

Documentación Scrum

En el presente documento quedarán recogidos y organizados los diferentes sprints del proyecto, con toda la información al respecto del Product Backlog, los diferentes Sprint Backlog y los avances y problemas de cada uno de los sprints.

Para más claridad y uniformidad, como equipo se tomó la decisión de realizar Sprints de dos semanas de duración, con reuniones presenciales todos los miércoles (sesiones que se aprovecharán para hacer sprint planning o sprint retrospective entre otras cuestiones). Además cada 1 ó 2 sprints se llevará a cabo una reasignación de los roles de *Scrum Master* y *Product Owner* para que todos podamos conocer, de primera mano, las ventajas e inconvenientes de cada rol.

Sprint 1

08/10/25 - 15/10/25

Sprint Planning

Este primer sprint, debido a las circunstancias propias del comienzo del proyecto, durará solamente una semana en lugar de las dos que queremos definir como estándar.

En esta primera reunión de equipo los objetivos eran claros: asignar el rol de cada uno de los integrantes, poner en común las ideas sobre cómo desarrollar y trabajar en el proyecto y empezar a designar tareas principales.

Dado que es un primer Sprint previo a la fase de desarrollo de la aplicación y de la documentación es un tanto distinto a los demás. Pese a ello se mantuvo una metodología ágil en todo momento.

Distribución de roles:

Como hemos comentado, cada uno/dos sprints se reasignará el rol de cada miembro del equipo. En este primer Sprint se decidió como *Scrum Master* a Santiago Moreno y a Iván Bezares como *Product Owner*. El resto de miembros formará el *equipo de desarrollo*.

Cabe aclarar que debido a la simplicidad del trabajo a realizar, sobre todo en estas primeras etapas del proyecto, se decidió que aquellos compañeros con estos roles destacados serían los designados para realizar precisamente las tareas de documentación de la metodología Scrum así como la planificación de las reuniones. Esto siempre en añadido a las funciones esenciales de comunicación con el cliente del *PO* y de facilitación del *SM*.

Designación de objetivos y tareas:

Al ser la primera reunión, antes de realizar una asignación de tareas es necesario crear un *Product Backlog* en el que poder basar los posteriores *Sprint Backlogs*.

En primer lugar pusimos en común una base de Scrum en la que estuviéramos todos de acuerdo al respecto de tener un *Product Backlog* distribuido en diferentes historias que además fuera flexible a cambios futuros etcétera.

Entonces pudimos poner en común las primeras historias necesarias para comenzar el proyecto. Entre las ideas concretadas aparecieron:

- Determinar con exactitud los requisitos funcionales y no funcionales de la aplicación en base a la transcripción de la charla.
- Generar un diagrama de casos de uso para apoyar los requisitos funcionales y empezar a plantear la estructura a nivel de uso de la aplicación.

Además de estas historias pertenecientes al propio Backlog, se propuso como tarea la documentación de las reuniones (SM) y la formalización del Backlog (PO).

Como es evidente, estas primeras historias no conforman la totalidad del Product Backlog. Una vez más las fases tan tempranas del desarrollo no permiten desarrollar más estos aspectos de Scrum hasta que se estudie el resto de modelos, diagramas y formas de desarrollo. A lo largo del presente documento se irán añadiendo, sprint a sprint, nuevas historias al Backlog principal.

Una vez determinadas algunas tareas pudimos generar una especie de Sprint Backlog, sin serlo del todo, en el que se determinan las tareas a realizar por el equipo a lo largo del Sprint. De esta forma la historia primera (requisitos funcionales) se distribuyó en:

1. Normalización y limpieza de la transcripción a formato pdf, separando preguntas de respuestas.
2. Obtención de requisitos funcionales.
3. Obtención de requisitos no funcionales.
4. Formalización en documento de todos los requisitos, organizados y numerados. Por otro lado, para el diagrama de casos se distribuyó en:
 1. Obtención de actores.
 2. Obtención de casos de uso principales.
 3. Relación entre los distintos casos.
 4. Creación de documentación del desarrollo y del diagrama.

De esta forma quedaban ya totalmente definidas las tareas a realizar. Así el equipo de desarrollo se dividió en dos grupos principales:

- Subgrupo 1: Iván, Adrián, Marcos, Alejandro, Daniel y Abel. Encargados de la parte de requisitos.
- Subgrupo 2: Blas, Santiago, Jesús, Fco. Javier y Julián. Encargados del diagrama de casos de uso.

En la propia reunión de planificación se empezó a trabajar en las tareas designadas.

Desarrollo del Sprint

A lo largo de la semana cada equipo fue desarrollando las tareas. Dada la dependencia entre ellas, finalmente se fueron desarrollando de forma secuencial (primero requisitos, después diagrama) y todo el equipo participó en ambas partes, aunque con mayor presencia en la designada durante la planificación.

Como es razonable para las dimensiones del proyecto, no fue necesario realizar una reunión diaria (daily) presencial. Sin embargo se creó un grupo de trabajo por el que se fueron comentando diariamente los avances en las diferentes tareas.

Sprint Review

Al final del sprint y al principio de la reunión semanal del Miércoles realizamos el sprint review en el que comentamos todas las tareas realizadas. Dada la facilidad de esta primera semana la mayoría de las tareas propuestas fueron completadas con éxito, de forma que los requisitos funcionales y no funcionales quedan plenamente definidos y se ha generado el diagrama de casos de uso. Sin embargo la documentación del diagrama de casos de uso queda pendiente para el siguiente Sprint.

Sprint Retrospective

Finalmente para cerrar el primer Sprint de todo el proyecto destinamos 10 minutos de charla para comentar cómo ha ido, qué cosas han sido positivas y qué cosas creemos que se podrían mejorar.

Uno de los detalles comentados es que una de las tareas del sprint, concretamente dentro del desarrollo de los casos de uso a la hora de definir las relaciones entre los distintos casos, había requerido más tiempo del programado inicialmente dado que poseía una mayor complejidad de la prevista.

Además de esto, a nivel de Scrum el desempeño del SM es positivo y el equipo está de acuerdo con la estructura de división de tareas por grupos planteada.

En general esta primera retrospective fue bastante breve dada la falta de tareas complejas a realizar por lo que comenzamos a plantear el segundo sprint.

Sprint 2

15/10/25 - 29/10/25

Sprint Planning

En este segundo sprint nuestra metodología de trabajo fue una extensión de la empleada en el primero, pero más refinada y mejor aplicada debido a una mayor cantidad de tareas a realizar lo cuál permitió una mejor división. Los roles de Product Owner y Scrum Master se mantuvieron con Iván Bezares Y Santiago Moreno respectivamente, pues debido a la ligereza del primer sprint no vimos necesario un cambio en este momento.

