro programa
- Redacción de la documentación necesaria.

De esta manera se podrá observar la evolución de nuestro proyecto en cuanto a este apartado se refiere.

Sprint 1

En este Sprint nos informamos y como resultado de esto conseguimos entender mejor la metodología Scrum. Además creamos las cuatro primeras historias de usuario (explicación detallada en el apartado [Historias de usuario](#)), que desarrollamos para obtener una versión básica del 'UNO'. Aunque, naturalmente, tuvimos que modificar el código relacionado con estas historias a lo largo del proyecto, en esta entrega implementamos un juego bastante funcional.

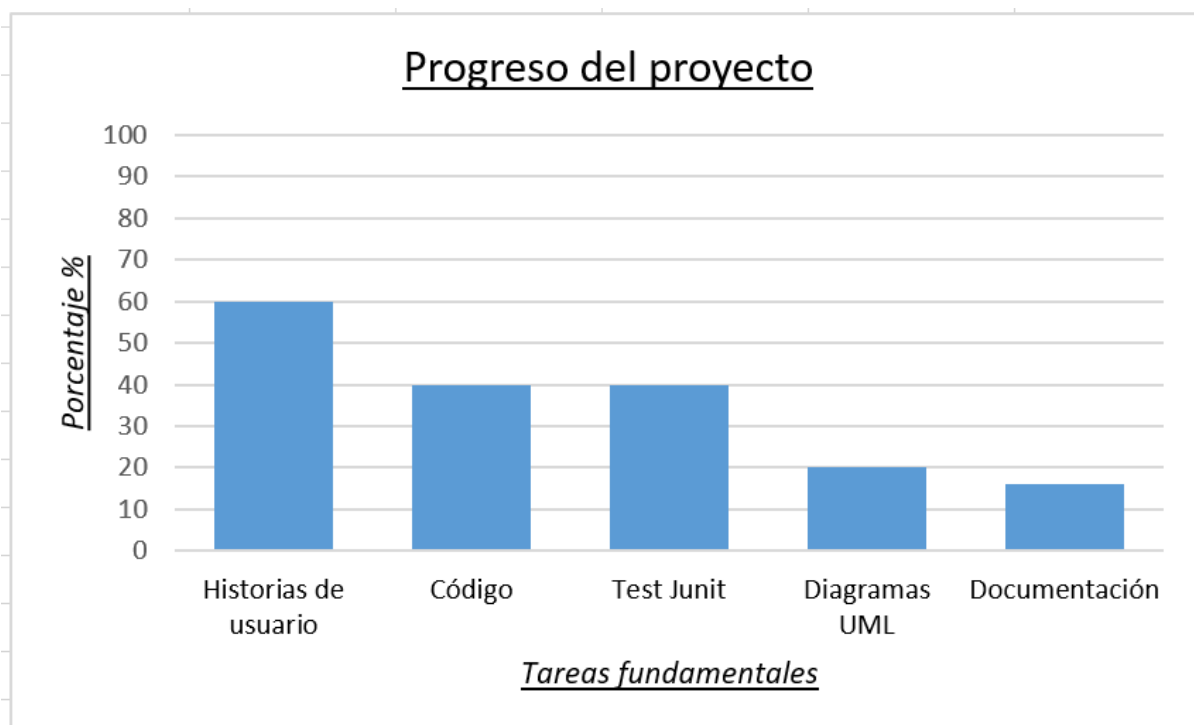


Sprint 2

Durante este Sprint desarrollamos las historias de usuario 5 y 6, que representaban la funcionalidad de guardar una partida en curso y conocer el nombre de los jugadores de la partida.

Se realizaron un gran número de pruebas, con el fin de depurar errores y mejorar el código del primer Sprint.

En estas dos semanas comenzamos a crear test Junit para comprobar el código, y diagramas de UML. Aunque estos últimos los tuvimos que modificar de forma radical en los Sprints siguientes debido a los cambios producidos en el código, esto nos sirvió para aprender la metodología de realización de estos diagramas, tanto de clases como de secuencias.

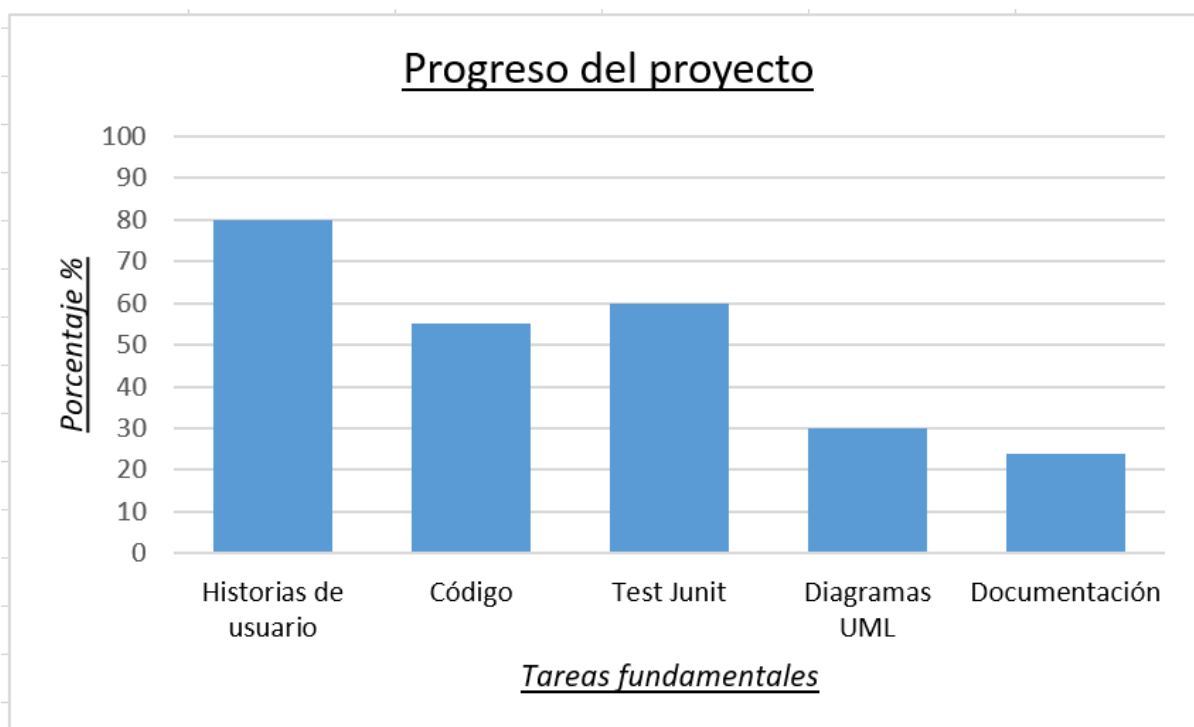


Sprint 3

En estas dos semanas desarrollamos las opciones de poder cargar una partida y de introducir comandos por teclado para agilizar el juego.

A su vez, realizamos un gran número de test JUnit al igual que en el segundo Sprint, lo que nos permitió probar el código.

Como ya habíamos reflejado en el Sprint anterior, algunos diagramas UML realizados no nos sirvieron para nada ya que el código de sus respectivas clases cambió. Durante este Sprint modificamos esos diagramas y creamos algunos nuevos, aunque el incremento fue menor como se puede observar en el gráfico, ya que la modificación de los anteriores nos requirió mucho tiempo.



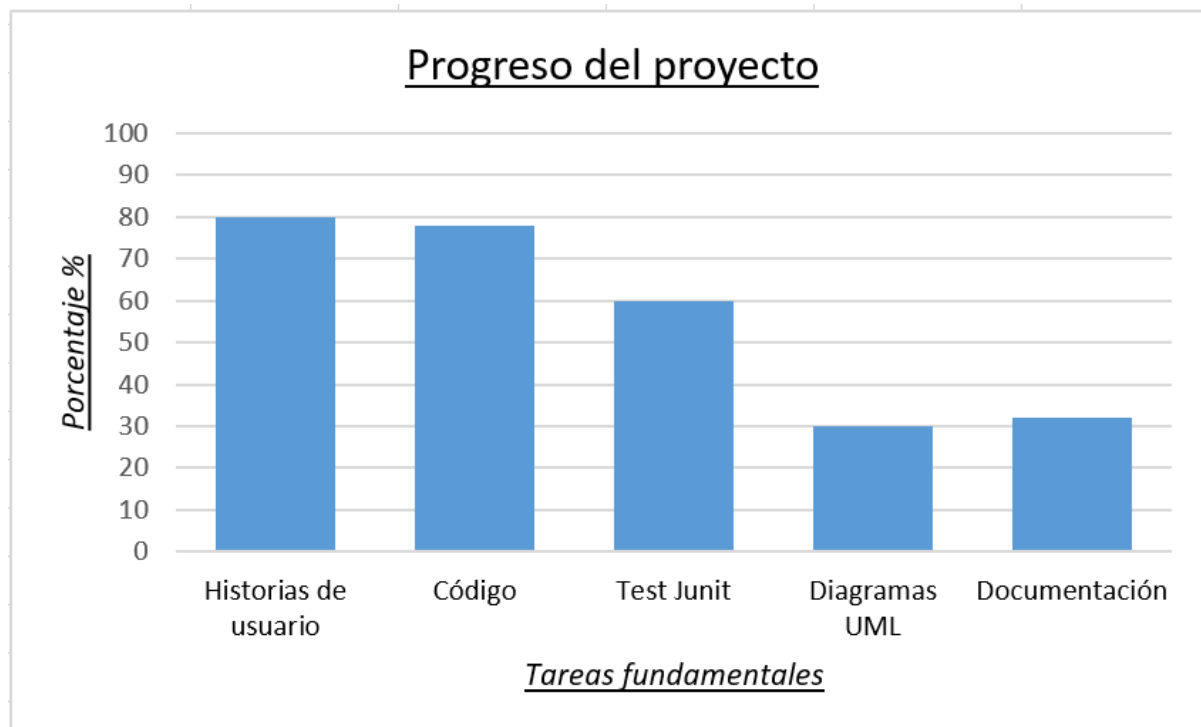
Sprint 4

Durante este Sprint no añadimos ninguna historia de usuario al Product Backlog.

