Universidad del Valle de Guatemala

Facultad de Ingeniería

Departamento de Ciencia de la Computación

Ingeniería de Software I

DESIGN BETTER

PLATAFORMA DE DISEÑO Y PERSONALIZACIÓN DE MODA

Pablo José Méndez Alvarado – 23975

Luis Fernando Palacios López – 239333

Roberto Samuel Nájera Marroquín – 23781

André Emilio Pivaral López – 23574

Catedrático: Erick Francisco Marroquín Rodríguez

Sección: 20

Nueva Guatemala de la Asunción, 28 de mayo de 2025

Índice

Índice ... 1
ENLACES DE REFERENCIA ... 4
Repositorios con el código desarrollado: ... 4
Enlace con carpeta de todos los diagramas en svg ... 4
CORTE 1: Design Thinking ... 4
Descripción de la Persona ... 4
Descripción de la Idea ... 4
Objetivo General ... 5
Objetivos Específicos ... 5
Declaración Guía ... 5
Metas ... 6
Tono de Voz ... 6
Objetivos Creativos ... 6
Objetivos Comerciales ... 7
Análisis de Mercado y Consumidores ... 7
Usuarios ... 8
¿Qué? ¿Cómo? ¿Por qué? ... 10
AEIOU ... 11
Perfiles ... 11
CORTE 1: Resumen de Entrevistas ... 12
CORTE 1: Design Thinking Posterior a las entrevistas: ... 20
Mapas Empatía ... 21
Insights ... 22
¿Cómo Podríamos? ... 25
CORTE 1: Definición Proyecto ... 26
CORTE 2: Aplicación de Design Studio y Pruebas de Prototipos ... 26
Ideas Generales ... 27
Oportunidad #5 ... 29
Ideas Votadas ... 29
Prototipos ... 37
CORTE 2: HISTORIAS DE USUARIO Y USER STORY MAPPING ... 39
Historias Usuario ... 39

Actores 41
Diagrama de Caso de Uso 44
Mapa Historias de Usuario 45
CORTE 3: Análsis y Diseño del Proyecto 45
Clases Preliminares 45
Casos de Uso y Clases Involucradas 47
Diagrama de Paquetes 52
Relaciones entre Paquetes 54
Persistencia de Datos 54
Estimaciones Requisitos No Funcionales 55
CORTE 3: Selección de Stack de trabajo 60
Selección de Tecnología Frontend 60
Selección de Tecnología Backend 63
Selección de Tecnología Persistencia de Datos 67
ACTIVIDAD: Lean UX Canvas 71
Capturas del proceso 71
Lienzo de estrategia 77
Enlace a Mural 77
SPRINT 1 78
Software Funcionando 78
Tareas Concluidas 78
Tareas planificadas, pero no concluidas 79
Gráfico de Sprint Burndown de puntos de historia 79
Gráfico de Sprint Burndown de Recuento de Incidencias 79
Reporte de Velocidad 80
Éxito del Sprint 80
Evidencia de Pruebas con Usuarios finales 81
SPRINT 2 82
Software Funcionando 82
Tareas Concluidas 83
Gráfico de Sprint Burndown de puntos de historia 83
Gráfico de Sprint Burndown de Recuento de Incidencias 84
Reporte de Velocidad 84

Éxito del Sprint 84
Evidencia de Pruebas con Usuarios finales 85
SPRINT 3 87
Software Funcionando 87
Tareas Concluidas 88
Gráfico de Sprint Burndown de puntos de historia 88
Gráfico de Sprint Burndown de Recuento de Incidencias 89
Reporte de Velocidad 89
Éxito del Sprint 89
Evidencia de Pruebas con Usuarios finales 90
SPRINT 4 92
Software Funcionando 92
Tareas Concluidas 92
Gráfico de Sprint Burndown de puntos de historia 93
Gráfico de Sprint Burndown de Recuento de Incidencias 93
Reporte de Velocidad 93
Evidencia de Pruebas con Usuarios finales 95
AVANCES TOTALES 96
Product Backlog 96
INFORME: Porcentaje de Avance del Desarrollo del Proyecto 100

ENLACES DE REFERENCIA

Repositorios con el código desarrollado:
https://github.com/Ultimate-Truth-Seeker/DesignBetterFrontend

https://github.com/Ultimate-Truth-Seeker/DesignBetterBackend

Enlace con carpeta de todos los diagramas en svg
https://github.com/Ultimate-Truth-Seeker/ProyectoIS/tree/main/SVG

Enlace de Jira: https://design-better.atlassian.net/jira/software/projects/SCRUMSITO/boards/1?atlOrigin=eyJpIjoiNjY0OGUzYjk5NzU0NGQ5YmFhMzk5NTZiYjZhMDVkMTAiLCJwIjoiaiJ9

Carpeta de grabaciones: Evidencias Scrum

Documentación Original del Proyecto, presentaciones, historial de versiones y LOGT de cada fase

https://github.com/Ultimate-Truth-Seeker/ProyectoIS

CORTE 1: Design Thinking

Descripción de la Persona

El proyecto está dirigido a una diseñadora de moda independiente que enfrenta desafíos al crear prendas personalizadas para distintos tipos de cuerpo. Su trabajo requiere precisión en la toma de medidas, visualización realista de los diseños y una comunicación fluida con los clientes para garantizar su satisfacción.

Dado que trabaja de manera autónoma o con un equipo reducido, la diseñadora necesita herramientas que le permitan agilizar su proceso de diseño, optimizar costos en prototipos y brindar experiencias de personalización más atractivas y eficientes para sus clientes.

Descripción de la Idea

La idea central del proyecto es desarrollar una solución tecnológica que facilite el diseño y personalización de prendas mediante herramientas de modelado y simulación digital. Actualmente, la diseñadora enfrenta dificultades para visualizar sus creaciones en diferentes tipos de cuerpo, lo que puede generar incertidumbre en los clientes y múltiples iteraciones en el diseño.

La ausencia de herramientas accesibles y realistas limita la capacidad de personalización y la eficiencia en la toma de decisiones. Implementar un sistema que permita visualizar de manera

precisa las prendas antes de su confección ayudará a optimizar tiempos, reducir costos y mejorar la experiencia tanto para la diseñadora como para sus clientes.

Objetivo General

Desarrollar una solución tecnológica que permita a la diseñadora de moda modelar, simular y personalizar prendas de manera eficiente, optimizando su proceso de diseño y mejorando la experiencia del cliente.

Objetivos Específicos

• Facilitar los procesos de toma de medidas y personalización de prendas para diferentes tipos de cuerpo.

• Implementar visualizaciones realistas en 3D para ayudar a los clientes a previsualizar las prendas antes de la confección.

• Agilizar el proceso de iteración en los diseños, reduciendo la cantidad de muestras físicas necesarias.

• Mejorar la comunicación entre la diseñadora y sus clientes, a través de herramientas de software interactivas.

• Reducir costos y tiempos en la creación de prototipos, optimizando el flujo de trabajo de la diseñadora.

Brief Problemática

Declaración Guía

Una diseñadora de vestuario se enfrenta a múltiples retos al buscar crear prendas a medida que celebren la individualidad de cada cliente y optimicen el proceso de diseño para diferentes tipos de cuerpo.

Necesita una forma de modelar, simular y personalizar de manera eficiente, alineándose con las preferencias de estilo de los clientes sin sacrificar la calidad o la creatividad.

Metas

• Variedad de formas corporales: Cada cliente tiene medidas únicas, lo que dificulta la creación de prototipos precisos y exige un proceso de personalización más ágil.

• Proceso de diseño complejo: Ajustar patrones, visualizar cambios y comunicarlos al cliente o al equipo con rapidez puede volverse engorroso y costoso.

• Necesidad de asesoría en estilo: Más allá del ajuste, los clientes buscan prendas que reflejen su personalidad. A menudo, la diseñadora carece de herramientas o referencias ágiles para proponer variantes de color, silueta o detalles de diseño que se adapten a sus gustos.

• Visualización realista: Resulta complicado mostrar de manera clara y atractiva cómo quedarán las prendas antes de la confección. La ausencia de simulaciones realistas puede generar dudas en los clientes y ralentizar la aprobación.

• Colaboración con el cliente: La diseñadora requiere medios efectivos para compartir avances y recopilar retroalimentación oportuna, evitando múltiples iteraciones y retrasos en la toma de decisiones.

Tono de Voz

El tono de la solución a proponer y la voz de la marca deben reflejar un espíritu amigable, solidario y creativo, que combine la experiencia de la moda de alta gama con la calidez y la accesibilidad de un estilista personal.

Objetivos Creativos

• Facilitar la personalización rápida: La diseñadora debe poder ingresar o importar medidas con facilidad. Requiere ver cambios en el patrón o la prenda de forma casi inmediata para iterar sin fricción.

• Acceder a sugerencias de estilo relevantes: Se busca orientación sobre posibles ajustes de diseño que se alineen con las preferencias y la forma corporal de cada cliente. La diseñadora necesita referencias o sistemas de recomendación que la ayuden a explorar diferentes opciones de manera más informada.

• Visualización realista: Hay un reto para implementar vistas de drapeado y renderizados de telas que sean lo suficientemente realistas para mostrar detalles propios de la prenda. Esto permitiría reforzar la confianza del cliente antes de producir una muestra física.

• Optimizar la colaboración: Es fundamental compartir avances de diseño con el cliente de manera segura y ágil, recibiendo comentarios que agilicen la aprobación y reduzcan iteraciones extensas.

Objetivos Comerciales

• Reducir costes y tiempos de prototipo: La diseñadora independiente busca minimizar la inversión en muestras físicas y acelerar la entrega, demostrando un retorno claro en su flujo de trabajo.

• Mejorar la satisfacción y fidelización del cliente: Garantizar que los clientes finales se sientan seguros al ver una representación fiel de la prenda antes de la confección. Contribuir a una experiencia positiva que impulse la repetición de servicios y el boca a boca.

• Diferenciación en un mercado competitivo: Es importante ofrecer un valor único y distintivo frente a herramientas o métodos tradicionales de diseño. La posibilidad de personalizaciones avanzadas y una experiencia cercana puede marcar la diferencia ante otros profesionales que no cuenten con estos recursos.

Análisis de Mercado y Consumidores

• Aumento de la demanda de moda personalizada: Los clientes valoran la unicidad y el ajuste perfecto, lo que incrementa la necesidad de procesos de diseño más flexibles y centrados en la persona.

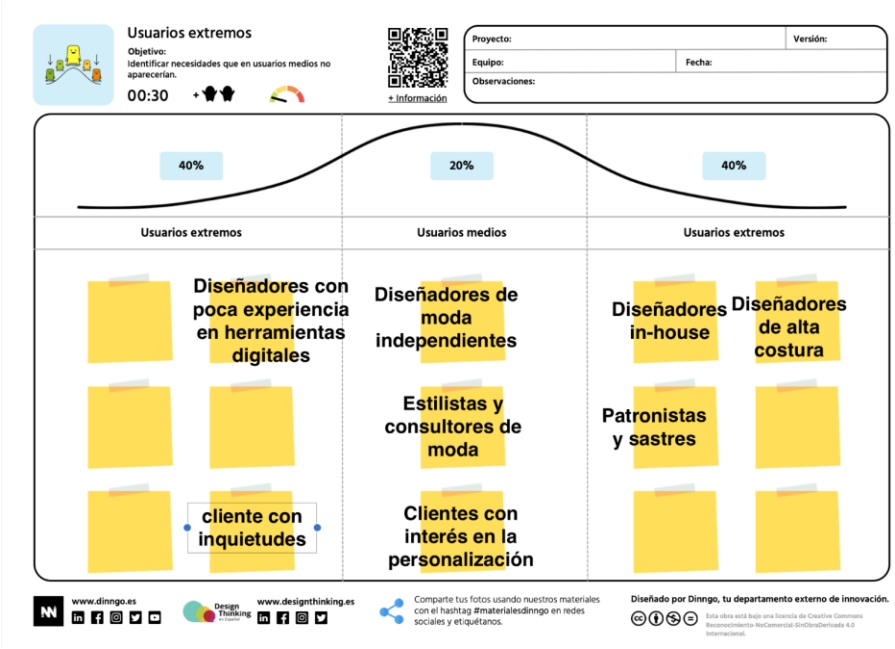
• Tendencia creciente del modelado 3D: En el sector moda, se prioriza cada vez más la digitalización para acelerar ciclos de desarrollo y reducir costes de muestras.

• Limitaciones de los diseñadores locales: Muchos profesionales independientes no tienen acceso a soluciones de modelado avanzadas o las encuentran demasiado complejas, costosas o dispersas.

• Expectativas de visualización del cliente: Los consumidores quieren ver cómo lucirá la prenda, especialmente en compras o procesos remotos, por lo que las simulaciones realistas son cada vez más relevantes para facilitar la toma de decisiones.

Usuarios

A continuación, les describimos los tipos de usuarios finales identificados, incluyendo los usuarios extremos que se analizarán:



Usuarios Generales

1.1 Diseñadores de moda independientes:

- Pueden trabajar de forma autónoma o tener un equipo muy reducido.
- Necesitan una herramienta ágil para crear y ajustar prototipos sin grandes costos en muestras físicas.
- Se enfrentan a la necesidad de administrar múltiples proyectos y clientes con recursos limitados.

1.2 Diseñadores de moda *in-house*:

- Forman parte del departamento de diseño de una marca o tienda local que ofrece prendas personalizadas.
- Requieren colaboración con demás profesionales del ámbito.
- Manejan volúmenes mayores de pedidos, por lo que buscan velocidad y exactitud en prototipado.

1.3 Patronistas y sastres tradicionales:

- Amplia experiencia en la elaboración manual de patrones y confección.
- En algunos casos, poca familiaridad con software de modelado 3D, lo que exige una interfaz sencilla.
- Valoran la precisión de las medidas y la posibilidad de verificar detalles técnicos en la prenda.

1.4 Estilistas y consultores de moda:

- Se centran en las recomendaciones de estilo y la asesoría de imagen.
- Buscan herramientas que ayuden a visualizar y comunicar opciones de color, silueta o combinaciones.
- Pueden proponer cambios que respondan a tendencias o preferencias del cliente.

1.5 Clientes finales con interés en la personalización:

- Personas que buscan prendas únicas a su medida, con poco o nulo conocimiento técnico de patronaje.
- Quieren previsualizar, hacer ajustes de estilo o medidas, y dar retroalimentación antes de la producción.
- Valoran la facilidad de uso y la claridad de la información al momento de aprobar un diseño.

Usuarios Extremos

2.1 Diseñadores con poca experiencia en herramientas digitales:

- Escasa familiaridad con tecnología de diseño asistido.

- Puede sentirse abrumado/a con sistemas complejos o interfaces poco intuitivas.

Clave de necesidad:

- Una interfaz sencilla, paso a paso, que permite iterar sin conocimientos avanzados.
- Tutoriales, ayudas contextuales y recursos formativos bien integrados.

2.2 Diseñadores de alta costura con alto nivel de exigencia:

- Necesitan control detallado de cada aspecto de la prenda.
- Buscan un realismo muy alto en la simulación y un soporte para telas avanzadas o procesos de producción de lujo.

Clave de necesidad:

- Modelos 3D de gran precisión, librería de telas sofisticadas y posibilidad de ajustes minuciosos.
- Capacidad de exportar o integrar sus diseños con otras herramientas de alta gama.

2.3 Cliente final con inquietudes especiales:

- Preocupación por la exactitud de las medidas y la representación realista de su cuerpo.
- Con frecuencia se sienten excluidos de las tallas estándar, por lo que buscan una experiencia que garantice inclusividad.

Clave de necesidad:

- Visualizaciones adaptadas a su tipo de cuerpo, con simulaciones confiables.
- Hacerlos sentir seguros de que la prenda se ajustará correctamente.

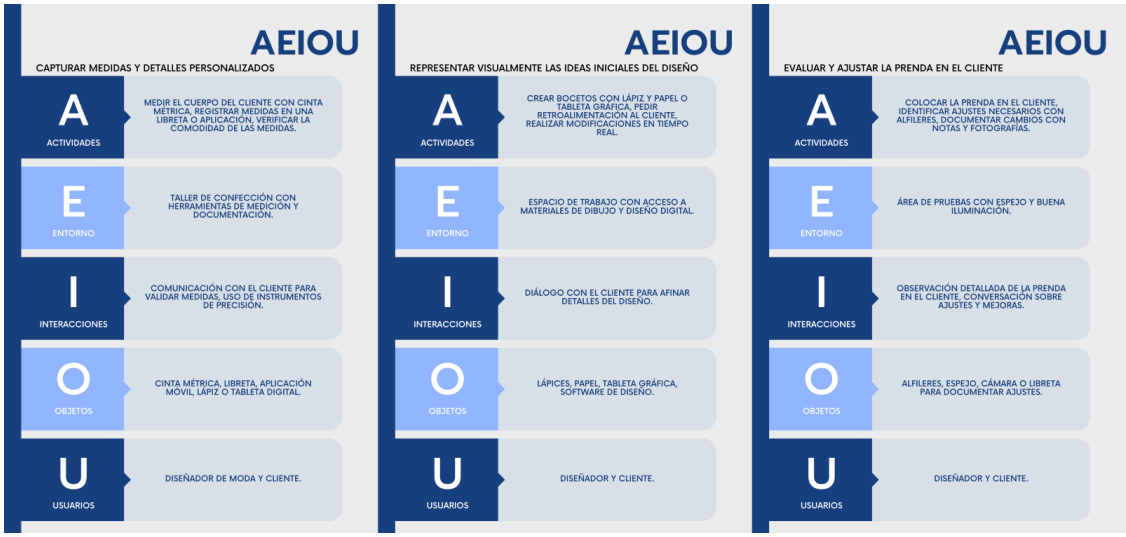
¿Qué? ¿Cómo? ¿Por qué?

A fin de analizar las actividades involucrada en el proceso de diseño y confección de prendas, se realizó el siguiente esquema:

Fotos		¿Qué está sucediendo?	¿Cómo está sucediendo?	¿Por qué está sucediendo?
	A	Capturar las dimensiones exactas del cuerpo del cliente.	Usando una cinta métrica para medir manualmente cada parte del cuerpo relevante.	Unas medidas precisas aseguran un ajuste adecuado en el diseño final.
	B	Usar herramientas precisas para garantizar la fiabilidad de las medidas.	Registrar las medidas en una libreta o aplicación específica para diseño.	Evita ajustes posteriores costosos o frustrantes para el cliente.
	C	Asegurarse de que las medidas incluyan detalles personalizados.	Confirmar con el cliente que todas las medidas se sienten correctas y cómodas.	Refleja un alto nivel de profesionalismo y atención al detalle.
	A	Representar visualmente las ideas iniciales del cliente.	Utilizar lápiz y papel o una tableta gráfica para hacer el boceto.	Un boceto claro y atractivo ayuda a comunicar la visión del diseño.
	B	Capturar los detalles clave de diseño como estilo, silueta y adornos.	Pedir al cliente retroalimentación sobre el diseño preliminar mientras lo desarrolla.	Involucrar al cliente en esta etapa refuerza su confianza en el proceso.
	C	Plasmar un diseño que sirva como base para futuras iteraciones.	Realizar modificaciones en tiempo real, asegurando que el cliente se sienta involucrado.	Reducir confusiones o malentendidos antes de avanzar al corte y confección.
	A	Evaluar cómo se adapta la prenda al cuerpo del cliente.	Colocar la prenda en el cliente frente a un espejo para análisis visual.	Este paso asegura que la prenda final cumpla con las expectativas del cliente.
	B	Identificar áreas que requieran modificaciones o mejoras.	Usar alfileres para marcar ajustes necesarios, como dobladillos o pinzas.	Minimiza el riesgo de insatisfacción y genera confianza en el proceso.
	C	Asegurarse de que la prenda sea funcional y estéticamente agradable.	Documentar los ajustes requeridos con notas detalladas y fotografías si es necesario.	Fortalece la reputación de la diseñadora como alguien que se preocupa por los detalles.

AEIOU

A fin de complementar el análisis anterior, se realizaron los siguientes esquemas AEIOU para cada actividad descrita:



Perfiles

<u>Clientes finales con interés en la personalización:</u>

Este grupo está formado por personas que anhelan prendas únicas que reflejen su personalidad y estilo. Aunque carecen de conocimientos técnicos en diseño o patronaje, desean involucrarse activamente en el proceso de creación. Valoran enormemente una interfaz sencilla y clara que les permita explorar opciones de telas, colores y estilos. La posibilidad de previsualizar sus elecciones en un modelo 3D les brinda confianza y emoción, asegurándoles que el producto final cumplirá con sus expectativas. Buscan una experiencia interactiva y personalizada, donde puedan ajustar detalles y ofrecer retroalimentación de manera sencilla antes de que la prenda entre en producción.

Diseñadores de moda independientes:

Son creativos emprendedores que suelen trabajar solos o en pequeños equipos, enfrentándose al desafío constante de destacar en un mercado competitivo. Necesitan herramientas ágiles que les permitan crear y ajustar prototipos rápidamente sin incurrir en altos costos asociados a muestras físicas. Manejan múltiples proyectos y clientes a la vez, por lo que valoran sistemas que optimicen su flujo de trabajo y les ayuden a gestionar eficientemente sus recursos limitados. Una plataforma que les ofrezca flexibilidad, eficiencia y capacidad de personalización es esencial para impulsar su creatividad y expandir su negocio.

Diseñadores con poca experiencia en herramientas digitales:

Estos profesionales poseen un talento innato para el diseño tradicional, pero pueden sentirse intimidados por la tecnología moderna. Buscan dar el salto al mundo digital sin enfrentarse a curvas de aprendizaje pronunciadas. Una interfaz intuitiva y fácil de seguir es crucial para ellos, así como la inclusión de tutoriales y ayudas contextuales que les guíen en cada paso. Recursos formativos integrados les permiten familiarizarse con la plataforma a su propio ritmo, facilitando la transición y evitando frustraciones. Al simplificarles el acceso a herramientas digitales, se les abre un abanico de posibilidades para innovar y mantenerse relevantes en la industria actual.

CORTE 1: Resumen de Entrevistas

Perfil 1. Diseñadores de moda independientes:

Guión

Sección 1: Contexto Profesional

1. ¿Podrías contarme un poco sobre tu experiencia como diseñador/a de moda?

2. ¿Trabajas de forma independiente o dentro de un equipo?

3. ¿Cómo describes tu proceso actual para diseñar y ajustar prendas personalizadas?

Sección 2: Retos en el Diseño y Prototipado

4. ¿Cuáles son los mayores desafíos al crear prototipos de prendas personalizadas?

5. ¿Cuánto tiempo y recursos inviertes en la creación de muestras físicas?

6. ¿Cómo gestionas la comunicación con tus clientes durante el proceso de diseño?

7. ¿Has utilizado herramientas digitales para modelado 3D o personalización de prendas? Si es así, ¿cómo ha sido tu experiencia?

Sección 3: Necesidades y Oportunidades

8. ¿Qué características te gustaría que tuviera una herramienta digital para mejorar tu proceso de diseño?

9. ¿Cuán importante sería para ti visualizar una prenda en 3D antes de la confección?

10. ¿Qué nivel de control te gustaría tener sobre los detalles de la prenda en un entorno digital?

Resumen

Entrevistada: Gabriela Nájera

1. El proceso de prototipado es costoso y consume tiempo

Observación, Cita o Historia

“A veces es el hecho de que puede ser un poco tardado, por ejemplo, porque a veces hay que hacer el prototipo dos o tres veces, dependiendo de cómo sea el cuerpo de la persona, entonces termina volviéndose a veces un proceso más largo, que sería interesante que pudiera ser más corto para reducir tanto el material con el que se crea y el tiempo tanto de la persona como mío.”

Posible Significado

El diseñador enfrenta dificultades al realizar prototipos debido a las múltiples iteraciones necesarias, lo que implica mayor inversión de tiempo y materiales. Esto sugiere una oportunidad para optimizar este proceso, reduciendo la cantidad de pruebas físicas y mejorando la precisión del diseño desde el inicio.

2. La comunicación con los clientes es variada y a veces ineficiente

Observación, Cita o Historia

“Bueno, eso se puede hacer de diferentes formas, ya que ellos pueden venir en persona, y a veces a través de las ilustraciones, que pueden comunicar las ideas, y también a través de referencias de imágenes que envíen, pero a veces no tienen tiempo de venir, entonces, pues es por llamada, yo les mando fotos (…)”

Posible Significado

El proceso de comunicación con los clientes no está estandarizado, lo que puede llevar a malentendidos y retrasos. Esto indica la necesidad de herramientas digitales que faciliten la comunicación y permitan visualizar ideas de manera clara y rápida.

3. Las herramientas digitales podrían optimizar el proceso, pero su acceso es limitado

Observación, Cita o Historia

“Realmente no las he usado porque es un programa que hay que comprar y que actualmente no lo tengo, pero sí es algo que me interesaría aprender porque de esa forma se puede ahorrar el proceso de muestras, ¿verdad?”

Posible Significado

Aunque el diseñador reconoce el valor de las herramientas digitales para optimizar su flujo de trabajo, el costo y la curva de aprendizaje representan una barrera. Esto resalta la oportunidad de desarrollar soluciones más accesibles y fáciles de usar para la industria de la moda personalizada.

Perfil 2. Clientes finales con interés en la personalización:

Guion

Sección 1: Experiencia con la Personalización de Prendas

1. ¿Has comprado prendas personalizadas antes? ¿Cómo fue tu experiencia?

2. ¿Qué te motiva a elegir prendas hechas a la medida en lugar de opciones estándar?

3. ¿Qué aspectos valoras más al personalizar una prenda (ajuste, diseño, exclusividad, etc.)?

Sección 2: Expectativas y Frustraciones

4. ¿Cuáles han sido los mayores retos al encargar ropa personalizada?

5. ¿Te ha resultado difícil visualizar cómo quedará la prenda antes de que esté lista?

6. ¿Qué tan importante es para ti poder hacer ajustes antes de la confección final?

7. ¿Te gustaría contar con herramientas digitales para ver una representación virtual de la prenda antes de su producción?

Sección 3: Oportunidades de Mejora

8. ¿Cómo sería tu experiencia ideal al solicitar una prenda personalizada?

9. ¿Qué elementos crees que podrían hacerte sentir más seguro/a al encargar una prenda?

10. ¿Qué tan cómodo/a te sentirías utilizando una plataforma digital para personalizar y visualizar tu prenda antes de la confección?

Resumen

Entrevistada: Victoria Chuj

1. Dificultad en expresar y visualizar lo que quiere

Observación, Cita o Historia

“No sé, tal vez como que transmitirle la idea de lo que quiero a la persona que lo está haciendo.”

Posible Significado

El cliente enfrenta barreras al comunicar sus expectativas, lo que puede generar frustración y falta de alineación con el diseñador. Esto indica la necesidad de herramientas que faciliten la visualización y personalización de prendas en un entorno digital.

2. Quiere seguridad en el proceso

Observación, Cita o Historia

“De nuevo comunicación, pero como que más allá como de actualización, así como que hay, o como que vamos, no sé, creo que ayuda mucho tener, digamos, un cronograma o como que tiempos de entrega para que uno se sienta más cómodo sabiendo cuándo y qué es lo que le van a estar entregando.”

Posible Significado

La falta de transparencia y seguimiento en el proceso de personalización genera inseguridad en el cliente. Esto sugiere la oportunidad de implementar herramientas que brinden actualizaciones en tiempo real sobre el estado del pedido.

3. Le interesa la tecnología, pero no como reemplazo del contacto personal

Observación, Cita o Historia

"Creo que podría ser interesante tener una plataforma digital, pero creo que al final del día como que sí le tendría que complementar como que algo más en persona."

Posible Significado

Si bien una herramienta digital podría mejorar la experiencia, no debe reemplazar la interacción con el diseñador. La clave está en un enfoque híbrido donde la tecnología complemente el proceso sin eliminar el contacto humano.

<u>Perfil 3. Diseñadores de moda con poca experiencia en herramientas digitales:</u>

Guion

Sección 1: Contexto Profesional

1. ¿Podrías contarme un poco sobre tu experiencia como diseñador/a de moda?

2. ¿Cómo sueles trabajar en la creación de tus diseños (bocetos a mano, patrones físicos, software, etc.)?

3. ¿Has utilizado alguna herramienta digital para el diseño de moda? Si es así, ¿cuál fue tu experiencia?