Designación de objetivos y tareas:

Los objetivos principales de este sprint responden a dos historias una vez más relacionadas con la fase de diseño de nuestra aplicación:

- Generar un diagrama Entidad-Relación con el que modelar la capa de datos de nuestra aplicación
- Generar un diagrama de Clases que modele el funcionamiento básico del sistema

Estas dos historias del product backlog fueron descompuestas en las siguientes tareas para el sprint backlog:

Diagrama Entidad-Relación:

1. Definición de entidades y sus atributos
2. Definición de las relaciones entre las mismas
3. Desarrollo del diagrama con el software Umbrello
4. Elaboración de la documentación correspondiente Diagrama de Clases:
 1. Definición de las clases fundamentales del sistema y su jerarquía basada en POO
 2. Desarrollo del diagrama con el software Umbrello
 3. Elaboración de la documentación correspondiente

Además de las historias ya tratadas, este sprint tuvo algunos objetivos adicionales. En primer lugar, era necesario finalizar la documentación del diagrama de casos de uso, tarea que quedó pendiente del sprint anterior. Por otro lado, sabíamos que al final del sprint nos enfrentaríamos a una primera evaluación de nuestro proyecto, motivo por el cual surgen un par de tareas adicionales

1. Preparación de la presentación
2. Unificación del trabajo realizado hasta el momento y ajustes para la entrega

De forma semejante al primer sprint, nos dividimos en 2 subgrupos para la realización de las tareas relacionadas con las dos historias principales:

- Subgrupo 1: Abel, Daniel, Jesus, Julian y Santiago. Encargados del diagrama de clases
- Subgrupo 2: Adrián, Alejandro, Blas, Ivan, Marcos y Fco. Javier. Encargados del diagrama de Entidad-Relación

Las tareas dentro de cada subgrupo fueron repartidas de manera independiente sin intervención de SM. Cuando los subgrupos lo veían oportuno, liberaban a miembros para

que comenzasen a trabajar en las tareas adicionales, como la documentación de los casos de uso de la que encargaron Alejandro y Daniel por poner un ejemplo.

Desarrollo del Sprint

En esta ocasión, ambos subgrupos pudieron paralelizar buena parte de su trabajo, pues estos diagramas, si bien están relacionados y fue necesario cierta comunicación entre ambos equipos para asegurarse que no existía ninguna incoherencia arbitrario entre los resultados finales, podían realizarse independientemente.

Como se ha sugerido en el apartado anterior, llegado cierto punto en el sprint algunos miembros del equipo abandonaron sus respectivos subgrupos, pues su contribución ya había sido completada, para asignarse una nueva tarea.

Sprint Review

Podemos decir que este sprint ha sido exitoso. No solo se ha avanzado en el Product Backlog con la realización de los diagramas de clase y de entidad-relación, sino que además ha sido posible saldar las tareas pendientes del sprint anterior y preparar una primera presentación con el cliente. No dejamos ninguna tarea sin terminar. No obstante, reconocemos la posibilidad de tener que en un futuro volver y ajustar algún aspecto, especialmente en lo relativo al diagrama de clase.

Sprint Retrospective

Este sprint retrospective resultó considerablemente más largo que el del primer sprint, lo cual era de esperar una vez considerada la mayor carga de trabajo y duración del mismo. En líneas generales, podemos decir que el equipo ha quedado contento con el trabajo realizado, sin embargo han habido algunos puntos que son susceptibles a mejoras.

En primer lugar, hemos de comentar la división de tareas en el sprint planning. Si bien hasta ahora ha sido funcional y hemos podido mejorar la división de tareas respecto al sprint anterior, consideramos que este aún no ha sido especialmente satisfactorio. La división en subgrupos ha resultado práctica, pero quizás nos haya impedido alcanzar la máxima eficiencia pues no todos los miembros tenían una tarea claramente designada en todo momento, sino que había puntos donde la división de tareas quedaba ofuscada por la división en subgrupos.

A pesar de todo, creemos que esta forma de organización ha sido la adecuada para esta primera fase de diseño de la aplicación, en la que era necesario que múltiples miembros compartiesen su opinión para así llegar a un diseño que nos satisfaga a todos. De cara al futuro, buscamos que la repartición de tareas resulte más clara, y esperamos que esto sea más fácil de lograr en la fase de desarrollo e implementación del diseño ya realizado.

Por otro lado, también se comentó la posibilidad de crear un nuevo canal de comunicación. Hasta ahora, más allá de las reuniones presenciales, solo hemos tenido un grupo de

Whatsapp para comunicarnos. Si bien este nos ha permitido organizarnos y tener conversaciones semejantes a lo que sería un daily Scrum, podríamos emplear un canal adicional, usando una aplicación como discord u otra semejante. Esto nos permitiría una comunicación más limpia y más cómoda, especialmente cuando se crean subgrupos de más de 3 personas para llevar a cabo una tarea.

Sprint 3

29/10/25 - 05/11/25

Sprint Planning

En el tercer Sprint los objetivos eran claros: por un lado preparar y refinar la presentación que habíamos estado preparando y comenzar con las tareas posteriores. La aparición de nuevas tareas permitió una mejor división que combinada con un cambio en los roles dio un soplo de aire fresco al equipo que afrontó con motivación la misión del sprint.

La división de tareas pudo entonces ser algo más fragmentada y conseguimos una mayor centralización además de procurar mejorar los fallos de los sprints anteriores.

Distribución de roles:

Como se comentó con anterioridad, cada cierta cantidad de sprints realizaremos cambios de roles. Tras dos sprints con una cantidad sustancial de trabajo decidimos cambiar tanto de Product Owner como de Scrum Master, en este caso siendo Adrián Romero el Product Owner y Abel Fernández el Scrum Master.

El resto del equipo se mantiene como un equipo de desarrollo por ahora unificado. Como sucedió en la asignación previa, en principio las tareas de PO y SM no son suficientes para ocupar todo su tiempo de forma que también formarán parte del equipo de desarrollo aunque desarrollando las tareas específicas de sus roles.

Designación de objetivos y tareas:

De cara al Product Backlog del proyecto surgieron nuevas historias importantes que deberán ser atendidas como lo puede ser la generación de los diagramas de secuencia. Añadido a esto quedaba pendiente terminar de preparar la presentación del proyecto para el 30/10.

De esta forma designamos a una parte del equipo (aquella que iba a formar parte de la presentación oral) a terminar de pulir los detalles de las propias diapositivas y a preparar el discurso mientras que el resto del equipo comenzaría a preparar los diagramas de secuencia.

Como se ha comentado anteriormente, decidimos hacer una distribución de tareas más granular y centralizada de forma que, en lugar de designar en general al otro grupo de desarrolladores la tarea del diagrama de secuencia decidimos poner en común todos los diagramas a realizar y asignar a cada miembro un diagrama. Así el sprint Backlog resultante lucía de la siguiente manera:

1. Formalizar la presentación y preparar el discurso. (M)
2. Realizar diagramas de secuencia:
 - Diagrama “Enviar Spam”. (S)
 - Diagrama “Recordatorio Evento”. (S)
 - Diagrama “Alta Usuario”. (S)
 - Diagrama “Crear Evento”. (M)
 - Diagrama “Comprar Entrada”. (L)

De esta forma cada miembro del equipo decidió una serie de tareas a realizar y daba comienzo el sprint.

