

UNIVERSIDAD DEL VALLE DE GUATEMALA

Ingeniería de Software 1, sección 30



Excelencia que trasciende

DEL VALLE
GRUPO EDUCATIVO

SEGUNDO SPRINT

Sofia Garcia - 22210

Julio Garcia Salas - 22076

Joaquin Campos - 22155

Juan Fernando Menéndez - 17444

Juan Pablo Solis - 22102

GUATEMALA, abril de 2024

Índice

| | |
|-----------------------------------|-----------|
| Índice..... | 2 |
| Resumen..... | 2 |
| Introducción..... | 2 |
| Product backlog..... | 2 |
| Pila del sprint..... | 12 |
| Resultados del sprint..... | 15 |
| Métricas del sprint..... | 16 |
| Discusión..... | 16 |
| Código desarrollado..... | 17 |
| Documentación..... | 17 |

Resumen

Durante el Sprint 2 del proyecto de gestión de inventarios y ventas, así como del sitio de comercio electrónico para Divino Seas, logramos importantes avances tanto en el backend como en el frontend. Se implementaron nuevas características y mejoras, centradas en aumentar la eficiencia operativa, mejorar la experiencia del usuario y expandir la funcionalidad de ambas plataformas. Además, se desarrollaron mejoras en la interfaz de usuario, haciendo que la navegación y el proceso de compra sean más intuitivos y rápidos para los clientes finales.

Introducción

El objetivo de este sprint era fortalecer la infraestructura de nuestro proyecto para Divino Seas, centrándonos en la robustez del backend y la usabilidad del frontend. Trabajamos en la optimización de la base de datos y en la implementación de nuevas APIs para soportar operaciones de inventario más complejas. En el frontend, se rediseñaron componentes críticos para mejorar la experiencia de compra, garantizando que los usuarios puedan realizar pedidos de manera eficiente y sin inconvenientes. Estos avances están alineados con nuestro objetivo de ofrecer una solución integral que no solo maneje las operaciones de manera efectiva sino que también proporcione una experiencia de usuario excepcional en el sitio de ecommerce de Divino Seas.

Product backlog

| ID | Descripción | Imp | Est | Demo | Notas |
|----|--|------|-----|--|--|
| 1 | Configurar entorno de desarrollo local | Alta | 3 | Mostrar IDE configurado con todas las herramientas necesarias. | Incluir Git, editor de código, y entornos de servidor local. |

| | | | | | |
|----|--|-------|---|--|---|
| 2 | Instalar framework backend (Vite y Express JS) | Alta | 2 | Crear un proyecto básico y ejecutarlo localmente. | Seleccionar Vite y Express JS por su estructura MVC y facilidad de uso. |
| 3 | Instalar framework frontend (React) | Alta | 2 | Configurar React y mostrar una página de prueba. | React es elegido por su simplicidad y documentación amigable. |
| 4 | Configurar gestor de base de datos (PostgreSQL) | Alta | 3 | Conexión a base de datos desde el backend. | Demostrar conexión exitosa y creación de tablas iniciales. |
| 5 | Crear repositorio Git y establecer flujo de trabajo | Media | 2 | Repositorio creado y primer commit. | Mostrar el repositorio en GitHub. |
| 6 | Implementar página de inicio estática | Media | 5 | Página de inicio accesible localmente. | Incluir elementos SEO básicos como meta tags (para la página web). |
| 7 | Crear estructura de proyecto y organizar archivos | Alta | 2 | Estructura de carpetas mostrada en la demo. | Seguir las mejores prácticas de estructuración de proyectos. |
| 8 | Configurar Docker para el entorno de desarrollo backend | Alta | 3 | Mostrar el entorno de desarrollo Docker funcionando. | Incluir Dockerfile y docker-compose.yml para el backend y la base de datos. |
| 9 | Configurar conexión a la base de datos en un contenedor Docker | Alta | 2 | Mostrar una conexión exitosa a la base de datos en Docker. | Utilizar docker-compose para levantar el servicio de base de datos. |
| 10 | Crear endpoints básicos de API para productos | Alta | 5 | Demostrar CRUD de productos vía API. | Incluir endpoints para listar, crear, actualizar y eliminar productos. |

