



PRÁCTICA 6 – SEGUNDO SPRINT

Sandra Alonso

Fernando Bellot

Andrés Casillas

Francisco García

Carlos González

Rogger Huayllaso

Boldureanu Daniela Nicoleta

Alejandro Ruiz

Pedro Sánchez

Daniel Villalba

Sergio Villarroel

Índice

1. Introducción	4
2. Mapa de historias	5
3. Product Backlog.....	5
4. Reunión de planificación	6
4.1 Propuesta del Product Owner	6
4.2 Estimación de las historias de usuario	7
4.3 Creación del Sprint Backlog	8
5. Sprint Backlog.....	14
5.1 Herramienta.....	14
5.2 Inicio Sprint Backlog.....	15
5.3 Durante Sprint Backlog.....	16
5.4 Final Sprint Backlog	17
5.5 Conclusiones	18
6. Desarrollo del Sprint	19
6.1 Estructura del control de versiones.....	19
6.2 Frecuencia de realización de pruebas.....	19
6.3 Sprint Backlog.....	20
6.4 Daily Scrum.....	21
6.5 Problemas durante el desarrollo.....	21
7. Reuniones diarias	22
8. Backlog de Impedimentos.....	25
9. Burndown Chart	26
10. Reunión de revisión	28
11. Reunión de retrospectiva	28
12. Comparativa con el primer sprint	30
13. Lecciones aprendidas durante el desarrollo del proyecto	30
14. Dinámica grupal.....	31
14.1 El barco.....	31
14.2 Repetir-Evitar	32

14.3 Reconocimiento	33
---------------------------	----

1. Introducción

Durante el desarrollo de esta práctica se continuó implementando las historias de usuario que fueron asignadas al segundo sprint.

En el periodo de tiempo que ha durado todo este proceso de desarrollo, nos han surgido determinadas complicaciones relacionadas con el formato que escoge Google para guardar los eventos que necesitábamos y la necesidad de adaptarlos al esquema de nuestra base de datos. Sin embargo, hemos identificado menos problemas que en el desarrollo del sprint anterior, en parte por el rodaje en la plataforma y en el lenguaje empleados al que nos tuvimos que enfrentar en la anterior entrega.

Para solucionar estas deficiencias, se emplearon los manuales que proporciona Google de manera online para desarrolladores, donde se detalla todos los atributos y funciones necesarias para desarrollar una extensión, así como el comportamiento que tienen. También, se llevaron a cabo reuniones entre los diferentes miembros del equipo para apoyar a aquellos que precisasen de una ayuda extra para solventar cualquier inconveniente.

Dada la situación actual de confinamiento, no se han podido incluir imágenes ni videos de las reuniones al no haber sido presenciales.

Todo esto se ha solucionado adoptando un sistema de reunión no presencial a través de plataformas de videollamada (Microsoft Teams) y mediante el uso intensivo de nuestros canales de comunicación en Telegram.

Por último, se detalla a continuación como queda estructurada la memoria:

- **Mapa de historias:** En este apartado se expone el mapa de historias validado en la entrega anterior, confirmando su idoneidad como versión final.
- **Product Backlog:** Se actualizará la versión del Product Backlog, con una explicación de los cambios realizados.
- **Reunión de planificación del sprint:** Se desarrollará el curso de la reunión, las historias propuestas por el Product Owner para incluir en el sprint, indicando cuáles fueron las que finalmente se incluyeron.
- **Sprint Backlog:** Se explicará la creación, el desarrollo y el resultado final del Sprint Backlog.
- **Desarrollo del sprint:** En este apartado, se analizará como se ha desarrollado el sprint.

- **Reuniones diarias:** Se indicarán las reuniones que ha mantenido el equipo diariamente, junto con un breve resumen de estas.
- **Backlog de Impedimentos:** Se mostrará de forma resumida los impedimentos que han surgido a lo largo del sprint.
- **Burndown Chart:** Se detallará la evolución del trabajo a lo largo del sprint.
- **Reunión de revisión:** Se expondrán los puntos clave de la reunión.
- **Reunión de retrospectiva:** Se detallará su duración, participantes y cuáles fueron los resultados de cada una de las fases.
- **Comparativa con el primer sprint:** Se analizarán las principales diferencias encontradas respecto al primer sprint.
- **Lecciones aprendidas durante el desarrollo del proyecto:** Se enumerarán las distintas lecciones aprendidas por el equipo durante el proyecto.
- **Dinámica grupal:** Se expondrán las actividades que se han realizado para mejorar la relación entre los miembros del equipo, así como una breve descripción del objetivo que se pretendía conseguir con cada una de ellas.

2. Mapa de historias

Al no haber introducido tareas nuevas, ni haber agrupado ninguna de las anteriores, no se ha producido ninguna modificación en el mapa de historias al considerar que seguía ajustándose al propósito del proyecto que solicitaba el cliente.

El mapa e historias se adjunta junto al resto de archivos que se solicitan para la entrega.

3. Product Backlog

A diferencia del mapa de historias, en el Product Backlog sí se ha visto modificada parte de las historias ya existentes para poder darles una definición más ajustada con su propósito.

Los cambios realizados afectan a las tareas de la release 3, siendo las que más a corto plazo van a ser ejecutadas. En concreto, las modificaciones afectan a sus criterios de aceptación. Algunos de ellos se han modificado tratando de definir de forma más clara su objetivo.

Todos los cambios relativos al Product Backlog han sido llevados a cabo en consenso entre el Product Owner y el Scrum Master, teniendo en cuenta la prioridad de las diferentes historias a la hora de prestar más atención en sus posibles correcciones.

Al igual que el mapa de historias, el Product Backlog se incluye en los archivos generados para esta entrega.

4. Reunión de planificación

La situación respecto al virus COVID-19 continúa evolucionando. Las clases presenciales fueron canceladas oficialmente y el confinamiento se mantiene. Seguimos adaptándonos a la situación y por tanto para todas las reuniones seguimos utilizando la herramienta Microsoft Teams.

4.1 Propuesta del Product Owner

Para empezar, el Product Owner presentó las tareas que quería que fueran implementadas durante este segundo sprint. En concreto propuso para este segundo sprint la realización de las siguientes 12 historias de usuario del Product Backlog. A partir de aquí, el equipo de desarrollo preguntó y resolvió todas las dudas que les surgieron relacionadas a las funcionalidades presentadas.

A continuación, se muestran la lista de tareas presentadas por el Product Owner y las dudas que le surgieron al equipo de desarrollo durante la presentación:

- **Permitir ordenación en vista de marcadores por nombre o URL:** Aparecieron dudas relacionadas con el filtrado de la información, pero el Product Owner las aclaró a todo el equipo.
- **Eliminar marcadores eliminados en el navegador:** No hubo ninguna duda respecto a esta funcionalidad, el equipo de desarrollo lo tenía claro.
- **Editar marcadores de forma manual y múltiple:** Un miembro del equipo de desarrollo planteó una forma para implementar esta funcionalidad y tanto el equipo de desarrollo como el Product Owner dieron el visto bueno.
- **Permitir búsqueda por URL en la vista del historial:** Debido a su similitud con la ordenación de marcadores por URL, no hubo ninguna duda respecto a esta funcionalidad, el equipo de desarrollo lo tenía claro.

- **Borrar historial de forma manual y múltiple:** Un miembro del equipo de desarrollo tuvo dudas, pero el Product Owner aclaró la funcionalidad rápidamente.
- **Sugerir el registro de usuario cuando un usuario no está registrado:** No hubo ninguna duda respecto a esta funcionalidad, el equipo de desarrollo lo tenía claro.
- **Cerrar la sesión tras 120 minutos de haber iniciado sesión:** No hubo ninguna duda respecto a esta funcionalidad, el equipo de desarrollo lo tenía claro.
- **Permitir borrar al usuario solicitando la contraseña:** El Product Owner explicó brevemente esta funcionalidad de nuevo para que el equipo de desarrollo lo tuviera claro.
- **Detectar pestañas abiertas y almacenarlas:** No hubo ninguna duda respecto a esta funcionalidad, el equipo de desarrollo lo tenía claro.
- **Permitir ver listado de pestañas almacenadas:** Debido a las similitudes con la anterior, tampoco hubo ninguna duda respecto a esta funcionalidad.
- **Detectar ventanas abiertas y almacenarlas:** Tampoco hubo ninguna duda respecto a esta funcionalidad, el equipo de desarrollo lo tenía claro.
- **Permitir ver listado de ventanas almacenadas:** No hubo ninguna duda respecto a esta funcionalidad, el equipo de desarrollo lo tenía claro.

4.2 Estimación de las historias de usuario

Debido a los buenos resultados del sprint anterior, a la hora de calcular la estimación de historias de usuario para este sprint se decidió utilizar:

La misma escala que sería la Escala de Cohn con los valores 0, 1, 2, 3, 5, 8, 13, 20, 40 y 100 y la misma equivalencia de puntos que en el sprint anterior donde cada punto de historia equivaldría a 3 días de trabajo ideal estimado.