Lamentablemente tuvimos que solucionar el siguiente problema: nos dimos cuenta de que nuestro código no se adaptaba muy bien al Modelo-Vista-Controlador.

En estas dos semanas realizamos grandes avances en cuanto al código se refiere, ya que conseguimos realizar prácticamente toda la interfaz gráfica.

En cuanto a los test JUnit y diagramas UML, no añadimos ninguno nuevo, sólo nos dedicamos a corregir errores de los ya existentes.



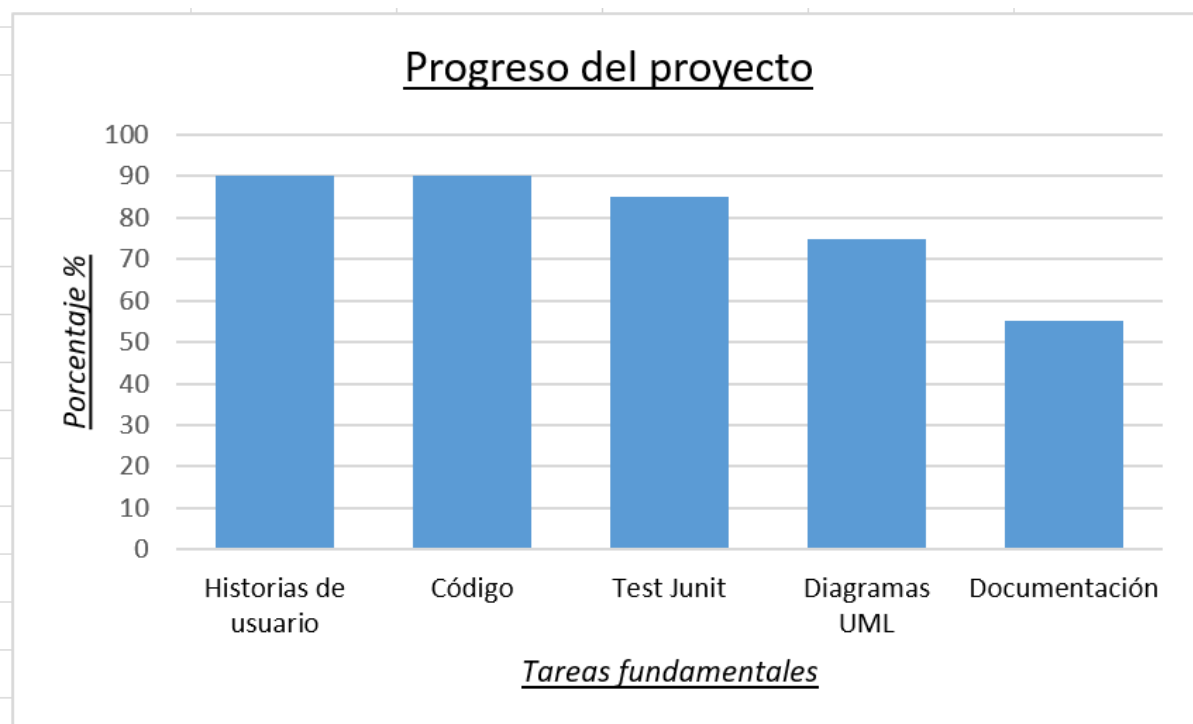
Sprint 5

En este Sprint decidimos implementar las estrategias del juego, añadiendo la historia de usuario 9. Nos dimos cuenta de algunos errores que habíamos cometido al desarrollar algunas historias de usuario, algo que tuvo como consecuencia la eliminación o fusión de algunas de ellas (detallado en el apartado "[Evolución del Product Backlog](#)").

Durante este Sprint sí que logramos crear tanto nuevos test de JUnit como diagramas UML.

Incluso comenzamos a redactar parte de la documentación de la entrega final.

En general en estas dos semanas realizamos grandes avances en el proyecto, ya que coincidieron con las vacaciones de Semana Santa, a pesar de los problemas de comunicación y organización que esto conllevó.



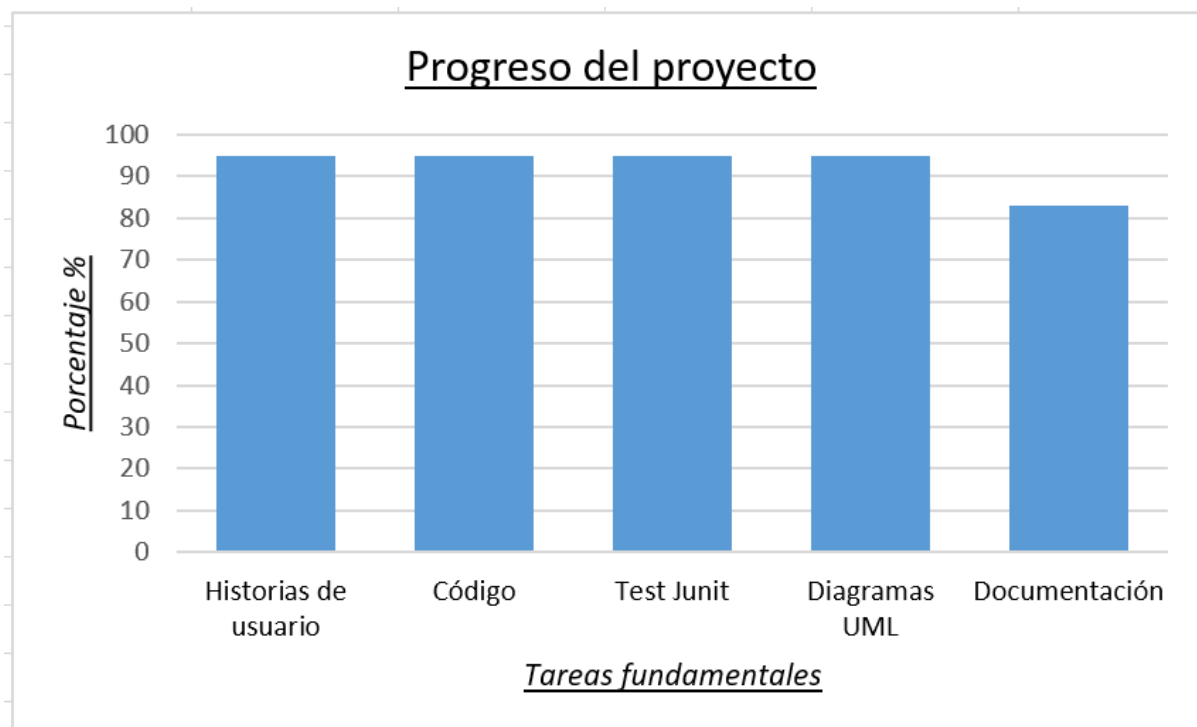
Sprint 6

Durante este Sprint conseguimos implementar la funcionalidad que nos quedaba por hacer, que era la de añadir la opción de jugar una partida en red (historia de usuario 10). Nuestro proyecto ya se podía empezar a ver como un juego totalmente real, a falta de algunas pruebas.

Como la funcionalidad del juego estaba prácticamente acabada, decidimos realizar todos los diagramas UML y test de JUnit que nos faltaban.

Además, seguimos redactando los documentos necesarios para la entrega final.

Como podemos ver en el gráfico, el proyecto estaba prácticamente acabado al final de este Sprint, a falta de realizar la documentación y las pruebas pertinentes para asegurar el correcto funcionamiento del juego.



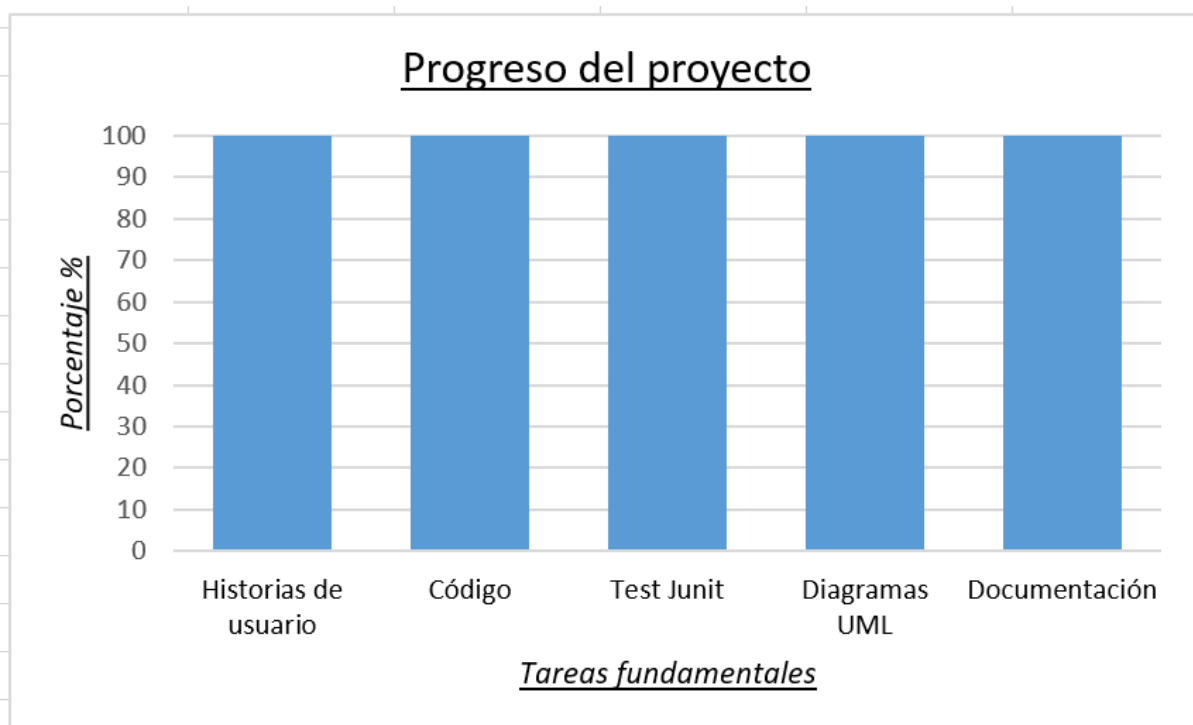
Sprint 7

En estas dos semanas conseguimos acabar el proyecto.