Sección 2: Retos con la Tecnología

4. ¿Cuáles son los principales desafíos que has enfrentado al intentar usar herramientas digitales en tu trabajo?

5. ¿Qué te ha impedido adoptar más tecnología en tu proceso de diseño? (Ejemplo: falta de capacitación, herramientas complejas, costo, etc.)

6. ¿Qué tan cómodo/a te sientes con el uso de programas de diseño en general (Photoshop, Illustrator, software de modelado 3D, etc.)?

7. ¿Cuándo necesitas hacer ajustes en tus diseños, prefieres hacerlo manualmente o crees que una herramienta digital podría facilitar el proceso?

Sección 3: Expectativas y Necesidades

8. Si una herramienta digital pudiera ayudarte en tu trabajo, ¿qué funciones debería tener para que fuera fácil de usar?

9. ¿Qué tipo de capacitación o apoyo te ayudaría a sentirte más cómodo/a usando tecnología en el diseño de moda?

10. ¿Cuán importante sería para ti que una herramienta digital tuviera tutoriales, soporte técnico o una interfaz sencilla?

Sección 4: Oportunidades de Mejora

11. ¿Te gustaría poder visualizar cómo quedará una prenda antes de confeccionarla?

12. ¿Si una plataforma digital pudiera mejorar tu proceso, qué aspectos te gustaría que simplificara?

13. ¿Crees que con el tiempo y la herramienta adecuada podrías sentirte más cómodo/a usando tecnología en el diseño de moda?

Resumen

Entrevistada: Gabriela Nájera

1.Usa herramientas digitales para ilustrar, pero no para patronaje o modelado 3D

Observación, Cita o Historia

“Bueno, yo hago los diseños en Procreate, que es como una herramienta para dibujar en iPad, y después otros dibujos que se hacen para producción los hago en plataformas como Illustrator.”

Posible Significado

El diseñador ha adoptado herramientas digitales para bocetos y diseño gráfico, lo que le permite agilizar su proceso creativo. Sin embargo, su uso de tecnología está limitado a la ilustración y no se extiende a la parte técnica del diseño, como el patronaje en 3D.

2. No ve desafíos en el uso de tecnología para diseño, pero sí en la representación realista

Observación, Cita o Historia

“Realmente no es un desafío. Creo que tal vez el hecho de que se mire lo más realista posible, que muchas veces un dibujo no se ve igual que la realidad, pero funciona muy bien realmente.”

Posible Significado

El diseñador se siente cómodo con herramientas digitales, pero reconoce que la visualización de las prendas en ilustraciones no siempre representa con precisión el producto final. Esto sugiere la oportunidad de incorporar herramientas de simulación más realistas.

3. El costo es una barrera para adoptar herramientas de modelado 3D

Observación, Cita o Historia

“Ese sí sirve más a veces verlo en 3D, que hay programas así, pero son un poco caros. Entonces no los he aprendido a usar.”

Posible Significado

Aunque el diseñador reconoce el valor del modelado 3D, el costo de las herramientas es una barrera para su aprendizaje y adopción. Esto indica la necesidad de soluciones más accesibles o alternativas de capacitación que faciliten su uso.

4. Visualizar una prenda antes de confeccionarla es clave, pero el patrón es más importante que el boceto

Observación, Cita o Historia

“Sí, realmente esa es toda la idea de hacer este proceso, digamos. Muchas veces, por ejemplo, si uno mira un dibujo, uno que es el diseñador sí se lo logra imaginar. Pero donde importa más a veces es en el patrón porque el patrón es el que al final le da la forma a la ropa, más que el dibujo.”

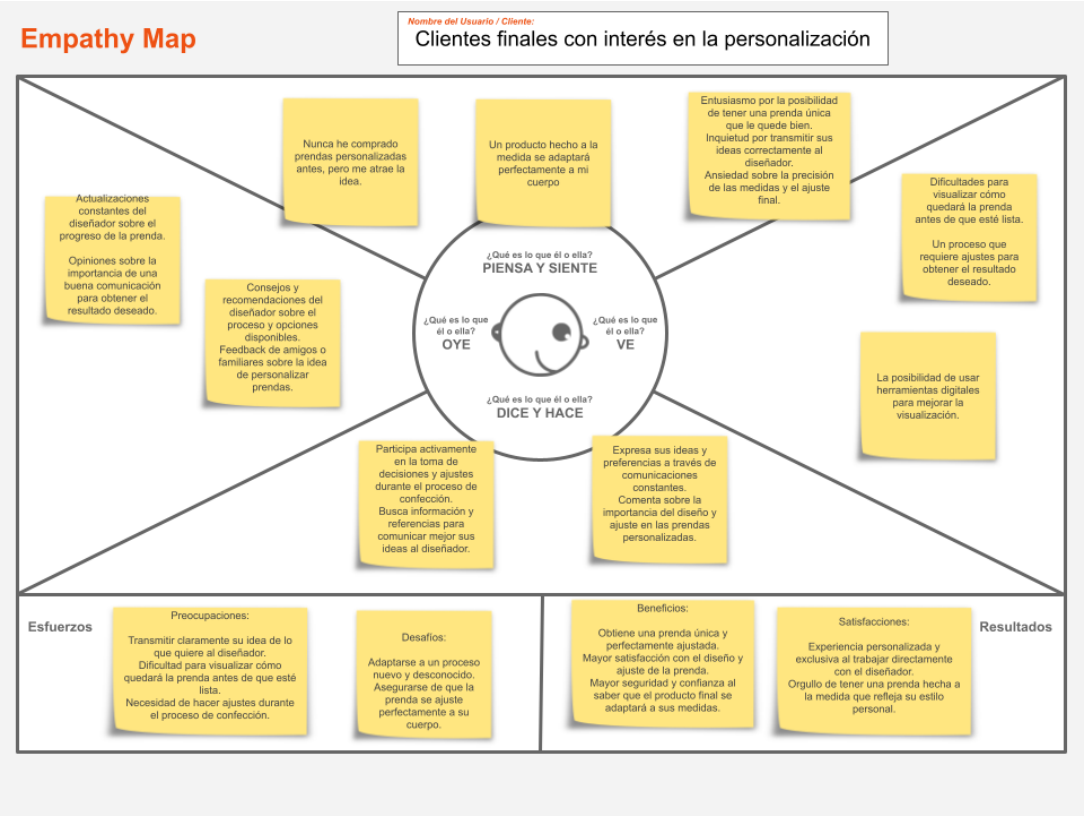
Posible Significado

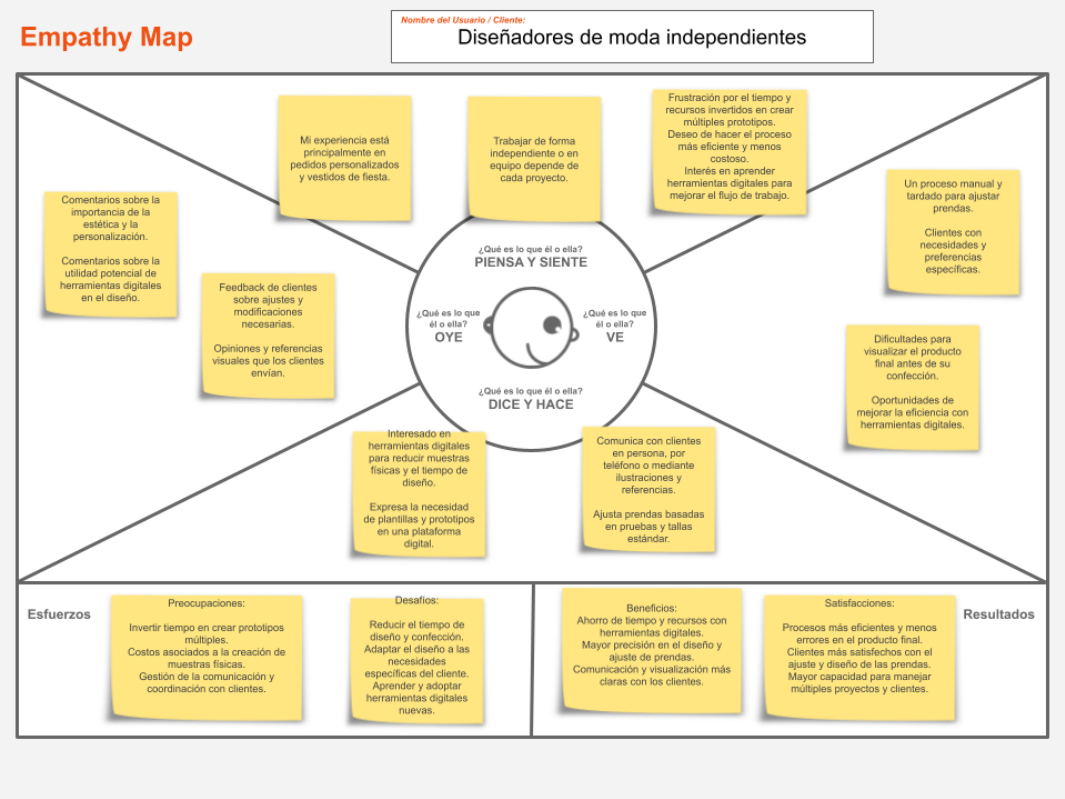
Aunque las ilustraciones ayudan a conceptualizar un diseño, lo que realmente define la prenda es el patronaje. Esto indica la oportunidad de desarrollar herramientas que prioricen la construcción y ajuste de patrones más que la simple visualización de bocetos.

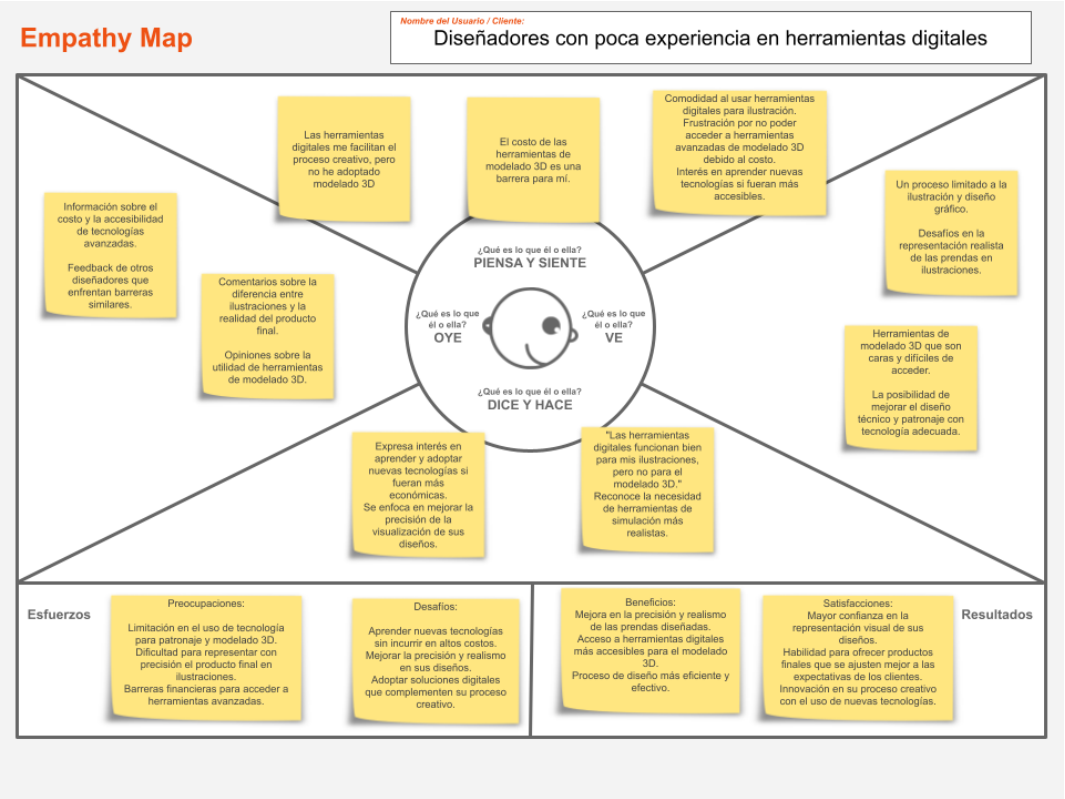
CORTE 1: Design Thinking Posterior a las entrevistas:

Mapas Empatía

Los mapas de empatía nos permiten entender mejor a nuestros usuarios y sus contextos, esto nos permite tener una visión clara y conjunta de las necesidades guiando el desarrollo de una plataforma que realmente satisfaga sus necesidades y supere sus expectativas.







Insights

Diseñador/a de Moda Independiente:

1. El proceso de prototipado es costoso y consume tiempo → Se requieren múltiples pruebas para alcanzar el ajuste ideal, lo que incrementa costos y desperdicio de material.
2. La comunicación con clientes es variada y a veces ineficiente → Utiliza ilustraciones, referencias y llamadas, pero esto puede generar malentendidos o retrasos.
3. Las herramientas digitales podrían optimizar el proceso, pero su acceso es limitado → No ha usado modelado 3D por el costo, pero ve potencial en reducir errores y tiempos de producción.

Cliente Final:

1. Dificultad en expresar y visualizar lo que quiere → No siempre puede comunicar su idea con claridad al diseñador.
2. Quiere seguridad en el proceso → Necesita actualizaciones y tiempos de entrega claros para sentirse más cómodo.
3. Le interesa la tecnología, pero no como reemplazo del contacto personal → Le gustaría visualizar la prenda digitalmente, pero también valora el proceso en persona.

Diseñador/a de Moda con poca Experiencia en Herramientas Digitales:

1. La ilustración digital agiliza la fase creativa, pero no reemplaza completamente la realidad → Herramientas como Procreate e Illustrator permiten generar ideas rápidamente, pero la representación visual no siempre coincide con el resultado final.
2. El modelado 3D es útil, pero su adopción es limitada por costos y necesidades específicas → Aunque reconoce su valor para la confección y el patrónaje, el diseñador prioriza su imaginación y considera que estas herramientas benefician más al consumidor que al diseñador.
3. Las texturas y materiales digitales suelen ser una limitación → Los dibujos no reflejan con precisión la caída o el comportamiento de la tela, lo que dificulta la visualización realista del diseño antes de producirlo.

4. Las herramientas digitales deben ser intuitivas y contar con recursos de aprendizaje accesibles → La disponibilidad de tutoriales y soporte es clave para la adopción de nuevas tecnologías, ya que no siempre es intuitivo aprenderlas sin ayuda.

Necesidades Identificadas

Diseñador/a de Moda Independiente:

• Reducir el número de iteraciones en el prototipado para ahorrar tiempo y materiales.

• Contar con una plataforma que permita visualizar diseños en 3D antes de hacer muestras físicas.

• Mejorar la comunicación con clientes mediante herramientas visuales más rápidas.

Cliente Final:

• Un proceso más claro y guiado para transmitir sus ideas al diseñador.

• Mayor seguridad en la personalización mediante actualizaciones y cronogramas definidos.

• Posibilidad de previsualizar la prenda antes de que se confeccione para reducir incertidumbre.

Diseñador/a de Moda con poca Experiencia en Herramientas Digitales:

- Acceder a una plataforma que facilite la visualización realista de texturas y materiales en los diseños.

- Contar con modelos base personalizables para optimizar el tiempo de diseño.

- Disponer de herramientas de aprendizaje integradas, como tutoriales paso a paso, para reducir la curva de aprendizaje en nuevas tecnologías.

- Evaluar soluciones accesibles para el modelado 3D que permitan visualizar el patrónaje sin costos elevados.

¿Cómo Podríamos?

1. ¿Cómo podríamos reducir el tiempo y costo en la creación de prototipos para diseñadores?
    a. Implementar herramientas de modelado 3D accesibles y fáciles de usar.
    b. Crear una biblioteca de plantillas digitales personalizables.

2. ¿Cómo podríamos mejorar la comunicación entre diseñadores y clientes para evitar malentendidos?
    a. Diseñar una plataforma donde los clientes puedan subir referencias y hacer ajustes visuales en tiempo real.
    b. Incorporar un asistente digital que guíe al cliente en la descripción de su idea.

3. ¿Cómo podríamos dar mayor seguridad a los clientes en el proceso de personalización?
    a. Integrar un sistema de seguimiento con actualizaciones sobre el progreso de su prenda.
    b. Definir tiempos de entrega claros y brindar retroalimentación sobre los ajustes en cada etapa.
4. ¿Cómo podríamos mejorar la experiencia de previsualización de prendas personalizadas?
    a. Ofrecer simulaciones en 3D con opciones de ajuste y cambios en tiempo real.
    b. Permitir que el cliente “pruebe” la prenda digitalmente con avatares que reflejen su tipo de cuerpo.
5. ¿Cómo podríamos equilibrar la personalización digital con el contacto humano necesario?
    a. Integrar una herramienta digital que sirva como primer punto de contacto y guía en el proceso, sin reemplazar las pruebas físicas.

b. 2Crear un espacio híbrido donde el diseñador pueda asesorar al cliente mientras usa la plataforma digital.

CORTE 1: Definición Proyecto

En el mundo de la moda personalizada, los diseñadores independientes enfrentan grandes desafíos al crear prendas a medida que reflejen la individualidad de sus clientes. La falta de herramientas accesibles para modelar, simular y personalizar prendas en diferentes tipos de cuerpo genera ineficiencias en el proceso de diseño, lo que puede traducirse en iteraciones prolongadas, costos elevados en muestras físicas y dificultades para comunicar la visión creativa en general.

Este proyecto busca desarrollar una plataforma de software que permita a los diseñadores de moda visualizar sus creaciones en digital, facilitando la personalización y mejorando la interacción con los clientes. La solución estará enfocada en la rapidez y precisión del ajuste de medidas, la simulación realista de telas y la optimización del flujo de trabajo mediante herramientas interactivas.

Al aplicar la metodología de *design thinking*, se han identificado necesidades clave dentro del perfil de una diseñadora independiente, tales como la falta de una visualización clara de los diseños antes de la confección, la dificultad para asesorar a los clientes en términos de estilo y la necesidad de reducir costos en prototipos físicos. A través de esta plataforma, se busca mejorar la experiencia del cliente y potenciar la competitividad de los diseñadores independientes en el mercado de la moda personalizada.

CORTE 2: Aplicación de Design Studio y Pruebas de Prototipos

A continuación, se presentan las evidencias de la implementación de la metodología Design Studio en nuestro proyecto.

Realizamos algunas reuniones entre los integrantes del grupo para realizar la técnica, escribiendo nuestras ideas en pósits para luego discutirlas y llegar propuestas finales para solucionar las oportunidades previamente analizadas.

Consideramos que fue un proceso efectivo donde logramos combinar las ideas más útiles en una solución convincente, como se aprecia en las imágenes adjuntas. Además, planteamos nuestras historias de usuario en base a las necesidades identificadas para el uso y funcionalidad requeridas de nuestros diferentes usuarios.

Ideas Generales

Oportunidad #1

¿Cómo podríamos reducir el tiempo y costo en la creación de prototipos para diseñadores?

• Crear una biblioteca de medidas que, en base al tipo de cuerpo, recomiende diseños.

• Herramienta con plantillas básicas de moldes de ropa editables pero directas.

• Tener plantillas prediseñadas para colocar solo la tela o utilización de IA para generar los bocetos o prototipos.

• Utilizar software especializado en prototipos, en especial si es gratuito. Apoyarse de programas similares existentes. Consultar con alguien experimentado en el área.

Oportunidad #2

¿Cómo podríamos mejorar la comunicación entre diseñadores y clientes para evitar malentendidos?

• Permitir la visualización del catálogo de plantillas disponibles a fin de ser un catálogo de referencias.

• Que el bosquejo inicial sea accesible y comentable, preferiblemente de un modo en vivo, que se hagan sugerencias individuales previas.

• Tener un apartado especial para solicitudes.

• Implementar chatbots para atender a varios clientes a la vez y que la información pueda ser vista por el diseñados para tener los datos recopilados.

Oportunidad #3

¿Cómo podríamos dar mayor seguridad a los clientes en el proceso de personalización?

• Recomendar un horario de fechas estimadas para los procesos de confección del producto en base al estilo de cuerpo seleccionado.

• Llevar un historial de ajustes y modificaciones de los prototipos, consultable.

• Que cada actualización se vaya mostrando al cliente, para que pueda ver el proceso.

• Añadir un tracking por cada fase del proceso, así como una fila por si hay trabajos pendientes.

Oportunidad #4

¿Cómo podríamos mejorar la experiencia de previsualización de prendas personalizadas?

• Utilizar modelos 3D en base a los diseños y plantillas previamente establecidos para los distintos tipos de cuerpo, a fin de poder realizar modificaciones en poco tiempo y mantener previsualización.

• Crear el catálogo de plantillas como lo habíamos pensado, y añadir una descripción de opiniones y recomendaciones del diseñador.

• Incorporar simulaciones que muestren cómo se comportan diferentes telas al moverse con realidad aumentada.

• Añadido al modelaje, que se le agreguen opciones de los detalles más comunes acerca del estilo.

Oportunidad #5

¿Cómo podríamos equilibrar la personalización digital con el contacto humano necesario?

• Realizar una página web y un software diseñado en base a las necesidades que se deben cubrir, tanto para la diseñadora como para el cliente; para el proceso de confección y el tracking.

• Que la herramienta se use en dos modalidades, una remota para el inicio y pedidos y una para interacciones personales con el diseñador.

• Desarrollar tutoriales y guías que expliquen paso a paso el uso de la plataforma o procesos de personalización.

• Hacer en plataformas diferentes lo que tendrá el diseñador y el cliente.

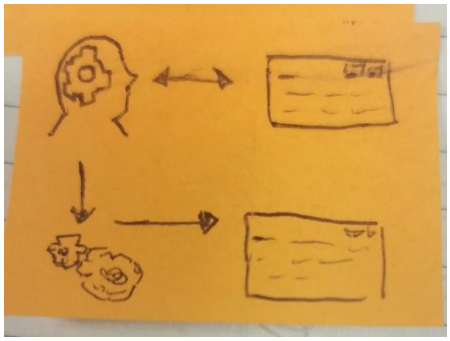
Ideas Votadas

Oportunidad #1

Utilizando proyectos ya hechos y el uso de IA puede ser útil, para que estos entrenen o “alimenten” al algoritmo. Además, se pueden combinar las ideas de la biblioteca de medidas con las plantillas y moldes.

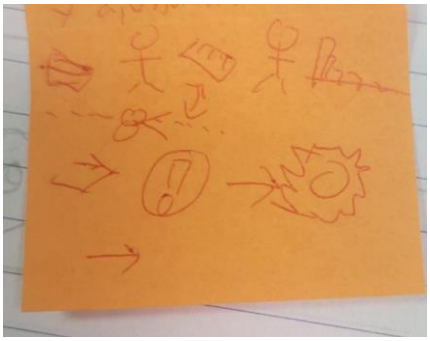
• Idea Pareja #1.

Utilizar proyectos ya hechos para entrenar el algoritmo de una IA



• Idea Pareja #2.

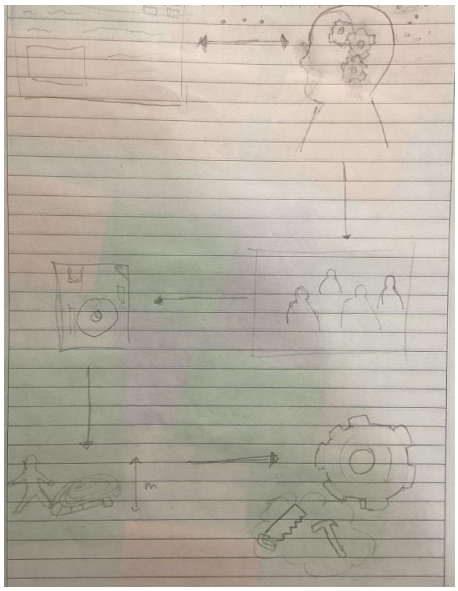
Biblioteca de medidas, clasificadas al tipo de cuerpo, relacionar moldes y plantillas comunes y ajustables.



Finalmente, se concluyó que era esencial el análisis de patrones en la información y sugerencias basadas en el mismo para simplificar el proceso de generación de prototipos

• Idea Final.

Uso de recursos digitales de dominio público para la generación de plantillas y su categorización en base al tipo de cuerpo. Almacenamiento de plantillas en una base de datos y en base a las medidas del cliente recomendar diseños acordes al tipo de cuerpo permitiendo ajustes y correcciones al gusto.

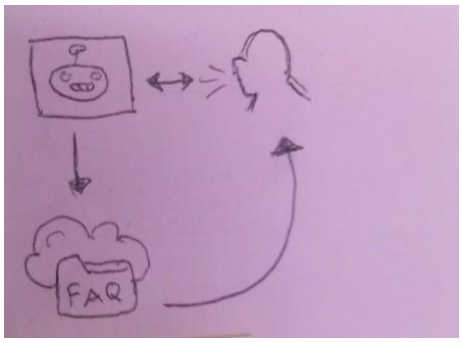


Oportunidad #2

Al tener plantillas o sugerencias anteriores, el utilizar un chatbox será útil para comunicarse con múltiples clientes al guardar las respuestas este dará una mejor respuesta, así como respuestas a preguntas básicas.

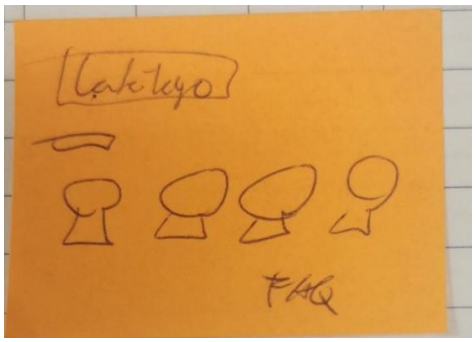
• Idea Pareja #1.

Cuando el cliente final hable con el chatbot, este le dejará una sugerencia, basándose en las solicitudes parecidas anteriores.



• Idea Pareja #2.

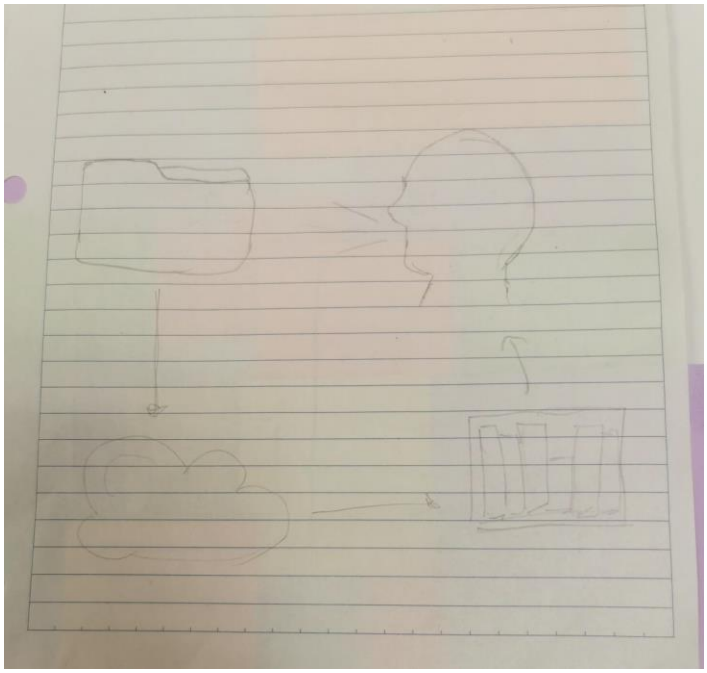
Ofrecer un catálogo de plantillas básicas modificables con sugerencias básicas, añadir FAQ.



La conclusión de la discusión fue que la disponibilidad de información de consulta y referencia, así como la facilidad de expresar ideas entre elc liente y el diseñador es esencial para mejorar la comunicación.

• Idea Final.

Catálogo con bocetos de las plantillas, adjuntar fotos de ejemplo, añadir sección de FAQ basada con sugerencias.

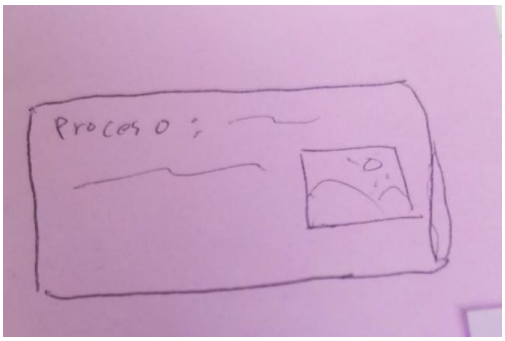


Oportunidad #3

Implementar un tracker, como los que tienen las aplicaciones de comida, es una herramienta útil para mostrar al cliente el estado de su prenda, de esta manera se le puede dar seguridad en el proceso que se esté llevando a cabo.