| | | | | | |
|----|--|-------|---|--|---|
| 11 | Documentar la API inicial | Media | 2 | Presentar documentación básica de la API. | Considerar herramientas compatibles con Docker para la documentación. |
| 12 | Investigación de palabras clave SEO | Alta | 4 | Lista de palabras clave objetivo. | Usar herramientas como SEMrush o Google Keyword Planner. |
| 13 | Crear la base de datos preliminar completa | Alta | 3 | Diagrama ER de la base de datos | Definir tablas y relaciones según el DER |
| 14 | Creación de productService | Alta | 3 | Mostrar productService.js implementando las operaciones CRUD. | Incluir conexión a la base de datos y métodos CRUD. |
| 15 | Creación de productRoute | Alta | 3 | Mostrar productRoute.js con rutas de API para productos. | Definir endpoints para operaciones CRUD y middleware. |
| 16 | Documentación de producto | Media | 3 | Documentar métodos y rutas de productos. | Detallar uso, parámetros, cuerpos de solicitud y respuestas. |
| 17 | Creación de categoryService | Alta | 3 | Mostrar categoryService.js con funciones CRUD para categorías. | Métodos para recuperar todas las categorías y gestionar una sola. |
| 18 | Creación de categoryRoute | Alta | 3 | Mostrar categoryRoute.js con rutas de API para categorías. | Incluir rutas CRUD para categorías. |
| 19 | Documentación de categorías | Media | 3 | Documentar métodos y rutas de categorías. | Explicar uso, parámetros y respuestas. |
| 20 | Creación de colorService | Alta | 3 | Demostrar colorService.js manejando colores. | Funciones CRUD para colores. |
| 21 | Creación de colorRoute | Alta | 3 | Exhibir colorRoute.js con rutas para gestionar colores. | Rutas de API para operaciones de color. |
| 22 | Documentación de | Media | 3 | Documentar métodos y | Detallar uso, |

| | | | | | |
|----|--|-------|---|--|--|
| | colores | | | rutas de colores. | parámetros, cuerpos de solicitud y respuestas. |
| 23 | Creación de collectionService | Alta | 3 | Presentar collectionService.js con lógica de negocio para colecciones. | Métodos CRUD para colecciones. |
| 24 | Creación de collectionRoute | Alta | 3 | Mostrar collectionRoute.js con rutas API para colecciones. | Rutas para interacción con información de colecciones. |
| 25 | Documentación de colecciones | Media | 3 | Documentar métodos y rutas de colecciones. | Instrucciones detalladas de uso y ejemplos de respuestas. |
| 26 | Creación de productColorService | Alta | 3 | Mostrar cómo productColorService.js relaciona productos con colores. | Asignación y eliminación de colores a productos. |
| 27 | Creación de rutas en productColorRoute | Alta | 3 | Enseñar productColorRoute.js gestionando asociaciones de colores de productos. | Rutas para crear y eliminar relaciones de productos y colores. |
| 28 | Documentación de productoColor | Media | 3 | Documentar métodos y rutas de asociación de colores de productos. | Proporcionar ejemplos de solicitudes y respuestas JSON. |
| 29 | Creación de clientService | Alta | 3 | Mostrar cómo clientService.js funciona en postman | Crud básico para cliente |
| 30 | Creación de clientRoute | Alta | 3 | Mostrar que si está funcionando en postman | Rutas para el crud de cliente |
| 31 | Documentación de cliente | Media | 3 | Documentar métodos y rutas | Proporcionar ejemplos de solicitudes y respuestas JSON |
| 32 | Creación de todos los archivos front end público | Alta | 3 | Crear los archivos de las secciones necesarias | Crear carpeta para guardar css |

| | | | | | |
|----|---|-------|---|---|--|
| 33 | Configuración de rutas | Alta | 3 | Configurar las rutas a todas las páginas creadas | |
| 34 | Conexión de rutas a la navbar | Baja | 1 | Hacer que desde la navbar se puedan acceder a las demás páginas | Asegurarse que los botones no se muestren como hipervínculos |
| 35 | Creación Componentes pagina Product | Media | 2 | Hacer los componentes necesario para la página de product | |
| 36 | Implementación componentes pagina Product | Media | 1 | Implementar los componentes creados en la pagina produc | |
| 37 | Edición pagina product con css | baja | 1 | Editar la página Product con css | |
| 38 | Creación Componentes página About us | Media | 2 | Hacer los componentes necesario para la página About us | |
| 39 | Implementación componentes página About Us | Media | 1 | Implementar los componentes creados en la página About Us | |
| 40 | Edición página About Us con css | baja | 1 | Editar la página About us con css | |
| 41 | Creación template básico Main Page (Empleados) para empleados | Alta | 1 | Crear el archivo Main Page para los empleados | |
| 42 | Creación Componentes Main Page (Empleados) | Media | 2 | Crear los componentes necesario para la Main page | |
| 43 | Implementación de componentes a la pagina | Alta | 2 | Implementar los componentes creados | |
| 44 | Configuración de Google Search Console | Alta | 2 | Sitio verificado en Google Search Console. | Configurar sitemap y verificar indexación. |
| 45 | Diseño de la arquitectura backend | Alta | 4 | Diagrama de arquitectura del backend. | Definir estructura de carpetas y módulos |