No añadimos ninguna historia de usuario nueva, tal y como estaba previsto. Nos hemos encargado de realizar las pruebas pertinentes para comprobar el código y los test JUnit. Hemos modificado algunas pantallas de la interfaz gráfica para que el juego quede más visual.

Asimismo, conseguimos acabar los documentos de Scrum y Diseño UML, necesarios para la entrega final.

Realizadas todas estas tareas, podemos decir que hemos terminado el proyecto, tal y como podemos observar en el siguiente gráfico.



Conclusión final

Como ya preveíamos los Sprint Reviews han resultado de gran utilidad especialmente teniendo en cuenta que hemos aplicado una forma de trabajo ágil basada en historias de usuario.

En cuanto al desarrollo que hemos visto a raíz de los Sprint Reviews has sido el siguiente: En un principio, como resulta evidente desarrollamos historias de usuario nucleares para el funcionamiento del juego, y dedicamos la amplia mayoría de nuestros esfuerzos a ello. A su vez, para la verificación del correcto funcionamiento de lo creado creamos varios test JUnit. Además, creamos los diagramas UML que resultaron de gran utilidad para el desarrollo del proyecto.

Más adelante, cuando ya disponíamos de una base sólida de código en torno a las historias de usuario principales nos vimos con el problema de no haber adecuado el proyecto al patrón modelo-vista-controlador. Por ello, tuvimos que dedicar una gran parte de nuestros esfuerzos a la refactorización del código lo cual nos ralentizó en lo demás.

Ligado a esto, una vez avanzados en el código, nos dimos cuenta de la errónea confección de algunas historias de usuarios que se vieron fusionadas o eliminadas en su defecto.

Asimismo, una vez implementada la funcionalidad general empezamos a dedicarle mayor parte del esfuerzo a completar la documentación del proyecto. Esto se acentuó en la parte final del proyecto, donde una vez terminado el código(salvo pequeños refinamientos) avanzamos en la realización de los documentos.

En conclusión, los Sprint Reviews han servido para llevar un riguroso control de lo que habíamos hecho en cada sprint para saber lo que teníamos terminado y tener una idea general de la evolución del proyecto.

Sprint Retrospectives



El **Sprint Retrospective** se basa principalmente en la autoevaluación por parte del Equipo Scrum acerca del trabajo realizado en cada Sprint.

Es una herramienta muy útil para poder detectar problemas en el desarrollo del proyecto, y así poder crear un plan de mejoras que serán abordadas durante el siguiente Sprint.

En nuestro caso, hemos utilizado el Starfish Retrospective durante todo el proyecto, basado en cinco puntos:

- **More of:** ¿Qué debemos mejorar?
- **Keep Doing:** ¿Qué debemos seguir haciendo?
- **Start Doing:** ¿Qué debemos empezar a hacer?
- **Stop Doing:** ¿Qué debemos dejar de hacer?
- **Less of:** ¿Qué debemos hacer menos?

Sprint Retrospective 1

More of

Deberíamos mejorar la comunicación entre los miembros del equipo, ser más claros y concisos en cuanto a la parte que va a realizar cada uno, para que no haya código duplicado u otros inconvenientes que nos puedan retrasar el trabajo.

Para el siguiente sprint, debemos unificar el código a la hora de programar, ya que en este sprint no hemos estado lo suficientemente cohesionados.

Deberíamos establecer una arquitectura clara y esquematizada, puesto que hasta ahora ha sido un poco caótico tratar de que cada uno haga su parte sin tener clara la de los demás.

Keep Doing

En este sprint nos hemos dado cuenta de lo necesarias que son las reuniones de equipo, por tanto debemos seguir haciéndolas e incluso aumentar el número de las mismas por semana. Estas reuniones mejoran la comunicación entre los miembros y hacen que el trabajo fluya mejor.

Start Doing

Ser más concretos en nuestras historias de usuario, pues esto dejaría claro el trabajo a realizar para la siguiente entrega.

Por la parte del código, estaría bien tratar de mantener la estética del mismo ordenando las funciones de las clases por visibilidad.

Stop Doing

Deberíamos dejar de tener ciertas expectativas demasiado altas para encontrarnos aún en el inicio de nuestro proyecto. Por ejemplo, tratar de que en una fase tan temprana nuestro código sea ya óptimo.

Less of

Deberíamos reducir las ocasiones en las que varios integrantes del grupo programen a la vez durante un periodo de tiempo extenso y sobre una misma clase.

Esto puede generar conflictos a la hora de mezclar el código.

Sprint Retrospective 2

More of

Deberíamos mejorar la distribución del tiempo a lo largo de la duración del Sprint para poder realizar cantidades de trabajo menores de una sola vez y no tratar de cumplir todos los objetivos en un tiempo menor del que tienen asignado.

Keep Doing

Como ya dijimos en el Sprint anterior, seguiremos dando importancia a la comunicación entre todos los miembros del equipo favoreciendo las reuniones frecuentes.

Start Doing

Debemos empezar a repartir las tareas de una manera más equilibrada, y no dejar el “mismo tipo” de trabajos para los mismos integrantes en cada Sprint.

Stop Doing

Debemos suprimir el número de tareas que realizamos sin haber adquirido unos conocimientos previos necesarios para poder realizar dicha tarea.

Less of

Debemos reducir lo máximo posible las ocasiones en las que los miembros del grupo no tienen claros los objetivos del Sprint.

Sprint Retrospective 3

More of

Consideramos que nos vendría bien realizar alguna reunión más.

De este modo, conseguiremos mejorar la comunicación y la cohesión en cuanto a grupo.

Keep Doing

Debemos seguir dividiendo el trabajo de una manera equilibrada, tal y como hemos hecho en este último sprint. Esto nos ayuda a trabajar de una manera más eficiente.

Start Doing

Todo el grupo se debe involucrar en las partes del trabajo en las que menos ha participado, intentando entender y comprender lo que han hecho los demás compañeros.

Stop Doing

Debemos dejar de realizar diagramas UML sin que el código esté totalmente acabado. Así evitaremos realizar cambios constantemente en los diagramas de clases y secuencias.

Less of

Tenemos que hacer menos cambios en los diagramas de secuencias, y esperar a que el código esté completamente modificado y correcto.

Sprint Retrospective 4

More of

Tenemos que fijar algo mejor los objetivos que queremos alcanzar durante un Sprint.

Keep Doing

Durante este Sprint nos hemos repartido el trabajo de una manera muy equilibrada y clara. Esto nos ha ayudado a trabajar de una manera más eficiente.

Start Doing

Debemos fijar mejor los plazos en los que cada integrante del grupo debe tener realizada sus tarea en el Sprint.

Stop Doing

Tendríamos que evitar realizar las tareas del Sprint con prisa, sólo por el hecho de que nos dé tiempo a entregarlo en el vigente Sprint, y así poder seguir con lo programado. Es mejor realizar las actividades a un ritmo un poco más lento, y así evitaremos futuras correcciones.

Less of

Aunque esta semana hemos intentado mejorar en este aspecto, deberíamos seguir aumentando nuestra implicación en la comprensión de las partes del proyecto que han realizado los compañeros, dejando de lado el desinterés acerca de este tema. Esto nos ayudaría a comprender mucho mejor nuestro proyecto.

Sprint Retrospective 5

More of

Debemos mejorar nuestra comunicación como grupo. Cada integrante del equipo debe comentar los avances que hace en su tarea, debe notificar si necesita algún tipo de ayuda...

Keep Doing

La organización que hemos llevado en este Sprint, en la que nos hemos repartido el trabajo en distintas secciones: código, diagramas, historias de usuario y documentación.

Start Doing

Debemos empezar a realizar los documentos necesarios para la entrega final, y así evitar agobios de última hora y acumulación de trabajo en las últimas semanas.

Durante este Sprint, en vez de responder a las dos siguientes cuestiones: **Stop Doing** y **Less of**, sólo queremos reflejar que, debido principalmente a la pausa vacacional, hemos tenido problemas de comunicación en el grupo ya que no nos hemos podido ver de manera recurrente como lo hemos hecho en los anteriores Sprints.

Sprint Retrospective 6

En este Sprint decidimos suprimir el Starfish Retrospective.

Vamos a realizar una valoración acerca de lo que nos ha parecido este Sprint, y de cómo debemos encarar el siguiente.

Debemos destacar el esfuerzo que ha hecho todo el equipo durante este Sprint. Hemos conseguido repartir el trabajo de una manera equilibrada y la comunicación del grupo ha sido muy buena.

Se podría decir que este ha sido nuestro mejor Sprint del proyecto en cuanto a aplicación de la metodología Scrum, organización...

De cara al próximo Sprint, que será el último, debemos seguir con esta dinámica positiva de trabajo para terminar el proyecto lo antes posible y así evitar agobios futuros. Debemos hacer un último esfuerzo para terminar el proyecto de la mejor manera posible.

Sprint Retrospective 7

En este Sprint decidimos suprimir el Starfish Retrospective.

Vamos a realizar una valoración acerca de lo que nos ha parecido este Sprint y cómo podríamos encarar un futuro proyecto, ya que en este ya no habrá más Sprints en este.

Debemos destacar el esfuerzo que ha hecho todo el equipo durante este Sprint. Pese a no tener nuevas historias de usuario, hemos conseguido repartir el trabajo de una manera equilibrada y la comunicación del grupo ha sido muy buena.

Se podría decir que a pesar de no haber tenido resultados muy vistosos, la importancia de este Sprint ha sido vital, en cuanto a que ha significado la culminación del proyecto.

De cara a próximos proyectos, debemos aplicar todos aquellos conocimientos, para nada escasos, respecto a la planificación de los mismos adquiridos a lo largo de este. Cabe mencionar que ya en el mismo transcurso del proyecto hemos notado una mejora causada por la aplicación de medidas surgidas en los mismos Sprint Retrospective.

Resumen Sprint Retrospectives

En este apartado se mostrará la evolución que han tenido a lo largo del proyecto los Sprints Retrospective.