Desarrollo del Sprint

La presentación fue realizada con éxito el día 30/10, en el que hicimos una pequeña reunión para poner un poco en común la primera parte del trabajo, aunque se comenta con más desarrollo ya en el sprint retrospective del sprint. Una vez terminada esa parte del sprint backlog la parte del equipo destinada a esta tarea pudo ayudar a realizar los diagramas de secuencia.

Gracias a la nueva organización planteada surgieron menos dudas en el desarrollo y cada miembro pudo realizar su parte del trabajo con más facilidad. De nuevo estos aspectos serían discutidos en el sprint retrospective.

Sprint Review

De cara a la reunión para la revisión de este sprint surgieron tanto noticias positivas como negativas.

Por una parte la presentación fue un éxito y estuvimos contentos con el resultado obtenido y el trabajo realizado, habiendo podido completar el objetivo de comentar todos los puntos importantes del desarrollo dentro del margen de los 10 minutos de presentación que teníamos.

Sin embargo, tras esta presentación hubo un decaimiento en el ritmo de trabajo que hizo que las tareas del diagrama de secuencias quedaran incompletas. Es cierto que la mayoría de diagramas excepto el de “Comprar Entrada” fueron completados aunque con algunos errores que deberán ser corregidos en el siguiente sprint.

De esta forma la mayoría de tareas fueron completadas pero quizás por una organización insuficiente las tareas más complejas quedaban pendientes para el siguiente sprint. Cabe destacar que durante la realización de los nuevos diagramas de secuencia se detectaron fallas en el diagrama de casos de uso, lo cual se comentó en la reunión de cara a prepararlo mejor para el siguiente sprint.

Sprint Retrospective

En este tercer sprint, debido a los cambios organizativos, de roles y a la compleción del primer gran hito del proyecto; el sprint retrospective fue aún más largo de lo habitual.

En primer lugar se comentó y se agradeció en general el uso de las nuevas herramientas de comunicación creadas, en este caso el canal de discord, que mejoró y facilitó en general la comunicación entre miembros del equipo.

Por otro lado los resultados de la organización planteada más específica y centralizada dieron sus frutos y en general había consenso en la mejora que provocaba de cara a tener claro quién hace qué, aunque aún se podía mejorar y quizás se debería plantear mejor la asignación de recursos a cada tarea dependiendo de la dificultad de estas.

Finalmente se propuso para el siguiente sprint dar más peso organizativo tanto al SM como al PO para la designación de tareas, consiguiendo así un trabajo más centralizado. Además estos roles, que hasta ahora apenas se dedicaban a la redacción del presente documento se volcaron más en su labor de ayudar al desarrollo de las tareas solventando todos los posibles problemas e inconvenientes que surjan.

Sprint 4

05/11/25 - 12/11/25

Sprint Planning

Para este cuarto sprint se presentaban varios retos por delante. Por un lado había que empezar a trabajar en la interfaz de usuario y por el otro había que completar todas las tareas pendientes alrededor del diagrama de casos de uso y los diagramas de secuencia.

Siguiendo lo comentado en el último sprint retrospective se procuró llevar a cabo una división de tareas más específica y clara de cara a facilitar el trabajo.

Designación de objetivos y tareas:

Una vez más, con el avance de las semanas van apareciendo nuevos requerimientos en nuestro Product Backlog, en esta ocasión relacionados con la interfaz de usuario, teniendo

que preparar unas maquetas con Penpot para exponer una idea inicial de los gráficos. Sumado a esto había que cubrir todas las tareas pendientes de sprints anteriores.

El sprint Backlog resultante fue el siguiente:

1. Completar el diagrama de secuencias de “Comprar Entrada”. (L)
2. Arreglar los fallos detectados en el diagrama de casos de uso junto con su anexo. (M)
3. Juntar todos los diagramas de secuencia en el proyecto de umbrello. (S)
4. Organizar el desarrollo con Penpot. (XL)

Procurando mejorar los fallos del sprint anterior se tuvo una charla tendida alrededor de la dificultad de las tareas y se procuró asignarlas correctamente para poder completarlas con facilidad.

En primer lugar era necesario completar el diagrama de secuencias pendiente. Para esta tarea se asignaron dos personas más el SM.

Tanto para completar el diagrama de casos de uso como para la unificación de los diagramas en umbrello se decidió destinar una persona a cada tarea respectivamente.

Finalmente el resto del equipo (6 personas) se encargaría del desarrollo en Penpot. Esta tarea además debía ser más desarrollada y dividida en subtareas para lo cual se tuvo una reunión específica en la que se discutió el tema con los miembros del equipo que tenían asignada esta tarea y el SM.

En esta reunión se decidió cuántas páginas había que crear para nuestra interfaz (página login, página evento, página de compra, versiones de accesibilidad, etc) y además se comentaron unas reglas de diseño comunes para estar coordinados como equipo y presentar una imagen homogénea en nuestro proyecto final.

Además de todo esto, para evitar lo ocurrido en el sprint pasado relacionado con la compleción de tareas en medio del sprint se decidió asignar a todo el mundo una segunda tarea a la que ayudar en caso de terminar la suya de forma prematura, evitando así que aquellos que terminan pronto su tarea no sepan exactamente dónde pueden ayudar.

Desarrollo del Sprint

A lo largo del Sprint aprovechamos una vez más los canales de comunicación instaurados para poder comunicarnos e informar de las tareas realizadas y los problemas de diseño que iban surgiendo a lo largo de la semana.

Como se había planteado durante la planificación del sprint algunas personas lograron completar sus tareas antes del fin del sprint por lo que pudieron ayudar a sus compañeros a planear y completar las suyas.

Debido a la planificación de la propia asignatura, la reunión que habríamos realizado normalmente el miércoles debió ser trasladada al jueves. Para ello se organizó a través del grupo de whatsapp de esta modificación y se buscó cuadrar un horario común a todo el mundo para poder asistir la mayor cantidad de compañeros posible.

Sprint Review

Llegada la reunión del jueves, revisando el trabajo realizado llegamos a la conclusión de que estaba incompleto y era insuficiente en general.

Ciertamente las tareas de compleción de diagramas, tanto el de secuencia como los de casos de uso habían sido completados y parcialmente documentados; sin embargo todo lo relacionado con el modelado gráfico había quedado o bien incompleto, faltando una cantidad considerable de páginas y requisitos funcionales no completados; o bien realizado de forma pobre, sin contemplar en ningún momento ningún esquema o principio de diseño. No había sensación de homogeneidad entre las páginas, el diseño era muy plano sin colores ni imágenes y la sección de accesibilidad se limitaba a aumentar el tamaño de la fuente.

Por ello toda la parte de Penpot debía ser revisada y estudiada. Algunas de las páginas podrían ser trabajadas por donde se habían quedado pero otras debían ser realizadas desde cero. Durante la Sprint Retrospective el equipo pudo hablar más en detalle de la situación.