Estas fueron las HU seleccionadas y la estimación realizada por el equipo:

HISTORIA DE USUARIO	PUNTOS DE HISTORIA
Permitir ordenación en vista de marcadores por nombre o URL	2
Eliminar marcadores eliminados en el navegador	5
Editar marcadores de forma manual y múltiple	5
Permitir búsqueda por URL en la vista del historial	2
Borrar historial de forma manual y múltiple	5
Sugerir el registro de usuario cuando un usuario no está registrado	2
Cerrar la sesión tras 120 minutos de haber iniciado sesión	2
Permitir borrar al usuario solicitando la contraseña	5
Detectar pestañas abiertas y almacenarlas	5
Permitir ver listado de pestañas almacenadas	5
Detectar ventanas abiertas y almacenarlas	5
Permitir ver listado de ventanas almacenadas	5

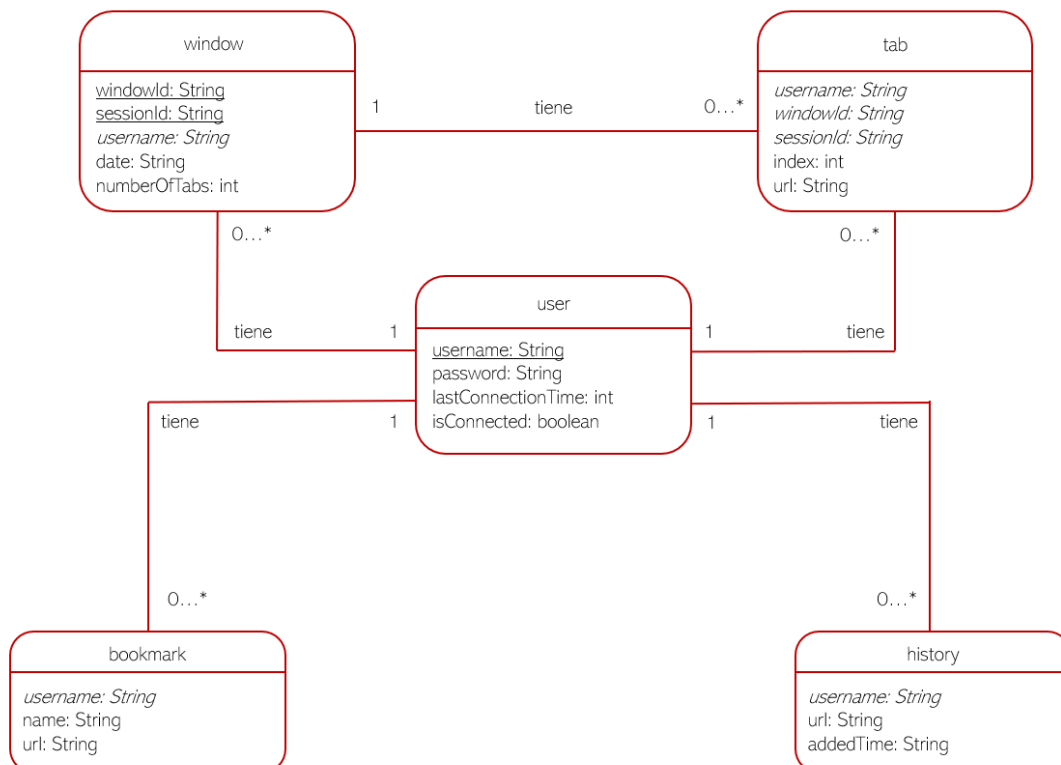
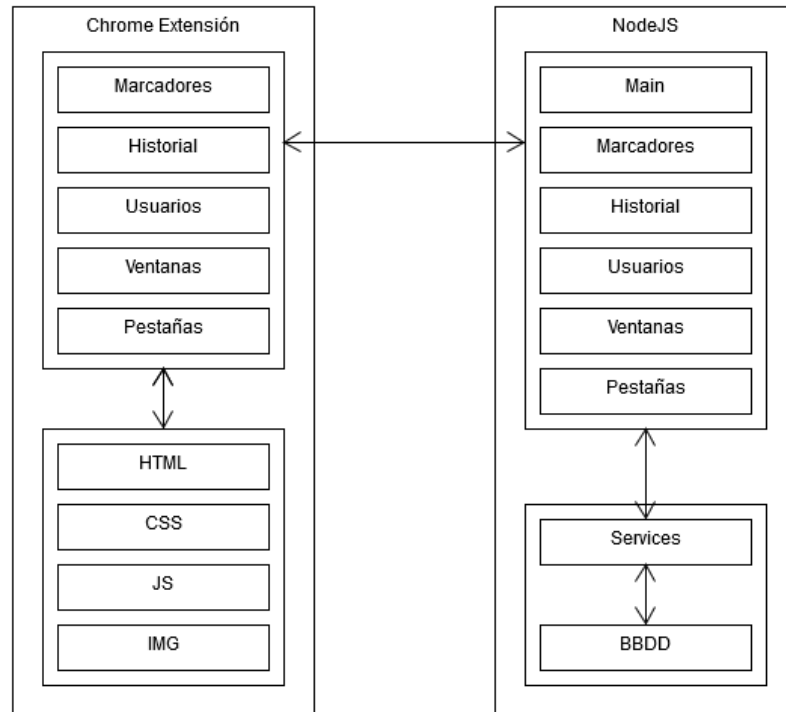
4.3 Creación del Sprint Backlog

Tras estimar y decidir las funcionalidades que se implementarán durante el sprint, el siguiente paso consiste en crear un diseño de alto nivel y descomponer las HU en tareas.

Diseño de alto nivel

Para continuar con el desarrollo de la extensión, el equipo de desarrollo mantuvo la utilización del Modelo-Vista-Controlador basado en capas. Debido a la facilidad para añadir nuevas funcionalidades que posee este modelo, el equipo de desarrollo tan sólo tuvo que ampliar en cada capa la extensión con el fin de añadir las nuevas funcionalidades deseadas para este sprint.

Para visualizarlo de forma más clara, aquí tenemos un pequeño esquema y un diagrama ER similares a los del primer sprint, pero actualizados con las nuevas funcionalidades:



Descomposición de tareas

Al igual que en el sprint anterior, de las 3 semanas disponibles para el desarrollo de este sprint solo se podían contar con los días entre semana, con lo cual en esta reunión se decidió que por cada punto asignado a las HU se estimarían unos 3 días de trabajo de un miembro del equipo de desarrollo, hasta un máximo de 5 puntos por cada miembro, es decir, 15 días.

A continuación, se muestran las tareas en las que se dividieron las historias de usuario:

TAREAS

Permitir ordenación en vista de marcadores por nombre o URL (HU_016)

Desarrollar una interfaz gráfica de usuario que permita ordenar por nombre o URL junto a su botón correspondiente.

Recoger el listado de marcadores.

Detectar tipo de ordenación solicitada por el usuario.

Devolver listado ordenado de marcadores.

Mostrar el listado de marcadores.

Eliminar marcadores eliminados en el navegador (HU_017)

Recoger el listado de marcadores.

Detectar los marcadores eliminados en el navegador.

Borrar los marcadores de la base de datos.

Devolver el listado actualizado de marcadores.

Editar marcadores de forma manual y múltiple (HU_018)

Desarrollar una interfaz gráfica de usuario que permita editar marcadores de forma manual o múltiple junto a su botón correspondiente.

Recoger el listado de marcadores.

Mostrar el listado de marcadores.

Detectar el marcador o marcadores que desea editar el usuario.

Borrar el marcador o marcadores de la base de datos.

Añadir el marcador o marcadores actualizado/s en la base de datos.

Devolver el listado de marcadores actualizado.

Mostrar el listado actualizado de marcadores.

Permitir búsqueda por URL en la vista del historial (HU_019)

Desarrollar una interfaz gráfica de usuario que permita buscar en el historial por URL junto a su botón correspondiente.

Recoger el listado de historial.

Detectar la URL introducida por el usuario.

Devolver el listado de historial que coincida con la URL introducida.

Mostrar el listado actualizado del historial.

Borrar historial de forma manual y múltiple (HU_020)

Desarrollar una interfaz gráfica de usuario que permita borrar el historial de forma manual o múltiple junto a su botón correspondiente.

Recoger el listado de historial.

Mostrar el listado de historial.

Detectar las visitas seleccionadas por el usuario.

Borrar las visitas seleccionadas de la base de datos.

Devolver el listado de historial actualizado.

Mostrar el listado actualizado del historial.

Sugerir el registro de usuario cuando un usuario no está registrado (HU_021)

Recoger el listado de usuarios.

Comprobar si el usuario esta registrado.

Envío de notificación de registro de usuario.

Cerrar la sesión tras 120 minutos de haber iniciado sesión (HU_022)

Comprobar si hay alguna sesión iniciada.

Recoger el listado de usuarios.

Detectar el usuario cuya sesión está iniciada.

Comprobar el tiempo activo desde el inicio de sesión de dicho usuario.

Envío de notificación de cierre de sesión una vez llegado al tiempo límite.

Permitir borrar al usuario solicitando la contraseña (HU_023)

Desarrollar una interfaz gráfica de usuario que permita borrar sus datos solicitando la contraseña junto a su botón correspondiente.

Recoger el listado de usuarios.

