

# Actividad integradora. Producto final U3

Asesora: Elizabeth Cristina Hernandez Hernandez. Presentarás el proyecto final y su documentación.

**EQUIPO**

**JAIME TONATIU BARAJAS VALENCIA**

**ANDREA AVENDAÑO CERVANTES**

Contenido

Logotipo

Descripción generada automáticamente

[Actividad integradora. Producto final U3 1](#_Toc135686003)

[Fases de los Sprint 3](#_Toc135686004)

[Actividades realizadas en el curso 4](#_Toc135686005)

[Los Sprint graficados 6](#_Toc135686006)

[Evidencia del funcionamiento de los Sprint 6](#_Toc135686007)

[Código de Sprint almacenados en Github 7](#_Toc135686008)

[El rol de cada participante: 9](#_Toc135686009)

[Calendario de Reuniones - mayo 2023: 9](#_Toc135686010)

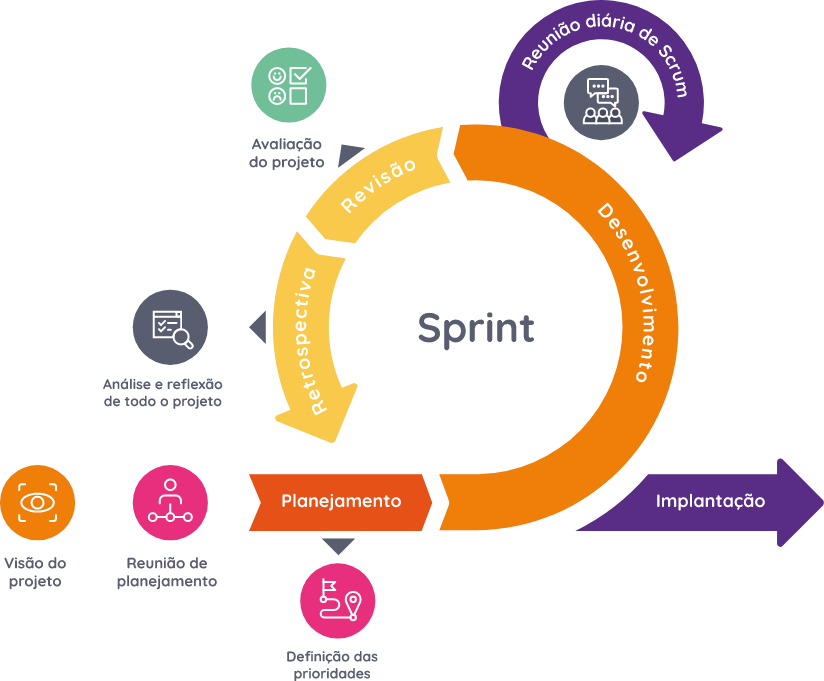
[Bitácora de modificaciones y cambios en GitHub 10](#_Toc135686011)

[Código Fuente: 13](#_Toc135686012)

[Reporte de las modificaciones realizadas 13](#_Toc135686013)

# Fases de los Sprint

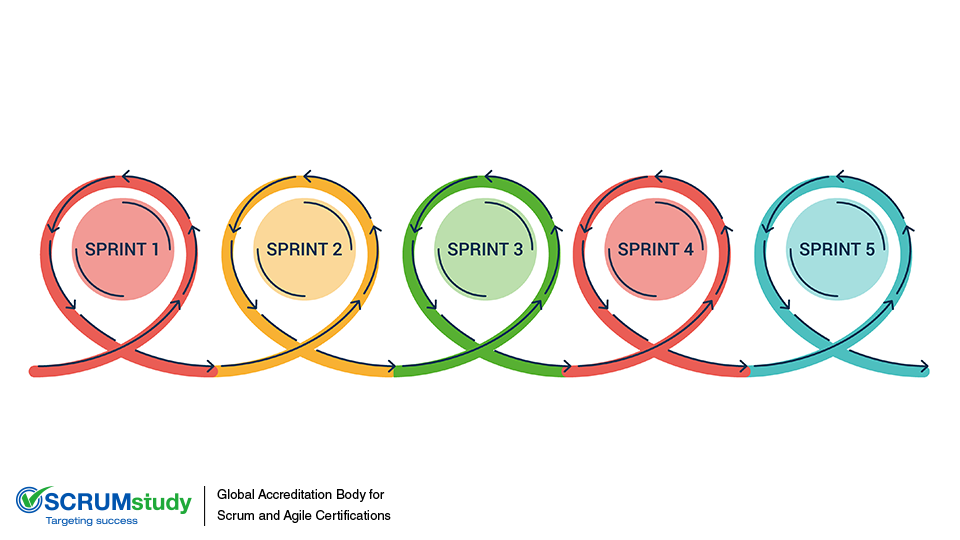
**Planificación:** En primer lugar, se debe realizar una planificación detallada del sprint. Se deben definir los objetivos y los requisitos específicos del proyecto. Además, se debe establecer un cronograma y asignar tareas a los miembros del equipo.