| | | | | | |
|----|--|------|---|--|--|
| | | | | | principales. |
| 46 | Configuración inicial de Vite para backend | Alta | 2 | Proyecto Vite corriendo con Express JS. | Mostrar servidor Express funcionando con Vite. |
| 47 | Creación de modelos de datos con Sequelize | Alta | 4 | Modelos para Producto, Categoría y Orden implementados. | Usar Sequelize como ORM para PostgreSQL. |
| 48 | Implementación de routing en React | Alta | 3 | Navegación entre páginas de inicio y gestión de inventarios. | Usar React Router para gestionar rutas. |
| 49 | Implementación de la UI en React | Alta | 8 | Componentes React para gestión de inventarios. | Convertir los mockups de Figma a componentes React. |
| 50 | Creación de la base de datos en PostgreSQL | Alta | 2 | Base de datos creada y configurada en PostgreSQL. | Ejecutar scripts SQL para crear la base de datos y tablas. |
| 51 | Diseñar sistema de navegación intuitivo | Alta | 4 | Sistema de navegación implementado. | Crear un menú lateral o superior para acceder fácilmente a todas las secciones de la app. |
| 52 | Implementar diseño responsivo | Alta | 6 | Demostración de la app en diferentes dispositivos. | Asegurar que la web app sea completamente usable en móviles, tablets y desktops. |
| 53 | Crear componentes UI reutilizables | Alta | 8 | Biblioteca de componentes UI. | Desarrollar botones, formularios, tarjetas, y otros elementos de UI en React para reutilización. |
| 54 | Desarrollar pantalla de gestión de inventarios | Alta | 8 | Pantalla de gestión de inventarios funcional. | Permitir a los usuarios ver, añadir, editar y eliminar inventarios |

| | | | | | |
|----|--|-------|---|--|---|
| | | | | | fácilmente. |
| 55 | Implementar tablas dinámicas para listado de productos | Alta | 6 | Tablas dinámicas en la sección de productos. | Usar una librería como React Table para mostrar productos con opciones de filtrado y paginación. |
| | | | | | |
| 56 | Diseñar e implementar dashboards de análisis | Alta | 8 | Dashboards de análisis para seguimiento de inventario. | Mostrar estadísticas clave, gráficos de ventas, y tendencias de inventario. |
| 57 | Integrar sistema de búsqueda avanzada | Alta | 6 | Funcionalidad de búsqueda avanzada. | Permitir a los usuarios buscar productos por diferentes criterios. |
| 58 | Desarrollar sistema de notificaciones | Media | 4 | Sistema de notificaciones en la app. | Notificar a los usuarios sobre cambios importantes, como bajos niveles de inventario. |
| 59 | Crear pantallas de carga y feedback al usuario | Media | 4 | Implementación de feedback visual. | Mostrar indicadores de carga y mensajes de éxito/error tras acciones del usuario. |
| 60 | Implementar modales para creación/edición de productos | Alta | 6 | Modales para añadir y editar productos. | Usar ventanas modales para formularios de productos, mejorando la interacción sin recargar la página. |
| 61 | Personalizar temas y estilos de la app | Media | 5 | Temas personalizados aplicados. | Permitir a los usuarios elegir entre temas claros y oscuros para mejorar la |

| | | | | | |
|----|--|-------|---|--|--|
| | | | | | experiencia visual. |
| 62 | Optimización de la interfaz para accesibilidad | Alta | 6 | Mejoras en accesibilidad verificadas. | Asegurar cumplimiento con estándares WCAG, incluyendo contraste de colores, navegación por teclado y etiquetas ARIA. |
| 63 | Pruebas de usabilidad con usuarios reales | Alta | 8 | Reporte de pruebas de usabilidad. | Recoger feedback de usuarios para iterar sobre el diseño y la funcionalidad de la UI. |
| 64 | Ajustes de UI basados en feedback de pruebas | Media | 6 | Implementación de ajustes en la UI. | Realizar mejoras en la interfaz basadas en los resultados de las pruebas de usabilidad. |
| 65 | Desarrollo de ayuda contextual y tooltips | Baja | 4 | Ayuda y tooltips implementados. | Proporcionar información útil y consejos sobre cómo usar diferentes partes de la app. |
| 66 | Implementación de endpoints para CRUD de productos | Alta | 6 | API endpoints para productos funcionando. | Crear, leer, actualizar, eliminar productos a través de la API. |
| 67 | Pruebas unitarias para modelos de datos | Alta | 4 | Pruebas unitarias para modelos pasando. | Usar Jest o Mocha para pruebas de modelos Sequelize. |
| 68 | Pruebas unitarias para endpoints de productos | Alta | 4 | Pruebas unitarias para API de productos pasando. | Asegurar que los endpoints retornan los datos esperados. |
| 69 | Configuración de seguridad básica en | Alta | 3 | Middleware de seguridad implementado. | Usar helmet y CORS en el |