Detectar el usuario que solicita borrar los datos.

Comprobar si la contraseña introducida coincide con la del usuario.

Eliminar el usuario de la base de datos.

Devolver el listado actualizado de usuarios.

Detectar pestañas abiertas y almacenarlas (HU_024)

Comprobar si hay alguna pestaña abierta en el navegador.

Recoger el listado de pestañas abiertas.

Añadir la nueva pestaña a la lista.

Devolver el listado actualizado de pestañas abiertas.

Permitir ver listado de pestañas almacenadas (HU_025)

Desarrollar una interfaz gráfica de usuario que permita ver el listado de pestañas.

Recoger el listado de pestañas abiertas.

Mostrar el listado de pestañas abiertas.

Detectar ventanas abiertas y almacenarlas (HU_026)

Comprobar si hay alguna ventana abierta.

Recoger el listado de ventanas abiertas.

Añadir la nueva ventana a la lista.

Devolver el listado actualizado de ventanas abiertas.

Permitir ver listado de ventanas almacenadas (HU_027)

Desarrollar una interfaz gráfica de usuario que permita ver el listado de ventanas.

Recoger el listado de ventanas abiertas.

Mostrar el listado de ventanas abiertas.

5. Sprint Backlog

Siguiendo el mismo procedimiento que en el primer sprint, usamos un archivo en línea que todos los miembros del equipo de desarrollo iremos modificando a diario o en los momentos en los que podamos ver algún avance en cada una de las HU asignadas, para así tener un control diario.

5.1 Herramienta

En el archivo usado se muestran todas las historias de usuario junto con sus descripciones, criterios de aceptación, tareas, puntos de historia (Escala Cohn) y el miembro del equipo al que se la ha asignado dicha historia. También se añadió el tiempo en el que se estima en que se terminará cada tarea asignada, así como el progreso que se va llevando de cada una.

En caso de que el progreso en alguna de las tareas sea 0 se marcará como roja, esto significa que la tarea aún no se ha empezado. Si el progreso de la tarea es mayor que cero y menor al estimado se marcará de color naranja, esto significa que la tarea está

en proceso. Por último, si el progreso de la tarea es igual al estimado se marcará de color verde, esto significa que la tarea se ha finalizado.

Si todas las tareas se han finalizado correctamente y se marcan de color verde, entonces se podrá marcar la historia de usuario como verde, es decir, finalizada. En caso contrario se marcará naranja (en progreso) o roja (no empezada).

También se incorporó como complemento extra para el seguimiento la herramienta de Trello, de esta manera tener más organización en cada tarea realizada.

5.2 Inicio Sprint Backlog

En el inicio del Sprint Backlog todas las tareas de cada una de las historias se marcaron en color rojo, es decir, no empezadas. Se corrigieron principalmente todos aquellos problemas que surgieron en el primer sprint ya que ahora el equipo tenía una base de trabajo y mayor conocimiento en las herramientas y el lenguaje de programación usados. Desde el primer día del sprint se decidieron hacer reuniones vía online a diario para observar el progreso de cada miembro y evitar o solventar cualquier tipo de problema que pueda surgir en el desarrollo de este sprint.

Historia de usuario	Descripción	Criterios de aceptación	Tareas	ESTIMADO TAREAS	PROGRESO TAREA	ESTIMADO HU	PROGRESO HU	ESCALA COHN	ASIGNADO
Permitir ordenación en vista de marcadores por nombre o URL (HU_015)	Como usuario identificado quiero que se muestren mis marcadores pudiendo ordenarlos por nombre o url tanto de forma ascendente como descendente de manera que sea más fácil visualizarlos.	Dado un usuario identificado, cuando acceda a la opción de ver marcadores, entonces podrá cambiar la ordenación del listado en función del nombre y la url. Tras ello, se mostrarán los datos en la forma solicitada por el usuario.	Desarrollar una interfaz gráfica de usuario que permita ordenar por nombre o URL junto a su botón	2	0	6	0	2	Alejandro
			Recoger el listado de marcadores	1	0				
			Detectar tipo de ordenación solicitada por el usuario	1	0				
			Devolver listado ordenado de marcadores	1	0				
			Mostrar el listado de marcadores	1	0				
Eliminar marcadores eliminados en el navegador (HU_016)	Como usuario identificado quiero eliminar marcadores eliminados en el navegador para poder así actualizarlo directamente en la extensión.	Dado un usuario identificado, cuando haya iniciado sesión accederá a gestión de marcadores > eliminar marcadores y entonces se podrán eliminar los marcadores uno a uno. Tras ello, los marcadores eliminados de forma individual no aparecerán.	Recoger el listado de marcadores	4	0	15	0	5	Roger
			Detectar los marcadores eliminados en el navegador	4	0				
			Borrar los marcadores de la base de datos	4	0				
			Devolver el listado actualizado de marcadores	3	0				
Editar marcadores de forma manual y múltiple (HU_017)	Como usuario identificado quiero poder editar varios marcadores a la vez de forma manual para poder ahorrar tiempo cuando tenga que modificar varios.	"Dado un usuario identificado, cuando haya iniciado sesión accederá a la opción de gestión de marcadores > editar marcadores y entonces podrá editar varios marcadores y emitir los cambios a la vez. Tras ello los marcadores editados estarán con los nuevos cambios. Las url deben estar en un formato correcto así como el campo de nombre no estar vacío. ----- Dado un usuario identificado, cuando en el apartado Gestión de Marcadores > editar marcadores ponga una url en formato incorrecto, entonces se mostrará un mensaje que indique que el campo url no presenta el formato idóneo."	Desarrollar una interfaz gráfica de usuario que permita editar marcadores de forma manual o múltiple junto a su botón correspondiente	2	0	15	0	5	Daniela
			Recoger el listado de marcadores	2	0				
			Mostrar el listado de marcadores	2	0				
			Detectar el marcador o marcadores que desea editar el usuario	2	0				
			Borrar el marcador o marcadores de la base de datos	2	0				
			Añadir el marcador o marcadores actualizado/s en la base de datos	2	0				
			Devolver el listado de marcadores actualizado	2	0				
			Mostrar el listado actualizado de marcadores	1	0				
Permitir búsqueda por URL en la vista del historial (HU_018)	Como usuario identificado quiero poder permitir la búsqueda dentro del historial, donde se muestra el listado, por url para ayudarme a encontrar el dato deseado.	Dado un usuario identificado, cuando haya iniciado sesión accederá a la opción de Gestión del Historial > Ver Historial, donde tras introducir un valor en el campo de búsqueda entonces se mostrará el listado del historial que cumple dicho criterio.	Desarrollar una interfaz gráfica de usuario que permita buscar en el historial por URL junto a su botón	2	0	6	0	2	Alejandro
			Recoger el listado de historial	1	0				
			Detectar la URL introducida por el usuario	1	0				
			Devolver el listado de historial que coincide con la URL introducida	1	0				
			Mostrar el listado actualizado del historial	1	0				
Borrar historial de forma manual y múltiple (HU_019)	Como usuario identificado quiero poder borrar el historial de forma manual y múltiple para poder eliminar entradas del historial de forma más ágil.	Dado un usuario identificado, cuando haya iniciado sesión, y tenga el historial almacenado, accederá a Gestión del historial > borrar historial y entonces se podrán eliminar varias entradas a la vez. Tras ello, las entradas ya no aparecerán.	Desarrollar una interfaz gráfica de usuario que permita borrar el historial de forma manual o múltiple junto a su botón correspondiente	3	0	15	0	5	Fran
			Recoger el listado de historial	2	0				
			Mostrar el listado de historial	2	0				
			Detectar las visitas seleccionadas por el usuario	2	0				
				1	0				

5.3 Durante Sprint Backlog

Durante el Sprint Backlog no hubo demasiadas complicaciones por parte del equipo, sin embargo, debido a gran cantidad de trabajo de otras asignaturas hubo algunos retrasos en algunas de las HU que finalmente se pudieron solventar gracias a otros miembros que ayudaron a terminirlas.

Aparte de las reuniones diarias también se plantearon dudas y problemas que surgían a través de Telegram con lo que siempre había buena comunicación dentro del equipo. Todo esto en conjunto ayudó a evitar los posibles retrasos y errores en la implementación.