Se realizará un breve resumen de cada uno de los Sprints Retrospective que hemos realizado.

Sprint 1

Durante el primer Sprint intentamos informarnos lo máximo posible para tratar de entender la metodología Scrum y poder sacarle el máximo partido.

En este Sprint llegamos a la conclusión de que era mejor no programar todos a la vez ya que a la hora de hacer el 'Merge' en Git surgían muchos conflictos. Detectar este error nos ayudó para evitar comentarlo en el futuro.

Sprint 2

Este Sprint nos ayudó a darnos cuenta de la importancia que tiene la organización y la distribución del trabajo a la hora de trabajar en equipo. Nuestro principal problema era que no concretábamos cuál era la tarea específica que debía realizar cada integrante del grupo en el Sprint, lo cuál nos generaba problemas de distribución de tiempo.

Sprint 3

El principal problema que detectamos en este Sprint fue la impaciencia que tuvimos en algunos momentos a la hora de realizar algunas tareas.

Un ejemplo de ello es lo que nos sucedió con los diagramas UML. En vez de esperarnos a que el código estuviese totalmente terminado por completo y revisado, nosotros empezábamos a hacer los diagramas de secuencias y de clases sin este paso previo, lo que nos generó que tuviésemos que modificar los diagramas en numerosas ocasiones debidos a los cambios que se producían en el código. Esto claramente nos ralentizaba a la hora de trabajar.

A su vez, este Sprint nos sirvió para reafirmarnos en la idea que llevábamos sosteniendo desde el primer Sprint: la importancia que tiene la buena organización y comunicación al trabajar en grupo.

Sprint 4

En este Sprint detectamos dos principales problemas. El primero de ellos fue que nos dimos cuenta de que en algunas ocasiones intentábamos hacer las tareas lo más rápido posible solo por el simple hecho de cumplir todos los objetivos marcados para el Sprint. Esto provocaba que nos equivocáramos en mayor medida, y por lo tanto nos ralentizaba.

Por otra parte, durante este Sprint decidimos solucionar un problema que llevábamos arrastrando desde el Sprint anterior. Se trataba de que cada integrante del equipo nos centráramos principalmente en nuestro trabajo, pero luego, ante la falta de tiempo, no nos podíamos implicar en la comprensión de las otras partes del proyecto que habían realizado el resto de compañeros, y que nos vendría bien conocerlas a todos.

Sprint 5

A pesar de que en los anteriores Sprints habíamos mejorado incrementalmente en el aspecto de comunicación como grupo, durante este Sprint retrocedimos un poco en ese sentido. Creemos que una de las causas principales de este hecho fue que este Sprint coincidió con el periodo vacacional de la Semana Santa.

Sin embargo, gracias a la gran distribución del trabajo que habíamos realizado pudimos completar los objetivos de este Sprint.

Sprint 6

Debemos destacar que la distribución de tareas, la comunicación entre miembros del equipo y, la organización en general, durante este Sprint, fue muy buena, siendo probablemente la mejor de todo el proyecto.

Todos los integrantes del grupo hicieron un gran esfuerzo para intentar adelantar el trabajo que nos quedaba y así ir un poco más relajados de cara al último Sprint.

Sprint 7

Durante este Sprint continuamos con la dinámica positiva que arrastrábamos de Sprints anteriores en cuanto a la organización y comunicación.

Todos los integrantes del grupo hicieron un gran esfuerzo para acabar todas las cuestiones pendientes que teníamos para finalmente dar por acabado el proyecto.

Conclusión final

Pese a nuestro escepticismo inicial sobre la utilidad real de los Sprint retrospective, cabe destacar que ha resultado de gran utilidad de cara a mejorar la dinámica de trabajo del grupo.

Se puede observar una clara evolución durante el desarrollo del proyecto en cuanto a cómo hemos trabajado. Inicialmente no estábamos un poco familiarizados con el funcionamiento de las ramas y cómo podríamos aplicar la programación simultánea por parte de distintos integrantes del grupo, lo cual nos llevó a tomar medidas en pos de solucionarlo. A raíz de esto empezamos a no programar todos a la vez, o al menos dos personas cuyas clases estuvieran interconectadas y pudieran causar problemas.

Sin embargo, más adelante nos dimos cuenta de que no concretamos la función de cada uno lo cual nos ralentizaba el desarrollo del proyecto. Fue a partir de este momento que empezamos a tener funciones claras para evitar los problemas anteriores.

No obstante, esta fragmentación del trabajo nos llevó al extremo contrario, el de no ser conocedores de lo que estaban desarrollando nuestros compañeros, lo cual también hizo que surgieran complicaciones en el desarrollo. Por tanto, también intentamos poner fin a este problema intentando aumentar la comunicación entre los integrantes del grupo.

En este sentido, se puede observar una tendencia natural de oscilación cuando adquirimos conocimientos: nos damos cuenta de que algo falla, y ponemos medidas demasiado drásticas para solucionarlo, sin tener en cuenta el balance, y por tanto las prácticas positivas que podemos estar dejando de hacer por centrarnos en lo nuevo. Por tanto, después de esto vemos que las medidas han sido excesivas hasta que encontramos el punto medio.

En cuanto a lo que hemos realizado correctamente durante el proyecto hemos intentado mantenerlas mientras intentábamos corregir los errores mencionados anteriormente.

En conclusión, el Sprint retrospective ha resultado ser de vital importancia para mejorar la dinámica de trabajo entre los integrantes del equipo y ha facilitado la incorporación de prácticas beneficiosas y la reducción de prácticas perniciosas.

Sprint plannings



El **Sprint Planning** es la reunión en la que se planifica el trabajo a realizar durante el Sprint.

Este apartado del documento lo vamos a dividir por Sprints. Por cada Sprint, realizaremos una descripción de los **objetivos** que hemos fijado. Se mostrarán una serie de tablas con las historias de usuario que tenemos pensado desarrollar para cada Sprint.

Sprint Planning 1

Objetivo del Sprint:

Para este Sprint, hemos decidido implementar lo básico en cuanto a código para poder jugar una partida básica de 'UNO' imprimiendo las cartas por la consola de Eclipse.

Para ello, el Equipo de Desarrolladores hemos llegado al acuerdo de crear las siguientes historias de usuario, que nos servirán para poder implementar lo descrito anteriormente:

ID	HISTORIA DE USUARIO
#1	Como usuario quiero jugar al UNO
#2	Como usuario quiero poder elegir el número de jugadores a participar
#3	Como usuario quiero jugar con las cartas (NORMALES y ESPECIALES)
#4	Como usuario quiero que las cartas se comporten adecuadamente

Además, intentaremos ampliar nuestros conocimientos acerca de la metodología de Scrum y de la aplicación GitHub, ya que nos servirá de gran ayuda durante todo el proyecto.

La distribución de tareas en el Equipo de desarrollo se podrá observar con más detalle en el apartado [Sprint Backlog 1](#).

Sprint Planning 2

Objetivo del Sprint:

En este Sprint, hemos fijado como objetivo principal depurar y mejorar el código realizado durante el Sprint anterior. Realizaremos pruebas para comprobar la consistencia del mismo y así poder detectar posibles errores.

A continuación vamos a describir estas pruebas de manera más detallada:

- Comprobaremos el funcionamiento de las cartas especiales, como puede ser la carta de cambio de la que color, en la que nos aseguraremos que, efectivamente, el color que predomina en ese momento en la partida se cambia al que el usuario desee.
- Debemos cerciorarnos de que se puede jugar una partida multijugador, es decir, que en la partida puedan jugar de 2 a 4 jugadores.
- Nos aseguraremos de que la partida acabe correctamente, es decir, finalice cuando hay un jugador que se queda sin cartas.

A su vez, aunque no se detalle en este documento, verificaremos que las funciones básicas del juego, como lanzar una carta, pasar al siguiente turno, robar una carta... se ejecutan correctamente.

Además de estas comprobaciones acerca del código, el Equipo de desarrollo ha decidido ha decidido crear las siguientes historias de usuario para este Sprint:

ID	HISTORIA DE USUARIO
#5	Como usuario quiero conocer los nombres de los otros jugadores de la partida
#6	Como usuario quiero poder guardar una partida en curso

Por otra parte, hemos decidido empezar a realizar los distintos diagramas UML del código que hemos realizado hasta ahora, tanto de secuencias, como de clases.

Asimismo, durante este Sprint nos documentaremos acerca de cómo hacer los test JUnit, e intentaremos realizar los primeros para ir comprobando el código.

La distribución de tareas en el Equipo de desarrollo se podrá observar con más detalle en el apartado [Sprint Backlog 2](#).

Sprint Planning 3

Objetivo del Sprint:

Teniendo en cuenta que la duración de este Sprint será de nuevo de dos semanas, hemos considerado que el trabajo principal durante este periodo de tiempo será realizar los diagramas de secuencias y de clases del código que hayamos hecho hasta el momento.

El Equipo de desarrollo ha decidido crear nuevas historias de usuario, que serán las siguientes:

ID	HISTORIA DE USUARIO
#7	Como usuario quiero poder cargar una partida que he guardado con anterioridad
#8	Como usuario quiero poder introducir mediante teclado un comando y que se ejecute en la partida

Con estas nuevas funcionalidades, el usuario podrá cargar una partida que ha guardado con anterioridad, y de este modo, continuará jugando esa partida. También podrá introducir un comando mediante teclado, y así agilizar el ritmo de la partida.

Además, el Equipo de desarrollo encargado del código, realizará las pruebas necesarias y los cambios pertinentes en el mismo para que éste funcione correctamente.

A su vez, para comprobar el código de una manera adecuada, seguiremos creando test de JUnit , y así podremos asegurarnos de que nuestro código es óptimo.

La distribución de tareas en el Equipo de desarrollo se podrá observar con más detalle en el apartado [Sprint Backlog 3](#).

Sprint Planning 4

Objetivo del Sprint:

Durante el Sprint anterior, aunque en un principio esta tarea no estaba programada, empezamos a realizar un prototipo en forma de boceto de lo que posteriormente será nuestra interfaz gráfica. Una vez acabado este boceto, hemos decidido que este Sprint lo dedicaremos a realizar la interfaz gráfica de nuestro juego.