| | | | | | |
|----|--|-------|---|--|---|
| | Express | | | | servidor Express. |
| 70 | Implementación de autenticación JWT | Alta | 4 | Sistema de autenticación con JWT funcionando. | Permitir login y proteger rutas de la API con JWT. |
| 71 | Documentación de la API con Swagger | Media | 3 | Documentación de la API disponible. | Usar Swagger para documentar los endpoints de la API. |
| 72 | Configuración inicial de SSL con Let's Encrypt | Media | 3 | HTTPS configurado para el sitio web. | Demostrar el sitio funcionando sobre HTTPS. |
| 73 | Optimización inicial de imágenes para la web | Media | 3 | Imágenes optimizadas en la página de inicio. | Usar herramientas como ImageOptim o servicios en línea. |
| 74 | Implementación de sistema de logging | Media | 3 | Logs de aplicación y errores configurados. | Usar Winston o Morgan para logging en el backend. |
| 75 | Preparación de entorno de pruebas para frontend | Media | 3 | Configuración de Jest para pruebas de React. | Mostrar pruebas básicas de componentes React. |
| 76 | Preparación y ejecución de pruebas de carga | Baja | 4 | Pruebas de carga ejecutadas en la API. | Usar Artillery o JMeter para simular tráfico. |
| 77 | Revisión de accesibilidad web | Baja | 3 | Auditoría de accesibilidad completada. | Usar herramientas como Lighthouse para evaluar accesibilidad. |
| 78 | Configuración de CI/CD con GitHub Actions | Media | 4 | Pipeline de CI/CD funcionando para pruebas y despliegue. | Automatizar pruebas y despliegue con GitHub Actions. |
| 79 | Creación de contenido SEO para la página de inicio | Alta | 4 | Contenido optimizado para SEO implementado. | Escribir y optimizar el contenido de la página de inicio. |
| 80 | Análisis y mejora de la velocidad de | Alta | 4 | Mejoras en la velocidad de carga demostradas. | Implementar técnicas de |

| | | | | | |
|----|--|-------|---|--|---|
| | carga de la página | | | | optimización como lazy loading. |
| 81 | Integración de pasarela de pago | Alta | 8 | Demostrar transacciones de prueba. | Integrar Stripe o PayPal para manejar pagos. |
| 82 | Desarrollo de funcionalidad de carrito de compras | Alta | 6 | Carrito de compras funcional en la web app. | Permitir a los usuarios agregar y modificar productos en el carrito. |
| 83 | Implementación de filtros de búsqueda para productos | Alta | 5 | Filtros de búsqueda funcionando. | Permitir a los usuarios filtrar productos por categoría, precio, etc. |
| 84 | Optimización de la base de datos | Alta | 4 | Mejoras en el rendimiento de consultas demostradas. | Indexar columnas clave y revisar consultas para optimización. |
| 85 | Implementación de respuestas caché en la API | Media | 3 | Caché de respuestas implementado para endpoints clave. | Usar Redis o memoria caché similar para mejorar la velocidad de respuesta. |
| 86 | Desarrollo de la sección de administración de inventarios | Alta | 8 | Sección de administración completa y funcional. | Permitir a los administradores gestionar inventarios directamente desde la web app. |
| 87 | Pruebas de seguridad de la aplicación | Alta | 6 | Reporte de pruebas de seguridad. | Realizar pruebas de penetración y corregir vulnerabilidades encontradas. |
| 88 | Configuración de backups automáticos para la base de datos | Alta | 3 | Sistema de backups configurado y probado. | Asegurar que los backups de la base de datos se realicen diariamente. |
| 89 | Preparativos finales | Alta | 5 | Checklist de lanzamiento | Revisar SEO, |