Historia de usuario	Descripción	Criterios de aceptación	Tareas	ESTIMADO TAREAS	PROGRESO TAREA	ESTIMADO HU	PROGRESO HU	ESCALA COHN	ASIGNADO
Permitir ordenación en vista de marcadores por nombre o URL (HU_015)	Como usuario identificado quiero que se muestren mis marcadores pudiendo ordenarlos por nombre o url tanto de forma ascendente como descendente de manera que sea más fácil visualizarlos.	Dado un usuario identificado, cuando acceda a la opción de ver marcadores, entonces podrá cambiar la ordenación del listado en función del nombre y la url. Tras ello, se mostrarán los datos en la forma solicitada por el usuario.	Desarrollar una interfaz gráfica de usuario que permita ordenar por nombre o URL junto a su botón Recoger el listado de marcadores. Detectar tipo de ordenación solicitada por el usuario. Devolver listado ordenado de marcadores. Mostrar el listado de marcadores.	2 1 1 1 1	2 1 0,5 0,5 0	6	4	2	Alejandro
Eliminar marcadores eliminados en el navegador (HU_016)	Como usuario identificado quiero eliminar marcadores eliminados en el navegador para poder así actualizarlo directamente en la extensión.	Dado un usuario identificado, cuando haya iniciado sesión accederá a la opción de gestión de marcadores > eliminar marcadores y entonces se podrán eliminar los marcadores uno a uno. Tras ello, los marcadores eliminados de forma individual no aparecerán.	Recoger el listado de marcadores. Detectar los marcadores eliminados en el navegador. Borrar los marcadores de la base de datos. Devolver el listado actualizado de marcadores.	4 4 4 3	4 2 0 0	15	6	5	Rogger
Editar marcadores de forma manual y múltiple (HU_017)	Como usuario identificado quiero poder editar varios marcadores a la vez de forma manual para poder ahorrar tiempo cuando tenga que modificar varios.	"Dado un usuario identificado, cuando haya iniciado sesión accederá a la opción de gestión de marcadores > editar marcadores y entonces podrá editar varios marcadores y emitir los cambios a la vez. Tras ello los marcadores editados estarán con los nuevos cambios. Las url deben estar en un formato correcto así como el campo de nombre no estar vacío." ----- Dado un usuario identificado, cuando en el apartado Gestión de Marcadores > editar marcadores ponga una url en formato incorrecto, entonces se mostrará un mensaje que indique que el campo url no presenta el formato idóneo."	Desarrollar una interfaz gráfica de usuario que permita editar marcadores de forma manual o múltiple junto a su botón correspondiente. Recoger el listado de marcadores. Mostrar el listado de marcadores. Detectar el marcador o marcadores que desea editar el usuario. Borrar el marcador o marcadores de la base de datos. Añadir el marcador o marcadores actualizado/s en la base de datos. Devolver el listado de marcadores actualizado. Mostrar el listado actualizado de marcadores.	2 2 2 2 2 2 2 1	2 2 1 1 1 0 0 0	15	7	5	Daniela
Permitir búsqueda por URL en la vista del historial (HU_018)	Como usuario identificado quiero poder permitir la búsqueda dentro del historial, donde se muestra el listado, por url para ayudarme a encontrar el dato deseado.	Dado un usuario identificado, cuando haya iniciado sesión accederá a la opción de Gestión del Historial > Ver Historial, donde tras introducir un valor en el campo de búsqueda entonces se mostrará el listado del historial que cumple dicho criterio.	Desarrollar una interfaz gráfica de usuario que permita buscar en el historial por URL junto a su botón Recoger el listado de historial. Detectar la URL introducida por el usuario. Devolver el listado de historial que coincida con la URL introducida. Mostrar el listado actualizado del historial.	2 1 1 1 1	2 1 1 0,5 0	6	4,5	2	Alejandro
Borrar historial de forma manual y múltiple (HU_019)	Como usuario identificado quiero poder borrar el historial de forma manual y múltiple para poder eliminar entradas del historial de forma más ágil.	Dado un usuario identificado, cuando haya iniciado sesión, y tenga el historial almacenado, accederá a Gestión del historial > borrar historial y entonces se podrán eliminar varias entradas a la vez. Tras ello, las entradas ya no aparecerán.	Desarrollar una interfaz gráfica de usuario que permita borrar el historial de forma manual o múltiple junto a su botón correspondiente. Recoger el listado de historial. Mostrar el listado de historial. Detectar las visitas seleccionadas por el usuario.	3 2 2 2	3 2 1 1	15	7	5	Fran

Sugerir el registro de usuario cuando un usuario no está registrado (HU_019)	Como usuario no identificado no registrado quiero que se pueda detectar cuando un usuario no está registrado y sugerirle al tratar de iniciar sesión que se registre para facilitar el proceso de registro.	Dado un usuario no identificado, cuando haya comprobado que un usuario no está registrado aparecerá la opción de registrar usuario. Entonces, podrá registrarse e iniciar sesión sin tener que volver a introducir el email y la contraseña. El email debe tener un formato correcto y la contraseña debe ser mayor a 8 caracteres.	Recoger el listado de usuarios.	2	2	6	5	2	Andres
		Dado un usuario no identificado, cuando en la página de registro de la extensión no introduzca un email para el registro entonces no se permitirá el inicio de sesión.	Comprobar si el usuario está registrado.	2	2				
		Dado un usuario no identificado, cuando en la página de registro de la extensión no introduzca una contraseña para el registro mayor a 8 caracteres, entonces entonces no se permitirá el inicio de sesión.	Envío de notificación de registro de usuario.	2	1				
Cerrar la sesión tras 120 minutos de haber iniciado sesión (HU_020)	Como usuario identificado quiero que se cierre la sesión después de 120 minutos para la protección de mi cuenta de usuario.	Dado un usuario identificado, cuando haya iniciado sesión entonces podrá usar su cuenta de usuario durante 120 minutos. Tras ello tendrá que volver a introducir usuario y contraseña para un nuevo inicio de sesión.	Comprobar si hay alguna sesión iniciada.	2	2	6	3,5	2	Andres
			Recoger el listado de usuarios.	1	1				
			Detectar el usuario cuya sesión está iniciada.	1	0,5				
			Comprobar el tiempo activo desde el inicio de sesión de dicho usuario.	1	0				
			Envío de notificación de cierre de sesión una vez llegado al tiempo límite.	1	0				
Permitir borrar al usuario solicitando la contraseña (HU_021)	Como usuario identificado quiero que se pueda borrar el usuario solicitando la contraseña para permitir dar de baja la cuenta con un cierto grado de seguridad.	Dado un usuario identificado, cuando haya iniciado sesión, podrá ir a la opción Gestión de Usuarios > borrar usuario, entonces introducirá la contraseña de la cuenta para poder eliminarla. Tras ello, el usuario habrá sido eliminado.	Desarrollar una interfaz gráfica de usuario que permita borrar sus datos solicitando la contraseña junto a su botón correspondiente.	3	3	15	9	5	Sergio
			Recoger el listado de usuarios.	3	3				
			Detectar el usuario que solicita borrar los datos.	3	2				
			Comprobar si la contraseña introducida coincide con la del usuario.	2	1				
			Eliminar el usuario de la base de datos.	2	0				
			Devolver el listado actualizado de usuarios.	2	0				
Detectar pestañas abiertas y almacenarlas (HU_022)	Como usuario identificado quiero que se vayan detectando las pestañas que se van abriendo en el navegador y se almacenen para mantener un registro de las mismas.	Dado un usuario identificado, cuando haya iniciado sesión y abra diferentes pestañas en el navegador entonces se irán almacenando en la extensión y se podrá observar un mensaje emergente que muestra dicha operación.	Comprobar si hay alguna pestaña abierta en el navegador.	4	4	15	10	5	Carlos
			Recoger el listado de pestañas abiertas.	4	4				
			Añadir la nueva pestaña a la lista.	4	2				
			Devolver el listado actualizado de pestañas abiertas.	2	0				
Permitir ver listado de pestañas almacenadas (HU_023)	Como usuario identificado quiero poder ver el listado de pestañas abiertas almacenadas para conocer las que hay.	Dado un usuario identificado, cuando haya iniciado sesión, abra pestañas y vaya a Gestión del navegador > Gestión de pestañas > Ver pestañas, entonces podrá ver el listado de pestañas que se han abierto y almacenado.	Desarrollar una interfaz gráfica de usuario que permita ver el listado de pestañas.	5	5	15	8	5	Sandra
			Recoger el listado de pestañas abiertas.	5	3				
			Mostrar el listado de pestañas abiertas.	5	0				
Detectar ventanas abiertas y	Como usuario identificado quiero poder detectar las ventanas abiertas para poder	Dado un usuario identificado, cuando haya iniciado sesión y abra ventanas en el navegador, podrá ir a la opción Gestión del navegador	Comprobar si hay alguna ventana abierta.	4	4				
			Recoger el listado de ventanas abiertas.	4	4				

5.4 Final Sprint Backlog

Al final del Sprint se pudieron acabar todas HU sin mayores complicaciones, con lo cual en la herramienta utilizada quedaron marcados de color verde tanto las HU como sus tareas. Se pudieron testear y comprobar que toda la implementación funcionaba correctamente y estaba libre de errores, tras esto, se unificó todo el trabajo que había subido cada miembro en el repositorio en un solo proyecto y como resultado final se obtuvo el prototipo de este segundo sprint.