**Diseño:** A continuación, se debe realizar el diseño de la interfaz de usuario. En este paso, se deben definir los componentes y la estructura de la aplicación.

**Configuración del entorno**: Luego, se debe configurar el entorno de desarrollo. Esto incluye la instalación de React y otras herramientas necesarias, como Webpack o Babel.

**Desarrollo de componentes:** A partir de aquí, se comienza con el desarrollo de los componentes de React. Cada

componente debe ser desarrollado y probado de manera individual para asegurarse de que funcione correctamente.

**Integración de componentes:** Una vez que se han desarrollado todos los componentes necesarios, se deben integrar en la aplicación. Esto implica la conexión de los componentes para formar la aplicación completa.

**Pruebas:** Después de la integración, se deben realizar pruebas exhaustivas para asegurarse de que todo funciona correctamente. Se deben probar los diferentes escenarios y casos de uso para garantizar que la aplicación sea estable y funcional.

# Actividades realizadas en el curso

Durante el inicio de cada sprint, se realiza una reunión de planificación en la que el equipo selecciona las tareas que se abordarán durante el sprint. Estas tareas se toman del backlog del producto, que es una lista priorizada de todas las características, mejoras y correcciones pendientes del proyecto.

Una vez que se seleccionan las tareas, el equipo trabaja en ellas durante el sprint, dividiéndolas en tareas más pequeñas si es necesario. El progreso se rastrea diariamente a través de reuniones cortas llamadas "daily scrums" o "stand-up meetings", donde cada miembro del equipo comparte qué ha hecho, qué planea hacer y si hay algún obstáculo que le impida avanzar.

Al final del sprint, se lleva a cabo una revisión del sprint, donde el equipo muestra los resultados de su trabajo al cliente, al producto owner y a otros stakeholders relevantes. También se realiza una retrospectiva del sprint, en la que el equipo reflexiona sobre el proceso y busca formas de mejorar en futuros sprints.

El ciclo de esprints se repite hasta que se completa el proyecto o se alcanza el objetivo final. La idea es entregar incrementos de software funcionales al final de cada sprint, lo que permite obtener retroalimentación temprana y realizar ajustes según sea necesario.

En resumen, un sistema implementado en sprint se basa en el enfoque ágil de Scrum, donde se realizan iteraciones cortas y enfocadas para desarrollar y entregar software de manera incremental, maximizando la colaboración y la adaptabilidad.

**Depuración y resolución de problemas:** Si se encuentran errores durante las pruebas, se deben depurar y resolver los problemas.

**Despliegue:** Por último, se debe desplegar la aplicación en un entorno de producción para que los usuarios puedan acceder a ella.