| | | | | | |
|----|--|-------|----------|--|---|
| | para el lanzamiento | | | completado. | seguridad, rendimiento y funcionalidades clave antes del lanzamiento. |
| 90 | Realización de pruebas A/B para la página de inicio | Media | 4 | Resultados de pruebas A/B implementados. | Usar herramientas como Google Optimize para realizar pruebas A/B. |
| 91 | Implementación de funcionalidades de reportes y análisis | Alta | 6 | Sistema de reportes implementado. | Desarrollar dashboards para seguimiento de ventas, inventario y análisis de usuarios. |
| 92 | Optimización final de rendimiento del frontend | Alta | 5 | Mejoras en el rendimiento del frontend demostradas. | Aplicar técnicas como code splitting, lazy loading y optimización de imágenes. |
| 93 | Revisión final de accesibilidad web | Media | 3 | Auditoría de accesibilidad completada y mejoras aplicadas. | Asegurar cumplimiento con WCAG y mejorar la accesibilidad general. |
| 94 | Preparación y ejecución de plan de lanzamiento | Alta | 4 | Plan de lanzamiento ejecutado. | Coordinar comunicados de prensa, publicaciones en redes sociales y campañas de email marketing. |
| 95 | Monitoreo post-lanzamiento y recopilación de feedback | Alta | On going | Informe de feedback inicial recopilado. | Establecer herramientas de monitoreo en tiempo real y canales de feedback. |
| 96 | Ajustes post-lanzamiento basados en | Media | On going | Ajustes implementados según feedback. | Priorizar y aplicar mejoras y correcciones |

| | | | | | |
|----|---|-------|---|--------------------------------------|--|
| | feedback | | | | rápidas post-lanzamiento. |
| 97 | Desarrollo de documentación para usuarios finales | Baja | 4 | Documentación de usuario disponible. | Crear guías de usuario, FAQs y documentación de ayuda. |
| 98 | Capacitación del equipo de soporte | Media | 4 | Sesión de capacitación completada. | Preparar al equipo de soporte con conocimientos sobre la aplicación y solución de problemas comunes. |

Pila del sprint

Historia de Usuario: Configuración del Servicio de Productos

Puntos de Historia: 8 horas

Tareas:

- Tarea ID 14: Creación de productService
 - Responsable: Julio García Salas
 - Fecha de Terminación: 18 de abril
- Tarea ID 15: Creación de productRoute
 - Responsable: Sofía García
 - Fecha de Terminación: 17 de abril
- Tarea ID 16: Documentación de producto
 - Responsable: Julio García Salas
 - Fecha de Terminación: 18 de abril

Historia de Usuario: Configuración del Servicio de Categorías

Puntos de Historia: 6 horas

Tareas:

- Tarea ID 17: Creación de categoryService
 - Responsable: Sofía García
 - Fecha de Terminación: 17 de abril
- Tarea ID 18: Creación de categoryRoute
 - Responsable: Julio García Salas
 - Fecha de Terminación: 18 de abril
- Tarea ID 19: Documentación de categorías

- Responsable: Sofía García
- Fecha de Terminación: 17 de abril

Historia de Usuario: Configuración del Servicio de Colores

Puntos de Historia: 6 horas

Tareas:

- Tarea ID 20: Creación de colorService
 - Responsable: Julio García Salas
 - Fecha de Terminación: 18 de abril
- Tarea ID 21: Creación de colorRoute
 - Responsable: Sofía García
 - Fecha de Terminación: 17 de abril
- Tarea ID 22: Documentación de colores
 - Responsable: Julio García Salas
 - Fecha de Terminación: 18 de abril

Historia de Usuario: Configuración del Servicio de Colecciones

Puntos de Historia: 6 horas

Tareas:

- Tarea ID 23: Creación de collectionService
 - Responsable: Sofía García
 - Fecha de Terminación: 17 de abril
- Tarea ID 24: Creación de collectionRoute
 - Responsable: Julio García Salas
 - Fecha de Terminación: 18 de abril
- Tarea ID 25: Documentación de colecciones
 - Responsable: Sofía García
 - Fecha de Terminación: 17 de abril

Historia de Usuario: Configuración del Servicio de Colores de Productos

Puntos de Historia: 4 horas

Tareas:

- Tarea ID 26: Creación de productColorService
 - Responsable: Julio García Salas
 - Fecha de Terminación: 18 de abril
- Tarea ID 27: Creación de rutas en productColorRoute
 - Responsable: Sofía García
 - Fecha de Terminación: 17 de abril

- Tarea ID 28: Documentación de productoColor
 - Responsable: Julio García Salas
 - Fecha de Terminación: 18 de abril