Historia de usuario	Descripción	Criterios de aceptación	Tareas	ESTIMADO TAREAS	PROGRESO TAREA	ESTIMADO HU	PROGRESO HU	ESCALA COHN	ASIGNADO
Permitir ordenación en vista de marcadores por nombre o URL (HU_015)	Como usuario identificado quiero que se muestren mis marcadores pudiendo ordenarlos por nombre o url tanto de forma ascendente como descendente de manera que sea más fácil visualizarlos.	Dado un usuario identificado, cuando acceda a la opción de ver marcadores, entonces podrá cambiar la ordenación del listado en función del nombre y la url. Tras ello, se mostrarán los datos en la forma solicitada por el usuario.	Desarrollar una interfaz gráfica de usuario que permita ordenar por nombre o URL junto a su botón	2	2	6	6	2	Alejandro
			Recoger el listado de marcadores.	1	1				
			Detectar tipo de ordenación solicitada por el usuario.	1	1				
			Devolver listado ordenado de marcadores.	1	1				
			Mostrar el listado de marcadores.	1	1				
Eliminar marcadores eliminados en el navegador (HU_016)	Como usuario identificado quiero eliminar marcadores eliminados en el navegador para poder así actualizarlo directamente en la extensión.	Dado un usuario identificado, cuando haya iniciado sesión accederá a gestión de marcadores > eliminar marcadores y entonces se podrán eliminar los marcadores uno a uno. Tras ello, los marcadores eliminados de forma individual no aparecerán.	Recoger el listado de marcadores.	4	4	15	15	5	Roger
			Detectar los marcadores eliminados en el navegador.	4	4				
			Borrar los marcadores de la base de datos.	4	4				
			Devolver el listado actualizado de marcadores.	3	3				
Editar marcadores de forma manual y múltiple (HU_017)	Como usuario identificado quiero poder editar varios marcadores a la vez de forma manual para poder ahorrar tiempo cuando tenga que modificar varios.	"Dado un usuario identificado, cuando haya iniciado sesión accederá a la opción de gestión de marcadores > editar marcadores y entonces podrá editar varios marcadores y emitir los cambios a la vez. Tras ello los marcadores editados estarán con los nuevos cambios. Las url deben estar en un formato correcto así como el campo de nombre no estar vacío. Dado un usuario identificado, cuando en el apartado Gestión de Marcadores > editar marcadores ponga una url en formato incorrecto, entonces se mostrará un mensaje que indique que el campo url no presenta el formato idóneo."	Desarrollar una interfaz gráfica de usuario que permita editar marcadores de forma manual o múltiple junto a su botón correspondiente.	2	2	15	15	5	Daniela
			Recoger el listado de marcadores.	2	2				
			Mostrar el listado de marcadores.	2	2				
			Detectar el marcador o marcadores que desea editar el usuario.	2	2				
			Borrar el marcador o marcadores de la base de datos.	2	2				
			Añadir el marcador o marcadores actualizado/s en la base de datos.	2	2				
			Devolver el listado de marcadores actualizado.	2	2				
			Mostrar el listado actualizado de marcadores.	1	1				
Permitir búsqueda por URL en la vista del historial (HU_018)	Como usuario identificado quiero poder permitir la búsqueda dentro del historial, donde se muestra el listado, por url para ayudarme a encontrar el dato deseado.	Dado un usuario identificado, cuando haya iniciado sesión accederá a la opción de Gestión del Historial > Ver Historial, donde tras introducir un valor en el campo de búsqueda entonces se mostrará el listado del historial que cumple dicho criterio.	Desarrollar una interfaz gráfica de usuario que permita buscar en el historial por URL junto a su botón	2	2	6	6	2	Alejandro
			Recoger el listado de historial.	1	1				
			Detectar la URL introducida por el usuario.	1	1				
			Devolver el listado de historial que coincide con la URL introducida.	1	1				
			Mostrar el listado actualizado del historial.	1	1				
Borrar historial de forma manual y múltiple (HU_019)	Como usuario identificado quiero poder borrar el historial de forma manual y múltiple para poder eliminar entradas del historial de forma más ágil.	Dado un usuario identificado, cuando haya iniciado sesión, y tenga el historial almacenado, accederá a Gestión del historial > borrar historial y entonces se podrán eliminar varias entradas a la vez. Tras ello, las entradas ya no aparecerán.	Desarrollar una interfaz gráfica de usuario que permita borrar el historial de forma manual o múltiple junto a su botón correspondiente.	3	3	15	15	5	Fran
			Recoger el listado de historial.	2	2				
			Mostrar el listado de historial.	2	2				
			Detectar las visitas seleccionadas por el usuario.	2	2				
				?	?				

5.5 Conclusiones

El Sprint Backlog ha ayudado a mantener una buena organización dentro del equipo. Todos los miembros del equipo lo iban consultando cada día lo cual permitió ver el avance que tenía el sprint día a día. En este segundo sprint hubo muy pocos problemas de conocimiento de lenguaje y herramientas respecto al primer sprint sin embargo surgieron nuevos problemas principalmente debido a la gran carga de trabajo por otras asignaturas. Otro de los problemas también presentes durante todo el sprint fue el no poder realizar reuniones presenciales debido al estado de alarma por el que estamos pasando, pero dado la gran cantidad de medios de comunicación que tenemos actualmente pudimos mantenernos en contacto en todo momento.

Al final el sprint se terminó dentro del tiempo que se había estimado previamente, sin embargo, no se llegó a implementar toda la funcionalidad de la segunda release ya que no se repartieron suficientes historias de usuario.

6. Desarrollo del Sprint

6.1 Estructura del control de versiones

En el desarrollo del siguiente sprint se ha seguido utilizando la estructura en GitHub, la herramienta de control de versiones utilizada en el sprint anterior. Esta estructura se divide en los siguientes campos:

- **Master:** Se corresponde con la rama principal del proyecto, en ella se almacenan las versiones terminadas y entregadas de la extensión.
- **Develop:** Se crea a partir de la rama master. En esta rama se almacenan las versiones de la aplicación que han pasado las pruebas y están listas para ser entregadas como producto final.
- **Feature/nombreHU:** Todas estas ramas surgen a partir de develop. Se han creado tantas ramas feature como historias de usuario se han implementado durante este sprint.

La finalidad de establecer esta estructura se basa en conseguir tener bajo control las versiones entregadas, las que están revisadas y las historias de usuario en proceso. Con esto se logra tener la versión final en la rama master sin crear conflictos a la hora de desarrollar nuevas funcionalidades, gracias a implementarse en ramas diferentes cada una de ellas.

6.2 Frecuencia de realización de pruebas

Pruebas unitarias y de integración

Para la ejecución de estas pruebas se han empleado las herramientas mocha y chai, con las que podemos ejecutar distintas pruebas sobre el código JavaScript de nuestra extensión.

Estas pruebas se han llevado a cabo cada vez que se ha hecho un commit en la rama feature de cada una de las funcionalidades a implementar. Gracias a la herramienta de integración continua Travis de GitHub se han ejecutado de manera automática.

Pruebas de interfaz

Se han llevado a cabo cada una de las veces que se ha unido una rama de historia de usuario en la rama develop de la extensión. Se realiza como último paso para validar de forma total la historia de usuario que se haya entregado.

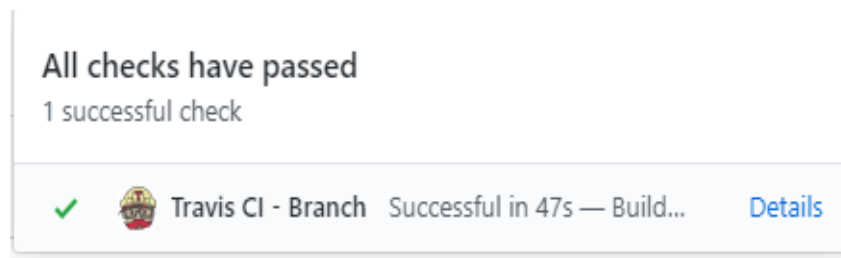
Pruebas de aceptación

Se realizarán dos días a la semana, tras haber pasado las pruebas automatizadas. Su realización nos permitirá controlar el ajuste de la extensión a las exigencias del cliente y el buen funcionamiento de las historias de usuario desarrolladas.

Pruebas de sistema

Al presentar nuestro proyecto una arquitectura cliente, servidor, realizaremos pruebas que afecten al correcto funcionamiento del software desarrollado correspondiente con la parte que interactuará con el usuario. Para ello, adoptaremos el papel de cliente y probaremos cada funcionalidad.

Por otra parte, para conocer el funcionamiento del servidor en el que se almacenan los datos, testaremos la correcta vinculación y conexión con este y el nivel de carga que soporta a la hora de demandarle operaciones con los datos que almacena.



6.3 Sprint Backlog

En este sprint se comienza con las historias de usuario de la segunda release. En esta había definidas un total de catorce historias de usuario, pero a causa de la carga académica que experimentan los miembros del equipo, se deciden implementar doce historias de usuario a repartir entre dichos miembros.

Una vez decididas las historias que se van a implementar, se van asignando a cada miembro del equipo una o dos historias dependiendo de su complejidad y de los puntos de historias dados según la escala de Cohn.

A cada una de las historias de usuario se les asignó un tiempo estimado de desarrollo que varía según su complejidad, además de un campo progreso para controlar la evolución de la implementación de esa tarea respecto a su tiempo estimado. En el caso de completar el progreso, la tarea se marcaría como DONE, adoptando un color verde.

Por el contrario, si estuviese en proceso, se marcaría como DOING y mantendría un color anaranjado.

Se han podido desarrollar la totalidad de las historias de usuario que se plantearon en un primer momento, pudiendo cumplir con los plazos que se establecieron al inicio del sprint.