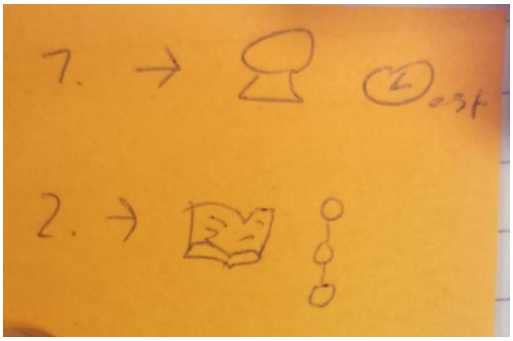
• Idea Pareja #1.

Implementación de un sistema de seguimiento visual de cada proceso. Incluirá una imagen representativa de cada proceso, pero no específicamente de cada pedido.



• Idea Pareja #2.

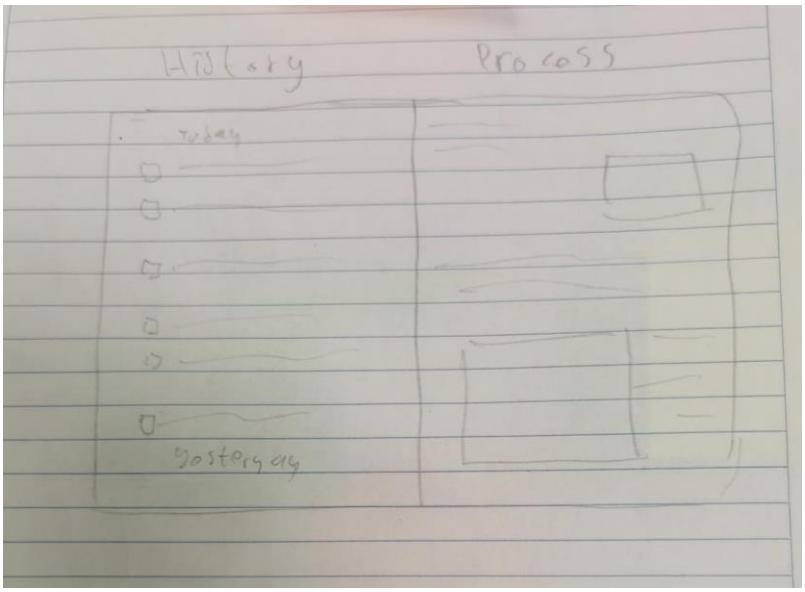
Sugerencias de tiempo estimado por plantilla, historial de ajustes al prototipo en diseño.



Nuestra discusión nos llevó al planteamiento de un sistema de seguimiento del producto, que se pueda consultar fácilmente y permita obtener el registro de cada fase. Además, consideramos útil en este aspecto añadir un estimado de tiempo inicial como referencia.

• Idea Final.

Hacer un tiempo estimado de proyecto. Dar un tracker de etapas de desarrollo e historial de ajustes, con mensajes de stock.

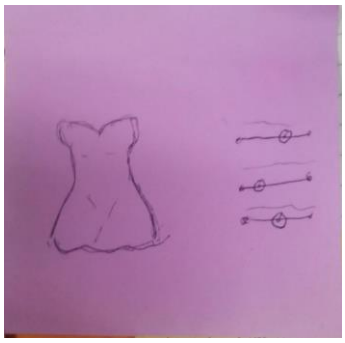


Oportunidad #4

Con simulaciones, se puede lograr ver el modelo 3D de una mejor manera, viendo sus comportamientos a movimientos. Sin embargo, también se ve la necesidad de la practicidad de implementar una herramienta de este tipo.

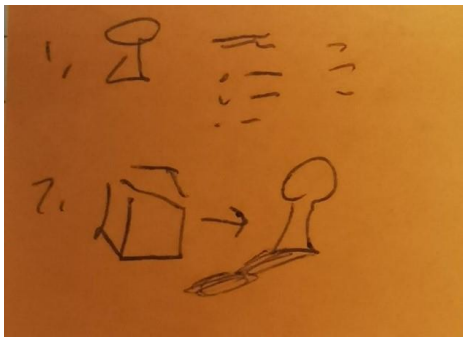
• Idea Pareja #1.

Hacer que el modelaje 3D pueda simular físicas y ediciones de tamaño pequeñas.



• Idea Pareja #2.

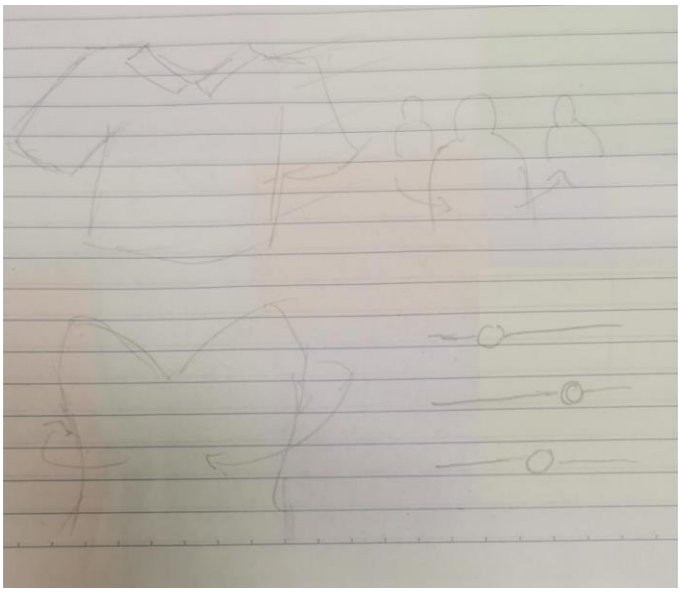
Descripción por plantilla, visualización básica en 3D del prototipo.



Sin lugar a dudas se concluyó sobre la necesidad crucial que tiene el diseñador de ofrecerle al cliente una representación fidedigna del diseño, de tal manera que este pueda confiar en la ruta que se estará tomando

• Idea Final.

Descripción por plantilla. Animación 3D del modelo en base a diferentes telas y proporciones, como opción de complemento.



Oportunidad #5

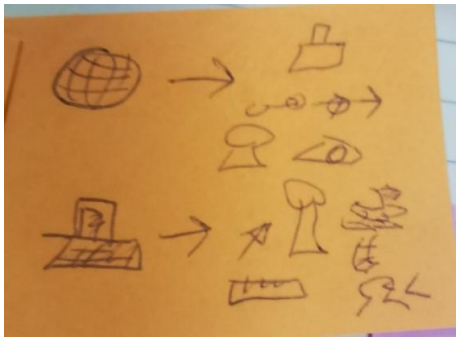
El utilizar guías o tutoriales hará que la plataforma utilizada sea más fácil de usar para los clientes. Además es necesario que cualquier solución ofrecida tenga una dinámica remota y una que permita el contacto humano.

• Idea Pareja #1.

Garantizar el equilibrio con 2 plataformas

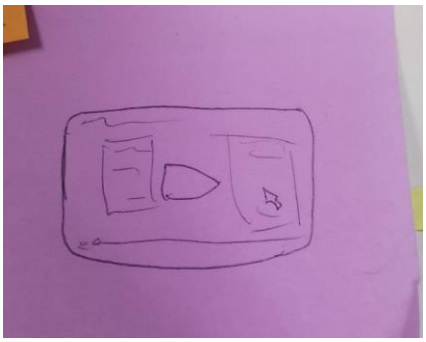
• Servicios al cliente por página web, seguimiento de órdenes e inicio de contacto

• Software de diseño, que recopila datos y agiliza el proceso de trabajo.



• Idea Pareja #2.

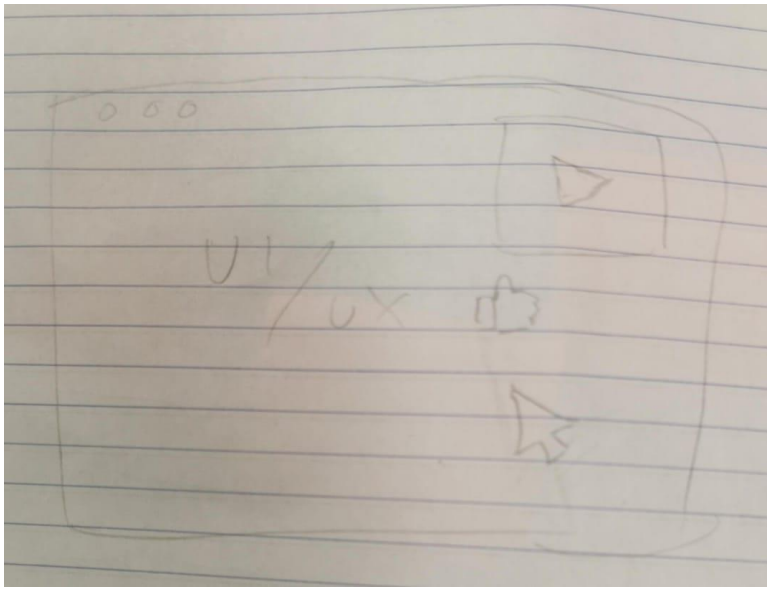
Agregarle tutoriales para que sea más fácil de hacer



La exposición de nuestras ideas significó una comprensión sobre el hecho de que el equilibrio entre la digitalización y el contacto humano radica en la división de ciertas labores, algunas de las cuales que se pueden realizar de paralelo de manera remota y otras que requieren un contacto humano directo. En base a esto se observó la necesidad de la división especializada de las modalidades de la solución. Además, conviene la inclusión de guías de uso, la cual se relaciona con la necesidad de la mejora en la comunicación.

• Idea Final.

Plataforma dividida en varios puntos: Una parte web remota accesible, que tiene el historial y trazos, así como el mismo manejo de pedidos, incluir tutoriales y guías de uso. Una parte local de admin, donde haya análisis de datos y controles del diseñador útil para el manejo en persona.



Prototipos

Para agilizar el proceso se decidió unificar los prototipos de todas las ideas en una sola presentación para cada iteración. Los audios de las pruebas se encuentran en el repositorio GitHub.

Versión 1: Basado en las ideas finales del Design Studio

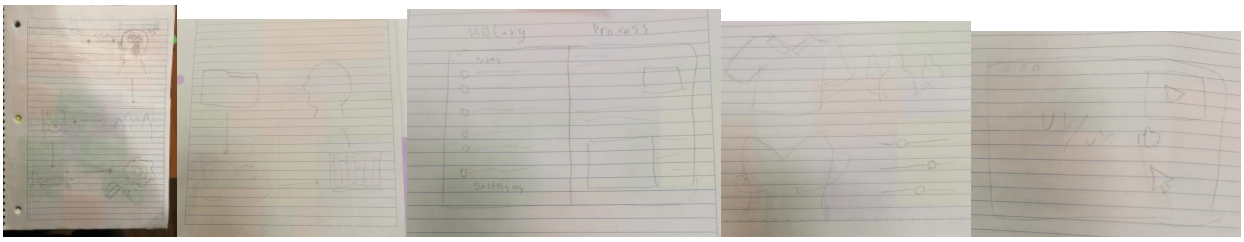
1. Biblioteca de medidas y plantillas: Un sistema que recomienda diseños en base al tipo de cuerpo del usuario.

2. Catálogo con bocetos y FAQ: Para mejorar la comunicación entre diseñador y cliente.

3. Sistema de seguimiento de pedidos: Un tracker visual con estimaciones de tiempo y actualizaciones del diseño.

4. Simulación 3D de prendas: Modelos en 3D con opciones de visualización de diferentes telas.

5. Plataforma con secciones diferenciadas: Una parte para interacción remota y otra para el diseñador con herramientas de control.



Resumen de retroalimentación:

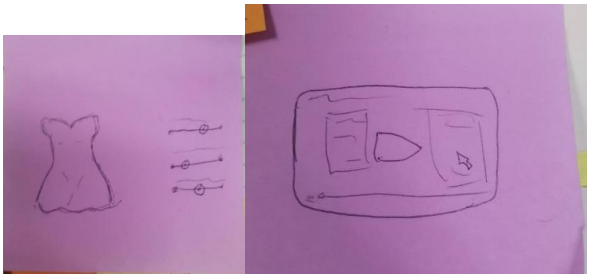
Hay un interés en biblioteca de medidas y plantillas con modelos estándar y medidas personalizadas que ayuda con la visualización del resultado. Además, conviene añadir una ayuda personalizada como un chatbot.

Versión 2: Basado en la retroalimentación recibida

Se añade

Modelos estándar predefinidos y posibilidad de ingresar medidas personalizadas para una recomendación más precisa

Integración de un asistente virtual o chatbot para facilitar la navegación y recomendaciones



Resumen de retroalimentación:

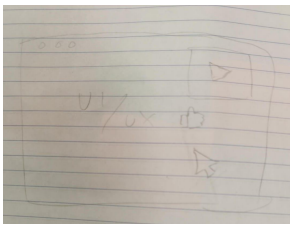
Se nota que el hecho de poder ingresar medidas personalizadas mejora la precisión del resultado obtenido, además que la integración de un chatbot con respuestas elaboradas puede facilitar el uso de esta herramienta.

Versión 3: Basado en la continuación de la retroalimentación

Se especifica que

La biblioteca de plantillas es basada en moldes básicos con opciones de personalización según volumen y proporciones del usuario.

El Chatbot inteligente viene con respuestas guiadas y posibilidad de derivar a un asesor humano cuando sea necesario.



Resumen de retroalimentación:

El prototipo se ve muy bien y es suficientemente bueno.

CORTE 2: HISTORIAS DE USUARIO Y USER STORY MAPPING

Historias Usuario

1. Diseñador de Moda Independiente

• Visualización de Diseños en 3D: Como diseñador independiente, quiero visualizar mis diseños en un modelo 3D realista, para reducir la cantidad de prototipos físicos y optimizar mi tiempo y materiales.

• Personalización de Prendas: Como diseñador independiente, quiero ajustar las medidas y estilos de mis diseños digitalmente, para ofrecer un servicio más eficiente y preciso a mis clientes.

• Optimización del Flujo de Trabajo: Como diseñador independiente, quiero una plataforma que almacene y organice mis diseños y medidas, para mejorar la gestión de mis proyectos.

• Colaboración con Clientes: Como diseñador independiente, quiero compartir avances de diseño de manera digital con mis clientes, para obtener su retroalimentación sin necesidad de reuniones presenciales.

2. Cliente Final con Interés en la Personalización

• Previsualización de Prenda en 3D: Como cliente, quiero ver una simulación digital de mi prenda en mi tipo de cuerpo, para asegurarme de que se ajuste correctamente antes de la confección.

• Personalización de Detalles: Como cliente, quiero poder elegir colores, telas y detalles de mi prenda a través de la plataforma, para reflejar mi estilo personal.

• Seguimiento del Pedido: Como cliente, quiero recibir actualizaciones en tiempo real sobre el estado de mi prenda, para sentirme seguro sobre los tiempos de entrega.

• Interacción con el Diseñador: Como cliente, quiero enviar comentarios sobre el diseño a través de la plataforma, para que mis sugerencias sean tomadas en cuenta antes de la producción final.

3. Diseñador con Poca Experiencia en Herramientas Digitales

• Interfaz Intuitiva: Como diseñador con poca experiencia digital, quiero una interfaz sencilla y amigable, para que pueda utilizar la herramienta sin necesidad de capacitación extensa.

• Tutoriales y Soporte: Como diseñador con poca experiencia digital, quiero acceso a tutoriales y guías paso a paso, para aprender a usar las herramientas sin complicaciones.

• Plantillas de Diseño: Como diseñador con poca experiencia digital, quiero contar con plantillas de diseño prediseñadas, para agilizar mi proceso de creación sin partir de cero.

• Simulación de Patronaje: Como diseñador con poca experiencia digital, quiero poder visualizar los patrones de mis diseños en un entorno digital, para reducir errores antes de la confección.

4. Administrador de la Plataforma

• Gestor de Usuarios: Como administrador, quiero gestionar los perfiles de diseñadores y clientes, para asegurar un uso adecuado de la plataforma.

• Control de Contenido: Como administrador, quiero moderar y verificar los diseños compartidos en la plataforma, para garantizar que cumplen con los estándares de calidad.

• Análisis de Uso: Como administrador, quiero acceder a estadísticas sobre el uso de la plataforma, para mejorar su rendimiento y funcionalidad.

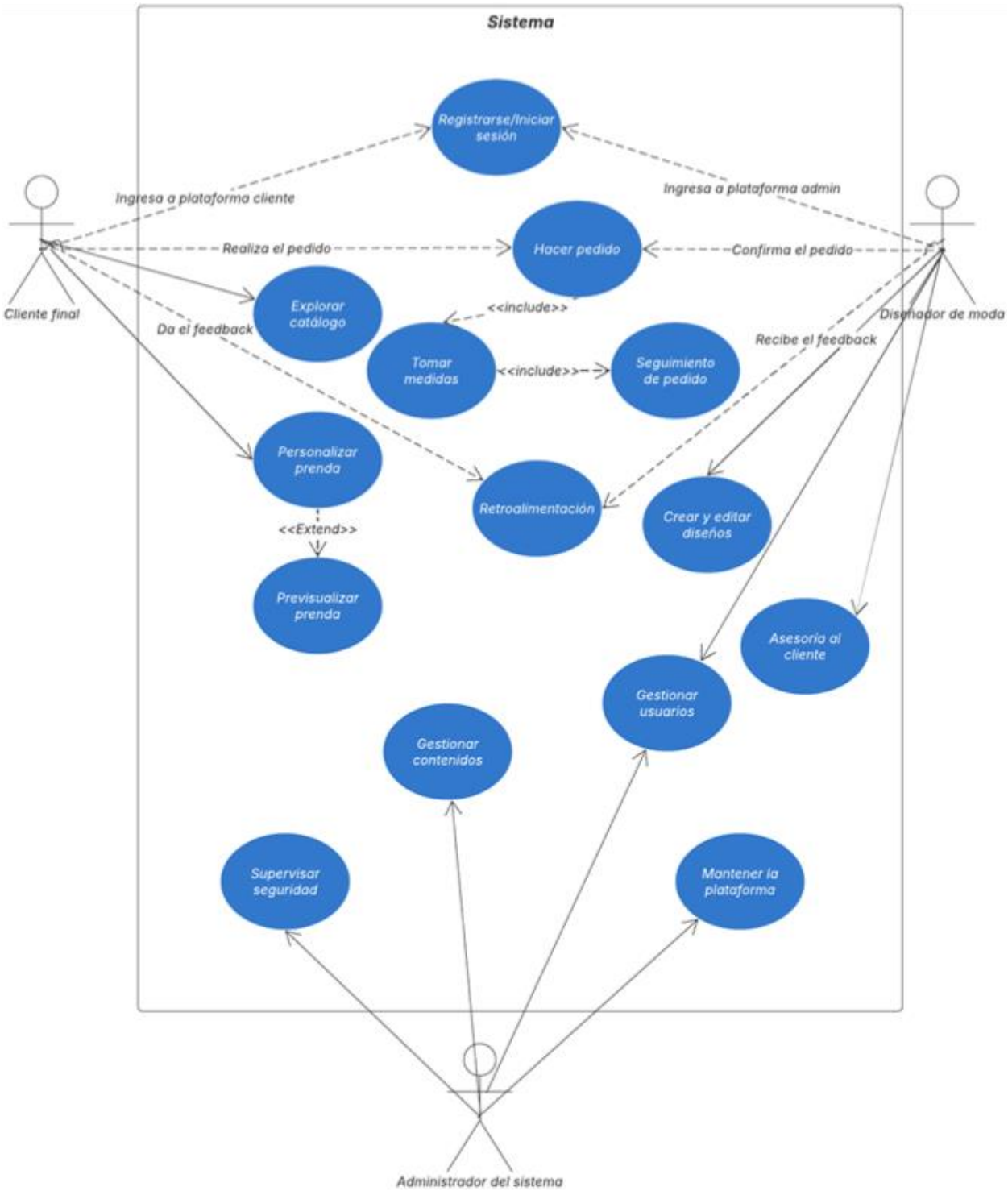
Actores

Actores	**Descripción**
Cliente final	Los clientes finales son individuos que buscan prendas personalizadas que se ajusten perfectamente a sus medidas y gustos. Estos usuarios pueden tener una variedad de perfiles, desde aquellos con un interés casual en la moda hasta personas apasionadas por la personalización y la exclusividad en su vestimenta. Perfil: Edad: Varía, pero generalmente entre 18 y 50 años. Nivel de Conocimiento: Varía desde usuarios con poca experiencia técnica hasta aquellos familiarizados con herramientas digitales básicas. Necesidades y Expectativas: Buscan prendas que reflejen su estilo personal y se ajusten perfectamente a sus medidas. Valoran la facilidad de uso de la plataforma, la capacidad de personalización y la posibilidad de visualizar el producto final antes de su producción. Motivaciones: Deseo de poseer prendas únicas y exclusivas. Interés en una experiencia de compra personalizada y satisfactoria. Valoran la transparencia en el proceso y la calidad del producto final. Desafíos: Pueden tener dificultades para tomar medidas precisas o comunicar sus ideas de diseño. Pueden sentirse inseguros sobre cómo quedará la prenda final sin una visualización previa.

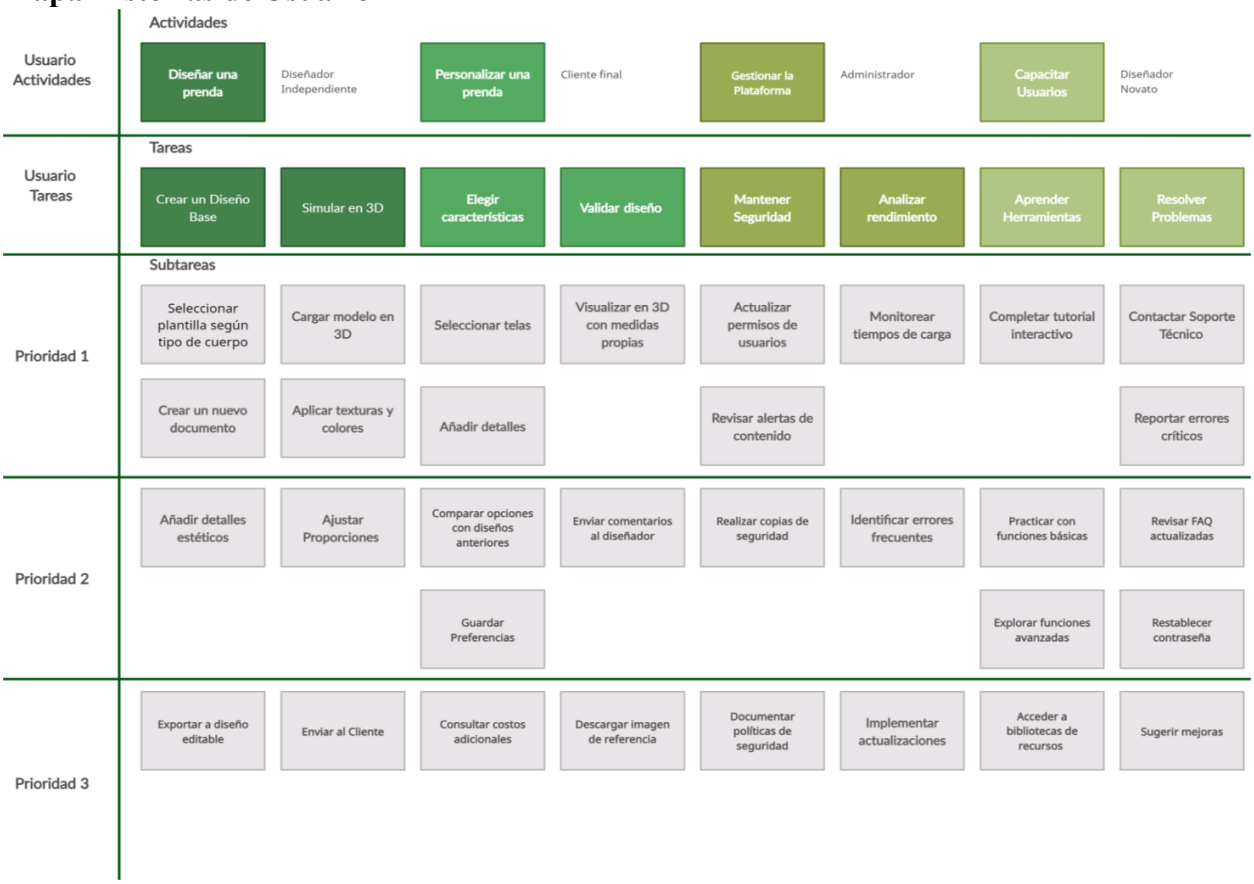
Diseñador de moda	Los diseñadores de moda son profesionales creativos que se encargan de la creación y ajuste de prendas personalizadas. Pueden trabajar de forma independiente o formar parte del equipo de diseño de una marca o tienda. Estos usuarios utilizan la plataforma para gestionar sus diseños, interactuar con clientes y supervisar el proceso de producción. Perfil: Edad: Varía, pero generalmente entre 25 y 45 años. Nivel de Conocimiento: Alta experiencia en diseño de moda, aunque su familiaridad con herramientas digitales puede variar. Necesidades y Expectativas: Necesitan herramientas ágiles y eficientes para crear, ajustar y visualizar diseños. Valoran la capacidad de comunicación con los clientes y la posibilidad de gestionar múltiples proyectos de manera eficiente. Motivaciones: Deseo de innovar y ofrecer diseños únicos. Interés en herramientas que reduzcan el tiempo y costo de producción. Valoran la satisfacción del cliente y la posibilidad de recibir feedback para mejorar sus diseños. Desafíos: Pueden enfrentar barreras tecnológicas si no están familiarizados con ciertas herramientas digitales. La gestión de la comunicación y coordinación con clientes puede ser un reto.
Administrador del sistema	El administrador del sistema es responsable de gestionar y mantener la plataforma, asegurando que funcione de manera eficiente y segura. Este actor se encarga de la administración de usuarios, la gestión de contenidos, la seguridad del sistema y la implementación de actualizaciones. Perfil: Edad: Generalmente entre 30 y 50 años.

	Nivel de Conocimiento: Alta experiencia técnica en administración de sistemas y seguridad informática. Familiaridad con herramientas de gestión y sistemas. Necesidades y Expectativas: Necesitan herramientas que les permitan gestionar usuarios y contenidos de manera eficiente. Valoran la seguridad del sistema y la capacidad de realizar auditorías y mantenimiento de forma regular. Motivaciones: Deseo de mantener la plataforma segura y funcional. Interés en implementar mejoras y actualizaciones que beneficien a todos los usuarios. Valoran la eficiencia y la confiabilidad del sistema. Desafíos: Pueden enfrentar desafíos relacionados con la seguridad de datos y la necesidad de mantener el tiempo de actividad del sistema. La implementación de actualizaciones sin interrumpir el servicio puede ser un reto.

Diagrama de Caso de Uso



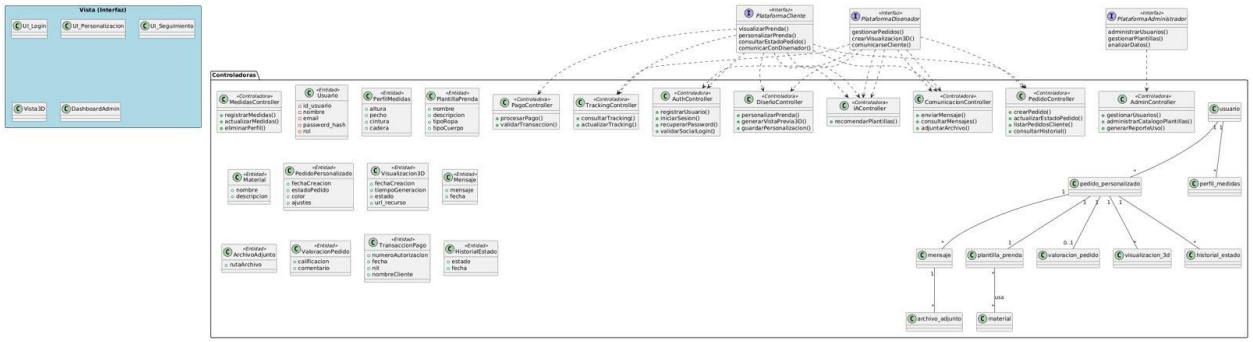
Mapa Historias de Usuario



CORTE 3: Análsis y Diseño del Proyecto

Clases Preliminares

Diagrama de Clases



Descripción de las Clases

Se realizó el diseño preliminar de las clases en base al modelo MVC que se estará implementando en el desarrollo de nuestro proyecto.