Esta interfaz estará integrada, en un principio, por una pantalla en la que podremos elegir la opción de comenzar una nueva partida o cargar una anterior, y la pantalla principal en la que se mostrará la partida que se está llevando a cabo, en la que aparecerán las cartas...

Este modelo es incremental, es decir, primero comenzaremos implementando las pantallas de una manera un poco más sencilla y rústica, y luego iremos mejorando las mismas, añadiendo nuevos botones, nuevos diseños...

Gracias a las explicaciones que nos proporcionó el profesor en el último día de laboratorio, nos ayudó a entender mucho mejor el Modelo-Vista-Controlador. Así que durante este Sprint, también intentaremos corregir y mejorar nuestro código para que se adapte al MVC.

Además, seguiremos confeccionando test de JUnit para comprobar nuestro código.

A su vez, continuaremos realizando los diagramas UML de las nuevas clases que hagamos a la hora de realizar la interfaz gráfica.

Esta última tarea queda en duda, ya que no sabemos exactamente cuánto tiempo nos llevará realizar la interfaz gráfica, y en consecuencia, tampoco el tiempo que podremos dedicar a realizar estos diagramas.

La distribución de tareas en el Equipo de desarrollo se podrá observar con más detalle en el apartado [Sprint Backlog 4](#).

Sprint Planning 5

Objetivo del Sprint:

El Equipo de desarrollo ha decidido que durante este Sprint, realizaremos las estrategias del juego, es decir, añadiremos distintos niveles de dificultad a la hora de jugar contra la máquina.

Esto se resume en la siguiente historia de usuario:

ID	HISTORIA DE USUARIO
#9	Como usuario quiero poder jugar contra la máquina en diferentes dificultades
	NIVEL FÁCIL
	NIVEL MEDIO
	NIVEL DIFÍCIL

NIVEL FÁCIL

Estrategia 1: Lanzar la primera carta posible de la lista.

NIVEL MEDIO

Estrategia 2: Priorizar lanzar cartas "normales" (con número y color) a las cartas con acciones especiales que pueden ayudar más al final de la partida.

Estrategia 3: Priorizamos echar una carta del mismo color que la del montón.

NIVEL DIFÍCIL

Estrategia 2: (desarrollado anteriormente)

Estrategia 4: Si en el montón hay una carta ante la cual podemos echar 2 o más cartas de diferentes colores, echamos la carta de la cual tengamos más de ese mismo color.

Además, seguiremos elaborando tes de JUnits, con los que probaremos el código.

Asimismo, continuaremos con la creación de diagramas de clases y de secuencias.

Este Sprint coincide en gran parte con la Semana Santa, por lo que intentaremos aprovechar este periodo vacacional para ir confeccionando los documentos necesarios para la entrega final.

La distribución de tareas en el Equipo de desarrollo se podrá observar con más detalle en el apartado [Sprint Backlog 5](#).

Sprint Planning 6

Objetivo del Sprint:

Hemos decidido que durante este sprint, el avance con respecto al Product Backlog será implementar la siguiente historia de usuario:

ID	HISTORIA DE USUARIO
#10	Como usuario quiero poder jugar una partida en la red

Esta nueva funcionalidad permitirá al usuario poder jugar de manera online al 'UNO'. Preveamos que la implementación de esta nueva opción de juego nos puede parecer algo difícil, ya que es algo que no hemos visto con anterioridad. Por eso, hemos decidido que le preguntaremos una serie de dudas al profesor durante la siguiente sesión de laboratorio para que nos intente ayudar.

Seguiremos con la creación de test JUnit para probar el código que hemos desarrollado.

Asimismo, continuaremos creando nuevos diagramas de secuencias y de clases.

Además, intentaremos seguir elaborando el documento de Scrum necesario para la entrega final, y comenzar a redactar el documento de Diseño UML para evitar agobios y complicaciones de última hora.

La distribución de tareas en el Equipo de desarrollo se podrá observar con más detalle en el apartado [Sprint Backlog 6](#).

Sprint Planning 7

Objetivo del Sprint:

Estas dos semanas equivalen al último Sprint del proyecto.

Con respecto al Product Backlog, hemos decidido no añadir ninguna historia de usuario más. Durante este Sprint repasaremos los últimos detalles de la implementación del juego y realizaremos numerosas pruebas para comprobar el buen funcionamiento del mismo.

A su vez, seguiremos el mismo criterio comentado anteriormente con los JUnit.

El objetivo principal de este Sprint será acabar toda la documentación del proyecto, que es lo que nos requerirá más tiempo.

Debemos acabar de redactar el documento de Diseño UML y el de Scrum.

Una vez acabados todos los apartados descritos con anterioridad, haremos un repaso general del proyecto entero y lo entregaremos en las mejores condiciones posibles.

La distribución de tareas en el Equipo de desarrollo se podrá observar con más detalle en el apartado [Sprint Backlog 7](#).

Resumen Sprint Plannings

En esta sección se hará un breve resumen de cada uno de los Sprints Planning que hemos ido realizando, donde se mostrará su evolución a lo largo del proyecto.

Sprint 1

Al comienzo de este Sprint, como todavía no teníamos muy claro cómo debíamos usar la metodología ágil de Scrum, nos centramos únicamente en la creación y desarrollo de las cuatro primeras historias de usuario, con un reparto de tareas lo más equitativo que pudimos.

Sprint 2

Durante este Sprint, una vez adquiridos los conocimientos necesarios para entender mejor la metodología Scrum, decidimos que la mejor forma de organizarnos sería dividirnos las distintas tareas o bloques de tareas en subgrupos de trabajo.

De esta manera, algunos miembros del equipo serían los encargados de desarrollar el código, otros se ocuparían de la documentación...

El objetivo principal de esta semana era hacer numerosas pruebas acerca del código creado en el Sprint anterior, para así tener una base sobre la que poder trabajar.

Sprint 3

En este Sprint seguimos con el reparto de tareas que habíamos acordado en el Sprint anterior, y además, intentamos establecer una cantidad de objetivos que se amolde al tiempo que disponemos en estas dos semanas para realizar el proyecto.

Entre los objetivos de esta semana se encuentran desarrollar las historias de usuario 7 y 8, realizar diagramas UML, test de JUnit...

Sprint 4

Continuamos realizando el reparto de tareas como en los Sprints anteriores, ya que este método nos estaba funcionando.

El objetivo principal de este Sprint fue adaptar nuestro código al Modelo-Vista-Controlador y comenzar a crear la interfaz gráfica.

En este Sprint decidimos ampliar ligeramente la cantidad de objetivos a cumplir, ya que durante estas dos semanas dispusimos de una cantidad de tiempo mayor para trabajar en el proyecto.

Sprint 5

En este Sprint llegamos a la conclusión de que debíamos darle un ligero impulso al proyecto, ya que coincidía con las vacaciones de Semana Santa, dónde dispondríamos de un mayor número de horas para realizar el trabajo.

De este modo, procuramos dejar el reparto de tareas lo más especificado y claro posible, ya que éramos consciente de que durante este Sprint la comunicación entre los miembros del grupo empeoraría por la razón anteriormente comentada.

El objetivo principal de este Sprint era desarrollar las estrategias del juego (Historia de usuario 9).

Sprint 6

Durante este Sprint continuamos realizando el reparto de tareas del mismo modo que lo llevábamos haciendo desde el Sprint 2.

En este Sprint decidimos comenzar a redactar los documentos necesarios para la entrega final, con el fin de adelantar el trabajo y evitar agobio de cara al último Sprint.

Acabar los diagramas UML, los test de JUnit, y desarrollar la funcionalidad para poder jugar en red, eran otros de los objetivos marcados para este Sprint.

Sprint 7

El Sprint 7 se trata del último Sprint del proyecto. Por lo tanto, nuestro objetivo principal será terminarlo para poder entregarlo en las mejores condiciones posibles.

En este Sprint no añadimos ninguna Historia de usuario nueva.

Nos dedicamos principalmente a hacer numerosas pruebas y corregir posibles errores para asegurar la fiabilidad tanto del código del juego, como de los JUnit.

Decidimos emplear una gran parte del tiempo de estas dos semanas a realizar la documentación necesaria para la entrega final, tanto del documento Scrum, como el de diseño UML.

Conclusión final

El Sprint planning, como ya se podía vaticinar, ha sido una de las piedra angulares del proyecto pues fue la base del reparto de tareas entre los integrantes del grupo.

En un principio no sabíamos exactamente cómo dividir las tareas por lo que realizamos un reparto temporal, aunque pronto hicimos un reparto más concreto y eficiente que seguimos durante la práctica totalidad del proyecto.

En un principio, como resulta evidente, nos centramos en realizar los casos de uso nucleares del proyecto y las tareas principales giraron en torno a esto.

Más adelante, una vez desarrollado una base sólida pudimos ceñirnos la planificación mencionada que se centraba tanto en el código, en la realización de documentos, diagramas UML y test JUnit.

Es verdad que al final del proyecto esta organización se vio ciertamente modificada motivado por lo que nos quedaba por hacer, pero grosso modo nos fue de gran utilidad para tener claro lo que cada uno debía hacer en cada Sprint.

En conclusión, el Sprint Planning ha resultado primordial para que cada integrante del grupo tuviera una función clara y pudiéramos avanzar correctamente. Asimismo, la especialización que cada integrante ha tenido en una parte concreta del proyecto ha hecho que fuéramos capaces de realizar cada una de ellas de forma más eficiente.

Product Backlog

Como ya sabemos, el Product Backlog es una lista ordenada de todo lo conocido que podría ser necesario en el producto.

Enumera todas las características, funcionalidades, requisitos, mejoras acerca del producto en este caso, en forma de Historias de usuario: pequeñas historias acerca de las necesidades del usuario mientras usa el software que estamos desarrollando. Dicho de otra manera, son el conjunto de requisitos que debe tener el sistema a desarrollar.

A continuación, vamos a mostrar una tabla que representa el número de historias de usuario que hemos desarrollado en cada Sprint.