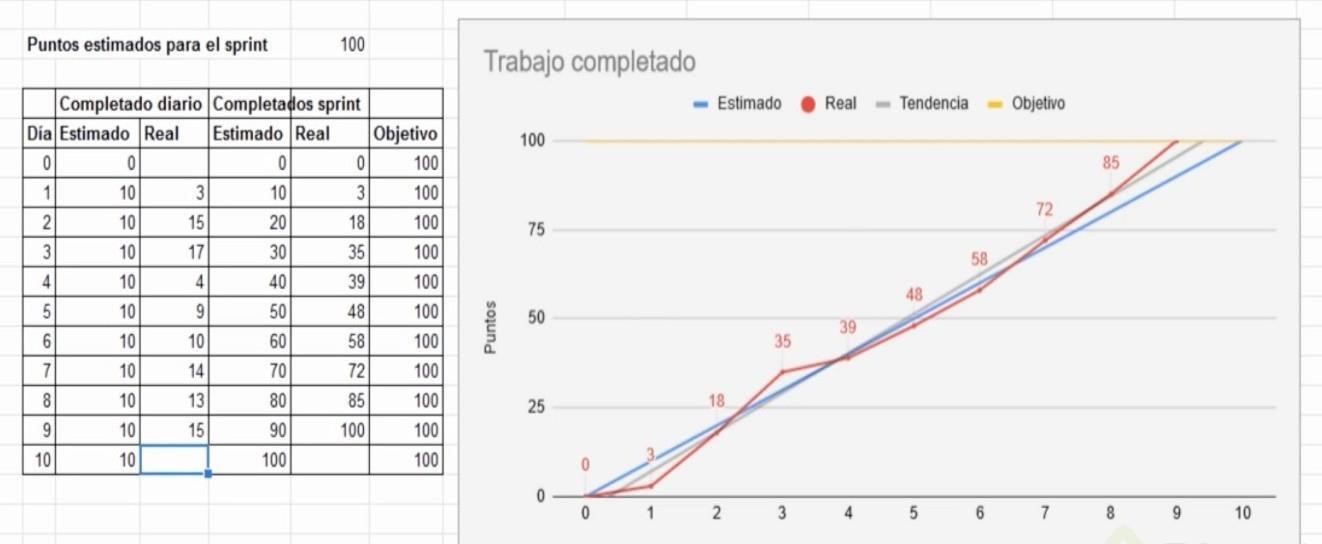
Estos son los pasos generales para desarrollar un sprint con React. Es importante recordar que cada proyecto puede tener requerimientos y características únicas, por lo que siempre se debe adaptar la metodología según las necesidades específicas de cada caso.

|  |  |  |  |
| --- | --- | --- | --- |
| **Sprint** | **Duración** | **Objetivo del Sprint** | **Elementos de trabajo seleccionados** |
| Sprint 1 | 2  semanas | Implementar el inicio de sesión de usuario | Crear la página de inicio de sesión, integrar con el backend, agregar validaciones de entrada de usuario |
| Sprint 2 | 3  semanas | Implementar la funcionalidad de búsqueda | Crear la página de búsqueda, integrar con el backend, agregar filtros de búsqueda |
| Sprint 3 | 4  semanas | Mejorar la experiencia de usuario en dispositivos  móviles | Rediseñar la interfaz para dispositivos móviles, optimizar la velocidad de carga en dispositivos móviles, agregar funciones específicas para dispositivos móviles |

En esta tabla se muestra la duración de cada Sprint, el objetivo que se busca lograr durante ese período, y los elementos de trabajo seleccionados para lograr ese objetivo. Cada uno de estos elementos de trabajo es una tarea específica que el equipo de desarrollo debe realizar durante el Sprint.

# Los Sprint graficados

Gráfico del sprint: trabajo pendiente, es un diagrama que muestra el trabajo que queda por hacer en comparación con el tiempo necesario para completarlo. Durante un sprint, el equipo usará estos gráficos para estimar cuánto tiempo demandará cada proceso iterativo.



Diagrama

Descripción generada automáticamente

# Evidencia del funcionamiento de los Sprint

|  |  |  |
| --- | --- | --- |
| ***Rol*** | ***Encargado*** | ***Estado*** |
| Product Owner | Jaime Tonatiu Barajas Valencia | Cumplido |
| ScrumMaster | Jaime Tonatiu Barajas Valencia | Cumplido |
| Desarrollador | Andrea Avendaño Cervantes | Cumplido |

|  |  |
| --- | --- |
| ***Necesidades*** | ***Cómo fueron cubiertas*** |
| Realizar ventas en línea | La aplicación permite realizar ventas en línea |
| Poder hacer reservaciones | Con la opción de reservaciones y el calendario que agregamos ahora es posible agendar reservaciones hasta con un añon de anticipación. |
| Recibir pedidos con anticipación | Ahora es posible enviar un pedido con anticipación y evitar la espera y aglomeración. |

|  |  |
| --- | --- |
| ***Necesidades*** | ***Retroalimentación del cliente*** |
| Realizar ventas en línea | * Fácil acceso. * Veloz. * Fácil de utilizar. * Aumentó las ventas. * Aumentó las ganancias. |
| Poder hacer reservaciones | * Mejoró la organización. * Evitó aglomeraciones. * Atrajo nuevos clientes * Mejoró el servicio |
| Recibir pedidos con anticipación | * Disminuyó la espera. * Mejoró el servicio. * Aumentó las ventas. * Aumentó las ganacias. |

# Código de Sprint almacenados en Github

<https://github.com/Andreaavencer/Producto-final-acitivdad-integradora.git>

**Agenda del trabajo realzado en equipo:**

* Revisión de los objetivos y metas del equipo.