Interfaces (Vista)

- PlataformaCliente: Permite al cliente visualizar prendas en 3D, realizar personalizaciones de estilo y medidas, consultar el seguimiento del pedido y comunicarse directamente con el diseñador. Se conecta directamente con las controladoras para realizar estas funciones.
- PlataformaDisenador: Proporciona al diseñador herramientas para gestionar pedidos personalizados, visualizar y modificar diseños en 3D, comunicación con clientes, y controlar el tracking de pedidos, optimizando así la gestión de su trabajo.
- PlataformaAdministrador: Proporciona gestión de usuarios, contenidos, plantillas y estadísticas de uso. Conectada a controladoras administrativas específicas.

Clases Controladoras (Controladores)

- AuthController: Responsable de gestionar registros, autenticaciones, validaciones y sesiones (incluyendo login social). Interactúa directamente con usuarios.
- PedidoController: Gestiona el ciclo completo de pedidos personalizados desde su creación hasta cambios de estado y almacenamiento histórico.
- DiseñoController: Se enfoca en la generación de visualizaciones previas 3D, personalización de diseños (colores, materiales, ajustes) y almacenamiento seguro de estas personalizaciones.
- ComunicacionController: Controla el intercambio de mensajes y archivos entre diseñadores y clientes relacionados con los pedidos específicos.
- TrackingController: Maneja la consulta y actualización del estado de los pedidos, proporcionando información clara y transparente al cliente.
- IAController: Utiliza algoritmos inteligentes para recomendar plantillas según medidas corporales y preferencias de estilo del cliente, basándose en datos históricos.
- PagoController: Administra pagos, transacciones seguras, y asegura el cumplimiento legal (facturación SAT Guatemala).
- AdminController: Gestiona tareas administrativas: manejo de usuarios, pedidos, auditorías y estadísticas generales del sistema.
- MedidasController: Gestiona las medidas ingresadas por el usuario y se encarga de registrarlas y actualizarlas segun los datos que se indican.

Clases de Entidad (Modelo)

- Usuario: Representa a los usuarios del sistema (clientes, diseñadores o administradores). Contiene información para autenticación y perfil básico.
- PerfilMedidas: Almacena perfiles específicos de medidas corporales asociadas a clientes, facilitando personalizaciones exactas para cada usuario.
- PlantillaPrenda: Guarda modelos predefinidos de prendas, categorizadas por tipo de ropa y cuerpo, sirviendo como base para personalizaciones.
- Material: Almacena los diferentes materiales disponibles para ser aplicados a las prendas. Vinculado con plantillas para recomendar combinaciones.
- PedidoPersonalizado: Guarda información detallada de pedidos realizados por los clientes, incluyendo personalizaciones específicas, materiales, colores, medidas y el estado actual del pedido.

- Visualizacion3D: Registra visualizaciones previas generadas en 3D para cada pedido personalizado, incluyendo estados y tiempo de generación.
- Mensaje: Almacena mensajes intercambiados entre cliente y diseñador, facilitando la comunicación efectiva relacionada a cada pedido específico.
- ArchivoAdjunto: Guarda referencias a archivos adjuntos enviados dentro de los mensajes (imágenes, PDFs), facilitando la comunicación visual.
- HistorialEstado: Registra cambios en el estado del pedido personalizado, proporcionando trazabilidad clara del proceso desde creación hasta entrega.
- ValoracionPedido: Permite almacenar la valoración (calificación y comentario) de cada pedido finalizado, contribuyendo al feedback y mejora del servicio.
- TransaccionPago: Gestiona y almacena datos de transacciones de pago asociadas con cada pedido, cumpliendo con los requisitos fiscales y legales establecidos (SAT).

Relaciones Principales (Entidades)

- Usuario ↔ PerfilMedidas: Un usuario puede registrar múltiples perfiles de medidas personales o de terceros.
- Usuario ↔ PedidoPersonalizado: Un usuario puede realizar múltiples pedidos personalizados, cada uno asociado claramente a un solo usuario.
- PedidoPersonalizado ↔ PlantillaPrenda: Cada pedido personalizado se basa en exactamente una plantilla inicial seleccionada por el usuario.
- PlantillaPrenda ↔ Material (Muchos a Muchos): Cada plantilla puede tener múltiples materiales recomendados, y cada material puede estar recomendado en varias plantillas.
- PedidoPersonalizado ↔ Visualizacion3D: Cada pedido puede tener varias visualizaciones 3D asociadas, dependiendo de las iteraciones y ajustes realizados.
- PedidoPersonalizado ↔ HistorialEstado: Cada pedido personalizado tiene múltiples registros históricos de cambio de estado.
- Mensaje ↔ PedidoPersonalizado: Cada mensaje está vinculado a un pedido específico, facilitando la comunicación directa y contextualizada.
- ValoracionPedido ↔ PedidoPersonalizado: Cada pedido finalizado puede tener una valoración única.
- TransaccionPago ↔ PedidoPersonalizado: Cada pago se asocia específicamente a un pedido personalizado.

Casos de Uso y Clases Involucradas

Requisitos Funcionales

1. Registro e Inicio de Sesión para Usuarios

Casos de Uso

- Registrar Usuario
- Iniciar Sesión
- Recuperar Contraseña
- Autenticación Social

Clases Involucradas

- AuthController: Gestiona lógica de autenticación, validación y registro.
- Usuario: Almacena información y credenciales del usuario.
- PlataformaCliente, PlataformaDiseñador, PlataformaAdministrador: Interfaces para registrar y autenticar usuarios.

2. Personalización de Prendas

Casos de Uso

- Personalizar Prenda
- Seleccionar Material y Color
- Ingresar Medidas Personalizadas
- Guardar Personalización

Clases Involucradas

- DiseñoController: Controla personalizaciones de prendas, ajustando diseños y generando vistas previas.
- PlantillaPrenda: Base del diseño personalizado.
- Material: Materiales seleccionados para personalización.
- PerfilMedidas: Almacena medidas específicas usadas para ajustar patrones.
- MedidasController: gestionar el uso de las medidas del cuerpo para la personalización
- PedidoPersonalizado: Guarda personalización específica y detalles adicionales (color, ajustes, notas).
- PlataformaCliente, PlataformaDiseñador: Interfaces para realizar ajustes y personalizaciones.

3. Previsualización de Prendas en 3D

Casos de Uso

- Generar Visualización 3D
- Modificar Visualización en Tiempo Real
- Crear Avatar Personalizado

Clases Involucradas

- DiseñoController: Genera vistas previas en 3D basadas en ajustes.
- Visualizacion3D: Almacena resultados visuales generados, recursos digitales, estados y tiempos.
- PedidoPersonalizado: Proporciona datos específicos sobre la personalización a representar.
- PlataformaCliente, PlataformaDiseñador: Interfaces que muestran vistas previas 3D y permiten modificaciones.

4. Seguimiento de Pedidos

Casos de Uso

- Consultar Estado del Pedido
- Actualizar Estado del Pedido
- Notificar Cambios en Pedido
- Consultar Historial de Pedidos
- Comunicarse verbalmente acerca del Pedido

Clases Involucradas

- TrackingController: Gestiona actualizaciones en estados de pedidos, comunicación y notificaciones automáticas.
- PedidoController: Coordina el seguimiento, creación y gestión general de pedidos.
- HistorialEstado: Registra históricamente cambios de estado en pedidos.
- PedidoPersonalizado: Almacena el estado actual del pedido.
- ComunicaciónController: Gestiona la comunicación entre el cliente y el diseñador, permitiendo establecer cont acto directo y personal sobre el estado del pedido
- Mensaje: Representa los mensajes enviados en el flujo de la comunicación
- Archivo Adjunto: Respresenta algun archivo adjunto al mensaje enviado
- PlataformaCliente, PlataformaDiseñador: Interfaces para consultar estados actuales e históricos.

Requisitos No Funcionales

5. Confiabilidad (Disponibilidad del sistema 99%)

Casos de Uso

- Restaurar Sistema tras Caída
- Monitorear Estado del Sistema

Clases Involucradas

- AdminController: Gestiona monitoreo, recuperación y reinicio de servicios.
- PlataformaAdministrador: Interfaz para monitorear estados del sistema y realizar acciones de recuperación.

6. Usabilidad (Interfaz intuitiva)

Casos de Uso

- Navegación Simplificada en Plataformas
- Uso de Ayudas y Tutoriales

Clases Involucradas

- PlataformaCliente, PlataformaDiseñador, PlataformaAdministrador: Interfaces diseñadas con alta usabilidad.
- IAController: Asistente inteligente que recomienda opciones basadas en interacciones previas del usuario.

7. Portabilidad

Casos de Uso

- Despliegue del Sistema en Distintas Plataformas (Linux, Windows, macOS)

Clases Involucradas

- AdminController: Configura y asegura que la aplicación pueda correr en distintos entornos.
- Indirectamente todas las clases, asegurando que su código fuente es compatible en distintos sistemas operativos.

8. Cumplimiento Legal (Facturación SAT Guatemala)

Casos de Uso

- Generar Factura Electrónica
- Validar Cumplimiento Fiscal

Clases Involucradas

- PagoController: Gestiona creación y validación de transacciones y facturas electrónicas.
- TransaccionPago: Información estructurada y almacenada para cumplimiento fiscal.

9. Escalabilidad

Casos de Uso

- Escalar Sistema Según Demanda de Usuarios
- Monitorear Rendimiento bajo Alta Demanda

Clases Involucradas

- AdminController: Monitorea rendimiento del sistema y gestiona ajustes de escalabilidad técnica.
- Todas las entidades persistentes involucradas en accesos concurrentes (Usuario, PedidoPersonalizado, etc).

10. Ética (No almacenamiento datos sensibles)

Casos de Uso

- Validar Transacción sin Almacenamiento de Información Sensible

Clases Involucradas

- PagoController: Responsable de garantizar que información sensible (tarjetas de crédito) no se almacene en la base de datos.

11. Rendimiento (Vista previa generada en ≤ 4 segundos)

Casos de Uso

- Generar Visualización 3D Eficientemente

Clases Involucradas

- DiseñoController: Garantiza generación rápida de vistas previas 3D.
- Visualizacion3D: Almacena tiempos de generación para monitoreo.

12. Mantenibilidad

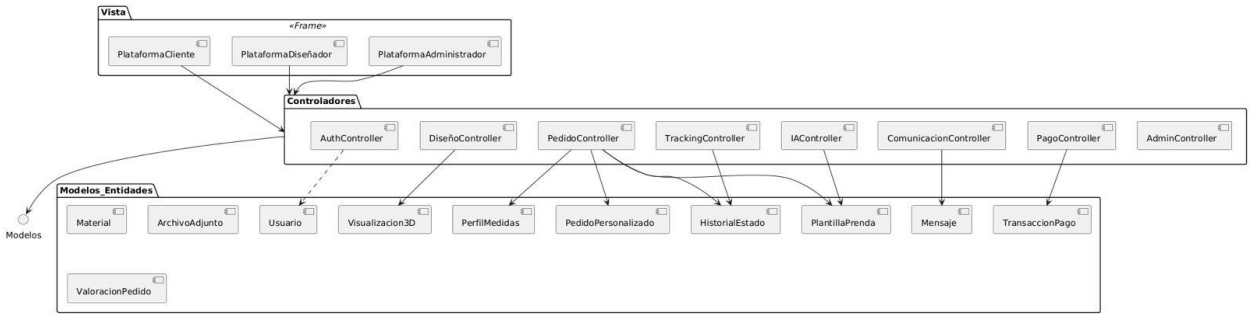
Casos de Uso

- Actualizar Plantillas y Diseños Rápidamente

Clases Involucradas

- AdminController: Gestión eficiente de actualización en plantillas.
- PlantillaPrenda y Material: Permiten modificaciones rápidas, estructura simple y clara.

Diagrama de Paquetes



Descripción de los Paquetes y sus Componentes

Vista (Interfaz)

Este paquete agrupa todas las interfaces gráficas y componentes que permiten la interacción directa con los usuarios del sistema (clientes, diseñadores y administradores). Aquí se encuentran:

- PlataformaCliente: Interfaz utilizada por clientes finales para visualizar prendas en 3D, realizar personalizaciones, consultar estados de pedidos, pagos y comunicarse con diseñadores.
- PlataformaDiseñador: Herramienta interactiva utilizada por los diseñadores para gestionar pedidos, visualizar personalizaciones, comunicarse con clientes, y administrar el proceso de creación y producción.
- PlataformaAdministrador: Herramienta utilizada por los administradores para gestionar usuarios, contenidos, revisar estadísticas de uso y mantener la plataforma segura y funcional.

Controladores

Paquete que agrupa las clases responsables de la lógica de negocio y la coordinación entre las interfaces (vistas) y los datos (entidades). Controlan operaciones como autenticación, gestión de pedidos, seguimiento, comunicación y recomendaciones con inteligencia artificial. Incluye:

- AuthController: Gestiona autenticación, registros, recuperación de contraseñas, validación de accesos mediante redes sociales.
- PedidoController: Responsable del ciclo de vida del pedido personalizado (creación, actualización de estados y gestión general).
- DiseñoController: Encargado de la personalización detallada de prendas, generación de vistas previas en 3D, y almacenamiento de ajustes específicos.
- ComunicacionController: Administra mensajes y archivos adjuntos entre usuarios para mejorar la comunicación y evitar malentendidos.
- TrackingController: Gestiona actualizaciones sobre el estado del pedido, seguimiento en tiempo real y proporciona claridad en tiempos estimados.
- IAController: Utiliza algoritmos inteligentes para recomendar diseños de prendas personalizadas basadas en perfiles corporales específicos.
- PedidoController: Centraliza la creación, modificación y administración general de los pedidos personalizados realizados por clientes.
- PagoController: Gestiona transacciones financieras y asegura cumplimiento legal con normativas de facturación (SAT Guatemala).

Entidades (Modelo)

Este paquete agrupa las clases que representan los datos principales del sistema, incluyendo la información de usuarios, prendas, pedidos, comunicación y pagos. Estas clases interactúan con la base de datos directamente, correspondiendo a las tablas relacionales definidas previamente. Contiene:

- Usuario: Almacena información básica y detalles de roles (clientes, diseñadores, administradores).
- PerfilMedidas: Representa diferentes perfiles con medidas específicas para personalizar prendas de clientes individuales.
- PlantillaPrenda: Modelos base de prendas que pueden ser personalizados por los usuarios.
- Material: Información sobre los materiales y telas utilizados en las prendas.
- PedidoPersonalizado: Información completa de pedidos personalizados realizados por clientes, incluyendo ajustes, materiales, colores y estado.
- Visualizacion3D: Registros sobre representaciones 3D generadas por el sistema para previsualización de prendas.
- HistorialEstado: Histórico detallado de cambios de estado en los pedidos para seguimiento y transparencia.
- Mensaje: Mensajes intercambiados entre clientes y diseñadores para facilitar la comunicación sobre los pedidos.
- ArchivoAdjunto: Archivos adicionales que acompañan a mensajes entre usuarios.

- ValoracionPedido: Comentarios y calificaciones finales de los clientes sobre los pedidos finalizados.
- TransaccionPago: Registro detallado de pagos realizados, asegurando cumplimiento fiscal y transparencia contable.

Relaciones entre Paquetes

- Interfaces (Vista) dependen directamente de las Controladoras, ya que invocan sus métodos para responder a las acciones de usuario.
- Controladoras interactúan directamente con Entidades (datos del Modelo) para acceder, modificar y persistir información en la base de datos.

Persistencia de Datos

Diagrama de Clases Persistentes

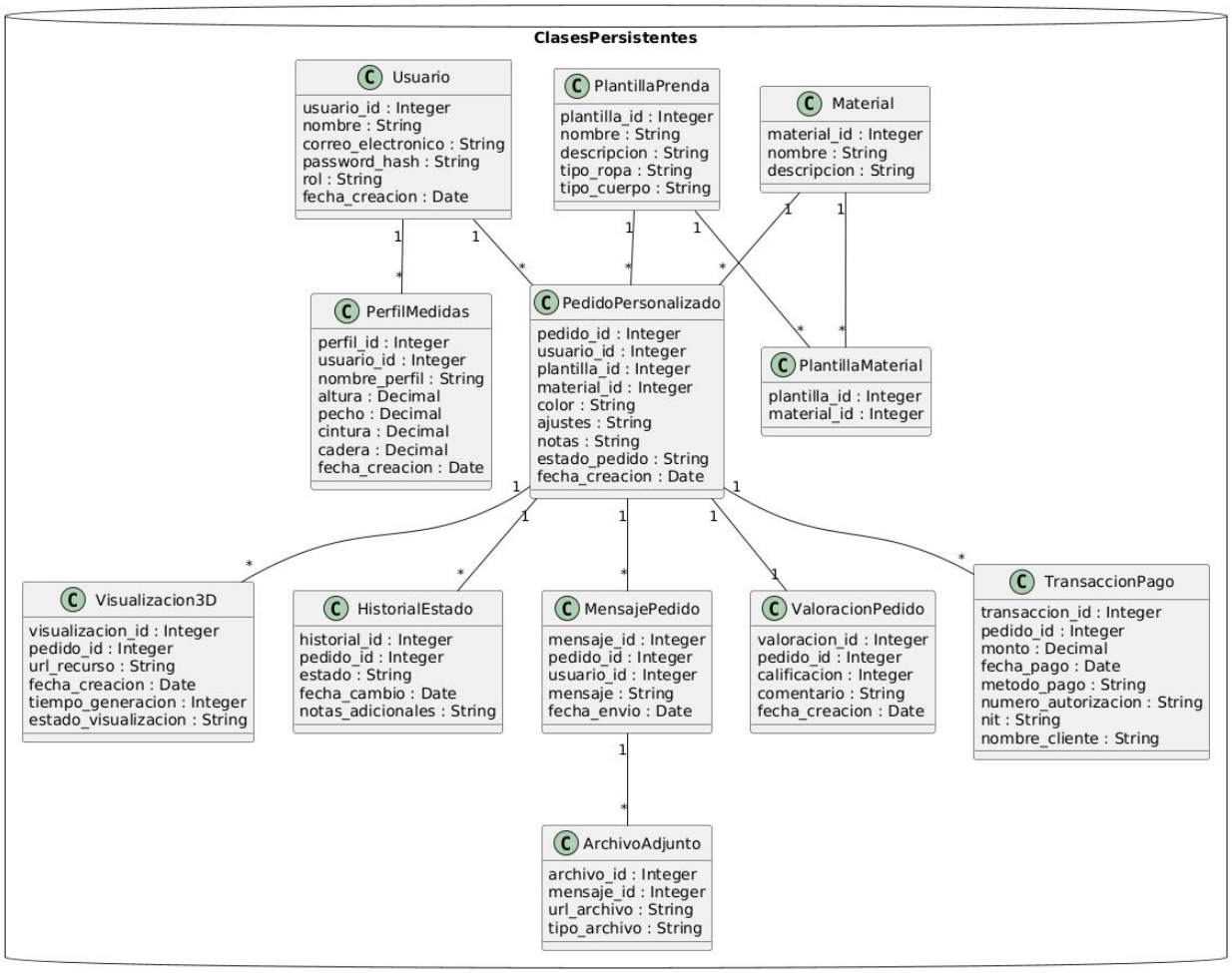
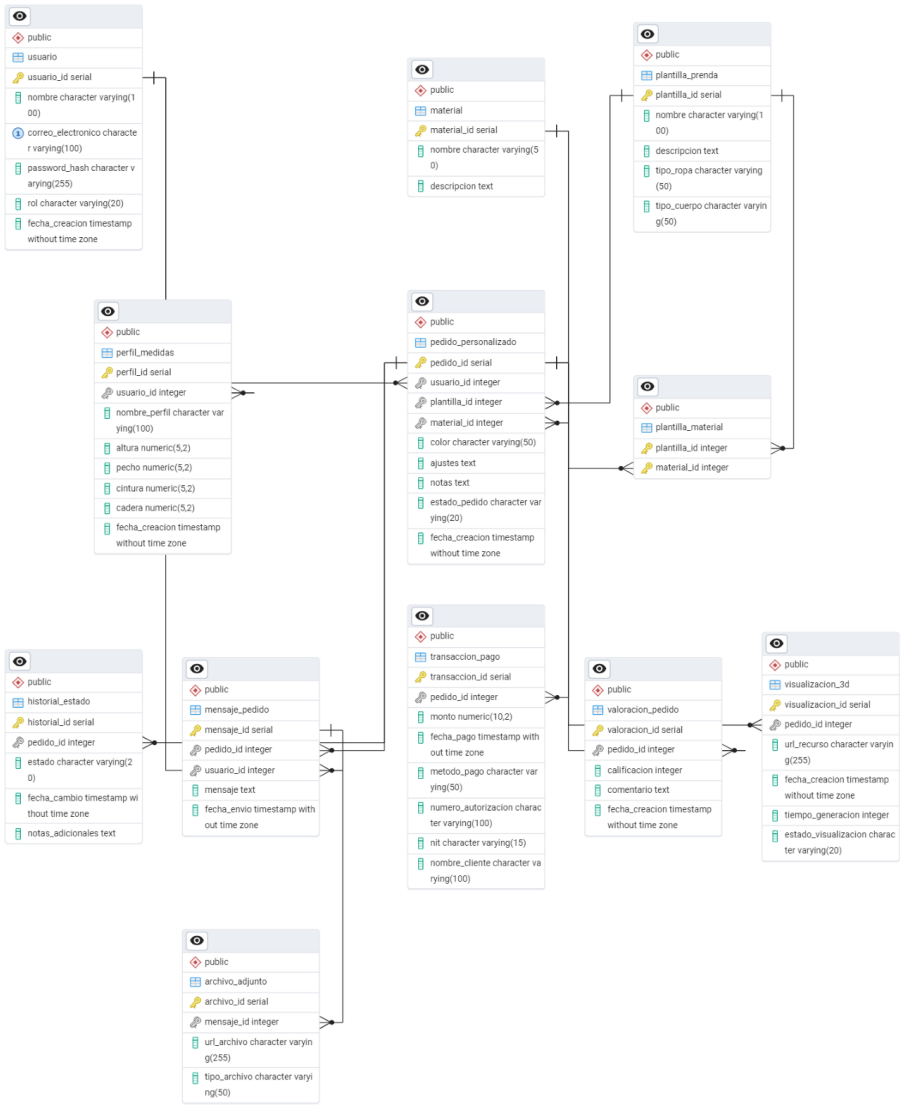


Diagrama Entidad Relación



Estimaciones Requisitos No Funcionales

Confiabilidad

1. El sistema debe garantizar una disponibilidad mínima del 99% en horario laboral para diseñadores y clientes. Se monitoreará el tiempo de inactividad semanal y se calculará el porcentaje de disponibilidad.

2. En caso de falla del servidor o pérdida de conectividad, el sistema debe reestablecer sus servicios críticos (carga de proyectos, modelado 3D y acceso a catálogos) en un máximo de 5 minutos. Se realizarán simulaciones de caída de servidor y se medirá el tiempo hasta restablecer la funcionalidad esencial.

Dimensiones:

- Infraestructura: Requiere servidores de alta disponibilidad, balanceadores de carga, backups automáticos, y monitoreo constante.
- Herramientas necesarias: Servicios en la nube como AWS EC2, Azure App Services, o Google Cloud. Sistemas de monitoreo como Datadog o New Relic.
- Escala en infraestructura: Dos servidores activos para redundancia + al menos un sistema de recuperación de emergencia (backup en caliente).
- Tiempo estimado de implementación: 2-3 meses para configurar, probar y validar la disponibilidad y tiempos de restauración en simulaciones.

Usabilidad

3. La plataforma debe contar con una interfaz intuitiva. Al realizar pruebas de usabilidad con al menos cinco usuarios representativos, la tasa de errores que impidan completar una tarea durante la navegación inicial no debe superar el 10%. Se medirá mediante heurísticas de usabilidad y conteo de errores críticos.

Dimensiones:

- Pruebas: Realizar al menos 2 ciclos de pruebas con grupos de 5-10 usuarios representativos.
- Tiempo dedicado: 2-3 semanas por ciclo de prueba incluyendo ajustes con base en el feedback.
- Herramientas necesarias: Prototipos interactivos (usando Figma o Adobe XD) y herramientas de pruebas de usabilidad como Hotjar o UserTesting.
- Recursos humanos: Diseñador UI/UX + moderadores para supervisar las pruebas.
- Escala del proceso: Al menos 10 sesiones de prueba, ajustes y validaciones finales.

Portabilidad

4. El sistema debe poder trasladarse y ejecutarse en diferentes entornos, incluyendo Windows, macOS y Linux. Se realizarán pruebas en distintos entornos de ejecución, asegurando que al menos el 90% de los entornos probados permitan la instalación y ejecución sin modificaciones significativas.

Dimensiones:

- Entornos de prueba: Versiones recientes de Windows 10/11, macOS Ventura/Monterey, y Ubuntu 20.04 o Fedora.
- Herramientas: Implementación multiplataforma con Docker o entornos virtualizados para pruebas.
- Tiempo dedicado: 3 semanas para pruebas en los tres sistemas operativos, ajustando dependencias según el entorno.
- Escala: Soporte probado en 90% de los entornos clave con configuraciones estándar.

Cumplimiento Legal y Normativo

5. La generación de facturas electrónicas debe cumplir con las normativas SAT de Guatemala. Se validará automáticamente con el sistema SAT, realizando pruebas de certificación, auditorías trimestrales sin incumplimientos y asegurando el almacenamiento de facturas por 5 años.

6. Toda imagen o plantilla utilizada en la plataforma debe contar con permisos y licencias adecuadas (libre uso, creative commons o contrato comercial). El 100% de los recursos deberán tener registros de fuente y tipo de licencia en el repositorio documental. Se auditarán los catálogos trimestralmente.

Dimensiones:

- Integración SAT: 2-3 semanas para desarrollo e integración de sistemas de facturación compatibles con la API del SAT.
- Auditorías de imágenes: Gestión de repositorios documentados con herramientas como Google Drive o sistemas de archivo dedicados (e.g., DocuWare).
- Tiempo dedicado: 1 semana por auditoría trimestral; hasta 2 meses iniciales para establecer estándares y procesos de licencia.
- Escala del proceso: Certificaciones iniciales y revisiones periódicas para mantener la conformidad.

Escalabilidad

7. El sistema debe manejar incrementos en la cantidad de usuarios sin degradar el rendimiento. Se realizarán pruebas incrementando progresivamente la carga de usuarios y monitoreando el rendimiento, asegurando que el sistema funcione correctamente con hasta 10 veces la carga esperada.

Dimensiones:

- Entornos de prueba: Versiones recientes de Windows 10/11, macOS Ventura/Monterey, y Ubuntu 20.04 o Fedora.
- Herramientas: Implementación multiplataforma con Docker o entornos virtualizados para pruebas.
- Tiempo dedicado: 3 semanas para pruebas en los tres sistemas operativos, ajustando dependencias según el entorno.
- Escala: Soporte probado en 90% de los entornos clave con configuraciones estándar.

Ética

8. El sistema de pago con tarjeta de débito/crédito no debe almacenar datos ingresados por el usuario. Se analizarán logs y bases de datos para detectar almacenamiento no autorizado, se realizarán pruebas de penetración para identificar vulnerabilidades y se certificarán los sistemas sin persistencia de información sensible post transacción.

Dimensiones:

- Integración con pasarelas de pago: Uso de Stripe, PayPal o una alternativa que maneje la seguridad del lado del proveedor.
- Pruebas: Pruebas de penetración trimestrales con herramientas como OWASP ZAP o Burp Suite para garantizar que no se almacenan datos sensibles.
- Tiempo dedicado: 2 semanas para configurar integraciones y pruebas iniciales.
- Escala del proceso: Auditorías de seguridad continuas cada trimestre.