SPRINT	Nº DE HISTORIAS DESARROLLADAS
#1	4
#2	2
#3	2
#4	0
#5	1
#6	1
#7	0

Como se puede observar, durante nuestro proyecto no hemos creado un gran número de Historias de usuario. Esto se debe a que hemos decidido realizar una menor cantidad de historias, pero sin embargo, desarrollarlas a un amplio nivel de detalle, abarcando cada una de ellas grandes rasgos de la funcionalidad del juego.

Naturalmente, aunque en algunos Sprints no hayamos añadido ninguna historia de usuario, no quiere decir que no hayamos hecho ninguna tarea relacionada con el desarrollo del Product Backlog. Durante estos Sprints, nos hemos decidido centrar más en la modificación y corrección de historias ya creadas, o simplemente la nueva funcionalidad que queríamos añadir era muy compleja y abarcaba gran parte del tiempo de nuestro Sprint, por eso hemos decidido desarrollar un número menor de ellas.

Aunque lo vayamos a comentar más en detalle en el próximo apartado “Evolución del Product Backlog”, cabe destacar que durante el proyecto, hemos eliminado o fusionado algunas Historias de usuario:

- En el Sprint 2 decidimos fusionar las historias 2 (Como usuario quiero poder elegir el número de jugadores a participar) y 5 (Como usuario quiero conocer los nombres de los otros jugadores de la partida), ya que compartían funcionalidad. Sin embargo en el Sprint 5 decidimos eliminarlas, ya que al implementar los niveles de dificultad del juego, nos dimos cuenta que la historia de usuario 9 (Como usuario quiero poder jugar contra la máquina en diferentes dificultades) abarcaba todo lo que recogían estas dos historias.
- En el Sprint 5 eliminamos la historia 8 (Como usuario quiero poder introducir mediante teclado un comando y que se ejecute en la partida) ya que no aportaba casi nada al proyecto.

Evolución del Product Backlog

En este apartado vamos a trazar cuál ha sido la evolución del Product Backlog a lo largo del proyecto. Para ello, vamos a realizar una descripción acerca de en qué momento hicimos las historias de usuario, en qué momento han sido eliminadas (en el caso en el que hayan sido eliminadas), qué modificaciones se han realizado, y por qué.

Sprint 1

Las historias de usuario que desarrollamos en este Sprint fueron las siguientes:

ID	HISTORIA DE USUARIO
#1	Como usuario quiero jugar al UNO
#2	Como usuario quiero poder elegir el número de jugadores a participar
#3	Como usuario quiero jugar con las cartas (NORMALES y ESPECIALES)
#4	Como usuario quiero que las cartas se comporten adecuadamente

En el Sprint 1 nos dedicamos a darle la forma básica al juego. El usuario podía jugar con una serie de comandos que ejecutaba para lanzar las cartas que tuviese en ese momento.

Inicialmente la partida se seguía únicamente desde la consola, con símbolos y colores característicos del juego pero sin mucha complejidad en los gráficos.

En este sprint aún no teníamos implementado el modelo-vista-controlador.

Habíamos pensado en desarrollar estas historias de usuario en estas dos semanas de Sprint y logramos completar este objetivo. Esto no quiere decir que ésta vaya a ser la versión final de cada una de estas historias de usuario y de su código, ya que podrán ser modificadas en un futuro.

La evolución de las historias de usuario en cuanto a código de este Sprint vienen explicadas detalladamente en el apartado “Diseño y evolución de las Historias de usuario” del documento Diseño UML.

Cada historia de usuario viene desarrollada detalladamente en el apartado [Historias de usuario](#).

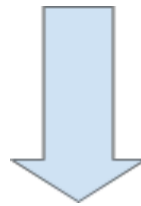
Sprint 2

Durante este Sprint desarrollamos las siguientes Historias de usuario:

ID	HISTORIA DE USUARIO
#5	Como usuario quiero conocer los nombres de los otros jugadores de la partida
#6	Como usuario quiero poder guardar una partida en curso

Al realizar la historia de usuario 5 (Como usuario quiero conocer los nombres de los otros jugadores de la partida), nos dimos cuenta de que compartía funcionalidades con la historia 2 (Como usuario quiero seleccionar el número de jugadores de la partida), por lo que decidimos fusionar las dos historias en una sola historia ya que en cuanto a código tienen grandes similitudes.

ID	HISTORIA DE USUARIO
#2	Como usuario quiero poder elegir el número de jugadores a participar
#5	Como usuario quiero conocer los nombres de los otros jugadores de la partida



ID	HISTORIA DE USUARIO
#5 (fusionada con #2)	Como usuario quiero conocer los nombres de los otros jugadores de la partida (aquí entra la selección de número de jugadores)

Conseguimos el objetivo de acabar de desarrollar estas historias de usuario en este Sprint.

La evolución de las historias de usuario en cuanto a código de este Sprint vienen explicadas detalladamente en el apartado “Diseño y evolución de las Historias de usuario” del documento Diseño UML.

Cada historia de usuario viene desarrollada detalladamente en el apartado [Historias de usuario](#).

Sprint 3

Desarrollamos las siguientes Historias de usuario durante este Sprint:

ID	HISTORIA DE USUARIO
#7	Como usuario quiero poder cargar una partida que he guardado con anterioridad
#8	Como usuario quiero poder introducir mediante teclado un comando y que se ejecute en la partida

Principalmente nuestra idea era implementar una funcionalidad que cargase una partida que habíamos guardado (Historia de usuario 7) con anterioridad gracias a la historia de usuario 6 (Como usuario quiero poder guardar una partida en curso) y crear una serie de comandos para que el usuario pudiera ejecutar la acción que desease con mayor facilidad y rapidez (historia 8).

Habíamos pensado que el desarrollo de estas dos nuevas historias de usuario nos iba a llevar las dos semanas del Sprint, cosa que así ha sido, ya que hemos logrado completar este objetivo.

Además, hemos retocado algunas otras partes del código ligadas a otras historias de usuario anteriores para mejorar el juego, pero no hemos cambiado las características principales de estas historias, por ello no las hemos considerado como modificadas.

La evolución de las historias de usuario en cuanto a código de este Sprint vienen explicadas detalladamente en el apartado “Diseño y evolución de las Historias de usuario” del documento Diseño UML.

Cada historia de usuario viene desarrollada detalladamente en el apartado [Historias de usuario](#).

Sprint 4

En estas dos semanas no añadimos ninguna historia de usuario nueva a nuestro Product Backlog. Esto se debe a que decidimos centrarnos en aprovechar este Sprint para adaptar nuestro código al Modelo-Vista-Controlador.

Esta tarea nos llevó más de lo esperado, ya que en un principio no entendíamos muy bien el concepto que representaba el Modelo-Vista-Controlador.

Fue en una sesión de laboratorio donde gracias a la explicación del profesor nos quedaron más claras las cosas que teníamos en nuestro código para adaptarlo a este patrón. Esto hizo que invirtieramos todo el tiempo de este Sprint en realizar esta tarea, cosa que nos impidió poder desarrollar nuevas Historias de usuario al Product Backlog.

Aunque no hayamos implementado funcionalidades nuevas durante estas dos semanas, hemos retocado el código perteneciente a historias de usuario anteriores para mejorar la calidad del juego.

Cabe destacar que al final de este Sprint decidimos añadir la siguiente historia de usuario al Product Backlog, pero no la comenzamos a desarrollar:

ID	HISTORIA DE USUARIO
#9	Como usuario quiero poder jugar contra la máquina en diferentes dificultades
	NIVEL FÁCIL
	NIVEL MEDIO
	NIVEL DIFÍCIL

La evolución de las historias de usuario en cuanto a código de este Sprint vienen explicadas detalladamente en el apartado “Diseño y evolución de las Historias de usuario” del documento Diseño UML.

Cada historia de usuario viene desarrollada detalladamente en el apartado [Historias de usuario](#).

Sprint 5

Durante este Sprint desarrollamos la siguiente historia de usuario:

ID	HISTORIA DE USUARIO
#9	Como usuario quiero poder jugar contra la máquina en diferentes dificultades
	NIVEL FÁCIL
	NIVEL MEDIO
	NIVEL DIFÍCIL

Como podemos comprobar en la anterior imagen, nuestra idea era tratar de implementar las estrategias del juego durante este Sprint.

Esta historia de usuario la habíamos definido y añadido al Product Backlog al final del Sprint anterior, pero hemos introducido algunas modificaciones que nos facilitan el desarrollo del código.

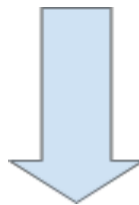
El desarrollo ha requerido una gran parte de nuestro tiempo debido a la amplia funcionalidad que se recoge en ella.

Hemos logrado cumplir el plazo de dos semanas que habíamos marcado para desarrollar esta historia, aunque hemos tenido grandes dificultades a la hora de comunicarnos como equipo, ya que este Sprint ha coincidido con el periodo vacacional de la Semana Santa.

Cabe recordar que en el Sprint 2, habíamos fusionado la historia de usuario 5 (Como usuario quiero conocer los nombres de los otros jugadores de la partida) con la historia de usuario 2 (Como usuario quiero seleccionar el número de jugadores de la partida).

Sin embargo, en este Sprint nos hemos dado cuenta de que la funcionalidad que recogían estas historias de usuario, que eran las de elegir el número de jugadores y sus nombres, las hemos contemplado al desarrollar la historia 9 (Como usuario quiero poder jugar contra la máquina en diferentes dificultades). Así que hemos decidido fusionar las historias de usuario 2 y 5 con la 9.

ID	HISTORIA DE USUARIO
#5 (fusionada con #2)	Como usuario quiero conocer los nombres de los otros jugadores de la partida (aquí entra la selección de número de jugadores)
#9	Como usuario quiero poder jugar contra la máquina en diferentes dificultades
	NIVEL FÁCIL
	NIVEL MEDIO
	NIVEL DIFÍCIL



ID	HISTORIA DE USUARIO
#9 (fusionada con la #2 y #5)	Como usuario quiero poder jugar contra la máquina en diferentes dificultades
	NIVEL FÁCIL
	NIVEL MEDIO
	NIVEL DIFÍCIL
	(Esta historia incluye la selección de número de jugadores y sus nombres)

A su vez, nos hemos dado cuenta de que la historia de usuario 8 (Como usuario quiero poder introducir mediante teclado un comando y que se ejecute en la partida), no añadía ninguna funcionalidad sustancial al proyecto, por lo que hemos decidido eliminarla.