Historia de Usuario: Configuración del Servicio de Clientes

Puntos de Historia: 4 horas

Tareas:

- Tarea ID 29: Creación de clientService
 - Responsable: Sofía García
 - Fecha de Terminación: 17 de abril
- Tarea ID 30: Creación de clientRoute
 - Responsable: Julio García Salas
 - Fecha de Terminación: 18 de abril
- Tarea ID 31: Documentación de cliente
 - Responsable: Sofía García
 - Fecha de Terminación: 17 de abril

Historia de Usuario: Configuración de rutas y template básico página pública

Puntos de Historia: 5 horas

Tareas:

- Tarea ID 32: Creación de archivos front end
 - Responsable: Juan Pablo Solis
 - Fecha de Terminación: 17 de abril
- Tarea ID 33: Configuración de rutas
 - Responsable: Juan Pablo Solis
 - Fecha de Terminación: 18 de abril
- Tarea ID 34: Conexión de rutas a la navBar
 - Responsable: Juan Pablo Solis
 - Fecha de Terminación: 18 de abril

Historia de Usuario: Creación página producto

Puntos de Historia: 4 horas

Tareas:

- Tarea ID 35: Creación de componentes necesarios
 - Responsable: Juan Fernando Menenez
 - Fecha de Terminación: 17 de abril
- Tarea ID 36: Implementación de los componentes en la pagina
 - Responsable: Juan Fernando Menendez
 - Fecha de Terminación: 18 de abril

- Tarea ID 37: Edición de la página con css
 - Responsable: Juan Fernando Menendez
 - Fecha de Terminación: 18 de abril

Historia de Usuario: Creación página About us

Puntos de Historia: 4 horas

Tareas:

- Tarea ID 38: Creación de componentes necesarios
 - Responsable: Juan Pablo Solis
 - Fecha de Terminación: 17 de abril
- Tarea ID 39: Implementación de los componentes en la pagina
 - Responsable: Juan Pablo Solis
 - Fecha de Terminación: 18 de abril
- Tarea ID 40: Edición de la página con css
 - Responsable: Juan Pablo Solis
 - Fecha de Terminación: 18 de abril