En cuanto al Backlog de los impedimentos de la reléase anterior, muchos se lograron solventar, entre ellos, el desconocimiento de los diferentes lenguajes que se utilizan en la implementación de la aplicación, el uso de la herramienta GitHub y el método de realización de las diferentes pruebas unitarias, que se solucionaron con ayuda de reuniones con un miembro del equipo experto en los diferentes temas planteados y de tutoriales en internet.

6.4 Daily Scrum

Durante el desarrollo de este sprint se llevaron a cabo 15 reuniones diarias. Estas reuniones sirvieron para planificar el desarrollo de las tareas que se iban a realizar, resolver dudas entre el equipo de desarrollo a medida que se iban implementando las historias de usuario designadas para este sprint, y consultando con el Product Owner si se iba ajustando a sus exigencias.

En el apartado dedicado a las reuniones diarias, se explicará de manera más detallada lo que se acometió en cada una de las reuniones que se mantuvieron.

6.5 Problemas durante el desarrollo

Gracias al sprint anterior, el equipo entendía perfectamente el funcionamiento de la herramienta GitHub, por lo que en este nuevo sprint no hubo tantos problemas a la hora de compartir el trabajo realizado por los miembros que componen el equipo. Sin embargo, si hubo algunos problemas de entendimiento en cuanto al funcionamiento de la API de Chrome y a la implementación de los diferentes interfaces de usuario. Estos problemas de entendimiento fueron solventados en reuniones en las que un experto del tema resolvía las dudas existentes.

Durante el desarrollo del sprint, un miembro del equipo borro el usuario de la Base de Datos que realizaba las pruebas automáticamente, por lo que las pruebas unitarias fallaban. Se volvió a crear al usuario para que volviesen a funcionar.

Otro de los problemas durante el desarrollo fue el poco tiempo del que disponía el equipo debido al confinamiento y a los exámenes venideros. También coincidieron

varios días de reunión con días festivos, por lo que el equipo no estaba centrado completamente.

A pesar de ello, los miembros del equipo pudieron acabar correctamente todo su trabajo y ayudaron a los otros miembros en caso de necesitarlo.

7. Reuniones diarias

A lo largo del sprint se han realizado 15 reuniones en las cuales se planteaban las dudas surgidas y se trataban de solventar los problemas encontrados. Estas reuniones, al seguir estando bajo la situación del COVID-19 no han podido ser presenciales y se ha utilizado para ello Microsoft Teams. Además, esto ha impedido al grupo grabar en video o tomar fotografías del equipo realizando las mismas.

Las reuniones decidimos hacerlas desde un inicio en horario de lunes a viernes de 13:00 a 13:20.

Se indica, a continuación, un resumen de lo tratado en cada una de las daily meetings.

Reunión 1: 29/04/2020

Al realizar la reunión por la mañana y ser el primer día del sprint no hubo problemas aún, por lo que se aprovechó para tratar en profundidad los atributos de las nuevas entidades del modelo de datos y planificar los siguientes pasos que seguir, como aumentar la base, esa misma mañana, acorde a las nuevas historias antes de comenzar las nuevas historias de usuario por la tarde.

Reunión 2: 30/04/2020

Tras realizar la base el día anterior y comenzar con las historias de usuario, se vio que alguna de las ideas iniciales de almacenar las URLs de las ventanas de Chrome no era posible de la forma prevista debido a la API que Chrome, por lo que la persona encargada de esa historia de usuario estuvo ese día buscando alternativas tal y como se acordó en la reunión. Respecto al resto de historias de usuario, ninguna persona había encontrado aún grandes impedimentos.

Reunión 3: 01/05/2020

A pesar de que parecía haber quedado clara la base, algunos integrantes entendieron mal la nueva ampliación y eso derivó en algún bloqueo en el avance de las historias.

Reunión 4: 04/05/2020

Ha habido problemas al implementar la funcionalidad de pestañas porque se obtenía un valor NULL en el evento producido al crear nuevas pestañas. La solución ha sido de reunirse el encargado de la historia de usuario con los más expertos en el proyecto para ver si en la daily del día siguiente se había solucionado.

Reunión 5: 05/05/2020

Ha habido problemas en la parte de node porque el token se devolvía NULL, de tal manera que dicho integrante no ha podido seguir con la implementación ese día espera encontrar una solución respecto a dicho problema.

Reunión 6: 06/05/2020

El problema del día anterior ha sido resuelto dado que al reunirse todos los desarrolladores se pudo aclarar dicho problema.

Reunión 7: 07/05/2020

La reunión transcurrió siguiendo las indicaciones del Scrum Máster de tal manera que se ha encargado de asegurar una correcta realización de la reunión. Los integrantes del equipo siguen implementando la historia que le corresponde a cada uno.

Reunión 8: 08/05/2020

Un compañero tiene un problema con su funcionalidad y no puede detectar las pestañas correctamente y como que otro compañero está bastante avanzado con su funcionalidad le ayudo para así poder avanzar.

Reunión 9: 11/05/2020

Debido a la saturación de trabajos y exámenes en otras asignaturas, varios compañeros llevan un retraso en su tarea, se busca la solución y se consensua que los que van más avanzados ayuden un poco a estos.

Reunión 10: 12/05/2020

Los integrantes que están agobiados comentan que les ha venido muy bien el pequeño apoyo de los compañeros y han dejado sus tareas al día, por otro lado, los demás tienen pequeñas dudas de funcionalidad que son respondidas en la misma reunión.

Reunión 11: 13/05/2020

Se comenta en la reunión que hay que tener cuidado con la BD, ya que las pruebas daban un error debido a que un compañero había eliminado un user de la BD con el cual se comprobaban varias funcionalidades.

Reunión 12: 14/05/2020

Un compañero había pasado un día atascado debido a que había un problema en los callback al hacer bucles FOR. Este mismo recibió apoyo por parte del Scrum Máster. Esta solución dio al equipo una visión nueva de implementación que ninguno conocía.

Reunión 13: 15/05/2020

Algún integrante tenía dudas respecto al Daily correspondiente al mismo día, dado que era festivo (San Isidro). El Scrum Master explico que nosotros íbamos a tomar los viernes como día laborable .Aun así había algunos retrasos en la implementación de algunas funcionalidades.

Reunión 14: 18/05/2020

Todos los desarrolladores han acabado y subido su trabajo a los repositorios de GitHub. El Scrum Máster decide que el día siguiente va a juntar todas las partes de los desarrolladores.





Reunión 15: 19/05/2020



Todos los miembros del equipo han conseguido acabar a tiempo cada historia asignada, dado que algunos miembros más expertos ayudaron a los otros miembros cuando surgían problemas.

De este modo se aseguraba que el sprint se acababa a tiempo y en las condiciones adecuadas. En ese mismo día el Scrum Máster reúne todas las partes de cada miembro del equipo.

8. Backlog de Impedimentos

A continuación, se muestra la tabla realizada para analizar la evolución de los impedimentos surgidos durante el segundo sprint: (*EQ ≈ Equipo de desarrollo)

Impedimento	Prioridad	Acción	Dueño	Fecha	Vencimiento	Resultado
Debido a exámenes y entregas de otras asignaturas había un pequeño retraso de calendario	Alta	Intentar adelantar trabajo para así no llegar a las fechas límites de las entregas cargados de trabajo	EQ	12/05/2020	19/05/2020	
NodeJS no es muy compatible con bucles FOR así que hubo que usar recursión	Alta	Investigar en internet mediante tutoriales y experiencias de otros usuarios como aplicar métodos recursivos en NodeJS	EQ	05/05/2020	08/05/2020	
Fallo en la interfaz gráfica, no cuadraba en ciertos formatos	Alta	Modificar mediante información encontrada en internet los formatos de la interfaz gráfica para que concordase	Daniela Fran Pedro Carlos	04/05/2020	05/05/2020	
Los test fallaron en un momento dado debido a que alguien había borrado el usuario que usaba los test en la BBDD	Alta	Encontrar el fallo en la BBD, crear de nuevo el usuario que ejecuta los test y volver a comprobarlo	Andrés	15/05/2020	16/05/2020	

Problemas para entender el formato DATE	Alta	Estudiar en manuales de internet como funciona el formato DATE	EQ	06/05/2020	07/05/2020	
La función Chrome no guardaba los valores deseados	Alta	Observar en la documentación de las APIs de Chrome una alternativa a esa función	Carlos	11/05/2020	13/05/2020	