ID	HISTORIA DE USUARIO
#8 (ELIMINADA)	Como usuario quiero poder introducir mediante teclado un comando y que se ejecuta en la partida

La evolución de las historias de usuario en cuanto a código de este Sprint vienen explicadas detalladamente en el apartado “Diseño y evolución de las Historias de usuario” del documento Diseño UML.

Cada historia de usuario viene desarrollada detalladamente en el apartado [Historias de usuario](#).

Sprint 6

En este Sprint desarrollamos la Historia de usuario número 10:

ID	HISTORIA DE USUARIO
#10	Como usuario quiero poder jugar una partida en la red

Durante este Sprint se nos ha requerido implementar esta historia de usuario. Añade la funcionalidad de poder jugar una partida en la red.

El desarrollo de esta historia nos ha resultado muy complejo ante la falta por parte de nuestro equipo de conocimientos previos relacionados con este tema.

En estas dos semanas no hemos modificado ninguna de las historias de usuario anteriores en términos de concepto o caracterización de las mismas, no de código.

La evolución de las historias de usuario en cuanto a código de este Sprint vienen explicadas detalladamente en el apartado “Diseño y evolución de las Historias de usuario” del documento Diseño UML.

Cada historia de usuario viene desarrollada detalladamente en el apartado [Historias de usuario](#).

Sprint 7

Durante este último sprint no hemos añadido ninguna historia de usuario nueva, por lo que el product backlog no ha aumentado. No obstante, dentro de los refinamientos llevados a cabo en este sprint, pese a no haber cambiado la funcionalidad de los mismos, lo cierto es que hemos retocado la implementación de algunos de ellos para dejarlos totalmente cerrados.

La evolución de las historias de usuario en cuanto a código de este Sprint vienen explicadas detalladamente en el apartado “Diseño y evolución de las Historias de usuario” del documento Diseño UML.

Cada historia de usuario viene desarrollada detalladamente en el apartado [Historias de usuario](#).

Sprint Backlog



Cabe recordar que el **Sprint Backlog** es el conjunto de tareas seleccionadas para realizar durante un Sprint.

Hemos decidido dividir esta sección por Sprints. Se mostrará una **tabla** con la distribución de las tareas que hemos realizado durante el Sprint, acompañado de los nombres de los integrantes del grupo que realizan la tarea, el estado en el que finalizó la tarea y una breve justificación de la distribución del trabajo realizado.

Debemos destacar que este formato de tablas lo comenzamos a hacer entre el Sprint 4 y 5, ya que el profesor nos recomendó que lo hiciéramos de esta manera para tenerlo más organizado. Así que decidimos pasar toda la información que teníamos de los Sprints anteriores a este formato de tablas para que todo quedara más unificado y visual.

Sprint Backlog 1

TAREAS	RESPONSABLE		ESTADO
#1 Como usuario quiero jugar al UNO			
	Mario	Lucía	ACABADO
	Beñat		ACABADO
#2 Como usuario quiero poder elegir el número de jugadores a participar			
Se ha añadido la funcionalidad de elegir el número de participantes en una partida	Mario	Beñat	ACABADO
	Lucía		ACABADO
#3 Como usuario quiero jugar con las cartas			
NORMALES	Javier	Lucía	ACABADO
ESPECIALES	Natalia	Lucía	ACABADO
#4 Como usuario quiero que las cartas se comporten adecuadamente			
Las cartas se deberán de comportar de una manera óptima, adaptándose a las reglas del juego	Mario	Alba	ACABADO
Redacción de la documentación para el vigente Sprint			
Se ha creado la documentación necesaria para el Sprint 1	Javier		ACABADO
	Natalia	Alba	ACABADO

A continuación dejamos un link que lleva a una tabla en la que se refleja la división de tareas y su estado.

[+ SPRINT BACKLOG 1](#)

Como hemos visto con anterioridad, en el [Sprint Planning 1](#), para este Sprint habíamos fijado como objetivo desarrollar las historias de usuario 1,2, 3 y 4 que se pueden observar en la tabla. Pensamos que la mejor manera de distribuir el trabajo sería desarrollar el código de cada historia en grupos de 2 o 3 personas, para que así todos realizáramos el mismo volumen de trabajo.

Sprint Backlog 2

TAREAS	RESPONSABLE		ESTADO
#5 Como usuario quiero conocer los nombres de los otros jugadores de la partida			
Se ha añadido la funcionalidad de poder ver el nombre de los otros usuarios que participan en la partida	Mario	Lucía	ACABADO
#6 Como usuario quiero poder guardar una partida en curso			
Se ha añadido la funcionalidad de poder guardar una partida que está en curso	Mario	Lucía	ACABADO
Creación de algunos JUnits			
Se han creado test de JUnit para probar el código	Beñat		ACABADO
Elaboración de los diagramas UML			
Diagramas de secuencias	Javier	Natalia	ACABADO
	Mario	Lucía	ACABADO
	Alba	Beñat	ACABADO
Diagramas de clases	Alba		ACABADO
Redacción de la documentación de la refactorización			
Se ha creado un documento donde se muestra la refactorización en forma de diagramas UML	Natalia		ACABADO
Redacción de la documentación para el vigente Sprint			
Se ha creado la documentación necesaria para el Sprint 2	Javier		ACABADO

A continuación dejamos un link que lleva a una tabla en la que se refleja la división de tareas y su estado.

📅 SPRINT BACKLOG 2

En el [Sprint Planning 2](#) se puede observar que para este Sprint nos habíamos fijado como objetivo en términos de Product Backlog, realizar las historias de usuario 5 y 6. Decidimos que lo mejor sería que dos integrantes del grupo se encargaran del desarrollo de estas historias, y los demás se ocuparan de realizar la documentación, los diagramas UML, test de JUnit...

Al final de este Sprint decidimos añadir la tarea de redactar el documento de la refactorización del código.

Sprint Backlog 3

TAREAS	RESPONSABLE		ESTADO
#7 Como usuario quiero poder cargar una partida que he guardado con anterioridad			
Se ha añadido la funcionalidad de poder cargar partidas guardadas anteriormente	Mario	Lucía	ACABADO
#8 Como usuario quiero poder introducir mediante teclado un comando y que se ejecute en la partida			
Se han añadido comandos para que el usuario pueda elegir la funcionalidad del juego que quiera ejecutar de una manera más ágil	Mario	Lucía	ACABADO
Creación de algunos JUnits			
Se han creado test de JUnit para probar el código		Beñat	ACABADO
Elaboración de los diagramas UML			
Diagramas de secuencias	Alba	Javier	ACABADO
Diagramas de clases		Alba	ACABADO
Creación de un primer prototipo de diseño			
Se ha creado un boceto o prototipo de lo que posteriormente será nuestra interfaz gráfica		Natalia	ACABADO
Redacción de la documentación para el vigente Sprint			
Se ha creado la documentación necesaria para el Sprint 3	Natalia	Javier	ACABADO

A continuación dejamos un link que lleva a una tabla en la que se refleja la división de tareas y su estado.

 [SPRINT BACKLOG 3](#)

Podemos apreciar los objetivos que habíamos definido para este Sprint en el [Sprint Planning 3](#). Nos hemos repartido el trabajo de una forma bastante similar a la del Sprint 2.

Dos miembros del equipo se han encargado de desarrollar las historias de usuario 7 y 8 mientras que el resto se han ocupado de labores como la creación de test de JUnit, diagramas UML, redacción de la documentación...

Al final de este Sprint decidimos añadir la tarea de realizar el prototipo de lo que será posteriormente nuestra interfaz gráfica.

Sprint Backlog 4

TAREAS	RESPONSABLE		ESTADO
Adaptacion al Modelo Vista Controlador			
Se ha corregido el código para que se adapte mejor al MVC	Mario	Lucía	ACABADO
Creación de GUI			
Clase ColorSelectionDialog	Mario	Lucía	ACABADO
Clase EndPanel	Mario	Natalia	ACABADO
Clase InitialPanel	Alba		ACABADO
Clase LoadPanel	Mario	Lucía	ACABADO
Clase MainGamePanel	Mario	Lucía	ACABADO
Clase MainWindow	Mario	Lucía	ACABADO
Clase MidSettingsPanel	Natalia		ACABADO
Clase NumPlayersPanel	Javier		ACABADO
Clase SavePanel	Lucía		ACABADO
Clase SettingsPanel	Natalia	Beñat	ACABADO
Clase UserNamePanel	Alba		ACABADO
Clase PanelBackground	Javier		ACABADO
Comprobación de algunos JUnits			
Se han refinado test de JUnit para probar el código	Beñat		ACABADO
Redacción de la documentación para el Sprint actual			
Se ha creado la documentación necesaria para el Sprint 4	Javier		ACABADO

A continuación dejamos un link que lleva a una tabla en la que se refleja la división de tareas y su estado.

[+ SPRINT BACKLOG 4](#)

En el apartado [Sprint Planning 4](#) aparecen los objetivos fijados para este Sprint.

Se puede ver que en este Sprint no hemos desarrollado ninguna historia de usuario nueva. Nos hemos distribuido el trabajo para realizar la interfaz gráfica, que ha sido nuestro principal objetivo durante este Sprint.

También hemos adaptado el código al M-V-C y hemos realizado otras tareas como el refinamiento de algunos test JUnits, la redacción de la documentación...