Sprint Retrospective

Como se ha comentado, la sprint review fue, en pocas palabras, un desastre. La gran mayoría de los integrantes del grupo no habían completado sus tareas o las habían hecho de forma insuficiente y poco trabajada.

Debido al resultado negativo de este Sprint el equipo decidió extender un poco la reunión de retrospectiva para intentar entender qué había ocurrido y qué justificaba ese desempeño tan negativo.

La razón principal apuntada era el hecho de que aquel Sprint juntaba dos aspectos contrarios: por un lado la mayoría de compañeros tenían exámenes a lo largo de la semana que les quitaron tiempo que pudieran haber dedicado para el proyecto, por el otro este Sprint poseía la mayor carga de trabajo hasta el momento, siendo precisamente el Penpot una tarea sumamente larga que pese a no ser especialmente difícil requería de mucho tiempo por parte de los desarrolladores.

Además de esto algunos compañeros destacaron una falta de conocimientos de cara al diseño de interfaces, de forma que no tenían muy claro cómo abordar la situación y, en muchas ocasiones, preferían no decir nada o pedir ayuda para realizarlo correctamente para no “molestar” al resto del equipo de desarrollo.

Además de todo esto surgieron algunas discusiones alrededor del diseño, puesto que había grandes diferencias en cuanto a la exigencia que había para el diseño entre unos compañeros y otros, de forma que lo que para unos parecía más que suficiente para otros no llegaba a los estándares mínimos de cómo debería verse la aplicación.

Dado que en otras partes del trabajo la carga de tareas era mucho menor, las ideas eran mucho más sencillas de compartir y si una persona tenía una mayor exigencia sobre cómo debería ser realizada una tarea podía mejorarla individualmente. Esta vez sin embargo, al

tener que colaborar todos para poder completar el diseño a tiempo, debíamos coordinarnos como grupo lo cual supuso dificultades considerables.

Una vez encontrado el problema era necesario buscar soluciones. En primer lugar se realizó una sesión fuera de horario de Scrum para poder poner en la misma página a todo el equipo, puesto que trabajando individualmente era ya prácticamente imposible completar el trabajo teniendo una semana por delante para la exposición. En esta reunión aclaramos un punto común de diseño y de estándares mínimos para entregar el trabajo, se decidió qué páginas se iban a realizar y cómo se iba a trabajar. Trataremos en más profundidad esta reunión en el Sprint Planning del quinto Sprint.

Además de esto decidimos que era necesario llevar a cabo un seguimiento más cercano y recurrente del trabajo realizado durante la semana. De esta forma lo que antes era enviar breves mensajes de actualización por los canales de comunicación se convirtió en reuniones “dailys” para revisar el progreso de las tareas. Para esta tarea adicional se decidió designar al Scrum Master que se encargaría de asegurar que todo el mundo estaba pudiendo completar sus tareas correctamente y se encargaría de organizar llamadas breves diarias con los miembros del equipo para poder realizar este seguimiento continuo del trabajo. En general la conclusión del equipo fue que el Scrum Master debería llevar a cabo un papel más de Scrum Master y descargarle el peso del rol secundario de Desarrollador puesto que cada vez eran más y más importantes las tareas que estaban relacionadas con él.

En añadido a esto, como grupo, se pidió en general a todos los compañeros volcarse un poco más en el proyecto y no limitarse a asistir a las reuniones semanales, de lo que pecaban algunos. Ciertamente los problemas de sprints anteriores respecto a la organización de tareas habían sido solucionados y nadie podía decir que no supiese qué tenía que hacer. Sin embargo una falta de compromiso por parte de algunos miembros del equipo llevó a que no trabajasen durante la semana a menos que fuera durante la reunión de los Miércoles, lo cual no hizo más que agravar la problemática.

Precisamente por esto se optó, una vez más, por aumentar aún más el peso directivo tanto del Scrum Master como del Product Owner que realizarían un seguimiento más de cerca de las tareas realizadas y el progreso en ellas. Además de esto como se ha comentado el

Scrum Master debería realizar reuniones personales con los miembros del equipo para asegurarse de que todos podían completar sus tareas y procurar ayudar en todo lo posible. Finalmente y aunque las tareas de cada desarrollador estaban claras, la nueva medida de asignar una segunda tarea optativa a los miembros del equipo por si terminaban sus tareas principales no se llevó a cabo de la forma que debería y en general no supieron cómo o en qué ayudar exactamente de forma que se decidió mejorar aún más este aspecto.

Sprint 5

12/11/25 - 19/11/25

Sprint Planning

Tal y como ya se ha explicado, el desarrollo del sprint anterior no fue el esperado por lo cual se realizó una sesión extra de gestión y organización antes de continuar con el sprint planning. La idea de esta reunión era entendernos como equipo y decidir cómo queríamos realizar el modelado de la interfaz, las ideas clave respecto a la paleta de colores, las fuentes y otras decisiones común para el diseño. Estas decisiones para el diseño son expuestas en mayor extensión y claridad en el documento anexo pertinente al modelado con Penpot.

Una vez realizada esta reunión excepcional dimos pie al quinto Sprint. Los objetivos eran claros y muy importantes. Por un lado era necesario realizar prácticamente en su totalidad el modelado con Penpot y su documentación y por el otro había que preparar la presentación.

Designación de objetivos y tareas:

Aprendiendo una vez más de las experiencias previas como grupo se decidió elaborar un Sprint Backlog detallado y muy claro en el que cada persona tuviera muy claro cuales eran sus tareas. De esta manera las grandes tareas ya explicadas se dividieron de la siguiente forma:

- Realizar páginas de compra de entradas (L) · Abel y Julián
- Adaptar las páginas para mayor accesibilidad (M) · Marcos
- Realizar interfaz pasarela de pago (S) · Alejandro
- Realizar interfaz lista de eventos (M) · Adrián y Daniel
- Redacción del anexo del modelado de interfaz (M) · Julián y Adrián
- Realizar interfaz de crear eventos (M) · Fco. Javier
- Redactar anexo de diagramas de secuencia (M) · Iván
- Normalizar y unificar el proyecto final de Umbrello (M) · Santiago
- Realizar presentación (parte de diagramas de secuencia) (S) · Jesús
- Realizar presentación (parte de modelado UI) (S) · Blas
- Realizar presentación (parte de Scrum) (S) · Abel

De esta manera quedaba el desarrollo de la interfaz completamente granulado en páginas independientes de forma que fuera más sencillo el desarrollo. Además de esto el Scrum Master llevaría a cabo las reuniones comentadas en el retrospective anterior para hacer un seguimiento profundo e intensivo del progreso del equipo.

Además de todo esto cada miembro del equipo recibió instrucciones claras de cómo ayudar en caso de completar su tarea respectiva. En nuestro caso decidimos que al terminar una tarea debía informarse de ello a través de WhatsApp o Discord y en ese momento todo el mundo debería de informar de su progreso y se decidiría en qué parte del desarrollo era necesaria qué cantidad de recursos y ayuda.