Rendimiento

9. La plataforma debe generar la vista previa 3D de un diseño (cuerpo + prenda) en un máximo de 4 segundos para plantillas de complejidad media (moldes con hasta 3 piezas principales). Se medirán tiempos promedio de renderizado en pruebas de carga con 10 usuarios concurrentes.

Dimensiones:

- Motor de renderizado: Herramientas como Three.js o Babylon.js optimizadas para modelos ligeros.
- Hardware del servidor: GPU o servicios de renderizado en la nube como AWS EC2 con GPU integrada.
- Pruebas: Pruebas de tiempo con cargas de 10, 50 y 100 usuarios concurrentes.
- Tiempo dedicado: 4-8 semanas para implementación y optimización del renderizado 3D.
- Escala del proceso: Capacidad para manejar plantillas de 3 piezas principales en cargas medias.

Mantenibilidad

10. El sistema debe permitir agregar o actualizar una plantilla o molde nuevo en un tiempo máximo de 3 horas, empleando procedimientos documentados. Se medirá el tiempo requerido para que un administrador capacitado complete el proceso y valide su correcto funcionamiento.

Dimensiones:

- Procedimientos: Documentación clara y entrenamiento para administradores.
- Herramientas: Interfaces intuitivas de administración como Django Admin o paneles personalizados.
- Tiempo dedicado: 1 mes inicial para diseñar el flujo de gestión y documentarlo.
- Escala del proceso: Proceso simplificado con pasos claros para cambios regulares.

CORTE 3: Selección de Stack de trabajo

Selección de Tecnología Frontend

Tecnologías Consideradas

Para la implementación del frontend, se han evaluado las siguientes opciones:

- Angular: Es un framework desarrollado por Google que utiliza TypeScript y sigue una arquitectura basada en componentes (Google Developers, 2023).
  Ventajas:
  - Ofrece una solución completa con herramientas integradas para manejo de estado, enrutamiento, validación de formularios y más.
  - Posee una estructura bien definida que facilita el desarrollo de aplicaciones grandes y complejas.

  Desventajas:
  - Tiene una curva de aprendizaje alta, lo que puede ralentizar el desarrollo inicial.
  - El binding bidireccional, si bien es potente, puede afectar el rendimiento en aplicaciones muy interactivas o de alta carga.
- Vue.js:
  Es un framework progresivo y ligero que se destaca por su sistema reactivo y su facilidad de integración en proyectos existentes (You, 2023).

  Ventajas:

  - Su sintaxis es sencilla y fácil de aprender, lo que permite un desarrollo rápido y eficiente.
  - Su ecosistema flexible facilita la integración con otras bibliotecas o herramientas.

  Desventajas:

  - La comunidad, aunque en crecimiento, es menor en comparación con Angular o React, lo que puede limitar el soporte y la disponibilidad de recursos en proyectos de gran envergadura.
  - En aplicaciones muy grandes, sin una arquitectura bien definida, pueden surgir problemas de escalabilidad y mantenimiento.
- React.js:
  Es una biblioteca desarrollada por Meta (Facebook) para la creación de interfaces de usuario, basada en componentes y en el uso del Virtual DOM (Meta Open Source, 2023).
  Ventajas:
  - Altamente flexible y eficiente en el renderizado, permitiendo una excelente experiencia de usuario.
  - Cuenta con un extenso ecosistema de herramientas (por ejemplo, Redux para gestión del estado) y una gran comunidad, lo que facilita encontrar soluciones a problemas comunes.

Desventajas:

- Al ser solo la capa de vista, requiere la integración con otras librerías para lograr funcionalidades completas como el enrutamiento o el manejo de datos en el lado del servidor, lo que puede incrementar la complejidad de la configuración inicial.

- Next.js:
  Es un framework construido sobre React que incorpora características avanzadas como renderizado del lado del servidor (SSR), generación de sitios estáticos (SSG) y soporte para API routes (Vercel, 2023).

  Ventajas:

  - Optimiza el rendimiento y la carga inicial gracias a sus técnicas de pre-renderizado, lo cual es crucial para cumplir con requisitos de rapidez en la visualización (por ejemplo, vistas 3D en menos de 4 segundos).
  - Mejora la escalabilidad y disponibilidad, facilitando la replicación y distribución de la información en la nube.
  - Aumenta la visibilidad en motores de búsqueda (SEO) gracias a su capacidad de generar contenido pre-renderizado.

  Desventajas:

  - Puede introducir una complejidad adicional en la configuración inicial, especialmente para desarrolladores menos familiarizados con conceptos como SSR y SSG.
  - Requiere una buena planificación en el manejo de rutas y datos, aunque esta complejidad se compensa con las ventajas en rendimiento y escalabilidad.

Evaluación Basada en Características del Framework y Estimaciones de los Requisitos

El proyecto debe cumplir con los siguientes criterios clave:

2.1 Escalabilidad

El sistema debe manejar un incremento en la cantidad de usuarios sin degradar el rendimiento. Next.js ofrece SSR y SSG, lo que optimiza la carga de datos y mejora la escalabilidad sin recargar el servidor.

2.2 Rendimiento

La plataforma debe generar vistas previas en 3D en un tiempo máximo de 4 segundos. Next.js mejora el rendimiento mediante su capacidad de pre-renderizado, reduciendo el tiempo de carga en comparación con soluciones SPA como Vue.js o React sin SSR.

2.3 Confiabilidad y Disponibilidad

El sistema debe garantizar una disponibilidad del 99% en horario laboral. Next.js facilita la replicación de información y recuperación rápida con su arquitectura optimizada para distribución en la nube.

2.4 Mantenibilidad

El sistema debe permitir agregar o actualizar plantillas en un tiempo máximo de 3 horas. La modularidad de React y Next.js permite una estructura escalable y de fácil mantenimiento.

2.5 Usabilidad y Portabilidad

Se requiere una interfaz intuitiva accesible desde diferentes entornos. React y Next.js permiten crear interfaces responsivas optimizadas para web y móvil.

Comparación de Tecnologías

Frontend

ANGULAR	VUE.JS	VS	REACT.JS	NEXT.JS
Framework completo	Framework progresivo	TIPO	Biblioteca de UI	Framework basado en React
Open Source (MIT)	Open Source (MIT)	LICENCIA	Open Source (MIT)	Open Source (MIT)
Arquitectura MVC integral y modular	Flexible, fácil integración y sintaxis sencilla	CARACTERÍSTICAS CLAVE	Componentes reutilizables y virtual DOM	SSR/SSG integrados y enrutamiento simplificado
Eficiente en aplicaciones empresariales, con cierta sobrecarga	Alto rendimiento en proyectos medianos	RENDIMIENTO	Muy bueno, optimizado a nivel de virtual DOM	Excelente rendimiento, ideal para carga rápida y SEO
Alta	Moderada	ESCALABILIDAD	Muy Alta	Óptima
Buena	Buena	SOPORTE Y COMUNIDAD	Excelente, Soporte de Facebook	Excelente
TypeScript	JavaScript/TypeScript	LENGUAJE BASE	JavaScript (JSX)/TypeScript	JavaScript/TypeScript

Justificación de la Elección

La combinación de React.js y Next.js se destaca frente a las demás tecnologías evaluadas por las siguientes razones:

1. Ventajas en Rendimiento y Escalabilidad:
    a. Next.js aprovecha el renderizado del lado del servidor y la generación estática, lo que permite tiempos de carga significativamente menores. Esto es crucial para cumplir con los exigentes tiempos de respuesta (menos de 4 segundos para vistas 3D) y para gestionar incrementos en la carga de usuarios sin afectar la experiencia del usuario.
    b. Frente a Angular y Vue.js, que dependen principalmente del procesamiento en el cliente, la arquitectura de Next.js ofrece una mejor gestión de recursos en aplicaciones de gran escala.
2. Flexibilidad y Ecosistema:
    a. React.js brinda una gran flexibilidad para construir interfaces de usuario a través de un sistema basado en componentes, lo que facilita el mantenimiento y la ampliación del sistema.
    b. La extensa comunidad y el ecosistema robusto de React permiten la integración de librerías y herramientas complementarias (por ejemplo, para gestión del estado o enrutamiento), superando a las soluciones que pueden resultar menos modulares o más rígidas en su estructura.
3. Optimización SEO y Experiencia de Usuario:
    a. La capacidad de Next.js para generar contenido pre-renderizado mejora la indexación en motores de búsqueda, lo cual es un valor agregado para cualquier plataforma digital.
    b. La experiencia de usuario se beneficia de interfaces interactivas y rápidas, esenciales para la personalización en tiempo real, algo en lo que React brilla gracias a su Virtual DOM.
4. Mantenibilidad y Desarrollo Ágil:
    a. La estructura modular de React junto con las características avanzadas de Next.js permite que nuevas funcionalidades y actualizaciones (como la incorporación de nuevas plantillas de diseño) se realicen de forma ágil y con menos riesgo de introducir errores, a diferencia de Angular, cuya complejidad puede ralentizar iteraciones de desarrollo.
    b. Aunque Vue.js presenta una curva de aprendizaje baja y buena flexibilidad, en comparación, la combinación de React y Next.js ofrece una solución más robusta y escalable para proyectos de alto rendimiento y disponibilidad.

Selección de Tecnología Backend

Tecnologías Consideradas

Para la implementación del backend se han evaluado las siguientes opciones, considerando aspectos como lenguaje, arquitectura, rendimiento, escalabilidad, facilidad de uso, seguridad y casos de uso:

A) Django:

Es un framework de alto nivel escrito en Python, diseñado para el desarrollo rápido y seguro de aplicaciones web (Holovaty & Kaplan-Moss, 2023). Ventajas:

- Desarrollo rápido y sencillo gracias a la filosofía “baterías incluidas”.

- Código limpio y mantenible, favorecido por la claridad y legibilidad de Python.
- Medidas de seguridad robustas integradas por defecto (protección contra CSRF, XSS, inyección SQL, etc.).
- Amplio ecosistema y documentación, lo que facilita la solución de problemas y la incorporación de nuevas funcionalidades.
- Escalabilidad adecuada para aplicaciones de mediana a gran escala, con buenas opciones de integración para caché y despliegue en la nube.

Desventajas:

- Puede requerir optimizaciones adicionales en escenarios de alta concurrencia.
- En proyectos extremadamente complejos, la estructura "baterías incluidas" puede resultar menos flexible en comparación con soluciones más modulares.

B) Ruby on Rails:
Es un framework basado en Ruby que enfatiza la convención sobre la configuración y la rapidez en el desarrollo.
Ventajas:

- Filosofía "convención sobre configuración" que acelera el desarrollo inicial.
- Código conciso y fácil de entender, lo que puede mejorar la productividad.
- Buenas medidas de seguridad integradas y una arquitectura que favorece la organización del código.

Desventajas:

- Desafíos en escalabilidad para aplicaciones de gran envergadura, requiriendo ajustes y estrategias adicionales.
- Menor flexibilidad en personalizaciones complejas debido a la fuerte dependencia en convenciones predefinidas.

C) Laravel:
Es un framework de PHP que se destaca por su elegante sintaxis y su enfoque en la simplicidad y la productividad (Otwell, 2023).
Ventajas:

- Sintaxis elegante y moderna que permite un desarrollo ágil.
- Gran comunidad y abundante documentación que facilitan el aprendizaje y la resolución de incidencias.
- Integración de múltiples herramientas que ayudan en tareas comunes como autenticación, enrutamiento y manejo de bases de datos.

Desventajas:

- Puede presentar limitaciones de rendimiento en escenarios de alta concurrencia si no se implementan correctamente estrategias de optimización y caché.
- La flexibilidad y potencia de PHP pueden depender en gran medida de la experiencia del desarrollador, lo que puede afectar la robustez del sistema.

D) Spring Boot (o Spring Root):
Es un framework basado en Java que simplifica la creación de aplicaciones independientes y productivas (Walls, 2022).
Ventajas:
- Altísimo rendimiento y solidez, especialmente en entornos empresariales de alta carga.
- Excelente escalabilidad, ideal para arquitecturas basadas en microservicios y aplicaciones de misión crítica.
- Amplias características de seguridad integradas, probadas en numerosos entornos críticos.

Desventajas:
- Curva de aprendizaje más pronunciada, debido a la complejidad inherente del ecosistema Java.
- Requiere una mayor inversión inicial en configuración y despliegue, lo que puede ralentizar el desarrollo en fases tempranas.

Evaluación Basada en Características del Framework y Estimaciones de los Requisitos

Para tomar una decisión informada, se han considerado los siguientes aspectos:

- Lenguaje: Python (Django) se destaca por su simplicidad y legibilidad, lo que favorece el mantenimiento y la colaboración.
- Arquitectura: Django utiliza el patrón MVT, ofreciendo una estructura clara y modular, mientras que las otras opciones utilizan MVC o arquitecturas basadas en microservicios.
- Rendimiento: Aunque Spring Boot ofrece un rendimiento excelente en entornos empresariales, Django se adapta de forma equilibrada a proyectos de mediana y gran escala sin complejidades excesivas.
- Escalabilidad: Django, con sus capacidades de caché y soporte para despliegue en la nube, permite escalar la aplicación de forma eficiente.
- Facilidad de Uso: La curva de aprendizaje de Django es menor en comparación con Spring Boot y ofrece una sintaxis limpia y una gran cantidad de recursos y documentación.
- Seguridad: Django incorpora de forma nativa medidas de seguridad robustas, lo que es esencial para proteger la información sensible de la plataforma.
- Casos de Uso: Para una plataforma que requiere rapidez en el desarrollo, mantenimiento ágil y alta seguridad, Django se alinea perfectamente con los requisitos del proyecto.

Comparación de Tecnologías

CRITERIO	DJANGO	RUBY ON RAILS	LARAVEL	SPRING BOOT
Lenguaje	Python	Ruby	PHP	Java
Arquitectura	Model-Template-View (MTV)	Model-View-Controller (MVC)	Model-View-Controller (MVC)	Model-View-Controller (MVC)
Rendimiento	Adecuado para aplicaciones con lógica compleja, pero requiere optimización para alto tráfico	Rendimiento decente, pero Ruby puede ser más lento en comparación con otros frameworks debido a su naturaleza interpretada	Rendimiento moderado, adecuado para aplicaciones pequeñas y medianas	Alto rendimiento, especialmente para aplicaciones empresariales complejas
Escalabilidad	Esscalable, pero puede requerir optimización para manejar grandes cargas	Escalable para proyectos medianos, pero puede ser un desafío en aplicaciones muy grandes o de tráfico intenso	Escalable, pero no tan eficiente como Django o Spring Boot para grandes cargas	Muy escalable, diseñado para apliaciones empresariales de gran escala
Facilidad de uso	Fácil de aprender y usar, especialmente para principiantes en desarrollo web	Intuitivo para desarrolladores familiarizado con Ruby, excelente para construir prototipos rápidamente	Fácil de usar para desarrolladores familiarizados con PHP	Más complejo, requiere experiencia en Java y su ecosistema
Seguridad	Alto nivel de seguridad con características integradas como protección contra CSFR, XSS y SQL Injection	Ofrece medidas de seguridad integradas, pero necesita configuraciones adicionales para ciertos ataques	Proporciona características de seguridad integradas, pero requiere configuración adicional	Muy seguro, con soporte para estándares empresariales de seguridad
Casos de Uso	Ideal para aplicaciones con lógica compleja, paneles de administración y APIs REST	Ideal para startups y proyectos donde la velocidad de desarrollo es clave	Adecuado para aplicaciones pequeñas y medianas, como blogs y sitios web	Perfecto para aplicaciones empresariales y sistemas distribuidos

Justificación de la Elección

Tras evaluar las ventajas y desventajas de cada tecnología en función de los criterios establecidos, se opta por Django por las siguientes razones:

- Lenguaje y Facilidad de Uso: Python es ampliamente reconocido por su simplicidad y legibilidad, lo que facilita el mantenimiento del código y permite que equipos de desarrollo colaboren de forma eficiente. Django, al estar basado en Python, aprovecha estas ventajas, reduciendo la curva de aprendizaje y acelerando el proceso de desarrollo.
- Arquitectura y Mantenibilidad: El patrón Model-View-Template (MVT) que utiliza Django ofrece una clara separación de responsabilidades, lo que facilita la organización y el mantenimiento del proyecto a largo plazo. Esta modularidad permite implementar y actualizar funcionalidades (como la integración de nuevas herramientas o módulos de seguridad) de manera ágil.
- Rendimiento y Escalabilidad: Django ofrece un rendimiento sólido adecuado para aplicaciones de mediana a gran escala, y su capacidad para integrarse con sistemas de caché y despliegue en la nube favorece una escalabilidad eficaz.

  Aunque frameworks como Spring Boot pueden ofrecer un rendimiento superior en entornos empresariales, la complejidad adicional y la mayor curva de aprendizaje no se justifican en este contexto, donde la agilidad y facilidad de desarrollo son prioritarias.

- Seguridad: La seguridad es un aspecto crítico en esta plataforma, y Django incorpora de forma nativa medidas contra vulnerabilidades comunes (como CSRF, XSS e inyección SQL), lo que garantiza una protección robusta sin necesidad de configuraciones excesivas.
- Casos de Uso y Flexibilidad: Django se adapta muy bien a los requerimientos de una plataforma que necesita desarrollo rápido, alta fiabilidad y facilidad para implementar cambios en función del crecimiento del sistema y las necesidades del negocio.
  Su amplio ecosistema y comunidad aseguran que existen soluciones y recursos para prácticamente cualquier necesidad que surja durante el ciclo de vida del proyecto.

Selección de Tecnología Persistencia de Datos

Tecnologías Consideradas

Para la persistencia de datos se han evaluado las siguientes opciones, analizando aspectos clave como integridad de datos, rendimiento, escalabilidad, facilidad de uso, seguridad y casos de uso:

A) PostgreSQL:

Es un sistema de gestión de bases de datos relacional (RDBMS) open-source que destaca por su robustez y capacidades avanzadas.

Ventajas:

◊ Alta integridad de datos gracias a las transacciones ACID.
◊ Excelente soporte para tipos de datos complejos (JSON, arreglos, tipos geométricos).
◊ Alto rendimiento mediante el uso de índices avanzados y optimización para consultas complejas.
◊ Gran capacidad de escalabilidad vertical y moderada escalabilidad horizontal.
◊ Amplia comunidad y abundante documentación disponible.
◊ Herramientas avanzadas, como triggers, funciones y procedimientos almacenados.

Desventajas:

◊ Requiere una configuración y mantenimiento adecuados para obtener el mejor rendimiento.
◊ Puede resultar inicialmente más complejo en comparación con soluciones más sencillas como SQLite.

B) MySQL:

Es uno de los sistemas de gestión de bases de datos más populares, ampliamente utilizado en proyectos web.

Ventajas:

◊ Excelente rendimiento en transacciones básicas y rápidas.
◊ Gran facilidad de configuración y administración.
◊ Herramientas robustas disponibles (MySQL Workbench, phpMyAdmin).
◊ Buena documentación y soporte de la comunidad.
◊ Uso extendido en múltiples proyectos web.

Desventajas:

◊ Menor soporte para tipos de datos avanzados y estructuras complejas.
◊ Escalabilidad limitada en consultas muy complejas.
◊ Menor flexibilidad comparado con PostgreSQL en consultas avanzadas.

◊ Algunas funcionalidades avanzadas pueden requerir versiones Enterprise o plugins comerciales.

C) SQLite:

Es un motor de bases de datos ligero que se integra directamente en la aplicación.

Ventajas:

◊ Muy ligera y sencilla de implementar.
◊ Ideal para prototipos rápidos o aplicaciones móviles/desktop de pequeño tamaño.
◊ No requiere infraestructura compleja.

Desventajas:

◊ Limitaciones importantes en rendimiento con grandes volúmenes de datos o usuarios concurrentes.
◊ Poco recomendable para aplicaciones con múltiples usuarios simultáneos.
◊ Baja escalabilidad y menor integridad en comparación con sistemas más robustos.

D) SQL Server (Relational):

Es una solución de bases de datos desarrollada por Microsoft, reconocida por su robustez en entornos empresariales.

Ventajas:

◊ Rendimiento excepcional en transacciones complejas y manejo de alto volumen de datos.
◊ Excelente integridad de datos mediante soporte completo de transacciones (ACID).
◊ Soporte robusto para tipos de datos complejos (JSON, XML).
◊ Herramientas gráficas completas (SQL Server Management Studio).
◊ Alta integración con tecnologías del ecosistema Microsoft (Azure, Power BI, .NET).
◊ Muy buena seguridad con auditorías avanzadas y controles estrictos de acceso.

Desventajas:

◊ Altos costos asociados a licencias comerciales, especialmente en funcionalidades avanzadas.
◊ Limitada escalabilidad horizontal sin configuraciones especializadas.
◊ Complejidad inicial y de mantenimiento relativamente alta.
◊ Dependencia del entorno Microsoft, lo que puede limitar la integración con soluciones open-source.

Evaluación Basada en Características del Framework y Estimaciones de los Requisitos

Para tomar una decisión informada, se han analizado los siguientes aspectos:

- Integridad de Datos:
  PostgreSQL y SQL Server sobresalen gracias a su robusto cumplimiento de transacciones ACID, mientras que MySQL y SQLite pueden presentar limitaciones en escenarios de datos complejos.
- Rendimiento:
  PostgreSQL ofrece un rendimiento alto en consultas complejas mediante el uso de índices avanzados, en tanto que MySQL brinda un excelente desempeño en transacciones básicas. SQLite es ideal para aplicaciones ligeras, y SQL Server se orienta a entornos de alta demanda, aunque con mayor costo y complejidad.

- Escalabilidad:
  PostgreSQL permite escalabilidad vertical y moderada horizontal de manera efectiva. MySQL puede escalar en ciertos escenarios, pero con limitaciones en consultas complejas. SQLite está pensado para cargas pequeñas, mientras que SQL Server destaca en entornos empresariales, aunque su configuración para escalabilidad horizontal puede ser compleja y costosa.
- Facilidad de Uso:
  MySQL y SQLite son conocidos por su simplicidad en la configuración inicial. PostgreSQL, aunque requiere una configuración cuidadosa para optimizar su rendimiento, cuenta con una documentación extensa. SQL Server puede ser más complejo y costoso de administrar, y su dependencia del ecosistema Microsoft añade otra capa de complejidad.
- Seguridad:
  Tanto PostgreSQL como SQL Server ofrecen robustas medidas de seguridad nativas. MySQL proporciona seguridad adecuada para muchas aplicaciones, mientras que SQLite, por ser un motor embebido, puede no ser la opción ideal para aplicaciones con alta concurrencia y necesidades avanzadas de seguridad.
- Casos de Uso:
  PostgreSQL es ideal para aplicaciones que requieren una alta integridad de datos, flexibilidad en consultas avanzadas y un manejo eficiente de datos complejos. MySQL es excelente para proyectos web de transacciones rápidas y fáciles de administrar, y SQLite es preferible para prototipos o aplicaciones de baja demanda. SQL Server es adecuado para sistemas empresariales críticos donde el costo no es una limitación.

Comparación de Tecnologías

Persistencia Datos

POSTGRESQL	MYSQL	VS	SQL SERVER	SQLITE
Relacional	Relacional	TIPO	Relacional	Relacional Embebida
Open Source	Open Source y Comercial	LICENCIA	Comercial	Open Source
Excelente	Buena	INTEGRIDAD REFERENCIAL	Excelente	Básica
Vertical: Alta Horizontal: Moderada	Vertical: Buena Horizontal: Baja	ESCALABILIDAD	Vertical: Alta Horizontal: Moderada	Vertical: Limitada Horizontal: Limitada
Alta	Moderada	COMPLEJIDAD DATOS	Alta	Baja
Excelente	Bueno	RENDIMIENTO CONSULTAS	Excelente	Moderado
• Linux • Windows • MacOS	Principalmente • Windows • MacOS	COMPATIBILIDAD MULTIPLATAFORMA	Principalmente • Windows	• Linux • Windows • MacOS
Excelente	Excelente	COMUNIDAD Y DOCUMENTACION	Buena	Excelente

Justificación de la Elección

Tras evaluar las ventajas y desventajas de cada tecnología en función de los criterios establecidos, se opta por PostgreSQL por las siguientes razones:

- Integridad y Flexibilidad en el Manejo de Datos: PostgreSQL destaca por su cumplimiento robusto de transacciones ACID, lo que garantiza una alta integridad de los datos. Su excelente soporte para tipos de datos complejos (como JSON y arreglos) permite manejar de forma eficiente la diversidad de información que requiere la plataforma.
- Rendimiento y Optimización: Gracias al uso de índices avanzados y optimización para consultas complejas, PostgreSQL ofrece un rendimiento sobresaliente en escenarios donde se requiere procesamiento intensivo de datos. Esto resulta fundamental para una plataforma que necesita manejar múltiples usuarios y consultas avanzadas sin comprometer la velocidad y la eficiencia.
- Escalabilidad y Adaptabilidad: La capacidad de PostgreSQL para escalar verticalmente y de forma moderada horizontal permite adaptarse al crecimiento del sistema, asegurando que la base de datos pueda evolucionar conforme aumente la carga y la complejidad de las consultas. Si bien SQL Server ofrece una escalabilidad excelente, los costos y la complejidad de su configuración no se ajustan tan bien a los requerimientos de una solución escalable y accesible.

- Facilidad de Uso y Comunidad de Soporte: Aunque PostgreSQL requiere una configuración cuidadosa, la abundante documentación y el soporte de una extensa comunidad facilitan la resolución de problemas y la optimización de la base de datos.
  Esto permite que el equipo de desarrollo se enfoque en mejorar la plataforma sin enfrentar barreras significativas de aprendizaje.
- Seguridad y Confiabilidad:
  Las medidas de seguridad integradas en PostgreSQL aseguran que la información sensible esté protegida, lo cual es esencial en una plataforma que maneja datos críticos y transaccionales. Esta característica, combinada con su robustez y fiabilidad, lo posiciona como la opción ideal frente a alternativas que pueden requerir configuraciones adicionales o depender de entornos comerciales.

ACTIVIDAD: Lean UX Canvas

Capturas del proceso

Cuadro 1. ¿Qué problema del negocio estás tratando de resolver?

1. Redacte su problema del negocio actual usando la siguiente plantilla:
[**Nuestro servicio/producto**] fue diseñado para alcanzar [**estas metas**]
Hemos observado [**de esta forma**] que el producto/servicio no está alcanzando estos objetivos por lo que está causando [**estos efectos adversos/problemas de negocio**] al negocio
¿Cómo podríamos mejorar el servicio/producto para que nuestros clientes sean más exitosos determinado por [**estos cambios medibles en el ,comportamiento del cliente**]

Nuestro servicio/producto
DesignBetter, plataforma de diseño y personalización de moda, fue diseñada para permitir a diseñadoras independientes y a sus clientes modelar, simular y personalizar prendas de manera eficiente y realista antes de la confección, optimizando tiempos y reduciendo costes.

Hemos observado
Mediante pruebas de prototipos y feedback de usuarios, que la falta de simulaciones 3D precisas y de recomendaciones automáticas de medidas personalizadas provoca múltiples iteraciones físicas y digitales, alargando los ciclos de diseño, elevando los costes de prototipado y generando retrasos, lo cual disminuye la satisfacción del cliente.

¿Cómo podríamos mejorar
la plataforma para que nuestros clientes sean más exitosos, por ejemplo, reduciendo un 50 % el número de iteraciones de prototipos (medido por revisiones solicitadas) y acortando en un 30 % el tiempo promedio de aprobación de diseño, de modo que aumente la eficiencia del proceso y la satisfacción del usuario?

2.Cuando termine, Copie su nueva declaración del problemas del negocio en su Lean UX Canvas

1

Cuadro 2. ¿Como se resolvió la declaración del negocio o el business statement?

Sugerencia: ¿qué harán las personas/ usuarios de manera diferente si sus soluciones funcionan? Considere las métricas que indican el éxito del cliente, como el valor promedio del pedido, el tiempo en el sitio y la tasa de retención

1) Trabajando individualmente, tome 5 minutos para responder esta pregunta:
¿Qué resultados está buscando su producto?¿Cómo se medirán?
Escriba un resulltado y una forma de medir en cada postit

- Reducir iteraciones de prototipado en un 50%

  Métrica: Nº promedio de revisiones solicitadas por pedido.
- Acortar el tiempo de aprobación de diseño en un 30 %

  Métrica: Tiempo medio (en horas) entre envío de diseño y aprobación final.
- Aumentar la tasa de conversión de diseño a pedido en un 20 %

  Métrica: % de diseños que se convierten en órdenes efectivas.
- Incrementar la retención de usuarios en un 15 %

  Métrica: % de usuarios que vuelven a crear un pedido en el mes siguiente.
- Disminuir el coste de prototipo en un 25 %

  Métrica: Coste medio por prototipo impreso/físico.
- Mejorar la satisfacción del cliente (NPS) en 10 puntos

  Métrica: Net Promoter Score recogido tras cada entrega.