Historia de Usuario: Creación Main Page admin

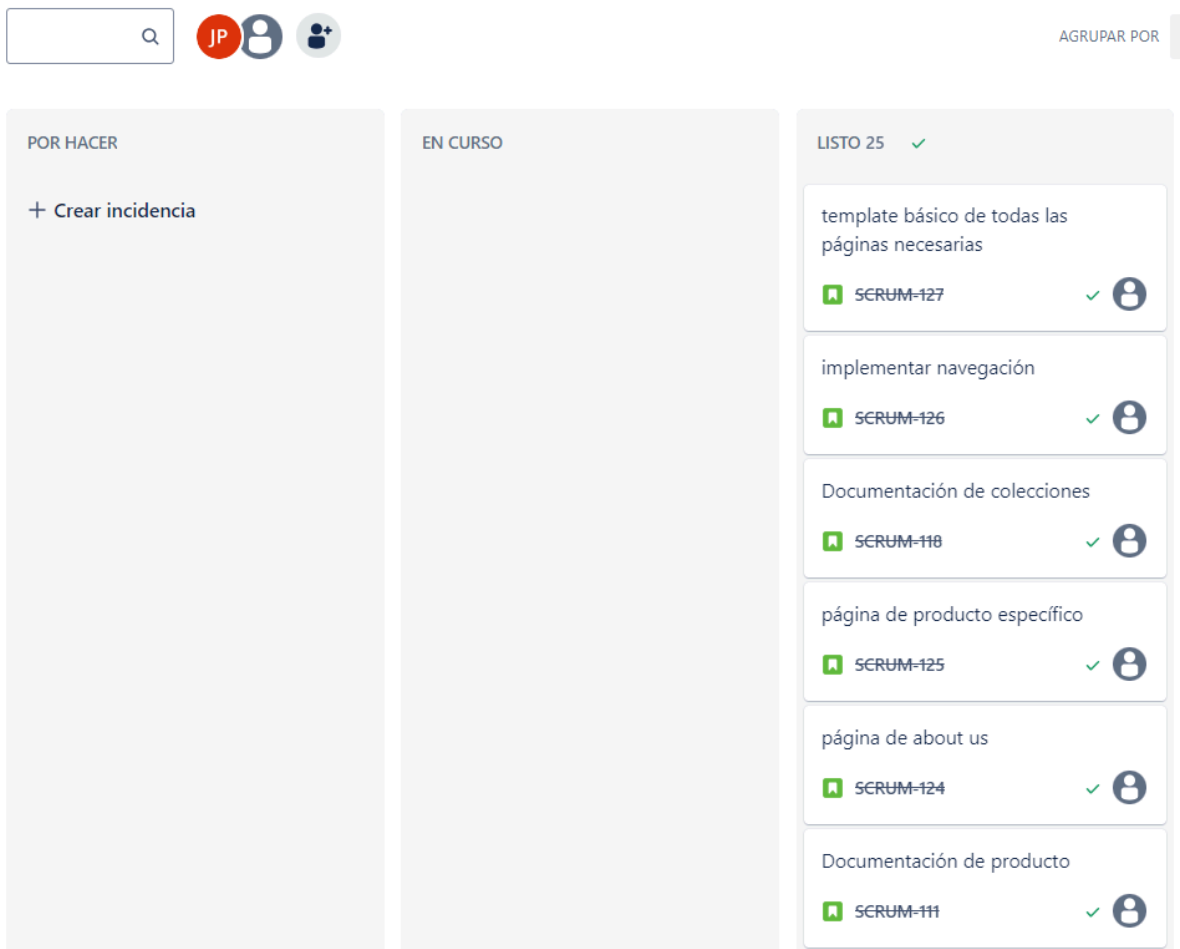
Puntos de Historia: 4 horas

Tareas:

- Tarea ID 41: Creación Template Básico Main Page
 - Responsable: Jose Joaquin Campos
 - Fecha de Terminación: 17 de abril
- Tarea ID 42: Creación de componentes necesarios
 - Responsable: Jose Joaquin Campos
 - Fecha de Terminación: 18 de abril
- Tarea ID 43: Implementación de componentes a la pagina
 - Responsable: Jose Joaquin Campos
 - Fecha de Terminación: 18 de abril

Resultados del sprint

Tablero Sprint 2

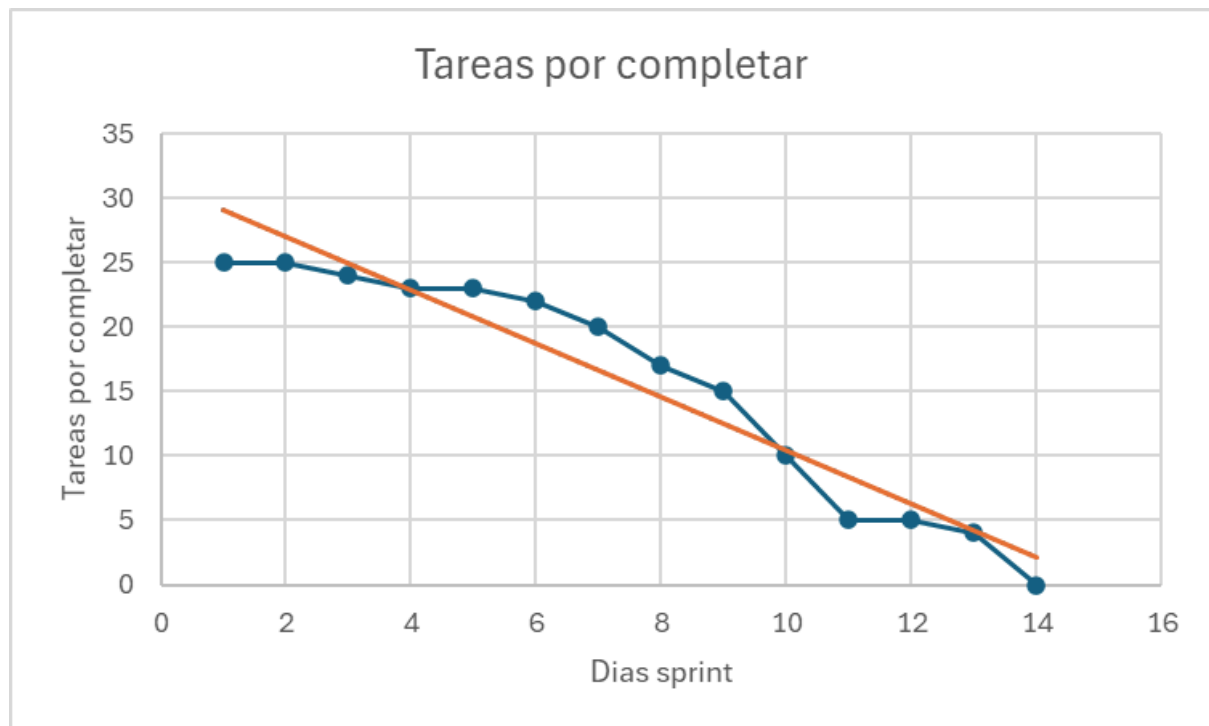


Tablero de jira:

<https://uvg-team-g8inmhj5.atlassian.net/jira/software/projects/SCRUM/boards/1?atlOrigin=eyJpIjoiZTgxMGJhMzkwNTNkNDI4MDkxNzk0N2Y2NWExN2E5YWliLCJwIjoiYj9>

Se lograron hacer todas las tareas planificadas para este sprint, y en cambio con el último sprint ya se logró colocar un mejor margen entre tareas asignadas y tiempo.