Durante la propia reunión semanal pudimos comenzar con el desarrollo y en general se consiguieron avances considerables, consiguiendo mejorar notablemente el aspecto de la página y dejando para el trabajo personal durante la semana tareas más sencillas.

Desarrollo del Sprint

Durante el Sprint se dieron las reuniones personales “dailys” que se han comentado previamente. En ellas el Scrum Master pudo ver cuál era el progreso real del equipo y de esta forma asignar correctamente los recursos a aquellas tareas que resultaron más costosas de lo previsto.

Además de esto durante el desarrollo del Sprint surgieron nuevas tareas o historias que debieron ser asignadas, esta vez mucho más específicas como por ejemplo la creación de los flujos de Penpot o el rediseño y unificación de las cabeceras de todas las páginas.

En añadido a ello, durante la semana gracias a las notificaciones de proyecto el equipo pudo poner en común el trabajo realizado en varias ocasiones lo cual permitió detectar de forma temprana disparidades y diferencias en el diseño que pudieron ser corregidas. Un ejemplo de ello es el diseño de la página en modo accesible, que inicialmente se había interpretado como un mero aumento de las fuentes y del tamaño de los recursos de la página pero que por idea de varios miembros del equipo se transformó en una accesibilidad más completa que envolvía cambios estructurales importantes en la página.

Otro ejemplo de esto era la página de la pasarela de pago, que pese a estar en tareas distintas estaba firmemente relacionada con la página de compra de entradas de forma que fue necesario poner en común el diseño para que surgiera un flujo homogéneo dentro de la página.

Más allá de esto, en general la presencia más cercana de los roles principales del Scrum parecieron dar sus frutos y en general se consiguió un progreso adecuado y prometedor.

Sprint Review

Al contrario que en el Sprint anterior, dado que la presentación sería el día miércoles tuvimos que adelantar al martes el sprint review para poder realizarlo antes de la presentación.

En general este quinto Sprint fue todo un éxito. Finalmente todas las tareas fueron completadas en su totalidad y el equipo estaba contento con el resultado final de todas y cada una de ellas.

Realizamos una pequeña demo con todos los flujos de la interfaz para ver el resultado al completo desde fuera del modelado de la interfaz y, en comparación con el resultado del Sprint anterior, en esta ocasión el trabajo realizado se reflejaba sin dudas. Todos los requisitos habían sido cumplidos y el diseño fue realizado correctamente.

Además de esto pudimos poner en común con todos los compañeros la presentación que realizaríamos el día siguiente y explicamos qué íbamos a comentar en líneas generales para estar de acuerdo en caso de querer cambiar algo de última hora. Sin embargo esto no fue necesario y hubo un fuerte consenso al respecto.

Sprint Retrospective

En este último Sprint previo a la presentación las sensaciones fueron mucho mejores. El equipo estaba contento con las decisiones tomadas y con la nueva organización, en la cual se delegaban mejor las distintas responsabilidades a los cargos “directivos” del equipo, lo cual facilitaba la tarea de los desarrolladores.

En general se tomaron de forma positiva las nuevas reuniones de progreso durante la semana, que no solo permitían al Scrum Master controlar el progreso para designar a miembros del equipo a aquellas tareas que necesitaban apoyo sino que además presentaban una ventana importante para los miembros del equipo para expresar sus dificultades o necesidades al Scrum Master de forma que se pudieran solventar estos problemas con mayor brevedad.

Por otro lado la estrategia de una designación de tareas granular y específica se asentaba una vez más como la mejor opción para distribuir los trabajos a realizar por la claridad y simplicidad que aportaba al proyecto.

Finalmente no se tomaron decisiones de grandes cambios para siguientes Sprints, más que pequeñas dificultades expresadas por algunos compañeros aprovechando la propia reunión.

SPRINT 6

26/11/25 - 02/12/25

Sprint Planning

El último sprint fue como se explica muy positivo y por lo tanto no quedó nada pendiente por hacer/señalar, lo que hizo que este sprint pudiese comenzar poco a poco desde el inicio. Se decidió que para esta parte del proyecto debería haber una preparación previa, ya que aunque todos conocemos java no todos teníamos soltura con algunas tecnologías como react o con el desarrollo de APIs. Por esta razón, aunque la presentación se realizó el día 19, se decidió no empezar con el primer sprint hasta el día 26, por lo tanto durante la primera semana NO se impuso ninguna tarea ni se avanzó en el proyecto, solo se hizo una organización de lo que cada uno haría y se realizó una preparación individual para afrontar los problemas que podían surgir de la forma más competente.

Asignación de roles

Ya que la presentación estaba hecha y comenzaba una nueva etapa del desarrollo del proyecto, decidimos como anteriormente cambiar de nuevo tanto al product owner como al

scrum master. En este caso se decidió que Julián Moreno de la Maza fuese el scrum master y que Jesús Quiñones de Las Peñas fuese el nuevo product owner. Aunque en principio siempre hemos hecho cambio de roles cada 2 semanas aproximadamente de trabajo, en este caso como ya tocaba la última parte que sería el desarrollo de código decidimos que estos roles se quedarían así hasta la última presentación. Con esto da comienzo la explicación del sprint número seis.

Designación de objetivos y tareas:

Tras esa semana de reflexión, teníamos muy claro la dirección que nuestro proyecto iba a tomar, y por ello también lo que cada uno debía hacer. Decidimos dividir el proyecto en “frontend” y “backend” (la parte de la interfaz gráfica y las llamadas a los datos y a la base de datos respectivamente) y cada uno se encargaría de algo. El grupo de frontend fue conformado por; Abel Fernández, Daniel Fernández, Alejandro Lobato, Marcos Du Welz y Adrián Romero. Por otro lado el backend estaba compuesto por; Iván Bezares, Santiago Moreno, Fco Javier Martín, Jesús Quiñones, Blas Carretero y Julián Moreno.

BACKEND:

Por una parte, en el backend hubo una división de tareas en la que cada uno se encargaría de crear una parte clave del código, siendo la siguiente:

- Ivan, Julián y Blas se encargan de implementar la API, entendiendo su funcionamiento
- Fco Javier, Jesús y Santiago se encargan de crear la base de datos con todo lo que se necesitaba (atributos, tipos, claves primarias y secundarias...)

Con esto cada uno sabía lo que tenía que hacer para la siguiente reunión, que en principio se estableció el miércoles siguiente, 03/12/25.

FRONTEND:

En la parte de frontend, se decidió seguir aprendiendo y probando cosas, creando componentes pequeños para entender cómo funciona y poco a poco implementarlo en el código real. En este caso como el scrum master estaba en el grupo de backend, se decidió crear una figura de organización y control que fue Abel, ya que era el que más experiencia tenía programando páginas web. Esto funcionaría además para enseñarle a los demás y explicar lo que no entendían.

Desarrollo del sprint

A la hora de empezar a trabajar, comenzaron a surgir ideas de cambio en la estructura del proyecto al completo, por ello se decidió hacer una mini reunión telemática 2 días después (28/11/25) para plantear la nueva estructura de todo.