2) Ahora trabajando en equipo, tome 5 minutos:
- Coloca tus notas debajo. Léelo en voz alta al equipo
- Ordénelos en temas que tengan sentido y asigne un nombre a cada tema

- Tema "Eficiencia de diseño"

  - Reducir iteraciones de prototipado
  - Acortar tiempo de aprobación
- Tema "Conversión & Retención"

  - Aumentar tasa de conversión diseño→pedido
  - Incrementar retención de usuarios
- **Tema "Costes & Calidad"**

  - Disminuir coste de prototipo
  - Mejorar NPS

2

3) En 5 minutos, vote por los 3 mejores resultados + métricas en las que cree que su equipo debería centrarse primero. Estos van en la casilla #2 de su Lean UX Canvas.

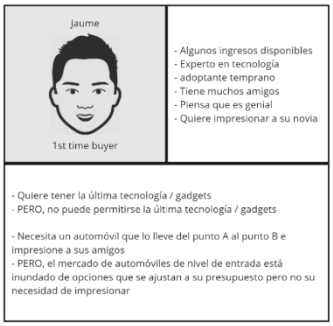
- Reducir iteraciones de prototipado en un 50 % / Nº promedio de revisiones solicitadas por pedido
- Aumentar la tasa de conversión de diseño a pedido en un 20 % / % de diseños que se convierten en órdenes efectivas
- 3. Acortar el tiempo de aprobación de diseño en un 30 % / Tiempo medio entre envío de diseño y aprobación final

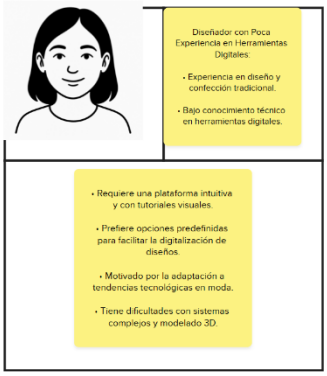
Cuadro 3. Usuarios

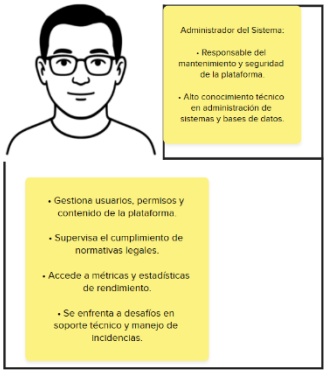
¿En qué tipos (es decir, personas) de usuarios y clientes debe centrarse primero? (Segerencia: ¿Quién compra su producto o servicio?¿Quén lo usa?¿Quién lo configura?

1) Trabaja con tu equipo para crear 2 Proto-Personas que creas que representan parte del público objetivo para tu producto / servicio. Estos deberían ser los segmentos o audiencia a los que se desea apuntar primero.
A la derecha hay un ejemplo de Protopersona y debajo hay dos plantillas de Proto-Persona para que completes.

2) Cuando haya terminado esta información irá al cuadro 3 del Lean UX Canvas







3

Cuadro 4. Resultados (Outcomes) y Beneficios a los usuarios

¿Por qué sus usuarios buscarían su producto o servicio?¿Qué beneficio obtendrían al usuarlo?¿Qué cambio de comportamiento podemos observar que nos dice que han logrado su objetivo?

Sugerencia: Ahorre dinero, obtenga una promoción, pase más tiempo con la familia

1) Comenzando con la sección inferior de tus Proto-Personas, asegúrate de tener tanto los obstáculos como los beneficios. Estos te ayudarán con esta actividad.
2) Declarar al menos 3 resultados de usuario: cambios medibles y valiosos en el comportamiento del usuario.
3) Para cada Resultado, determine el Beneficio asociado preguntando:
- ¿Qué están tratando de lograr por sí mismos?
- ¿Por qué buscarían nuestro producto o servicio?

4) Esta información irá en el Cuadro 4 del Lean UX Canvas

Nuestros usuarios acudirán a DesignBetter porque quieren agilizar y abaratar todo el proceso de diseño y confección de prendas personalizadas, evitando las múltiples idas y vueltas propias de los prototipos físicos tradicionales. Con nuestra plataforma, observaremos tres cambios de comportamiento clave que demuestran el éxito del producto:

1. Menos prototipos físicos: los diseñadores y clientes reducirán en un 50 % la cantidad de muestras impresas que piden antes de dar por válido un diseño.
Beneficio: Ahorran dinero en materiales y mano de obra, y dedican mucho menos tiempo a esperar y revisar prototipos.

2. Aprobaciones ultrarrápidas: al menos el 80 % de los diseños se aprobarán en menos de 24 horas tras subirse a la plataforma.
Beneficio: Reciben sus prendas con mayor rapidez, evitan reuniones o envíos interminables y mejoran su productividad.

3. Menos devoluciones y correcciones: las solicitudes de ajustes o devoluciones por mal encaje caerán un 40 %.
Beneficio: Ganan confianza en que el primer pedido encajará bien, reducen el estrés post-venta y construyen una experiencia de compra más satisfactoria.

4

Cuadro 5. Soluciuones

¿Qué podemos hacer para resolver nuestro problema comercial y satisfacer las necesidades de nuestros clientes al mismo tiempo? Enumere aquí las ideas de producto, características o mejoras.

Recordando nuestro problema comercial, los usuarios que hemos priorizado, sus motivacionees y el comportamiento que nos gustaría vere en ellos ...
1) Trabaja con tu equipo para resolver la siguiente pregunta:
¿Qué solución servirá a nuestras personas y creará los resultados deseados?
2) Cree un dibujo simple, boceto o pizarra para aumentar sus ideas.
3) Estas ideas irán al cuadro 5 del Lean UX Canvas

Para dar respuesta a nuestro problema comercial y cubrir las necesidades de diseñadores y clientes, proponemos estas soluciones de producto y mejoras:

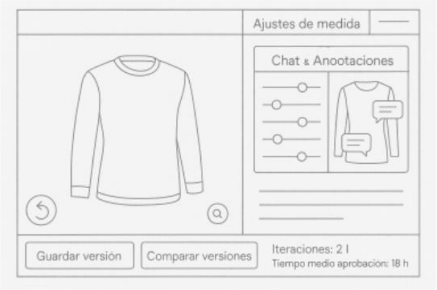
1. Editor 3D interactivo de prendas
Un módulo donde el usuario pueda ver y rotar un modelo tridimensional de su diseño en tiempo real, aplicando materiales y texturas. De este modo se reducen drásticamente los prototipos físicos, pues el simulador refleja el caimento y los volúmenes con alta precisión.

2. Generador de tallas asistido por IA
Un motor que, a partir de medidas básicas (altura, contorno de pecho/cintura/cadera), sugiera un patrón de tallas personalizado. Así evitamos devoluciones por mala adaptación y aceleramos la aprobación: o el cliente acepta la propuesta automática o la ajusta mínimamente.

3. Feedback colaborativo y marcado directo sobre el 3D
Un chat integrado y la posibilidad de dibujar anotaciones ("pinches") sobre el modelo 3D para dejar comentarios precisos (por ejemplo: "subir este hombro 2 cm"). Esto facilita la comunicación entre cliente y diseñador y evita ambigüedades.

4. Historial de versiones con comparador visual
Cada vez que el diseñador o el cliente aprueban un cambio, se guarda una versión en la cronología. Un comparador "antes/después" permite revisar evoluciones y volver a una iteración previa sin perder trabajo.

5. Panel de métricas de prototipado
Un dashboard que muestre en tiempo real los indicadores clave (número de iteraciones, tiempo medio de aprobación, tasa de conversión de diseño a pedido). Gracias a esto, nuestro equipo y el cliente ven al instante si estamos cumpliendo los objetivos de reducción de ciclos y aceleración de aprobaciones.



5

Cuadro 6. Hipótesis

Combine los supuestos de 2,3,4 y 5 en la siguiente declaración de hipótesis:
"Creemos que este [resultado comercial] se logrará si [el usuario] alcanza [beneficio] con [función]".
(Sugerencia: cada hipótesis debe centrarse solo en una característica).

Divide en parejas. Cada pareja escribe una declaración de hipótesis que cree que puede resilver su problema comercial para su persona objetivo

Cosas claves para recordar:

- Ya ha indicado los supuestos que vamos a utilizar para formar el enunciado de hipótesis en los cuadros 2,3,4,5. Recombinarlos en este formato nos ayuda a ver si incluso lo creemos.
- Cada hipótesis debe centrarse solo en una característica / funcionalidad

Luego haga el Canvas de priorización de hipótesis que está más abajo.

We believe that [business outcome] will be achieved if [user] attains [outcomes & benefits] with [feature].

Examples:

We believe that sales of Ford's existing cars will increase if Nadine, the Mr of family sedans at Ford successfully drives more consumers to Ford dealerships with the ability to buy Fiore + C thatbor installed at the dealership

We believe an increase in peak usage of active rides will be achieved if Carlos the a dairy driver attains driving confidence or if the ability to see his grandkids more with 1-tap "Navigate to Sawac Locations"

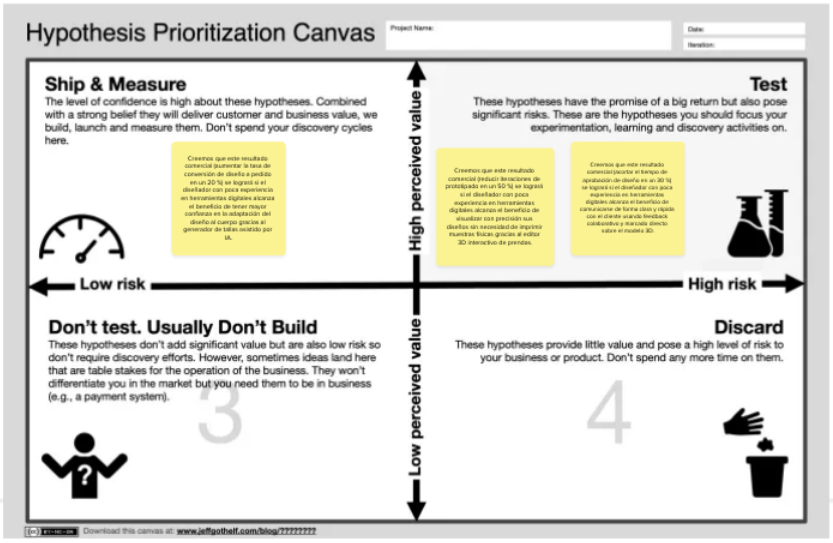
Creemos que este resultado comercial (reducir iteraciones de prototipado en un 50 %) se logrará si el diseñador con poca experiencia en herramientas digitales alcanza el beneficio de visualizar con precisión sus diseños sin necesidad de imprimir muestras físicas gracias al editor 3D interactivo de prendas.

Creemos que este resultado comercial (aumentar la tasa de conversión de diseño a pedido en un 20 %) se logrará si el diseñador con poca experiencia en herramientas digitales alcanza el beneficio de tener mayor confianza en la adaptación del diseño al cuerpo gracias al generador de tallas asistido por IA.

Creemos que este resultado comercial (acortar el tiempo de aprobación de diseño en un 30 %) se logrará si el diseñador con poca experiencia en herramientas digitales alcanza el beneficio de comunicarse de forma clara y rápida con el cliente usando feedback colaborativo y marcado directo sobre el modelo 3D.

6

Cuadro 6. Hipótesis



6

Cuadro 7. ¿Qué es lo más importante que debemos aprender?

Cómo equipo, elija qué hipótesis le gustaría probar primero.
Luego pregunte: ¿Qué es lo más importante que debemos aprender?¿Qué suposición causaría un fracaso total si nos equivocamos?

Creemos que podemos reducir las iteraciones de prototipado en un 50 % si el diseñador con poca experiencia alcanza confianza visualizando su diseño mediante un editor 3D interactivo.

¿Qué debemos aprender primero?

¿Los diseñadores sin experiencia en herramientas digitales pueden confiar en un editor 3D para validar sus diseños sin *necesidad de imprimir una muestra física*?

Si esta suposición es incorrecta, el modelo completo de reducción de costos y tiempos pierde sentido, y la adopción del sistema podría fracasar.

Hipótesis:
Aumentar la tasa de conversión de diseño a pedido en un 20 %
Supuesto: Las medidas básicas son suficientes para lograr un buen ajuste. Si las sugerencias no son suficientemente precisas, aumentan las devoluciones y se pierde credibilidad.

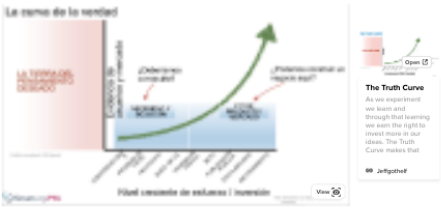
Hipótesis:
Acortar el tiempo de aprobación de diseño en un 30 %
Supuesto: Ambas partes revisan y responden al feedback rápidamente. Si alguno no entra a tiempo o ignora los comentarios, no se acorta el proceso.

Cuadro 7. ¿Qué es lo más importante que debemos aprender?

Mirando su respuesta en el Cuadro 7 del lienzo Lean UX, pregunte:
¿Cuál es la menor cantidad de trabajo que debemos hacer para aprender eso?¿Qué técnicas podríamos usar?

Discuta las opciones en equipo y diseñe diferentes experiencias en cada uno de los siguientes plazos. Cada uno debe "montar la Curva de la Verdad"

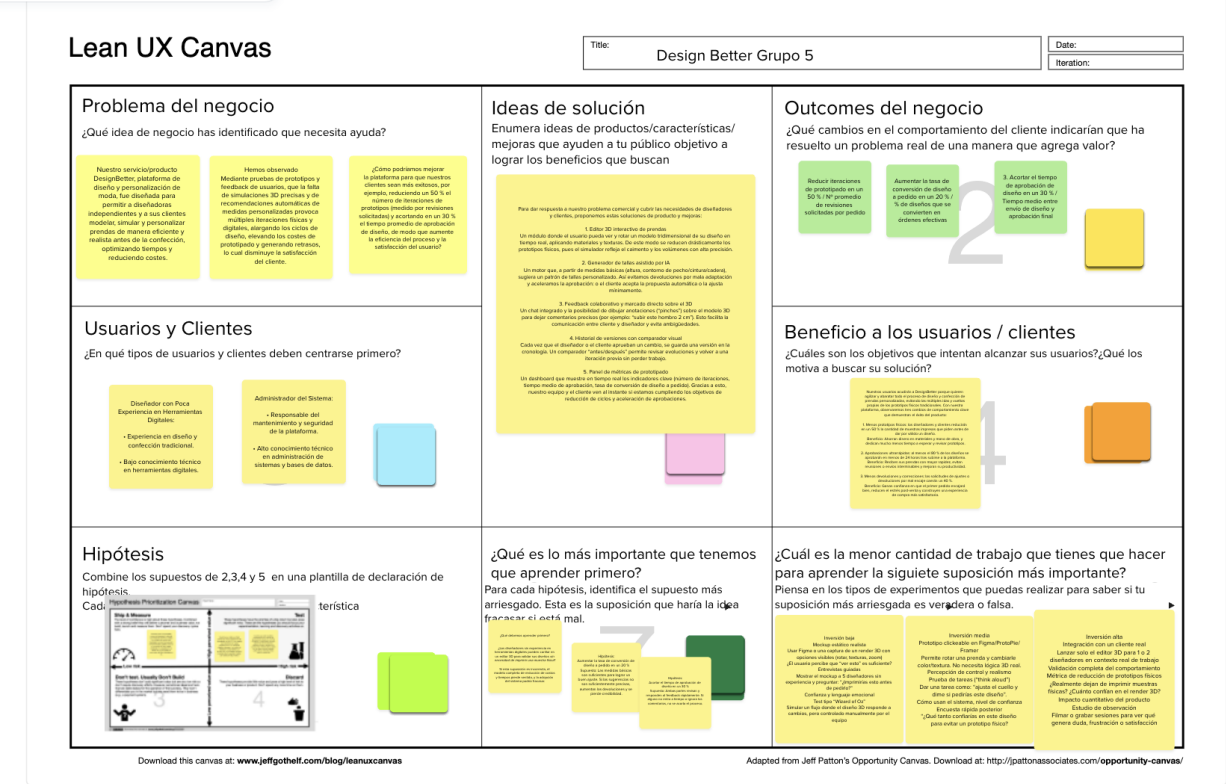
Pase luego los postit al cuadro 8 del Lean UX Canvas



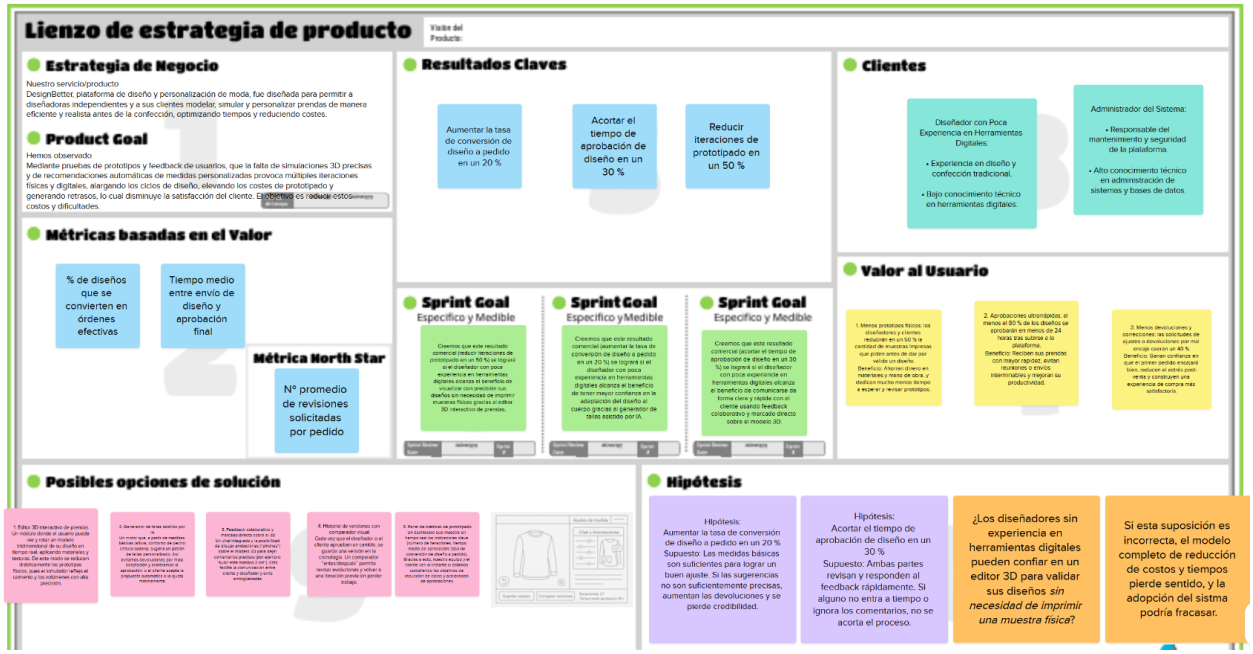
Inversión baja
Mockup estático realista
Usar Figma o una captura de un render 3D con opciones visibles (rotar, texturas, zoom)
¿El usuario percibe que "ver esto" es suficiente?
Entrevistas guiadas
Mostrar el mockup a 5 diseñadores sin experiencia y preguntar: "¿Imprimirías esto antes de pedirlo?"
Confianza y lenguaje emocional
Test tipo "Wizard of Oz"
Simular un flujo donde el diseño 3D responde a cambios, pero controlado manualmente por el equipo

Inversión media
Prototipo clickeable en Figma/ProtoPie/Framer
Permite rotar una prenda y cambiarle color/textura. No necesita lógica 3D real.
Percepción de control y realismo
Prueba de tareas ("think aloud")
Dar una tarea como: "ajusta el cuello y dime si pedirías este diseño".
Cómo usan el sistema, nivel de confianza
Encuesta rápida posterior
"¿Qué tanto confiarías en este diseño para evitar un prototipo físico?

Inversión alta
Integración con un cliente real
Lanzar solo el editor 3D para 1 o 2 diseñadores en contexto real de trabajo
Validación completa del comportamiento
Métrica de reducción de prototipos físicos
¿Realmente dejan de imprimir muestras físicas? ¿Cuánto confían en el render 3D?
Impacto cuantitativo del producto
Estudio de observación
Filmar o grabar sesiones para ver qué genera duda, frustración o satisfacción



Lienzo de estrategia

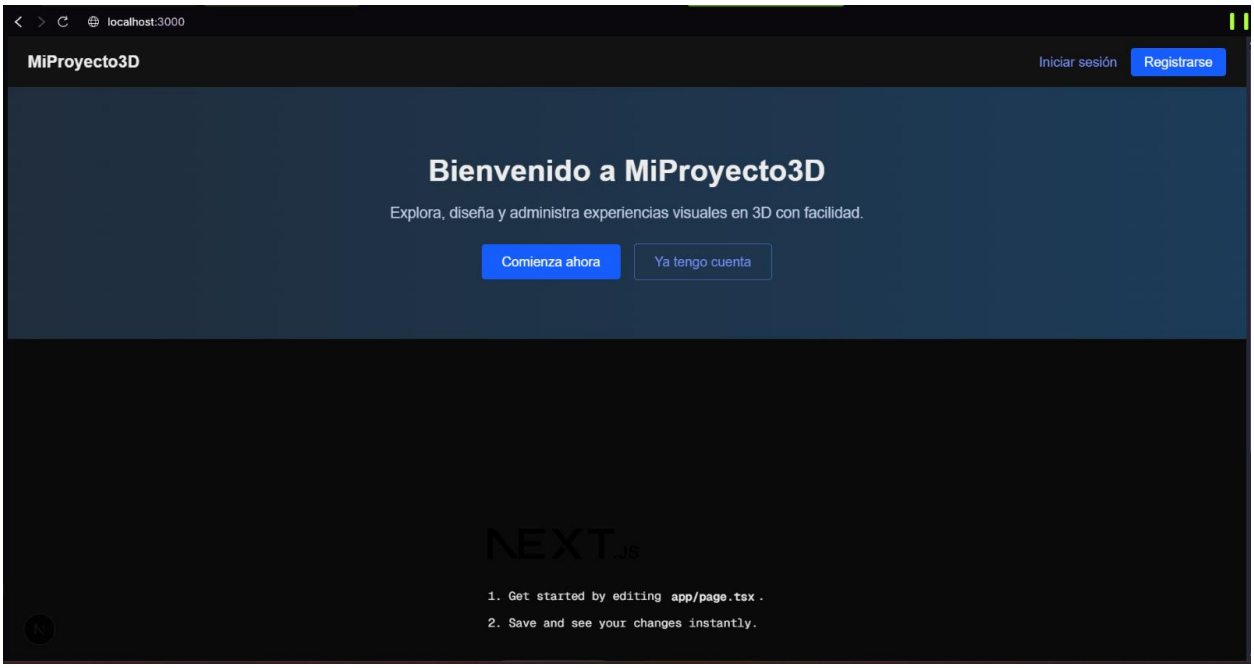


Enlace a Mural

[Mural de Lean UX Canvas]

SPRINT 1

Software Funcionando



Tareas Concluidas

Tipo	Clave	Resumen	Persona asignada	Informador	Prio	
	~~SCRUMSITO-5~~	Solucionar coordinacion de arranque inicial de contenedores y hacer migraciones automaticamente	LUIS FERNANDO...	ROBERTO SAMU...	Medium	
	~~SCRUMSITO-3~~	pruebas de contenerización	PABLO JOSE ME...	ROBERTO SAMU...	Medium	
	~~SCRUMSITO-1~~	interfaces diferenciadas para cliente, diseñador y administrador	PABLO JOSE ME...	ROBERTO SAMU...	Medium	
	~~SCRUMSITO-1~~	recuperación de contraseña	ANDRE EMILIO P...	ROBERTO SAMU...	High	
	~~SCRUMSITO-1~~	Autenticación via correo con confirmación	ROBERTO SAMU...	ROBERTO SAMU...	Medium	
	~~SCRUMSITO-1~~	Formulario de registro de usuarios	ROBERTO SAMU...	ROBERTO SAMU...	Highest	
	~~SCRUMSITO-1~~	Documentacion	PABLO JOSE ME...	ANDRE EMILIO P...	Medium	
	~~SCRUMSITO-1~~	Métricas Sprint	LUIS FERNANDO...	ANDRE EMILIO P...	Medium	
	~~SCRUMSITO-9~~	Resultados Sprint	ANDRE EMILIO P...	ANDRE EMILIO P...	Medium	
	~~SCRUMSITO-8~~	README con los comeandos necesarios	ANDRE EMILIO P...	ANDRE EMILIO P...	Low	
	~~SCRUMSITO-7~~	Creación de Archivos Configuración	ROBERTO SAMU...	ANDRE EMILIO P...	Highest	
	~~SCRUMSITO-6~~	Herramienta Contenerización	ROBERTO SAMU...	ANDRE EMILIO P...	High	
	~~SCRUMSITO-5~~	Eleccion Contenedor	LUIS FERNANDO...	ANDRE EMILIO P...	High	
	~~SCRUMSITO-4~~	Calendario	ANDRE EMILIO P...	ANDRE EMILIO P...	Medium	
	~~SCRUMSITO-3~~	Lista de Historias de Usuario	ANDRE EMILIO P...	ANDRE EMILIO P...	Medium	

Tareas planificadas, pero no concluidas

Backlog (38 incidencias) 7 15 0 Crear sprint

Crea una pizarra para planificar tu trabajo PRUÉBALO

SCRUMSITO-14	autenticación via redes sociales	GESTIÓN DE...	TAREAS POR ...	05 APR	2	RM
SCRUMSITO-33	panel de administración con gestión ...	ADMINISTR...	EN CURSO	06 APR	2	RM
SCRUMSITO-45	Como usuario (diseñador o cliente), q...	GESTIÓN DE...	EN CURSO		8	
SCRUMSITO-50	Como administrador, deseo gestionar...	ADMINISTR...	EN CURSO		5	
SCRUMSITO-17	plantillas en base de datos y moldes ...		TAREAS POR ...	07 APR	-	PA
SCRUMSITO-34	herramienta para moderación de dise...	ADMINISTR...	TAREAS POR ...	07 APR	2	AL
SCRUMSITO-35	modulos de análisis estadístico de uso	ADMINISTR...	TAREAS POR ...	08 APR	3	LL

Gráfico de Sprint Burndown de puntos de historia

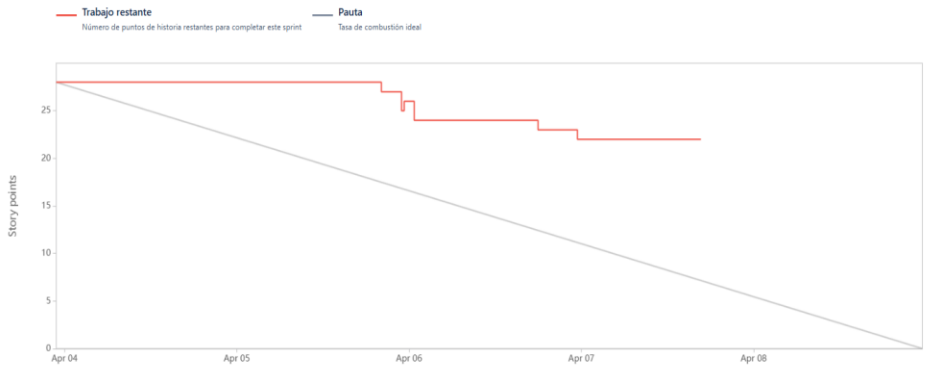
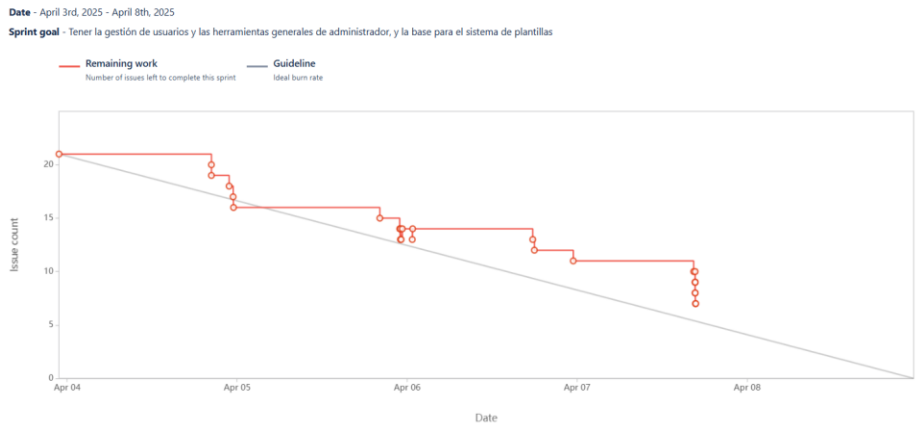


Gráfico de Sprint Burndown de Recuento de Incidencias



Reporte de Velocidad

Sprint	Commitment	Completed
Tablero Sprint 1	28	6

Éxito del Sprint

Para evaluar de manera objetiva el rendimiento del equipo durante el Sprint I, se utilizó un indicador de éxito basado en los puntos de historia comprometidos y completados.