* Análisis del progreso de los proyectos en curso.

* Identificación y análisis de problemas o dificultades en el trabajo.

* Planificación de próximas acciones y tareas a realizar.

* Comentarios finales y cierre de la reunión.
* **Desarrollo de la reunión:**

* Se revisó el progreso de los proyectos en curso, destacando los logros y las dificultades encontradas.

* Se identificaron algunos problemas y dificultades en el trabajo, discutiendo las posibles soluciones.
* Se planificaron las próximas acciones y tareas a realizar, asignando responsabilidades y estableciendo plazos.

* Se permitió un tiempo para comentarios finales y se agradeció a todos los asistentes por su participación.

* Se dio por concluida la reunión y se acordó la fecha y hora para la próxima reunión.

* Espero que esta minuta de reunión te sea útil como ejemplo para realizar una para tu propio equipo de trabajo.





# El rol de cada participante:

|  |  |
| --- | --- |
| **Participante** | **Rol** |
| Jaime Barajas valencia | Facilitador |
| Andrea Avendaño Cervantes | Minutista |

El facilitador es responsable de liderar la reunión, mantener el orden del día, asegurarse de que todos los puntos sean discutidos y tomar decisiones si es necesario. También se encarga de hacer un seguimiento de los tiempos y moderar cualquier debate o discusión.

El minutista es responsable de tomar notas durante la reunión, registrar los puntos importantes discutidos, las decisiones tomadas y las tareas asignadas. Al finalizar la reunión, el minutista deberá compartir las actas o minutas con los participantes para su revisión y referencia futura.

Ten en cuenta que estos roles son solo ejemplos y pueden ajustarse según las necesidades y preferencias de los participantes.

# Calendario de Reuniones - mayo 2023:

Reunión de inicio del proyecto

Fecha: 2 de mayo

Hora: 9:00 a.m. - 10:00 a.m.

Reunión de seguimiento semanal

Fecha: 9 de mayo

Hora: 11:00 a.m. - 12:00 p.m.

Reunión de revisión de hitos

Fecha: 15 de mayo

Hora: 10:00 a.m. - 11:30 a.m.

Reunión de planificación adicional

Fecha: 18 de mayo

Hora: 2:00 p.m. - 3:00 p.m.

Reunión de seguimiento semanal

Fecha: 23 de mayo

Hora: 9:30 a.m. - 10:30 a.m.

Reunión de cierre del proyecto

Fecha: 30 de mayo

Hora: 3:00 p.m. - 4:30 p.m.

Estos son solo ejemplos y las fechas y horas pueden ajustarse según las necesidades y disponibilidad de los participantes. Además, es importante enviar invitaciones y confirmar la disponibilidad de los participantes antes de establecer definitivamente las fechas y horarios de las reuniones.

# Bitácora de modificaciones y cambios en GitHub

### Fecha: 28/04/2023

En esta bitácora se registrarán las nuevas modificaciones realizadas en GitHub, incluyendo los cambios en el backlog y en el propio repositorio.

[Fecha actual]

Se realizó una revisión del backlog para identificar nuevas tareas y funcionalidades que deben ser implementadas en el proyecto.

Se agregaron tres nuevas tareas al backlog: "Implementar autenticación de usuarios", "Mejorar la interfaz de usuario" y "Optimizar rendimiento del sistema". Se asignaron responsables a las nuevas tareas en el backlog: Juan será responsable de "Implementar autenticación de usuarios", María se encargará de "Mejorar la interfaz de usuario" y Carlos estará a cargo de "Optimizar rendimiento del sistema".

### Fecha: 1/05/2023

Se realizó una reunión de equipo para discutir los cambios propuestos en el backlog y asignar prioridades.

Después de la discusión, se decidió reordenar el backlog para dar prioridad a la tarea "Optimizar rendimiento del sistema" debido a su impacto en la experiencia del usuario.

Se actualizó el backlog y se reasignaron las responsabilidades: Juan se encargará de "Optimizar rendimiento del sistema", María trabajará en "Implementar autenticación de usuarios" y Carlos se encargará de "Mejorar la interfaz de usuario".

### Fecha 3/05/2023

Se creó un nuevo proyecto en GitHub para organizar las tareas relacionadas con las modificaciones propuestas.