Lo que se planteó fue cambiar la topología del github ya que en un principio teníamos un repositorio del código completo (incluyendo backend y frontend) pero esto era un problema ya que a la hora de hacer commits todo se mezclaba, haciéndolo incomprensible y además

difícil para cambiarlo tantas personas a la vez. La opción que surgió inicialmente fue crear 2 repositorios, pero esto también era un problema ya que a la hora de entregar el trabajo no se verían todos los commits, o se tendrían que entregar varios proyectos. Por ello Abel decidió crear una estructura de github detallada a continuación.

Lo inicial es una organización, llamada CudeClarke, en la que todos somos contribuyentes, y dentro de esta organización se encuentran 3 repositorios, ClarkeAPI que hace referencia al backend, ClarkeWeb que habla del frontend y ClarkeDocs que contiene todos los documentos relevantes del proyecto (diagramas, anexos, informes...). Esto soluciona el problema de tener varios repositorios “desperdigados” y además al realizar cambios en el proyecto de un repositorio, los demás no se verían afectados.

También se crearon “teams”, frontend, backend y admins, que se le asignaba a cada miembro del grupo para que pudiese acceder al repositorio que le tocara. Esto además de darle un toque de realismo al proyecto, también daba una sensación de limpieza y organización al proyecto, consiguiendo un mejor método de trabajo.

Dentro del código, se debían hacer los push de una cierta forma. Todo se haría desde develop, significando que main solo se cambiará una vez el proyecto esté terminado, o por lo menos hasta que se tenga un mvp (minimum viable product). Cuando alguien quiere hacer un cambio crea una rama con git, que esté dentro de develop con un esquema “x/y” donde x es lo que se está haciendo (feature, fix, refactor, documentation) e “y” se trata de lo que se está haciendo. Por ejemplo si queremos crear una clase objeto la rama resultante sería “feature/creation-of-Object-class”. Esto además de dar una seguridad extra al proyecto por si se produce algún cambio no deseado, da una organización al código, sabiendo exactamente qué se está haciendo en cada momento.

A la hora de hacer un push, es necesario hacer una “pull request”, que es una petición para que ese código sea implementado en el repositorio. Esto se debía hacer desde tu rama creada anteriormente a develop, y solo podía ser aceptado por 1 persona de cada grupo (Iván desde el backend y Abel desde el frontend) y por el scrum master que eran todos los que estaban dentro del team admins. Esto se decidió así para que en cada grupo siempre haya una persona atenta a los cambios y además que el scrum master que se ocupa de organizar ambos grupos también tenga control. Cabe destacar que además de ser obligatorio el uso de “pull request” por todos los miembros del grupo, tampoco se podía aceptar una pull request a uno mismo. Por ejemplo el scrum master que tenía permisos para aceptar pull request, NO puede aceptar su propia pull request, lo que hace que siempre debe haber alguien que revise las implementaciones además del propio implementador y también para proporcionar un realismo y parecido a como trabajan las empresas reales.

Cabe destacar que esto fue realizado por Abel Fernández Palomo que se encargó de la creación de todos los repositorios, la organización y la implementación de las reglas.

Después de este cambio, las tareas ya se habían completado, por lo que se hizo una nueva asignación para todos los miembros del grupo.

Por una parte a los del backend se les encargó seguir con el desarrollo del código. Para esto se decidió seguir una estructura DAO (Data Access Object) que como su nombre indica se encargaría de acceder a la base de datos para sacar esos datos. Debido a la cantidad de tareas tan extensa decidimos seguir una metodología en la que pusimos todas las cosas que quedaban pendientes en nuestro grupo de discord, y cada vez que alguien hacía algo decía que se lo asignaba y una vez terminado se tachaba de la lista. Esto se hizo ya que no se podían realizar reuniones cada dos por tres y además porque se confiaba en que cada uno haría lo que le toca, realizando al menos alguna tarea además de la asignada en la reunión previa. Esto llevó a la siguiente distribución

- Julián y Santiago se encargaban de hacer las clases que necesitaban consultas a la base de datos. Esto incluye creación del factory method, UserDAO y TicketDAO, además de las interfaces iTicketDAO, iUsuarioDAO y las conexiones TicketDAOMySQL y UsuarioDAOMySQL. Además Santiago se encargó de la creación de clases de datos para entrada, usuario, evento y ticket
- Fco Javier se encargó de la implementación de Evento y todas sus clases (Evento, EventoCarrera, EventoConcierto...)
- Ivan se encargó de refactorizar algunas clases para mejorar su rendimiento y además de la creación del endpoint de usuario (durante el proyecto en general y al igual que Abel, se encargó de actuar como organizador del proyecto, lo que hace que se encargue en su mayoría de resolver dudas y conflictos, además de refactorizar código para que esté implementado de manera correcta).
- Blas y Jesús se encargaron de crear los endpoints de ticket y entrada/evento respectivamente.

Más adelante, el día 1 de diciembre se realizó una reunión telemática rápida en la que se habló de algunos cambios en el backend relacionado con las clases de datos, lo que hizo que se tuvieran que refactorizar algunas clases por lo que se volvieron a asignar tareas simples, que cada persona que hubiera hecho una clase la refactorizase para que implemente de forma correcta los nuevos cambios, quedando así:

Paco y Jesús: Refactorizar DAO evento y endpoints evento

Santi: Refactorizar clases de datos

Julián: Refactorizar DAO tickets

Iván: Implementar DAO factory y refactorizar UserHandlers

Blas: Endpoints Tickets y mirar JUnit y testing

Por otro lado, el frontend seguía con la organización previamente relatada. Todos estaban programando en react, aprendiendo como hacer cada cosa pero sin tocar el código original. Mientras tanto Abel que era el que más conocía el tipo de programación se fue encargando de crear la página de lista de eventos y comenzó a crear la de comprar entradas. Posteriormente Daniel hizo algunas aportaciones al código, como cambiar los estilos de las cabeceras (todo se puede observar en los commits y pull requests dentro del repositorio ClarkeWeb).

Cabe destacar que hubo una reunión telemática rápida el día 30 para ver como iba cada miembro, opinando sobre el trabajo de los demás y aconsejando a cada uno como debía seguir.

La intención de la asignación de estas tareas era aprender a como crear la página web y poco a poco ir implementando algunos componentes y código dentro del repositorio antes de la reunión siguiente, que sería el miércoles siguiente (03/12/25).

Sprint Retrospective

Una vez llegó el día de la reunión se habló en primer lugar lo que cada parte había hecho y hubo un descontento general. Muchas de las tareas pendientes se terminaron, pero aún quedaban bastantes y algunas personas no habían terminado ni sus tareas básicas. Esto provocó que el proyecto llevase algo de retraso y además se tuvo que hacer un cambio de rumbo, pues como algunas personas no habían hecho algunas cosas se tuvo que cambiar la asignación de estas acciones pendientes.