Ecuación

Indicador de Éxito = (Puntos completados / Puntos comprometidos) × 100

= (19 / 28) × 100 ≈ 67.8%

- Puntos comprometidos al inicio del sprint: 28
- Puntos efectivamente completados: 19

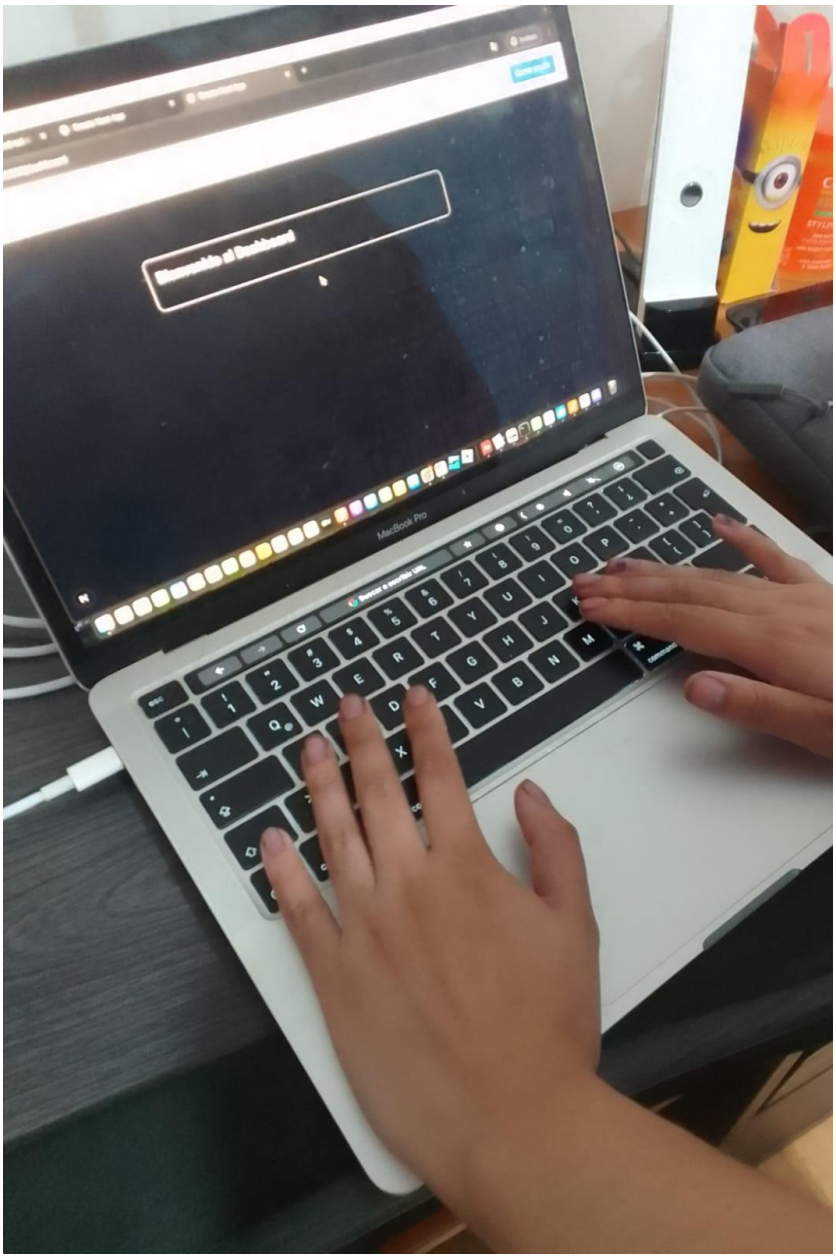
Basándose en este indicador de éxito, la calificación del Sprint sería de 7 de 10. El principal problema observado durante este primer Sprint fue la inexperiencia de todos los miembros del equipo para las herramientas utilizadas.

Iniciando en el ámbito de la contenerización, ya que nadie había usado antes Docker y adaptarse a esta herramienta sigue siendo desafiante incluso con el Sprint ya finalizado. Sumado a ello, de igual forma nadie tenía conocimiento acerca del uso de Jira, y aprender a usarlo resto días que se pudieron haber utilizado para las tareas que más tarde fueron asignadas.

Otro matiz que se podría considerar para un resultado no óptimo fue el hecho de asignar muchas tareas por cada punto de historia, por lo que en la Gráfica de Burndown de puntos de historia pareciese que no se alcanzó un buen porcentaje de tareas completas frente a las que se tenía esperado completar, sin embargo, esto se pudo evitar agregando menos tareas por punto de historia, lo cual habría mejorado no solo el gráfico sino la efectividad del Sprint.

Finalmente, un aspecto que afectó al rendimiento de este Sprint fue la subestimación de algunas tareas, ya que estas en muchos casos resultaron siendo más desafiantes y tardadas de lo que se tenía previsto, y esto afectó a que los tiempos planificados resultaran siendo imprecisos.

Evidencia de Pruebas con Usuarios finales



Usuario de prueba: diseñadora Gaby Nájera

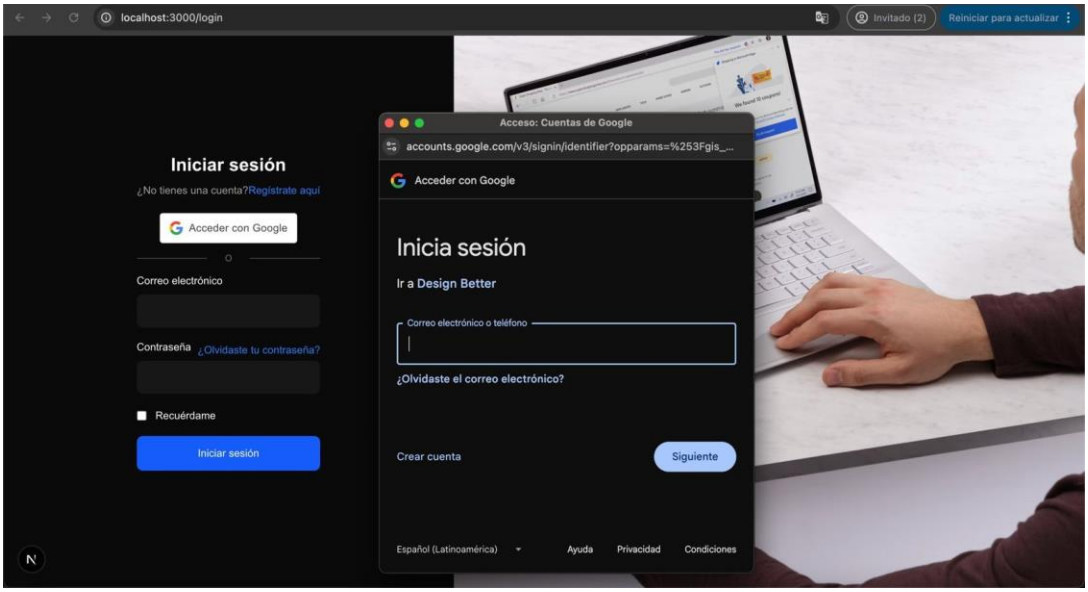
Fecha: 8 de abril a las 20:30

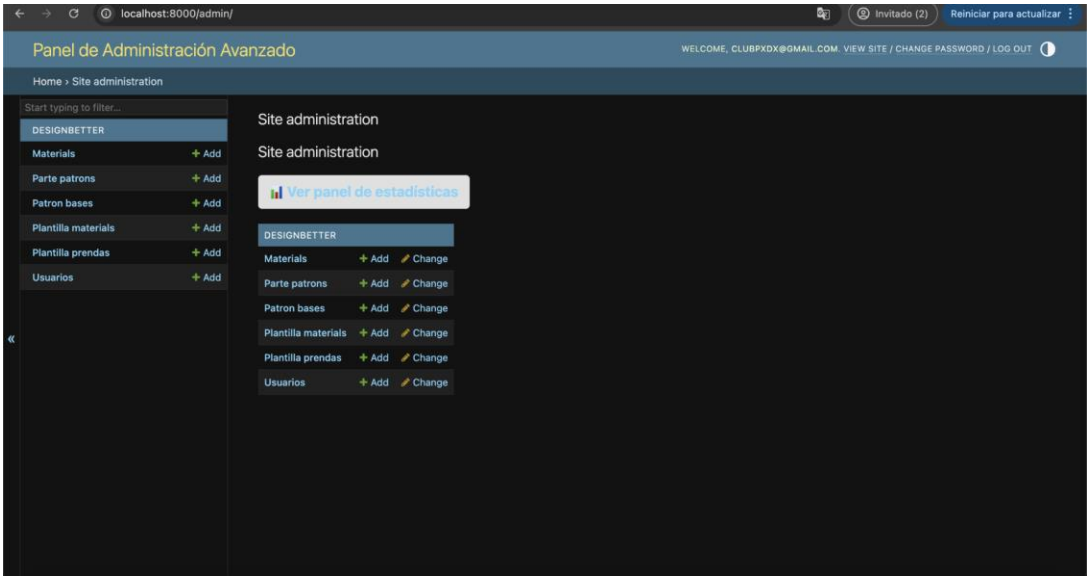
Resumen de comentarios:

- Cuando se hizo el registro, preguntó si sí era seguro y si iba a recibir algo en su correo
- Preguntó si el enlace de activación funcionaba y le expliqué que de momento solo en la computadora de pruebas.
- Muy bien, bastante básico y hay que mejorar diseño gráfico.

SPRINT 2

Software Funcionando





Tareas Concluidas

Tipo	Clave	Resumen	Estado	Comentarios	Sprint
	SCRUMSIT...	Implementar diseño nuevo y más agradable	FINALIZADA	Añadir comentar	Tablero Sprint 2
	SCRUMSIT...	Modularizar código actual en el frontend	FINALIZADA	Añadir comentar	Tablero Sprint 2
	SCRUMSITO-56 SCRUMSI...	Herramienta para moderación de diseños publicados	FINALIZADA	Añadir comentar	Tablero Sprint 2
	SCRUMSITO-56 SCRUMSI...	Como administrador, deseo gestionar usuarios, contenido...	FINALIZADA	Añadir comentar	Tablero Sprint 2
	SCRUMSITO-56 SCRUMSI...	Modulos de análisis estadístico de uso	FINALIZADA	Añadir comentar	Tablero Sprint 2
	SCRUMSIT...	Resolver problemas con interfaces diferenciadas	FINALIZADA	Añadir comentar	Tablero Sprint 2
	SCRUMSITO-72 SCRUMSI...	Plantillas en base de datos y moldes estandar personaliza...	FINALIZADA	Añadir comentar	Tablero Sprint 2
	SCRUMSITO-56 SCRUMSI...	Panel de administración con gestión de usuarios	FINALIZADA	Añadir comentar	Tablero Sprint 2
	SCRUMSIT...	Reorganizar directorios del frontend para autenticar acces...	FINALIZADA	Añadir comentar	Tablero Sprint 2
	SCRUMSITO-55 SCRUMSI...	Añadir dependencias al backend para autenticación por re...	FINALIZADA	Añadir comentar	Tablero Sprint 2
	SCRUMSITO-55 SCRUMSI...	Como usuario (diseñador o cliente), quiero poder registrar...	FINALIZADA	Añadir comentar	Tablero Sprint 2
	SCRUMSIT...	Añadir .gitignore al backend	FINALIZADA	Añadir comentar	Tablero Sprint 2
	SCRUMSITO-55 SCRUMSI...	Ajustar archivo settings.py para incorporar autenticación ...	FINALIZADA	Añadir comentar	Tablero Sprint 2
	SCRUMSITO-55 SCRUMSI...	Implementar decodificación y manejo de jwt en frontend	FINALIZADA	Añadir comentar	Tablero Sprint 2
	SCRUMSIT...	Resolver lógica para desglose de interfaces diferenciadas	FINALIZADA	Añadir comentar	Tablero Sprint 2
	SCRUMSIT...	Investigar propuestas para mejorar interfaz gráfica	FINALIZADA	Añadir comentar	Tablero Sprint 2
	SCRUMSITO-55 SCRUMSI...	Autenticación via redes sociales	FINALIZADA	Añadir comentar	Tablero Sprint 2
	SCRUMSITO-56 SCRUMSI...	Añadir modelos de plantillas al admin.py	FINALIZADA	Añadir comentar	Tablero Sprint 2
	SCRUMSIT...	Resultados del sprint en el documento	FINALIZADA	Añadir comentar	Tablero Sprint 2
	SCRUMSIT...	Pila del producto y del sprint	FINALIZADA	Añadir comentar	Tablero Sprint 2

No quedaron tareas pendientes ni en proceso en este Sprint

Gráfico de Sprint Burndown de puntos de historia

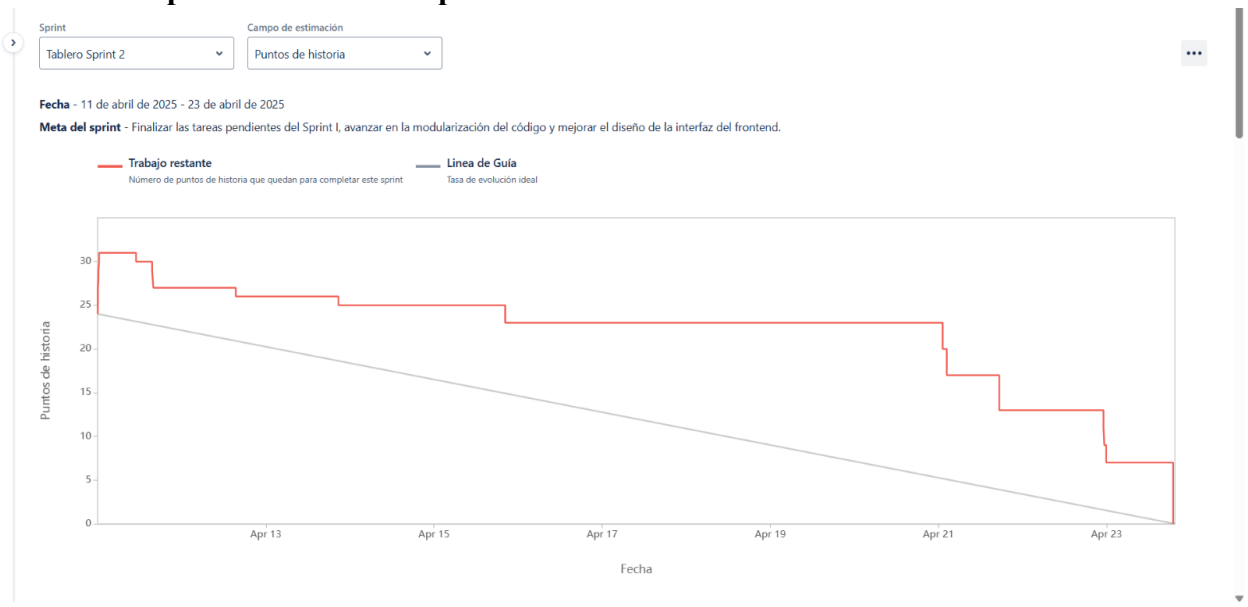


Gráfico de Sprint Burndown de Recuento de Incidencias

Reporte de Velocidad

Sprint	Confirmado	Completado
Tablero Sprint 1	28	6
Tablero Sprint 2	24	31

Éxito del Sprint

Para evaluar de manera objetiva el rendimiento del equipo durante el Sprint I, se utilizó un indicador de éxito basado en los puntos de historia comprometidos y completados.

Ecuación

Indicador de Éxito = (Puntos completados / Puntos comprometidos) × 100

= (31 / 31) × 100 = 100%

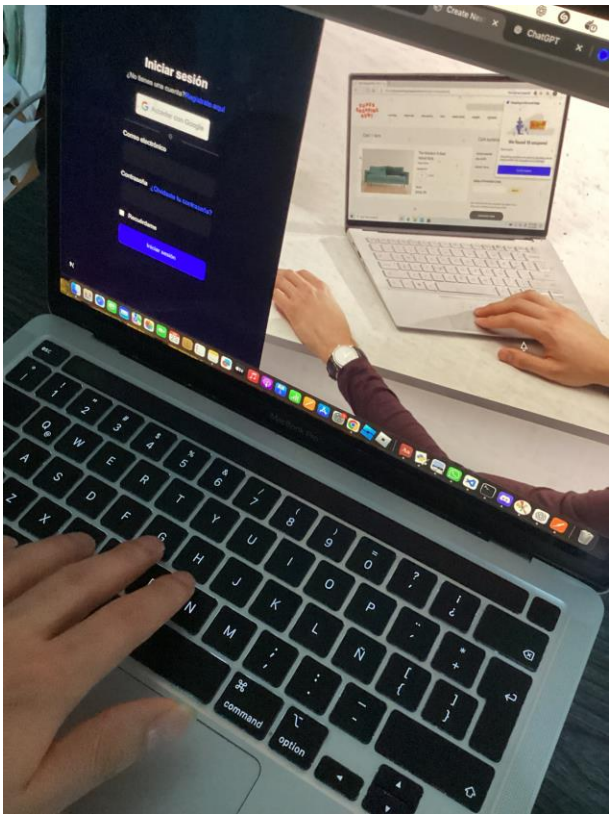
- Puntos comprometidos al inicio del sprint: 31
- Puntos efectivamente completados: 31

Durante el Sprint II se evidenció una mejora significativa en comparación con el Sprint I, especialmente en cuanto a la capacidad del equipo para cumplir los compromisos establecidos. El indicador de éxito alcanzado fue del 100%, ya que se completaron los 31 puntos de historia comprometidos al inicio del sprint. Este resultado refleja un alto nivel de organización, esfuerzo y adaptación por parte del equipo.

No obstante, el análisis más profundo de las gráficas revela matices importantes en la distribución del trabajo a lo largo del sprint. A pesar de que el trabajo se completó en su totalidad, las gráficas muestran que una gran parte del avance ocurrió en los últimos días del sprint. Este fenómeno puede atribuirse, en gran medida, a la pausa operativa ocasionada por la Semana Santa, lo que generó un espacio de baja productividad entre el 14 y el 20 de abril.

Este lapso fue evidentemente un punto muerto en el flujo de trabajo, como se puede observar en ambas gráficas de burndown, donde la línea roja permanece plana durante varios días consecutivos. Sin embargo, una vez retomadas las actividades tras el receso, se logró una recuperación eficiente del ritmo de trabajo, completando todas las tareas en un corto periodo de tiempo y alcanzando así el objetivo del sprint.

Evidencia de Pruebas con Usuarios finales



Fotografía con permiso de la entrevistada

Usuario de prueba: diseñadora Gaby Nájera

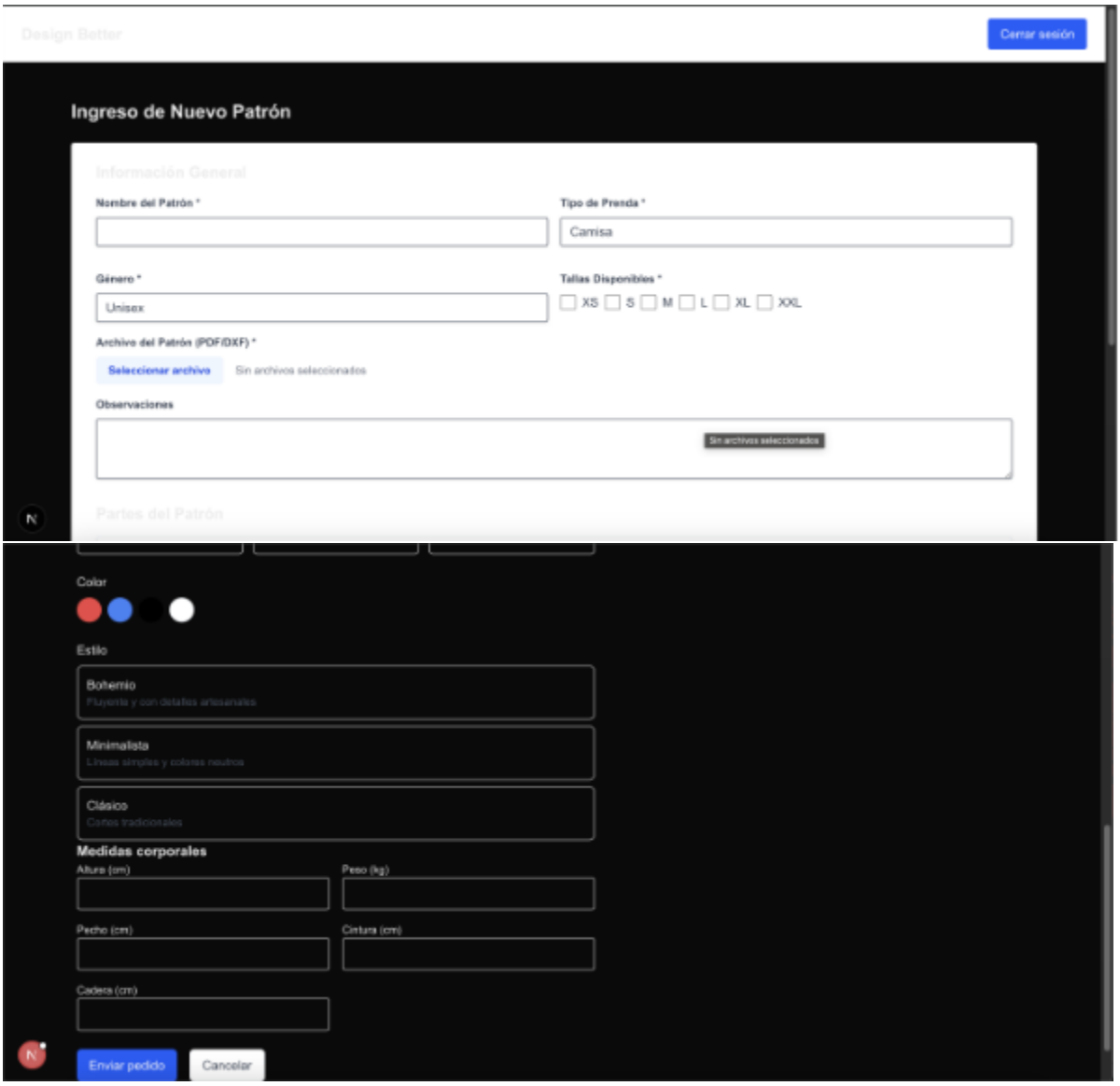
Fecha: 23 de abril a las 21:30

Resumen de comentarios:

- Le agradó bastante la idea de iniciar sesión con cuenta de Google, ya que es una manera práctica y fácil de recordar sus cuentas en sitios
- Preguntó sobre las imágenes, si eran de placeholder porque no le parecieron muy explicativas de momento y no le transmitian mucha información
- Comentó que sería razonable ver diferentes diseños de ejemplo en las imágenes demostrativas, aunque con una también es suficiente
- Respondió que los colores le parecían bien por ser neutros cuando se le preguntó específicamente por ese aspecto
- Le pareció una interfaz bastante simple de usar, aunque la vista de la página de inicio no estuvo tan clara.

SPRINT 3

Software Funcionando



Tareas Concluidas

SCRUMSIT...	apis para almacenar dxf y para guardar valores de patr...	DONE	Add comment	Tablero Sprint 3	ROBERTO SAMUE...	May 3, 2025
SCRUMSIT...	librerias ezdxf y svgwrite en backend	DONE	Add comment	Tablero Sprint 3	PABLO JOSE MEN...	Apr 30, 2025
SCRUMSIT...	carga de archivo dxf-aama desde el frontend	DONE	Add comment	Tablero Sprint 3	LUIS FERNANDO P...	May 3, 2025
SCRUMSIT...	Formulario para ingreso de patrones segun modelo de d...	DONE	Add comment	Tablero Sprint 3	PABLO JOSE MEN...	May 5, 2025
SCRUMSIT...	script para convertir de dxf a patron base a svg desde e...	DONE	Add comment	Tablero Sprint 3	ROBERTO SAMUE...	May 6, 2025
SCRUMSIT...	método para leer información del dxf o del patrón base ...	DONE	Add comment	Tablero Sprint 3	ROBERTO SAMUE...	May 6, 2025
SCRUMSIT...	apis para obtener datos del patron y conversion a svg, ...	DONE	Add comment	Tablero Sprint 3	ROBERTO SAMUE...	May 7, 2025
SCRUMSIT...	layout basico de la vista de personalización	DONE	Add comment	Tablero Sprint 3	LUIS FERNANDO P...	May 8, 2025
SCRUMSIT...	conexión frontend con backend usando REST API para ...	DONE	Add comment	Tablero Sprint 3	ANDRE EMILIO PI...	May 12, 2025
SCRUMSIT...	sistema de ingreso de medidas personalizadas y almac...	DONE	Add comment	Tablero Sprint 3	ROBERTO SAMUE...	May 10, 2025
SCRUMSIT...	interfaz para eleccion de tipos de prendas, telas, colore...	DONE	Add comment	Tablero Sprint 3	PABLO JOSE MEN...	May 9, 2025
SCRUMSIT...	Como cliente, quiero personalizar prendas según mis pr...	DONE	Add comment	Tablero Sprint 3		
SCRUMSIT...	Pagina de bienvenida con selección de rol	DONE	Add comment	Tablero Sprint 3	ANDRE EMILIO PI...	May 2, 2025
SCRUMSIT...	Corregir desglose de interfaces en frontend	DONE	Add comment	Tablero Sprint 3	LUIS FERNANDO P...	May 1, 2025

No quedaron tareas pendientes ni en proceso en este Sprint

Gráfico de Sprint Burndown de puntos de historia

Gráfico de Sprint Burndown de Recuento de Incidencias

Reporte de Velocidad

Sprint	Confirmado	Completado
Tablero Sprint 1	28	6
Tablero Sprint 2	24	31
Tablero Sprint 3	21	21

Éxito del Sprint

Para evaluar de manera objetiva el rendimiento del equipo durante el Sprint I, se utilizó un indicador de éxito basado en los puntos de historia comprometidos y completados.