Sprint Backlog 5

TAREAS	RESPONSABLE		ESTADO
#9 Como usuario quiero poder jugar contra la máquina en diferentes dificultades			
Nivel FÁCIL	Mario	Lucía	ACABADO
Nivel MEDIO	Mario	Lucía	ACABADO
Nivel DIFÍCIL	Mario	Lucía	ACABADO
Creación de algunos JUnits			
Se han creado test de JUnit para probar el código		Beñat	ACABADO
Elaboración de los diagramas UML			
Diagramas de secuencias		Alba	ACABADO
Diagramas de clases		Alba	ACABADO
Revisión y modificación de las historias de usuario			
Se han corregido algunas historias de usuario que no estaban bien desarrolladas		Natalia	ACABADO
Redacción de la documentación para el vigente Sprint y el proyecto en general			
Se ha creado la documentación necesaria para el Sprint 5 y se ha empezado a redactar la de la entrega final		Javier	ACABADO

A continuación dejamos un link que lleva a una tabla en la que se refleja la división de tareas y su estado.

 [SPRINT BACKLOG 5](#)

Durante este Sprint, como podemos comprobar en el [Sprint Planning 5](#), definimos el objetivo (en términos de Product Backlog) de desarrollar la historia 9. Para esta tarea se han encargado dos integrantes del grupo.

El resto del equipo se ha ocupado de la creación de test JUnit y diagramas UML, así como de la revisión y modificación de algunas historias de usuario.

A su vez, hemos decidido empezar a redactar el documento Scrum perteneciente a la entrega final, debido a que hemos gozado de un mayor cantidad de tiempo libre, ya que este Sprint ha coincidido con un periodo vacacional (Semana Santa).

Sprint Backlog 6

TAREAS	RESPONSABLE		ESTADO
#10 Como usuario quiero poder jugar una partida en la red			
Se ha diseñado el código necesario para poder jugar una partida online	Mario	Lucía	ACABADO
Creación de algunos JUnits			
Se han creado test de JUnit para probar el código	Beñat		ACABADO
Elaboración de los diagramas UML			
Diagramas de secuencias	Alba		ACABADO
Diagramas de clases	Natalia		ACABADO
Creación de historias de usuario			
Se ha creado una nueva historia de usuario	Natalia		ACABADO
Se ha redactado la funcionalidad de la nueva historia	Lucía		ACABADO
Redacción de la documentación para el Sprint actual			
Se ha creado la documentación necesaria para el Sprint 6	Javier		ACABADO
Redacción de la documentación de Diseño UML			
Comentar patrones y hacer dibujos	Natalia		ACABADO
Redactar la parte de clases y añadir fotos de la vista	Lucía		ACABADO
Redactar la parte de las Historias de usuarios	Alba		ACABADO
Corrección del documento	Mario	Beñat	ACABADO
Redacción del documento de Scrum Final			
Se han realizado avances en el documento de Scrum Final	Javier		ACABADO

A continuación dejamos un link que lleva a una tabla en la que se refleja la división de tareas y su estado.

[+ SPRINT BACKLOG 6](#)

En cuanto a términos de Product Backlog se refiere, durante este Sprint, tal y como viene reflejado en el [Sprint Planning 6](#), se ha desarrollado la historia de usuario 10, que consiste en poder jugar una partida en la red. De esta tarea se han encargado dos integrantes del equipo.


Hemos completado otras tareas como la creación de diagramas UML o de test JUnit o la redacción de la documentación del vigente Sprint.

Cabe destacar que, hemos comenzado a redactar el documento de Diseño UML, y hemos continuado con el de Scrum. De este modo, intentamos evitar que se nos acumule el trabajo de cara al último Sprint.

Sprint Backlog 7

TAREAS	RESPONSABLE		ESTADO
Realización de pruebas y revisión del código			
Realizaremos numerosas pruebas para comprobar el correcto funcionamiento del juego	Mario	Lucía	ACABADO
Realización de pruebas y revisión de los test JUnit			
Se probarán minuciosamente los test	Beñat		ACABADO
Redacción del documento Diseño UML			
Se ha conseguido terminar el documento Diseño UML	Lucía		ACABADO
	Natalia	Alba	ACABADO
	Beñat	Mario	ACABADO
Revisión de detalles de la interfaz gráfica			
Se refinan detalles para mejorar la interfaz gráfica	Natalia	Alba	ACABADO
Redacción de la documentación para el Sprint actual			
Se ha creado la documentación necesaria para el Sprint 7	Javier		ACABADO
Redacción del documento de Scrum Final			
Se ha conseguido acabar el documento Scrum Final	Javier		ACABADO

A continuación dejamos un link que lleva a una tabla en la que se refleja la división de tareas y su estado.

 **SPRINT BACKLOG 7**

En términos de Product Backlog, no hemos añadido ninguna historia de usuario nueva, tal y como habíamos propuesto en el [Sprint Planning 7](#).

Hemos acabado los documentos de Scrum y Diseño UML necesarios para la entrega final.

A su vez hemos refinado y comprobado algunos detalles del código para asegurarnos de que el juego funciona correctamente.

Trabajo realizado por cada componente del equipo

- Mario Calvarro Marines:

Encargado del diseño arquitectónico del programa. Ha diseñado en términos generales el MVC y se ha encargado de asegurar que el patrón Cliente-Servidor ha sido implementado correctamente.

En cuanto al código, ha realizado la implementación de las clases relacionadas con el controlador, algunos métodos de la clase *Game*, el servidor y algunas partes de la vista por interfaz gráfica.

En cuanto a documentación, ha escrito los comentarios de Javadoc sobre *Controller*, los *Builders*, el servidor, ciertos métodos de *Game* y la clase *Storage*.

- Lucía Alonso Mozo:

Scrum Master del proyecto. Encargada de la programación de la funcionalidad y lógica del juego, así como de gestionar el proceso Scrum y eliminar impedimentos que puedan afectar a la entrega del producto. Para ello, ha organizado al equipo en funciones y tareas, con la finalidad de que todos trabajen de forma organizada, permitiendo cumplir plazos de entrega y un buen resultado final.

En cuanto al código, ha realizado la implementación de las clases relacionadas con los *gameobjects*, las estrategias de la inteligencia artificial, la vista por interfaz gráfica y algunos métodos de *Game*.

En cuanto a la documentación, ha escrito los comentarios de Javadoc sobre la clase *Game*, y los *gameobjects*. Además, la parte del documento de diseño de UML de la arquitectura, diseño y evolución de la interfaz del usuario y diseño y evolución de las clases principales del modelo.

- Natalia Rodríguez Caballero:

Encargada de la estética del juego, así como de la documentación de los sprints, además de la realización de diagramas de clase ilustrativos para la evolución del proyecto.

En cuanto al código ha mejorado la vista por interfaz gráfica para hacer el juego más atractivo visualmente e implementado la lógica de algunos de los paneles pertenecientes a la misma.

En cuanto a documentación, ha escrito los comentarios de Javadoc sobre las clases pertenecientes a los subpaquetes de *view*. Por otra parte redacta las partes del documento de interfaces y patrones, así como ha completado ciertas

partes pobres en la evolución de diseño y evolución tanto de historias de usuario como de las clases principales del modelo.

- **Javier Amado Lázaro:**

Encargado de la documentación de cada Sprint. En términos generales, se ha ocupado de la documentación de todos los Sprints a lo largo del proyecto.

En las primeras etapas del proyecto creó alguna clase de la interfaz gráfica y elaboró algunos de los primeros diagramas UML del proyecto.

Su función principal ha sido la redacción de la documentación relacionada con cada una de las reuniones que hemos realizado en cada Sprint (Sprint Review, Sprint Planning, Sprint Retrospective...).

Ha sido el encargado de redactar el documento **Scrum** necesario para la entrega final.

- **Alba Bautista Nuñez:**

Encargada del diseño UML del proyecto, su función principal en cada Sprint era realizar los diagramas de los cambios realizados en el Sprint anterior.

Se ha encargado de hacer los diagramas de secuencia de las historias de usuario, mostrando la evolución y los cambios realizados en el juego, utilizando la extensión PlantUML de Eclipse la cual se basa en un código bastante sencillo para realizarlos. Además ha comentado cada uno de ellos de la forma más detallada posible para su mejor comprensión tanto de funcionalidad como de estructura.

Por otra parte, ha realizado gran parte de los diagramas de clases, tanto de la parte de las historias de usuario, como de la parte de la evolución de las clases, mostrando los cambios realizados en cada Sprint. Para realizar los diagramas de clases ha utilizado la aplicación de Modelio y se ha encargado de enseñar a alguno de sus compañeros de proyecto como se realizaban, para que ellos mismos pudiesen hacer alguno si era necesario.

Por último, ha realizado los diagramas necesarios para mostrar la Vista de Uso y redactó las instrucciones con sus respectivas imágenes de como descargar el juego y como inicializarlo para que el usuario pueda jugar.

- **Beñat Perez de Arenaza Eizaguirre:**

Encargado de la implementación de los test JUnit además de colaborar en crear ciertas clases del *Game* y refinar otras.

En un principio implementó algunas de las clases nucleares del funcionamiento del juego, como es el caso de *Game* o *CardList*, además de crear sus correspondientes diagramas de clase y de secuencia.

Además, parte de su función ha residido en compaginar la simultaneidad de realizar los test de las nuevas clases con la colaboración en ciertos aspectos del código como la creación y refinamiento de componentes visuales así como el ajuste de algunas de las estrategias.

Asimismo, este rol también se ha podido ver en la realización de la documentación del proyecto, quien tuvo la responsabilidad de completar partes del documento “Scrum final” además de otras partes del “Diseño UML”, centrado en las clases con las que tuvo mayor contacto ya sea en su creación o en su refinamiento.

Incidencias

En este apartado, queremos comentar un aspecto que queremos aclarar acerca de nuestro proyecto.

Se trata de que en el apartado de Git - Insights - Contributors, los datos que aparecen no reflejan la realidad de lo que ha sido el trabajo de los integrantes del grupo a lo largo del proyecto.

Esto se debe a que han habido ocasiones en las que un grupo desarrollaba código en un mismo ordenador, y por lo tanto solo le contabilizaba la realización del trabajo al propietario del ordenador. Además, en algunos casos no se nos han contado todos los commits que hemos realizado debido a que teníamos esta opción desactivada.

Por ello, queremos dejar claro que durante todo el proyecto hemos intentado repartir el trabajo de la manera más equilibrada posible, y por lo tanto, estos datos no reflejan una imagen real de nuestro proyecto.