En la parte de backend la mayoría de clases se hicieron a tiempo y fueron correctas, aunque algunas se entregaron poco antes de la reunión, lo que provocó que no se pudiese dejar trabajo hecho antes del próximo sprint. Aún así fue un resultado muy positivo que provocó un significativo avance en este equipo.

Por la parte de frontend, hubo una baja implicación. Todos se esforzaron en aprender a crear la página web pero no hubo aportaciones reales al proyecto de casi nadie antes de la reunión del día 30, esto provocó un leve retraso en el desarrollo de la página, además de una sobrecarga para algunos miembros del equipo que tuvieron que trabajar más de la cuenta.

A pesar de los problemas que nos encontramos, todavía teníamos tiempo para seguir programando por lo que no era un problema muy grave el retraso del proyecto. Aunque en un principio el sprint tomó un cambio drástico en la estructura del proyecto todos nos adaptamos de forma correcta y rápida, consiguiendo así una continuidad en el proyecto sin necesidad de pararse a explicar o cambiar nada.

SPRINT 7

03/12/25 - 09/12/25

Sprint Planning

La reunión del 3 de diciembre trajo muchas dudas sobre el proyecto con respecto a implementaciones. Esto provocó que se tuvieran que asignar muchas tareas ya que se hicieron muchos cambios en clases del código del backend por lo que se tuvo que plantear la organización de tareas pendientes que se asignaría a cada persona, mientras que el frontend seguía avanzando con lo que ya tenía claro que debía hacer.

Designación de objetivos y tareas:

Para el backend, el product backlog de este sprint sería muy repetitivo ya que al final no había tanta implementación como refactorización. Este quedó tal que así:

- Julián -> Refactorizar TicketDAOMySQL
- Santiago -> Refactorizar UsuarioDAOMySQL
- Fco Javier -> Refactorizar EventoDAOMySQL
- Jesús -> Refactorizar EntradaDAOMySQL
- Iván -> Refactorizar clases para establecer estructura Singleton
- Blas -> Creación de Managers para cada clase

Con esto el plan era hacer cada uno su parte para el domingo 7, para ese día realizar otra reunión telemática rápida y volver a asignar tareas.

Para el frontend como se explica en la sprint retrospective del sprint anterior, no hubo mucho avance en el código por parte del grupo, lo que hizo que en este sprint se pusieran las pilas y empezasen a asignar tareas que debían terminar antes de la siguiente reunión. Con esto, el sprint quedó de la siguiente forma:

- Alejandro se encargó de crear los componentes relacionados con la pasarela de pago, además de la implementación de esta página.
- Daniel siguió encargándose de crear componentes pequeños, siguiendo modificando la cabecera de las páginas y además creando los PopUps que se utilizarían en todas las páginas
- Marcos se encargó de crear el footer utilizado en todas las páginas y además se encargó de la página crear evento
- Adrián se encargó de la última página en la que se entrega la entrada, con los datos de la entrada, el qr y demás.
- Abel siguió con la creación de la página para comprar entradas, recibiendo ayuda de algunos miembros y además siguió actuando como director del equipo, ayudando con las implementaciones y encargando tareas posteriormente.

Todos tenían claro lo que debían hacer por lo que así comenzó esta nueva semana de sprint en la que cada uno debería esforzarse para terminar su parte.

Desarrollo del Sprint

Para el desarrollo del sprint lo explicaré separando en frontend y backend como anteriormente.

En el backend las tareas estaban claras, se hicieron todas sin problema alguno y además se fueron implementando soluciones a errores minúsculos que no habían sido avistados antes, como tipos de variables en la base de datos incorrectos, variables de más, métodos que podían simplificarse...

Posteriormente se hicieron implementaciones que sí afectaban a todo el código, como la conversión de algunas clases a singleton y además la eliminación de factories individuales para crear una abstract factory. Esto hizo que la tarea de refactorizar se volviese más larga y compleja, pero como cada uno tenía una parte asignada a refactorizar solo se tenía que volver a revisar para ver si había que cambiar algo, sin necesidad de asignar nuevas tareas.

Hubo algunas personas que terminaron antes por lo que se asignaron nuevas tareas que fueron las siguientes:

- Jesús -> Crear managers para ayudar a Blas y posteriormente implementar métodos necesarios en clases de datos (toString y hashCode)
- Fco Javier -> Encargarse de los cambios que iban surgiendo en la base de datos
- Santiago -> Creación de tests para las clases DAO

El avance del backend iba muy bien y estaba por terminar, por lo que se decidió hacer una mini-reunión en la que se haría un cambio de tareas para asignar a personas a otras ocupaciones que era necesario terminar y no eran parte de frontend ni de backend. Esto se hizo el domingo 7 de diciembre de forma telemática. En ella se decidió que se debían cambiar los diagramas de clases hechos anteriormente, ya que a pesar de que en ese momento eran correctos, la creación del código conllevó un aumento en las clases y métodos, que debían verse reflejados en los diagramas. Por ello el nuevo sprint backlog es el siguiente:

- Jesús y Blas se encargan de la modificación de los diagramas de clases
- Iván, Santiago y Fco Javier se quedaron encargados de darle los últimos retoques al backend, siendo los dos primeros encargados de refactorizar clases de datos y métodos y el tercero encargado de hacer los pertinentes cambios a la base de datos.
- Julián a partir de esta parte se ciñó a realizar el scrum (como este documento) pero también estaba encargado de ayudar con el código en backend si era necesario.

Por otra parte, el frontend siguió con sus tareas, las cuales se extendieron hasta el día de la reunión. No hubo cambios de tareas en general ya que cada uno necesitó toda la semana para terminar sus implementaciones. Al igual que en el backend, hubo una reunión telemática para comprobar el progreso de cada miembro el mismo día (7 de diciembre) y en vista de los avances de algunos miembros se decidió asignar nuevas tareas.

Una vez finalizado todo esto, quedaba seguir trabajando y esperar a la reunión presencial del día 10.

Sprint Retrospective

En este sprint las cosas fueron mucho mejores ya que, por una parte el backend estaba muy cerca de tener un mvp y el frontend por otro lado llevaba muy buen camino. El problema es que algunas tareas se habían atrasado en el frontend y esto provocó que las implementaciones encargadas para esta reunión se entregasen muy cerca de la fecha

(como la noche anterior) o incluso que no se entregasen. En este punto, aunque seguíamos teniendo tiempo, la fecha de la presentación estaba cada vez más cerca y esto era otro tema que también se tenía que discutir en esa reunión. Esto provocó que algunos miembros tuvieran que trabajar más de la cuenta.

SPRINT 8 (ÚLTIMO SPRINT)

10/12/25 - 17/12/25

Sprint Planning

Como se explica en el sprint retrospective del sprint anterior, backend iba bastante bien pero frontend llevaba un retraso notable. El problema es que en este punto del proyecto rotar a miembros de backend para que aprendiesen react y además ayudasen a los integrantes del frontend era algo altamente difícil pues solo disponíamos de una semana, por lo que decidimos hacer una asignación de tareas amplia para que, ya que no podíamos quitar trabajo en el propio frontend, al menos pudiéramos liberar obligaciones tales como los diagramas o la presentación.