Ecuación

Indicador de Éxito = (Puntos completados / Puntos comprometidos) × 100

= (21 / 21) × 100 = 100%

- Puntos comprometidos al inicio del sprint: 21
- Puntos efectivamente completados: 21

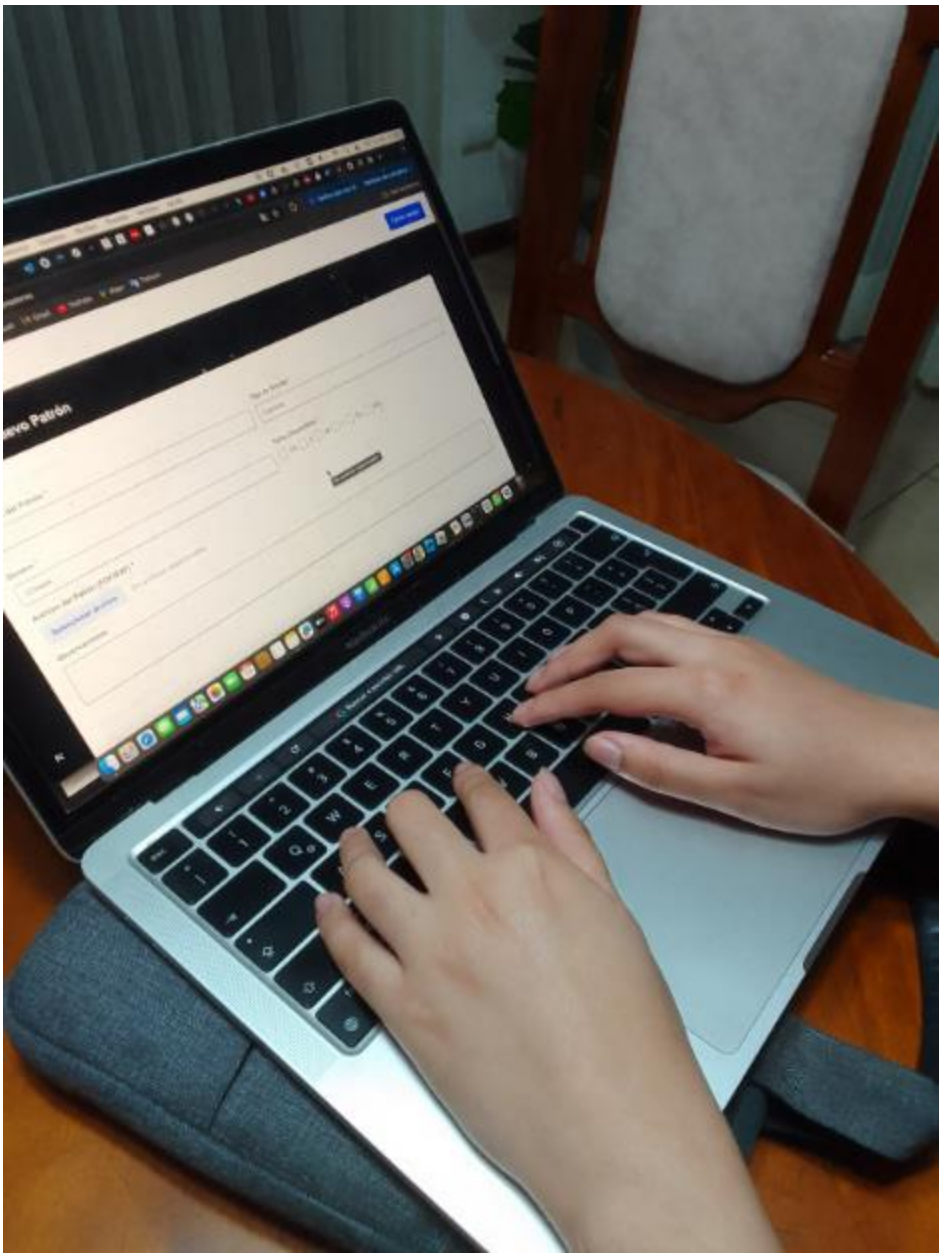
Durante el Sprint III el equipo comenzó a completar puntos desde el día 1, lo cual muestra una disposición inicial proactiva, sin esperar acumulaciones ni dejar el trabajo para fases tardías. Esto contrasta con sprints anteriores, donde hubo pausas iniciales o arranques lentos. Aquí se evidencia una mejor preparación previa al sprint, con refinamiento y definición de tareas claras.

El gráfico muestra la línea de trabajo completado de forma escalonada con múltiples puntos de incremento diario. Esto sugiere que varias tareas se fueron completando de manera continua, empezando sin estancarse por bloqueos técnicos ni de gestión, manteniendo un ritmo estable en la entrega. Más adelante la línea se mantiene plana, aunque no representan un problema crítico, esos tramos podrían verse como tareas más complejas en ejecución, las cuales requieren más tiempo; también pudo haber posibles bloqueos que se resolvieron luego y se invirtió tiempo en pruebas o revisiones.

El equipo cerró con un repunte final justo para alcanzar los 21 puntos, demostrando el control que se llevaba del progreso teniendo un cierre preciso. Durante la mayor parte del sprint, la línea de trabajo completado se mantuvo ligeramente por debajo o sobre la directriz; lo cual se toma como un indicador de realismo en la estimación. No hubo grandes desviaciones negativas, lo que implica que hubo una buena capacidad de planificación, un correcto desglose de tareas y los riesgos fueron controlados o anticipados con éxito.

Evidencia de Pruebas con Usuarios finales

Fotografía con permiso de la entrevistada



Usuario de prueba: diseñadora Gaby Nájera

Fecha: 14 de mayo a las 21:30

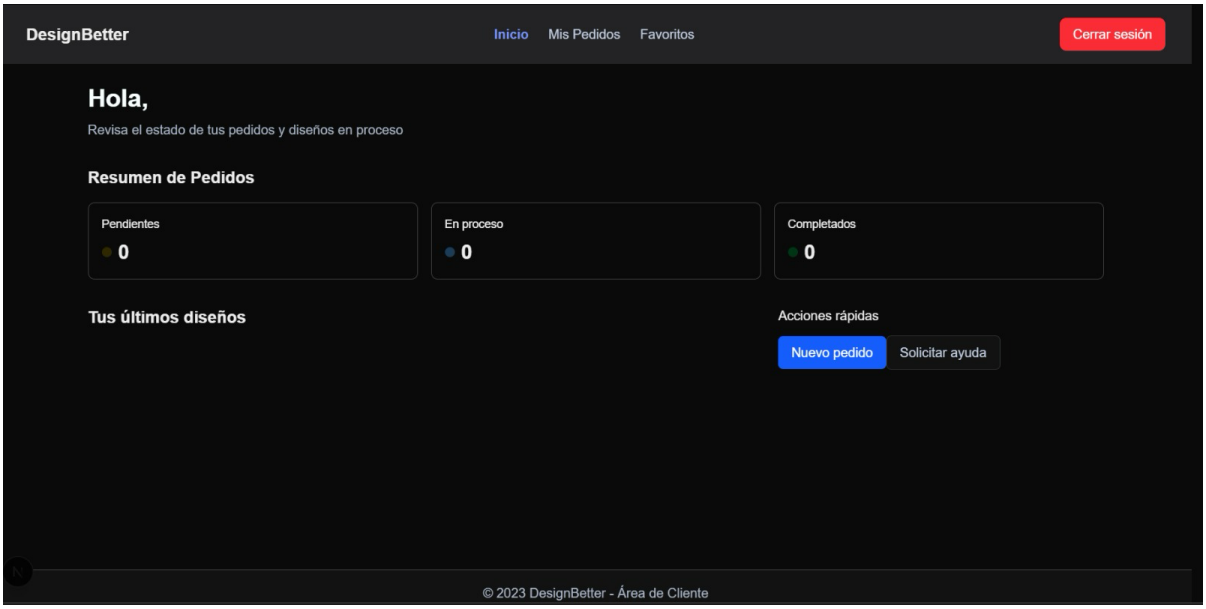
Resumen de comentarios:

- Le entusiasmaron los futuros planes de interpretar el patrón dentro del sistema, aunque siente que podría ser muy difícil para el proyecto
- Aun sabiendo que los formularios actuales son principalmente de ejemplo le pareció bastante acertada la idea general mostrada
- Sugirió que para los clientes sería recomendable alguna guía para obtener sus medidas correctamente, además de que es un dato que ella prefiere conservar cómo referencia para el proceso de diseño

- Mencionó que hay algunas medidas adicionales que son importantes para la personalización, como los largos del tronco y las extremidades

SPRINT 4

Software Funcionando



Tareas Concluidas

☐	☑	SCRUMSIT...	Cambiar formato de json en el ingreso del patrón a un mo...	FINALIZADA	Añadir comentari	Tablero Spri...
☐	☑	SCRUMSIT...	Arreglar la barra de navegación del cliente	FINALIZADA	Añadir comentari	Tablero Spri...
☐	☑	SCRUMSIT...	Reparar selección de opciones en pedido para que no se r...	FINALIZADA	Añadir comentari	Tablero Spri...
☐	☑	SCRUMSIT...	Corregir contraste de colores de los textos en el modo os...	FINALIZADA	Añadir comentari	Tablero Spri...
☐	☑	SCRUMSIT...	Arreglar presentación de opciones en solicitud de pedido	FINALIZADA	Añadir comentari	Tablero Spri...
☐	☑	SCRUMSIT...	Añadir una vista de cargado cuando se hacen solicitudes ...	FINALIZADA	Añadir comentari	Tablero Spri...
☐	☑	SCRUMSIT...	En el formulario de registro de patrón, incluir entrada de g...	FINALIZADA	Añadir comentari	Tablero Spri...
☐	☑	SCRUMSIT...	Arreglar el desglose de la vista del dashboard por rol para...	FINALIZADA	Añadir comentari	Tablero Spri...
☐	☑	SCRUMSIT...	Añadir pantalla donde avisa el correo de confirmación al r...	FINALIZADA	Añadir comentari	Tablero Spri...
☐	☑	SCRUMSIT...	Añadir un menú de opciones, algunas sin implementar tod...	FINALIZADA	Añadir comentari	Tablero Spri...
☐	☑	SCRUMSIT...	Implementar APIs para los formularios en el frontend	FINALIZADA	Añadir comentari	Tablero Spri...
☐	☑	SCRUMSIT...	Corregir ingreso de medidas	FINALIZADA	Añadir comentari	Tablero Spri...

No hubieron tareas planificadas y no concluidas para este Sprint

Gráfico de Sprint Burndown de puntos de historia

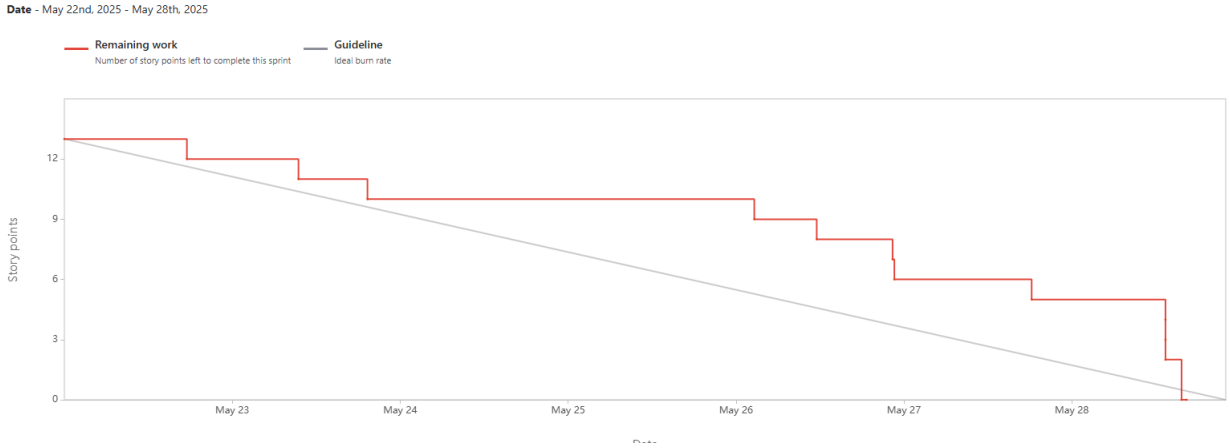
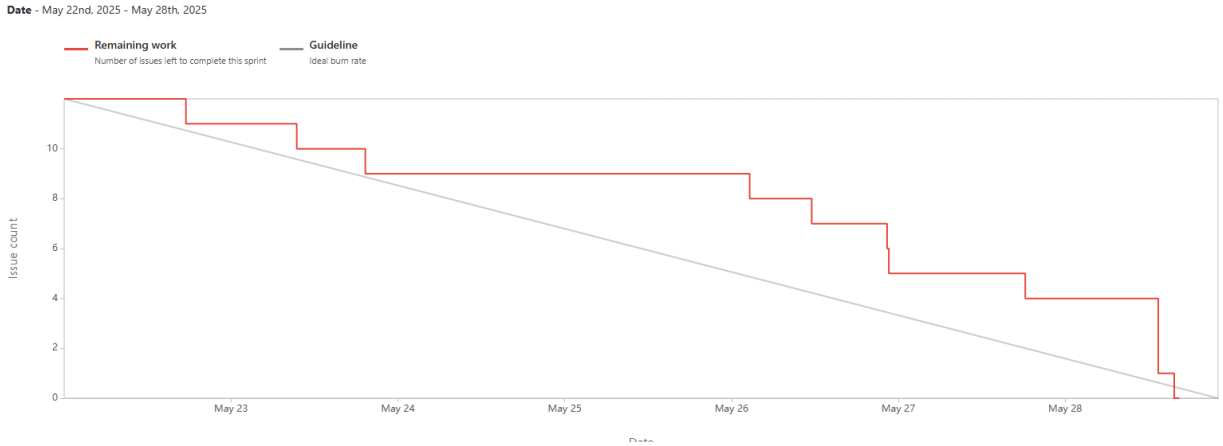
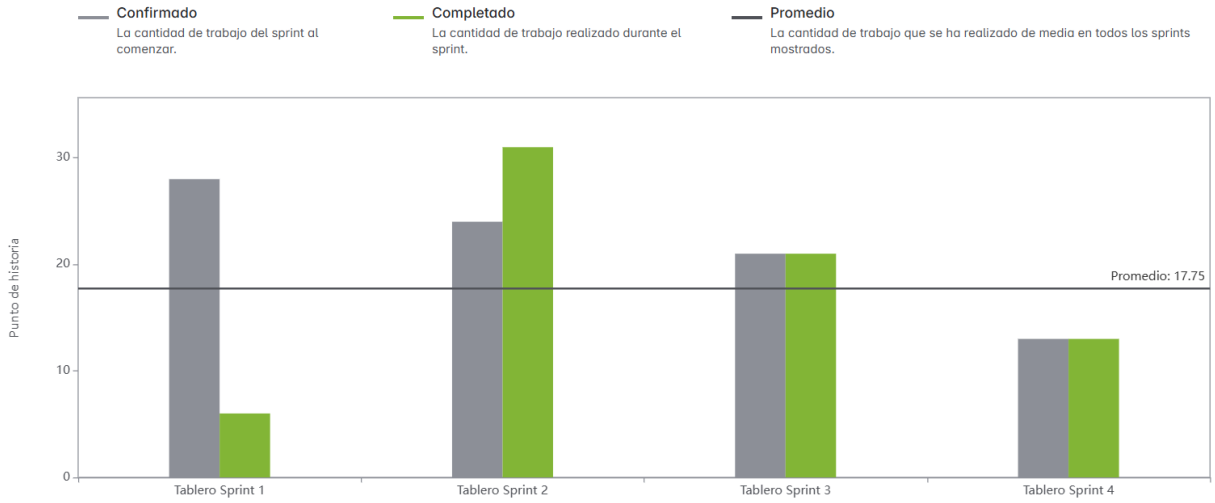


Gráfico de Sprint Burndown de Recuento de Incidencias



Reporte de Velocidad



Éxito del Sprint

Para evaluar de manera objetiva el rendimiento del equipo durante el Sprint, se utilizó un indicador de éxito compuesto por dos factores: el cumplimiento de todas las tareas (60%) y la entrega puntual de cada una (40%).

Ecuación

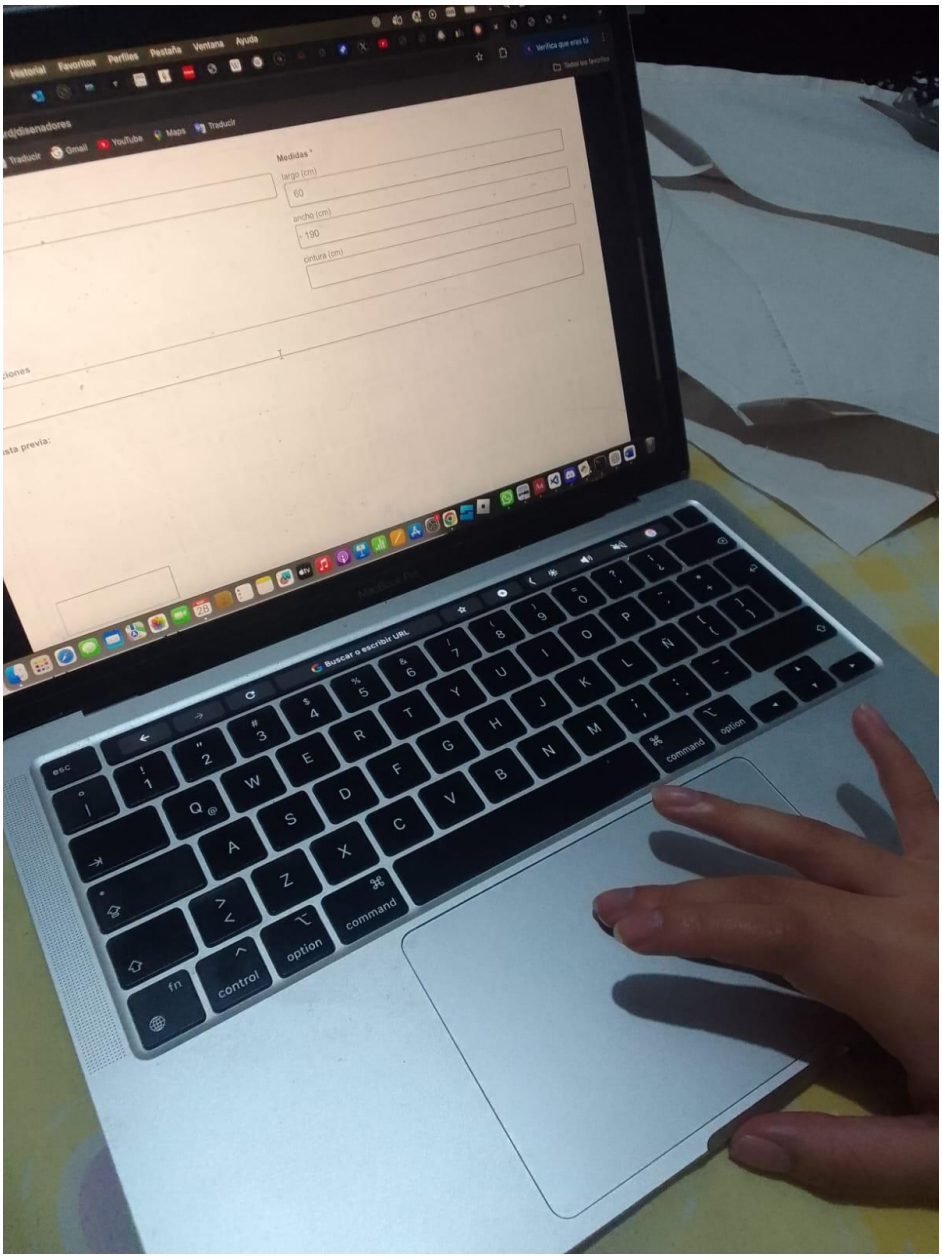
Indicador de Éxito = (Tareas completadas × 0.6) + (Promedio de puntualidad × 0.4) = (100 × 0.6) + (89.17 × 0.4) = 95.67%

- Tareas comprometidas al inicio del sprint: 12
- Tareas completadas: 12
- Puntaje promedio por cumplimiento de fechas: 89.17/100

El trabajo no comenzó desde el primer día del sprint, lo que se traduce en el tiempo dedicado a la planificación técnica, entendimiento de requisitos y reasignación de tareas. Este tipo de arranque puede ser positivo si se utiliza para afinar enfoques y resolver incertidumbres técnicas antes de ejecutar. Se identificó una meseta de inactividad, causada por tareas técnicas más complejas que requirieron más tiempo de lo estimado o bloqueos temporales o de revisión de decisiones. Sin embargo, no afectó el cumplimiento final, lo cual indica una recuperación oportuna y planificación sólida.

El equipo logró completar el 100% de los puntos comprometidos, lo que refleja un cierre disciplinado, compromiso fuerte con los entregables y un buen manejo de tiempo restante tras los días de pausa.

Evidencia de Pruebas con Usuarios finales



Usuario de prueba: diseñadora Gaby Nájera

Fotografía usada con su permiso

Fecha: 28 de mayo a las 22:00

Resumen de comentarios:

- Le agradó que implementáramos una vista previa del patrón y le gustaría poder descargarla al generarla en base a las medidas
- Menciona que es bastante importante que la escala del documento que se genere del patrón es importante, porque cuando se vaya a querer imprimir debe de tener el tamaño correcto

- Indíca que a futuro es probable que debamos de trabajar juntos la parte de cómo adaptar el tamaño de los diseños porque es información técnica de diseñador cuales son los largos que se ajustan en el vestuario para la persona.

AVANCES TOTALES

Product Backlog
Las tareas subrayadas en rojo son las que fueron completadas durante este Sprint y anteriores.

Épica 1: Gestión de Usuarios
Prioridad: Alta

Historia de usuario: Como usuario (diseñador o cliente), quiero poder registrarme e iniciar sesión para acceder a funcionalidades específicas de la plataforma.

Tareas:

- Crear formulario de registro de usuarios.
- Implementar autenticación vía correo electrónico (con confirmación).
- Integrar inicio de sesión con redes sociales (Google, Facebook).
- Crear sistema para recuperación de contraseña.
- Desarrollar interfaces diferenciadas para cliente, diseñador y administrador.

Épica 2: Personalización de Prendas
Prioridad: Alta

Historia de usuario: Como cliente, quiero personalizar prendas según mis preferencias y medidas corporales para recibir productos únicos.

Tareas:

- Crear base de datos de plantillas y moldes estándar personalizables.
- Implementar interfaz interactiva para elección de tipos de prendas, telas, colores y estilos.
- Desarrollar sistema para ingresar medidas personalizadas y almacenarlas.
- Crear el layout básico de la vista de personalización (frontend).
- Implementar selección interactiva de tipos de prendas (menú desplegable, botones).
- Desarrollar selección de colores y materiales usando paletas visuales dinámicas.
- Desarrollar sistema de recomendación inteligente (IA) de estilos según medidas.
- Implementar persistencia temporal de selecciones del usuario usando almacenamiento local (localStorage).
- Conectar frontend con el backend mediante llamadas REST API para almacenar personalizaciones permanentemente.

Épica 3: Visualización Realista (3D)
Prioridad: Alta

Historia de usuario: Como usuario, quiero previsualizar digitalmente mis prendas personalizadas antes de su confección.

Tareas:

- Investigar e integrar biblioteca para visualizaciones en 3D (por ejemplo, Three.js, Babylon.js).
- Investigar documentación técnica y tutoriales de biblioteca elegida (Three.js o Babylon.js).
- Configurar entorno inicial básico para renderizado 3D en frontend.
- Crear modelo básico estático para prueba inicial de visualización.
- Implementar carga dinámica de modelos 3D desde backend.
- Programar interacción en tiempo real con controles básicos (zoom, rotación, desplazamiento).
- Desarrollar integración con personalizaciones elegidas por usuario (color, material, tamaño).
- Crear función para actualizar visualización inmediatamente al cambiar opciones del usuario.
- Optimizar rendimiento de carga de modelos (carga progresiva, caching).
- Realizar pruebas de rendimiento y usabilidad de la visualización 3D.
- Desarrollar función para visualización dinámica en tiempo real (actualización inmediata tras cambios).
- Implementar creación de avatares personalizados según medidas del cliente.

Épica 4: Seguimiento de Pedidos
Prioridad: Media-alta

Historia de usuario: Como cliente, deseo conocer en tiempo real el estado de mi pedido desde su creación hasta la entrega.

Tareas:

- Crear un sistema visual de tracking del estado de pedidos.
- Desarrollar notificaciones automáticas sobre cambios de estado.
- Implementar historial detallado de modificaciones en cada pedido.

- Definir estados específicos del pedido (e.g., Recibido, En diseño, En producción, Enviado, Entregado).
- Diseñar visualmente el componente gráfico para tracking en el frontend (barra de progreso visual).
- Implementar la lógica en backend para almacenar y modificar estados del pedido.
- Crear API REST para consultar y actualizar estado de pedidos desde frontend.
- Desarrollar frontend para reflejar en tiempo real cambios en los estados del pedido.
- Configurar sistema automático de notificaciones (email y/o push) al usuario al cambiar estados.
- Desarrollar módulo de historial visual para mostrar la trazabilidad completa del pedido.

Épica 5: Comunicación y Colaboración
Prioridad: Alta

Historia de usuario: Como usuario, quiero una comunicación efectiva y directa entre cliente y diseñador.

Tareas:

- Desarrollar un módulo interno para mensajería y envío de archivos adjuntos.
- Implementar chatbot inteligente para responder preguntas frecuentes y derivar a un asesor cuando sea necesario.
- Investigar soluciones tecnológicas existentes para chatbots (por ejemplo, DialogFlow).
- Crear cuenta y configurar entorno básico del chatbot en la plataforma seleccionada.
- Programar flujo básico de conversación con respuestas predefinidas (FAQ inicial).
- Implementar lógica del chatbot para derivar consultas complejas a un asesor humano.
- Integrar chatbot en el frontend del sitio (widget flotante).
- Crear módulo básico de mensajería interna entre usuarios (cliente-diseñador).
- Desarrollar backend para almacenar mensajes y archivos adjuntos en base de datos.
- Crear interfaz gráfica en frontend para intercambio de mensajes en tiempo real (tipo chat).
- Programar notificaciones instantáneas al recibir nuevos mensajes.
- Integrar sección FAQ para consultas rápidas.

Épica 6: Administración y Gestión de Plataforma
Prioridad: Media

Historia de usuario: Como administrador, deseo gestionar usuarios, contenido y realizar análisis de datos sobre el uso de la plataforma.

Tareas:

- Desarrollar panel de administración con gestión de usuarios.
- Implementar herramienta para moderación de diseños publicados.
- Crear módulo de análisis estadístico de uso (dashboard de métricas).

Épica 7: Infraestructura Técnica
Prioridad: Alta

Historia de usuario: Como desarrollador, quiero un entorno de desarrollo replicable usando contenedores para facilitar el despliegue y mantenimiento del sistema.

Tareas:

- Instalar y configurar Docker o Podman.
- Crear Dockerfile para el servicio web y base de datos.
- Preparar archivo docker-compose para orquestación de servicios.
- Configurar variables de entorno necesarias para ambos servicios.
- Probar construcción local de imágenes Docker (docker build).
- Escribir documentación técnica (README.md) con pasos de ejecución de contenedores.

Épica 8: Gestión de Pagos
Prioridad: Media

Historia de usuario: Como cliente, quiero pagar de forma segura y rápida por los pedidos realizados.

Tareas:

- Investigar plataformas de pago compatibles (PayPal, Stripe, MercadoPago).
- Implementar módulo de pagos integrando la plataforma elegida.
- Configurar entorno backend para comunicación segura con la API de pagos.
- Desarrollar módulo backend para crear transacciones y manejar callbacks/respuestas de pagos.
- Implementar validación de datos y cifrado para garantizar seguridad en transacciones.
- Asegurar cumplimiento legal (facturación SAT Guatemala).
- Documentar claramente los procesos técnicos para futuras auditorías.

Épica 9: Despliegue y Publicación del Sistema
Prioridad: Media-alta

Historia de usuario: Como usuario final, deseo acceder fácilmente a la plataforma desde cualquier dispositivo.

Tareas:

- Seleccionar proveedor de hosting en la nube.
- Desplegar aplicación usando servicios en la nube (AWS, Heroku, DigitalOcean).
- Realizar pruebas en entorno de producción.

Épica 10: Mantenimiento y Escalabilidad
Prioridad: Baja-media

Historia de usuario: Como administrador del sistema, quiero garantizar que el sistema sea mantenible y escalable a largo plazo.

Tareas:

- Establecer procesos de backup automático de datos.
- Documentar claramente el código y procedimientos técnicos.
- Implementar monitoreo automático de rendimiento y alertas de caídas.

INFORME: Porcentaje de Avance del Desarrollo del Proyecto

Segun lo indicado por las tareas completadas del Backlog, El proyecto presenta un avance del 29.58%, con un total de 21 tareas completadas de las 71 tareas planificadas en el backlog general. Este progreso refleja un desarrollo sólido en áreas clave como la gestión de usuarios, la personalización de prendas, la infraestructura técnica y la administración de la plataforma, lo que indica que se han priorizado funcionalidades esenciales para la operatividad inicial del sistema. A medida que se avanza en las siguientes épicas, se espera un incremento sostenido en el porcentaje de cumplimiento.

Porcentaje: 29.58%