# Estado Actual del Proyecto

**Resumen General:**

El proyecto se encuentra en un 30%, en este punto tenemos la base de datos funcionando y ya se pueden guardar datos de registro de usuarios, además, el carrito funciona para agregar y eliminar prendas. La página cuenta con un poco de lógica con respecto a si no hay una prenda en stock o si el carrito se encuentra vacío.

**Funcionalidades Completadas:**

* Carrito: Agregar y eliminar elementos del carrito
* Registro: El usuario se puede registrar
* Login: Con esos mismos datos el usuario puede entrar a la página
* Filtrado: el usuario puede filtrar por el tipo de ropa que necesite
* Catálogo: La página cuenta con un catálogo de prendas disponibles en el momento en la página web

**Funcionalidades en Progreso:**

* Eliminar y Modificar: El usuario puede eliminar sus datos de la página o modificarlos
* Estilo: Estamos trabajando en el estilo real que debería tener la plataforma
* Carrito: Con el carrito lleno poder ir a comprar y realizar un pedido, resumen de costos, agregar prendas y poder agregar cupones de descuento

**Principales Dificultades o Bloqueos:**

* Problema 1: El primer problema que hemos encontrado es que no podemos compartir la demo de página web porque tiene una base de datos funcional, github no permite subir archivos sql
* Problema 2: Tenemos la página completa en WordPress y es un problema a la hora de pasar todo eso a código

# Comparación con la Planificación Inicial

Revisión del Plan: Vuelve a revisar tu cronograma original (Diagrama de Gantt, lista de hitos, backlog inicial, etc.).

Análisis de Desviaciones: Compara el estado actual (Sección 1) con lo que debería estar completado según la planificación original para esta fecha.

¿Están adelantados, atrasados o según lo previsto?

Identifica las principales causas de las desviaciones (si las hay). ¿Fueron subestimaciones, problemas técnicos inesperados, cambios en los requisitos, etc.?

Lecciones Aprendidas (Opcional pero recomendado - puntos extra 👆): ¿Qué han aprendido sobre su proceso de estimación y planificación?

# Alcance Definitivo para la Entrega Final

Funcionalidades Comprometidas: Basándose en el estado actual, la velocidad de desarrollo observada y el tiempo restante (5 semanas), enumere de forma clara y específica las funcionalidades exactas que se comprometen a entregar y demostrar en la fecha final (13 de mayo de 2025).

Sean realistas. Es preferible comprometerse a menos funcionalidades bien terminadas que a muchas a medio hacer.

Describe brevemente qué hará cada funcionalidad comprometida.

Funcionalidades Descartadas (si aplica): Indica qué funcionalidades planificadas originalmente no se incluirán en la entrega final, si es el caso. Justifica brevemente por qué se descartan (falta de tiempo, complejidad, menor prioridad).

# Stack Tecnológico Detallado

Describe de la forma más completa posible las tecnologías, lenguajes, frameworks, librerías y herramientas que componen su proyecto. El objetivo es tener suficiente detalle para poder recomendar estrategias y herramientas de testing adecuadas. Incluye:

Frontend:

Lenguaje(s): (Ej: JavaScript, TypeScript, HTML5, CSS3)

Framework/Librería(s): (Ej: React v18, Angular v17, Vue.js v3, Svelte, Next.js, Nuxt.js, etc. o "Vanilla JS")

Librerías UI/Componentes: (Ej: Material UI, Bootstrap, Tailwind CSS, Ant Design)

Backend:

Lenguaje(s): (Ej: Java 17, Python 3.11, Node.js v20, PHP 8.2, C# .NET 7, Go 1.21)

Framework(s): (Ej: Spring Boot 3, Django 5, Flask, Express.js, NestJS, Laravel 10, ASP.NET Core, Gin)

Arquitectura (si relevante): (Ej: Monolito, Microservicios, Serverless)

Base de Datos:

Tipo: (Ej: Relacional, NoSQL, Documental, Clave-Valor)

Sistema Gestor (SGBD): (Ej: PostgreSQL 15, MySQL 8, MongoDB Atlas, Oracle DB, SQL Server, Redis, Firebase Firestore)

ORM/ODM (si aplica): (Ej: Hibernate, SQLAlchemy, TypeORM, Mongoose, Entity Framework Core)

APIs y Servicios Externos:

¿Consumen APIs de terceros? (Ej: API de Google Maps, Stripe, OpenWeatherMap). ¿Cuáles?

¿Exponen APIs propias? ¿Usan REST, GraphQL, gRPC?

Autenticación/Autorización: (Ej: JWT, OAuth2, Auth0, Firebase Auth, Spring Security)

Infraestructura y Despliegue (si ya lo han considerado/implementado):

Proveedor Cloud: (Ej: AWS, Google Cloud, Azure, Heroku, Vercel)

Contenedores: (Ej: Docker, Docker Compose)

Orquestación: (Ej: Kubernetes - K8s)

Servidor Web/Aplicaciones: (Ej: Nginx, Apache, Tomcat, Kestrel)

CI/CD: (Ej: GitHub Actions, GitLab CI, Jenkins)

Otras Herramientas Relevantes: (Ej: Gestor de paquetes como npm/yarn/pip/maven, herramientas de build como Webpack/Vite/Gradle, sistemas de mensajería como RabbitMQ/Kafka)