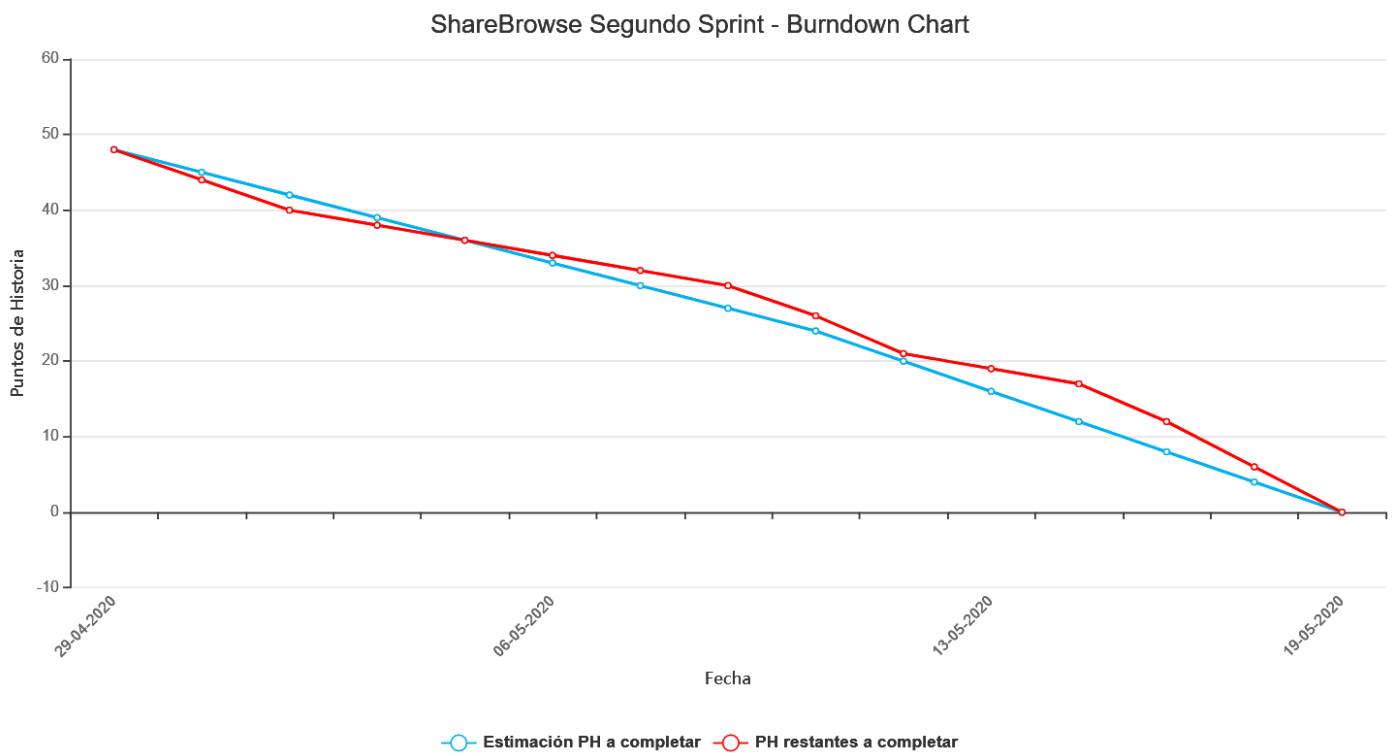
Este elemento Scrum permite al equipo monitorizar los impedimentos surgidos durante el transcurso del segundo sprint. Se ha creado esta tabla con el dueño del impedimento, fecha en la que surgió, fecha en la que debería de estar resuelta, impedimento, plan de acción y su resultado. El Scrum Master fue el encargado de determinar las fechas de vencimiento de cada uno de los impedimentos.

9. Burndown Chart

El gráfico de Burndown Chart tiene como objetivo observar la evolución de trabajo a lo largo del sprint de una forma muy clara y visual. Además, es posible comprobar si el equipo es capaz de cumplir los objetivos fijados dentro del sprint.

Fecha	Estimación PH a completar	PH restantes a completar
29/04/2020	48	48
30/04/2020	45	44
01/05/2020	42	40
04/05/2020	39	38
05/05/2020	36	36
06/05/2020	33	34
07/05/2020	30	32
08/05/2020	27	30
11/05/2020	24	26

12/05/2020	20	21
13/05/2020	16	19
14/05/2020	12	17
15/05/2020	8	12
18/05/2020	4	6
19/05/2020	0	0



Podemos observar que durante este sprint la evolución de trabajo de los primeros días fue muy positiva, ya que los miembros más expertos del equipo de desarrollo realizaron sus tareas más rápido de lo esperado.

Pero apenas unos días después esta ventaja se perdió debido a los miembros menos expertos ya que su avance era más lento y solicitaban ayuda a los miembros más avanzados. Aun así, la evolución de trabajo siguió su curso correctamente.

Durante los últimos días sufrimos un leve repunte, ocasionado por la gran cantidad de trabajo del resto de asignaturas, pero que no impidió la correcta finalización de todas las historias de usuario previstas para este segundo sprint.

10. Reunión de revisión

Al igual que con el sprint anterior, en este tampoco hemos podido hacer la reunión de revisión de manera presencial, de manera que, tal y como hicimos anteriormente, la hicimos a través de Microsoft Teams.

En la reunión se le presentaron las nuevas características producto del sprint al Product Owner, así como la integración de ellas con lo que teníamos anteriormente.

En esta ocasión además de ampliar las funcionalidades anteriores, habíamos implementado dos nuevos apartados, que son la gestión de pestañas y de ventanas.

Feedback

El producto del sprint se seguía adaptando a la idea del Product Owner con las nuevas funcionalidades.

En esta ocasión, después de ver que las funcionalidades de la extensión iban ampliándose, el Product Owner nos ha sugerido que se diseñe una interfaz más clara y ordenada para hacerla más intuitiva.

11. Reunión de retrospectiva

En la reunión de retrospectiva, al igual que la anterior vez, se analizaron los puntos que han hecho que el sprint funcione y los que han podido obstaculizarlo.

En general, el equipo de desarrollo ha mejorado con respecto al anterior, ya que en esta ocasión había aprendido que la organización y la buena repartición del trabajo era clave.

Primera semana

Como para este sprint ya se disponía del proyecto estable y funcional del anterior sprint, el comienzo del desarrollo ha sido mucho más rápido y eficaz, además el equipo de desarrollo ya poseía un cierto conocimiento sobre el lenguaje de programación de manera que todos sabían cuál era su parte y como organizarse, además de que surgieron muchas menos dudas sobre el lenguaje de programación que obstaculizaran el desarrollo, aunque el hecho de encontrarnos a finales de cuatrimestre ha hecho que

todos los miembros tengan otros compromisos y entregas de otras asignaturas lo que ha hecho que no todos hayan podido trabajar a la vez de manera conjunta.

Segunda semana

En la segunda semana el equipo de desarrollo fue avanzando con el código, pero de forma más lenta, debido a que en este sprint las funcionalidades que tenían que implementar eran más complejas y requerían conocimientos más técnicos, hubo ciertos problemas debido a ellos.

Entre ellos estaban en el desconocimiento de la API de Chrome que en esta ocasión era más necesario un dominio de ella, también el hecho de que NodeJS no es tan compatible con los bucles FOR a los que están acostumbrados los miembros del equipo de desarrollo y había que utilizar funciones recursivas que sustituyeran los iterativos con las que los miembros del equipo no estaban tan familiarizados.

También hubo que arreglar algunos elementos de la interfaz ya que no cuadraban ciertos formatos en ella.

Tercera semana

En la tercera semana, en cierto momento fallaron los test de integración debido a que se había borrado el usuario que usaban los test de la base de datos, cuando fue localizado y reparado el error los test volvieron a funcionar.

Debido a que durante este sprint ha habido 2 viernes festivos, y en la planificación se contaba con estos dos días como lectivos, si algún miembro no trabajaba esos días tenía que doblar su trabajo al lunes siguiente.

Después de analizar el desarrollo del equipo en este sprint se llegaron a las siguientes conclusiones:

- El equipo ya tenía cierto dominio sobre el lenguaje de programación, por lo tanto se empezó a programar con mucha más eficiencia, pero esto no hizo que surgieran ciertos problemas y dudas con las tecnologías debido a que había aumentado su complejidad.
- El desarrollo nunca está exento de problemas y cuestiones que pueden hacer que se paralice el desarrollo, pero es importante saber solventarlos lo antes posible para que no se retrase la entrega.
- El equipo ya tenía experiencia en programar en casa y hacer reuniones de manera telemática lo que ha dado cierta agilidad al desarrollo.

12. Comparativa con el primer sprint

Después de analizar el rendimiento del equipo en estos dos sprints se ha podido ver una mejora notable probablemente debido a lo siguiente:

En el primer sprint el equipo no tenía la experiencia suficiente con el lenguaje de programación ni con el desarrollo Scrum, para este sprint el equipo ya había adquirido cierta experiencia en ambas cosas y sabían lo que tenían que hacer tanto personalmente como en grupo para desarrollar eficientemente.

La organización también ha sido un punto importante, ya que en el primer sprint el equipo empezó más desorganizado y separado, lo que hizo que el comienzo fuese muy lento y poco eficiente, para este sprint se ha dejado claro desde el primer día el trabajo y función en el equipo de cada uno, por lo tanto, el equipo comenzó a codificar de manera eficiente desde el primer momento con suficiente tiempo para no ir tan ajustados con el día de la entrega.

Por último destacar también la solidaridad del equipo, que en el primer sprint ya fue notable y se vio como era clave para el desarrollo, ya que no todos los miembros del equipo de desarrollo tienen el mismo conocimiento sobre las tecnologías y lenguajes de programación de la extensión y para este sprint, además, las historias de usuario habían sido repartidas de otra manera y en muchas ocasiones el código que tenían que aprovechar los miembros había sido escrito por otro integrante del equipo y, los miembros que eran conscientes de la importancia de la solidaridad, dedicaron el tiempo necesario para explicar su trabajo anterior.