Métricas del sprint



Velocidad del Sprint

- Se concluyó el sprint con un puntaje de 72 puntos completando las 25 tareas asignadas previamente para el sprint.
- En cuanto a velocidad media del Scrum total podemos decir que con la velocidad del sprint pasado, con el actual podemos concluir que se tuvo una velocidad de 53.5 puntos promedios de manera que hubo un gran aumento en velocidad en este segundo sprint y estamos por arriba del promedio investigado.

Discusión

Cumplimiento de Objetivos

- Positivo: El equipo completó las 25 tareas asignadas, lo cual indica que se cumplieron los objetivos del sprint en términos de volumen de trabajo.
- Para mejorar: La distribución de la carga de trabajo a lo largo del sprint fue desigual, con un esfuerzo considerable concentrado hacia el final. Esto indica una planificación o estimación inicial no óptima.

Gestión del Tiempo y Distribución del Trabajo

- Positivo: La capacidad del equipo para completar todas las tareas, incluso con un impulso final, demuestra flexibilidad y compromiso.
- Para mejorar: Una distribución más uniforme del trabajo podría ayudar a evitar el estrés y el riesgo asociados a la acumulación de tareas hacia el final del sprint.

Calidad del Trabajo

- A considerar: Aunque el gráfico muestra que se completaron todas las tareas, la calidad de las tareas completadas también es importante, lo que se debe mejorar para próximos sprints, por lo tanto, es importante detallar las tareas realizadas para tener la mejor calidad posible.

Comunicación y Colaboración en el Equipo

- Positivo: La finalización exitosa de las tareas sugiere una buena colaboración dentro del equipo, especialmente en la fase de alta presión hacia el final.
- Para mejorar: Sería útil analizar si la comunicación efectiva dentro del equipo y con los stakeholders podría mejorarse para identificar cuellos de botella o redistribuir tareas antes en el sprint.

Sostenibilidad del Esfuerzo

- Preocupación: La concentración del trabajo hacia el final del sprint plantea preguntas sobre la sostenibilidad de este enfoque. Es crucial considerar el bienestar del equipo y si tal patrón de trabajo es viable a largo plazo.

Aprendizajes y Mejoras para Sprints Futuros

La reflexión sobre este sprint debe llevar a ajustes en la planificación y ejecución de sprints futuros. Considerar métodos para una mejor estimación de tareas, técnicas de priorización más efectivas, y estrategias para fomentar una distribución más equitativa del trabajo a lo largo del sprint.

Código desarrollado

https://github.com/JPS4321/React_Proyecto

Lista de funcionalidades terminadas:

- API para producto
- API para categoría
- API para color
- API para productColor
- API para cliente
- API para colecciones
- Base de datos completa preliminar hecha
- Product Page
- Versiones Alternativas Hero Section
- Pagina "About Us"
- Admin Main Page
- Componente Navbar para Admin
- Componente Statbar para Admin
- Componente Banner

Documentación

- Retrospectiva del sprint:

- El segundo sprint del proyecto para Divino Seas mostró un cumplimiento total de las tareas planificadas, pero con una distribución desigual del trabajo a lo largo del periodo, lo que indica la necesidad de mejorar la planificación inicial para evitar acumulación de tareas hacia el final del sprint y garantizar la sostenibilidad del equipo a largo plazo.

Aunque se completaron todas las tareas, es esencial mantener un enfoque en la calidad del trabajo realizado. La colaboración dentro del equipo fue positiva, pero pensamos que una comunicación más efectiva puede ayudar a identificar y abordar cuellos de botella de manera eficiente. En conclusión, este sprint nos dejó lecciones importantes para ajustar la planificación y ejecución de futuros avances, priorizando una distribución equitativa del trabajo y la calidad del producto final.

Link al documento:

https://docs.google.com/document/d/1SReFis7Z8rOUs-hoo3IAeLpdkSSI_8BsQV_qxV8_KaE/edit