Se agregaron las tareas del backlog en el proyecto de GitHub, asignándolas a los responsables correspondientes.

Se crearon nuevas ramas en el repositorio para trabajar en paralelo en cada una de las tareas asignadas.

Se notificó a los miembros del equipo sobre los cambios realizados en GitHub y se les proporcionó acceso a las ramas correspondientes.

### Fecha 5/05/2023

Se completó la tarea "Optimizar rendimiento del sistema" y se fusionó la rama correspondiente con la rama principal del repositorio.

Se cerró el "issue" en GitHub relacionado con la tarea completada.

Se realizó una revisión del código y se realizaron ajustes menores para asegurar la calidad del código.

### Fecha 6/05/2023

Se completó la tarea "Implementar autenticación de usuarios" y se fusionó la rama correspondiente con la rama principal del repositorio.

Se cerró el "issue" en GitHub relacionado con la tarea completada.

Se realizaron pruebas exhaustivas para garantizar el correcto funcionamiento de la autenticación de usuarios.

### Fecha 8/05/2023

Se completó la tarea "Mejorar la interfaz de usuario" y se fusionó la rama correspondiente con la rama principal del repositorio.

Se cerró el "issue" en GitHub relacionado con la tarea completada.

Se realizó una revisión final de la interfaz de usuario para verificar que todas las mejoras hayan sido implementadas correctamente.

Se realizó una reunión de equipo para evaluar el progreso del proyecto y discutir futuras modificaciones.

Se agregaron nuevas tareas al backlog basadas en los comentarios y sugerencias recibidos durante la reunión.

Se asignaron responsables a las nuevas tareas y se actualizaron en GitHub.

**Fuentes de consulta:**

"Scrum: A Revolutionary Approach to Building Teams, Beating Deadlines, and Boosting Productivity" de Jeff Sutherland.

"The Scrum Field Guide: Practical Advice for Your First Year" de Mitch Lacey.

"Scrum Mastery: From Good to Great Servant-Leadership" de Geoff Watts.

"Essential Scrum: A Practical Guide to the Most Popular Agile Process" de Kenneth S. Rubin.

# Código Fuente:

El Código fuente es la denominación que tiene cualquier colección de código, con o sin comentarios, que está escrito usando el lenguaje de programación legible y entendible por los humanos, como texto sin formato.  Está creado como una manera de facilitar el trabajo a los programadores de ordenadores.

En otras palabras, podemos decir que el código fuente es la fuente de un programa de ordenador, el cual contiene declaraciones, instrucciones y funciones que funcionan que actúan como base para que el programa pueda funcionar.

El código fuente tiene que ser generado a través de un programa, en base a uno de los lenguajes de programación que existen.

Sin embargo, en un primer momento, este no es ejecutable. Por esta razón, debe ser traducido a otro lenguaje o a un código binario, para que el equipo informático pueda usar el hardware y ejecutar las instrucciones que se indican en él. La interpretación es realizada por compiladores, intérpretes, ensambladores u otros sistemas de traducción.

La compilación es una parte imprescindible que hay que realizar sobre un código fuente, para que este pueda tener algún uso práctico en el equipo informativo sobre el que se ejecute.

***Código fuente utilizado***: Markdown

# Reporte de las modificaciones realizadas

En esa parte de nuestro trabajo agregaremos cambios realizados a través de la creación del producto final. Comenzamos analizando las fases de los Sprint, comprendiendo nuestro proyecto y dividiendo nuestras tareas dependiendo de la función de cada integrante del equipo. Los Sprint sufrieron varios cambios dependiendo de las necesidades y la retroalimentación que nuestra asesora nos brindó, un ejemplo de esto son los cambios que realizamos para mejorar la experiencia de los usuarios en los dispositivos móviles e implementar la funcionalidad de búsqueda.

Posteriormente graficamos los Spring, en este parte del proceso tuvimos que hacer tres modificaciones hasta poder graficar de forma correcta y mostrar todos nuestros datos.

De la misma forma modificamos el calendario de la reuniones debido a que en ese momento necesitábamos más reuniones para arreglar ciertos detalles que se nos presentaron en el proceso

Otra modificación que realizamos fue crear un nuevo proyecto en GitHub para poder tener mejor organizada nuestra información recientemente corregida, agregamos todas las tareas del backlog y todos los documentos correspondientes. De la misma forma realizamos todas las mejoras que nuestras asesora nos dio en la retroalimentación atamos de mantener todo en orden.