13. Lecciones aprendidas durante el desarrollo del proyecto

En el desarrollo de este proyecto se ha utilizado la metodología Scrum; una metodología ágil muy enfocada al desarrollo incremental y colaborativo con la que el equipo no tenía experiencia ya que nunca habían desarrollado de esta manera, lo que nos ha hecho aprender mucho sobre Scrum.

Hemos aprendido la importancia de invertir el tiempo que sea necesario a organizar y planificar para cumplir con las fechas, sobre todo, porque en el caso de esta asignatura los grupos son de 11 personas y cuesta trabajar en conjunto con un equipo grande y sincronizarlas para que todas trabajen acorde a lo previsto. Además, en un principio no nos conocíamos apenas entre nosotros lo que podía crear un ambiente de trabajo más frío, pero después de realizar varias reuniones y, sobre todo, dinámicas grupales, los

miembros fuimos conociéndonos y ganando confianza entre nosotros, lo que hizo que estuviésemos mucho más a gusto trabajando y con más motivación para hacer un producto de calidad en conjunto.

14. Dinámica grupal

Hemos realizado diferentes dinámicas de grupo aplicando un enfoque concreto a cada una de ellas. El objetivo de estas dinámicas ha sido diverso.

Algunas se han basado en mejorar la comunicación, motivación o efectividad del equipo, mientras que otras se han puesto en marcha para mejorar la efectividad de la retrospectiva del actual y futuros sprints.

14.1 El barco

Descripción de la actividad

Una dinámica muy conocida y efectiva. Se trata de poder expresar de una manera metafórica qué tal ha transcurrido el sprint. Siendo el equipo el barco, hemos de identificar qué nos ha frenado en el desarrollo del sprint (el ancla), qué riesgos hemos encontrado (las piedras), qué nos ha ayudado a avanzar (el viento) y en qué medida hemos alcanzado nuestros objetivos (la isla).

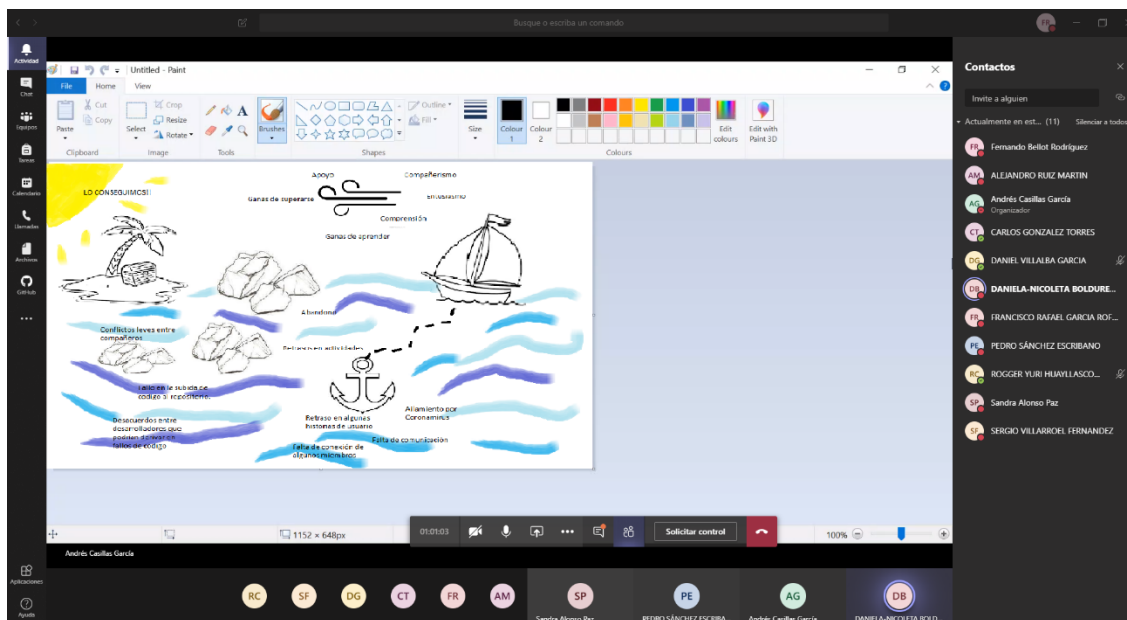
Cada uno de los miembros del proyecto podrá detallar un ejemplo de cada uno de ellos y se generará una conversación para lograr entendimiento de los diferentes puntos de vista.

Objetivo

El Barco es una técnica apropiada para poder tomar un feedback de equipo además de ser utilizada para trabajar la eficiencia dentro de un grupo de trabajo.

De esta manera podemos saber cada aspecto de complacer o disgusto de cada miembro del equipo y mejorar el ambiente, comunicación y trabajo en el equipo.

Desarrollo y resultado



14.2 Repetir-Evitar

Descripción de la actividad

Repetir-Evitar es una actividad grupal en la cual se entrega un bolígrafo a cada participante

Se pide a todos que escriban en orden en la pizarra, en nuestro caso virtual, una cosa que seguirían haciendo (Repetir) y otra cosa que dejarían de hacer en el proyecto (Evitar).

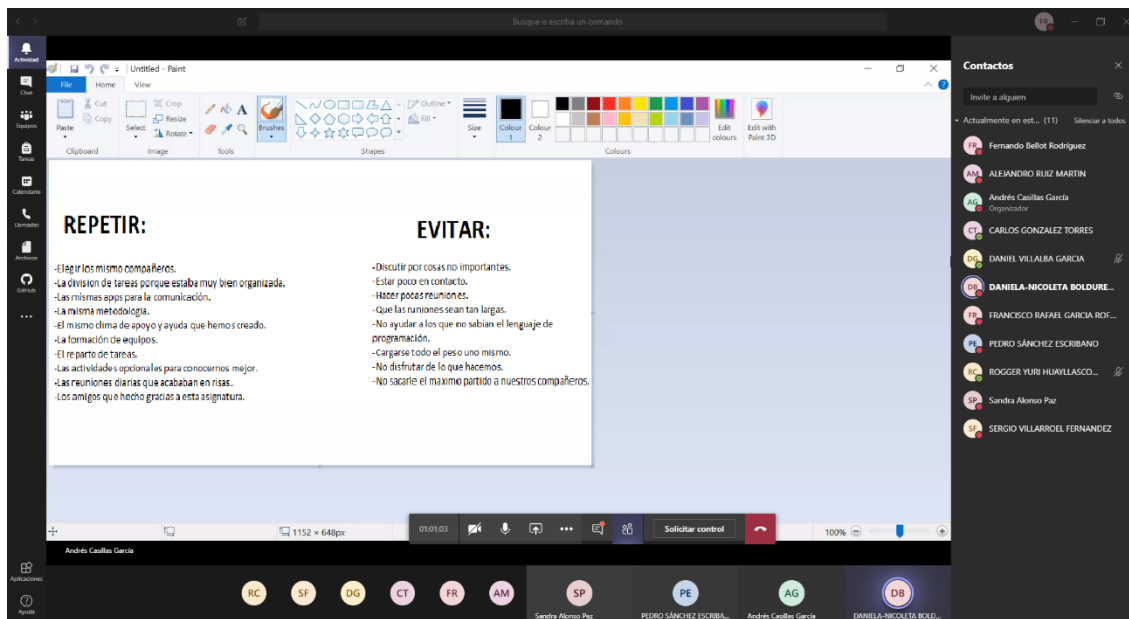
De esta manera y una vez hayan escritos en la pizarra todos los participantes podemos ver si el grupo está de acuerdo en Repetir-Evitar ciertos procesos en el futuro mediante una discusión grupal.

Objetivo

Repetir-Evitar es una actividad de recopilación de datos grupales simple y efectiva, también apropiada para poder tomar un feedback de equipo. Se basa en dos preguntas muy específicas: ¿Qué repetir? ¿Qué evitar?

De esta manera podemos dejar de ejecutar ciertos procesos que no benefician al grupo y seguir repitiendo aquellos que sí lo hacen, aumentando así la eficiencia grupal.

Desarrollo y resultado



14.3 Reconocimiento

Descripción de la actividad

Reconocimiento consiste en dar un sincero agradecimiento a uno de los participantes. Puede ser cualquier cosa con la que haya contribuido: ayudarte a ti, a un miembro del equipo, resolver un problema, etc.

Una integrante del proyecto al azar comenzará la actividad y detallará su reconocimiento. Seguidamente invitará a otros a que hagan lo mismo y se esperará a que alguien rompa el hielo.

La retrospectiva terminará cuando nadie hable durante unos cuantos minutos.

Objetivo

El objetivo de la actividad grupal Reconocimiento es fortalecer la cohesión del equipo y las relaciones personales. También muestra las habilidades comunicativas de los integrantes y se basa en el apoyo entre participantes durante las etapas de desarrollo del proyecto.

También deja que los miembros del equipo reconozcan algo positivo de otros para terminar la retrospectiva de una forma positiva.

Desarrollo y resultado