También cabe destacar que, aunque los miembros de frontend se dedicasen exclusivamente a crear código, se les asignó a algunos partes de la presentación oral del día 18.

Designación de objetivos y tareas:

En primer lugar el backend, aunque ya estuviese casi terminado todavía necesitaba algunas pinceladas. El sprint backlog quedó tal que así:

- Julián, Santiago, Iván y Fco Javier se encargaron de la implementación de variables que guardasen el ID de cada clase, además de los métodos necesarios para recuperar estos valores.
- Blas y Jesús como establecido anteriormente siguieron con las modificaciones necesarias a los diagramas de datos

Aunque esta asignación varió durante la semana.

En segundo lugar, el backend se centró en terminar de manera urgente las tareas pendientes, y además asignar nuevas tareas para terminar la página web. Con esto el sprint backlog se veía de la siguiente manera:

- Abel y Daniel se encargaron de la implementación de la accesibilidad en todas las páginas
- Marcos seguía con la creación de la página para crear un evento
- Adrián seguía con la implementación de la página en la que se enseña el ticket que se ha comprado
- Alejandro se encargó de terminar de diseñar la página de la pasarela de pago, con ayuda de Abel.

Cabe destacar que, al igual que en el sprint anterior, Abel siguió actuando como organizador para tanto ayudar, como corregir y explicar todo lo que era necesario.

Por último faltaba la asignación de turnos de la presentación. Aquí era obvio que cada parte debía ser explicada por miembros de esta, por lo tanto se decidió que los miembros de cada equipo eligiesen ellos mismos quien hablaría y de que parte. Tras unos cuantos días, el resultado fue el siguiente:

- Iván y Fco Javier se encargarían de explicar el backend a nivel de código, contando también su estructura.
- Alejandro y Marcos se encargarían de mostrar la página web durante la presentación para que todos pudieran observarla, mientras explicaba las funcionalidades más importantes.
- Abel se encargaría de mostrar y explicar la estructura utilizada en github
- Julián (scrum master) se encargaría de explicar el desarrollo del proyecto, mostrando la metodología scrum que se ha utilizado.

Con esto se termina la asignación de tareas y comienza el desarrollo del sprint

Desarrollo del sprint

Empezaremos explicando el desarrollo por parte del equipo de backend. En este lado, faltaban algunas implementaciones dentro del código, pero muy cortas por lo que la distribución de tareas seguiría como estaba.

Más adelante se vio que estas tareas ya estaban completadas, con lo que se decidieron hacer otros cambios que cada miembro implementaría. Entre estos cambios estaban añadir más métodos, pero también crear testing para probar el código al completo y además crear pruebas de integración con el frontend (que aunque no esté conectado se observaba la salida para comprobar si eran correctas). Además se tendría que asignar a alguien el trabajo de levantar la página web para que fuese funcional una vez todo estuviese implementado.

Por una parte el testing fue realizado enteramente por Santiago, mediante mockito. Con esto generaba objetos de mentira para probar las distintas funcionalidades (como la conexión a la base de datos o simplemente objetos como evento o ticket). Esto se alargó hasta aproximadamente el fin de semana de esa misma semana.

Por otro lado, las pruebas de integración utilizando los handlers las realizó Iván, comprobando que las consultas se realizaban correctamente y además la propia creación de esos handlers.

Lo último era levantar la página web. Esto en un principio fue una idea que tuvimos para mantener la página con los puertos abiertos en un ordenador y que así cualquiera tuviera

acceso, esto se le encargó a Blas, por lo que Fco Javier le sustituyó ayudando a Jesús con los diagramas.

Esto se hizo mediante el uso de 2 máquinas virtuales, una se encargaría de mantener la base de datos de forma online continuamente, pudiendo hacerse consultas que se quedarían en la base de datos y que posteriormente se integraría al backend. Por otra parte, la segunda máquina virtual se encarga de mantener la página abierta, actuando de servidor. A todo esto se le sumó la compra de un dominio (cudeclarke.es) el cual serviría para acceder de forma más cómoda, en lugar de usar la ip de la máquina virtual. Cabe destacar que esto se hizo de forma rápida y eficiente gracias a Blas que se encargó del lanzamiento de las máquinas virtuales y su configuración y a Abel que se encargó de la compra del dominio y del enlazado de la máquina con el frontend.

Una vez tuvimos esto y el backend estaba terminado se obtuvo una versión casi completa, a la que solo le faltaba implementar el frontend.

Luego, en cuanto a la parte del frontend, aunque con algo de retraso las cosas avanzaban. Cada uno tenía tareas asignadas por lo que el desarrollo era muy directo.

Aquí ya que estábamos tan cerca de la entrega, decidimos que quien terminase su tarea se encargase de ayudar a quien todavía no había terminado, por ello una vez Daniel terminó se encargó de ayudar a Marcos.

Abel por otro lado se encargó de añadir funcionalidades extra a la página una vez estaba casi terminada, como una barra de búsqueda, o arreglos a ciertas partes como la página en la que se enseña el ticket comprado o la pasarela de pagos.

Con todo esto dicho, el desarrollo de frontend llegó a su mvp el día 14, por lo que a partir de ese día todos los miembros del grupo se encargaron de añadir funcionalidades extra ya que lo básico estaba hecho y además ayudar con el levantamiento de la página web.

Cabe destacar que el día 14 de diciembre, ya que todo había quedado terminado se decidió hacer una reunión de todos los miembros, que sirvió para realizar las pruebas más grandes de conexión a API, base de datos y además se decidiría lo que cada uno iba a decir en la presentación, ya que como se explica en la planificación esto era una tarea pendiente que se hizo este día y se decidió tal cual aparece en el apartado mencionado.

Sprint Retrospective y conclusión

Con esto el trabajo llegó a su fin. Aunque pudiéndose haber realizado de una forma más rápida y con más ayuda de algunos miembros, estábamos orgullosos ya que habíamos terminado todas las tareas necesarias, al punto de tener tiempo para realizar implementaciones extra que no eran necesarias para darle un estilo más bonito a la página o utilidades.

La sensación era de plenitud, pues habíamos hecho un gran trabajo, en el tiempo asignado y en equipo, tal y como trabajan empresas reales en el sector de la informática. Aunque

haya habido momentos mejores y peores la conclusión es que lo hemos hecho lo mejor que hemos podido y estamos muy orgullosos tanto del resultado como del proceso en sí, sintiendo que estamos listos para hacer cosas incluso de mayor calibre.

Cabe destacar que todo lo contado es simplemente una explicación desde el punto de vista del scrum master, hablando de como se ha organizado todo. Para revisar cosas como lo que cada uno ha hecho o para ver los resultados del código de alguna parte se puede visitar el repositorio o comprobar los archivos entregados junto a este documento, donde aparece la información detallada.