**Informe de Avance Fase 2**

*Proyecto Capstone: “Agenda Barber”*

Integrantes:

Danyel Pereira

Benjamín Durán

Javier Sereño

[**Introducción**](#_o9zbos66l0mf) **3**

[**Datos del Documento**](#_suvh7etj6x76) **4**

[**Información del Proyecto**](#_xtmm8i95wnvy) **5**

[**Propósito del Plan de Proyecto**](#_1xwz04nlx6c7) **5**

[**Visión del Proyecto Scrum**](#_mql8y4yuhni9) **5**

[**Alcance del Proyecto**](#_mlaba88qb2e2) **6**

[**Metodología de Desarrollo**](#_6w0lkprb0hkh) **7**

[**Definición de Roles y Responsabilidades**](#_x1lnb3o99v6) **8**

[**Costos (Estimación en CLP)**](#_p9i4b7sr4kuz) **8**

[**Resumen de Riesgos**](#_vaahpp692p8c) **9**

[**Épicas para el Proyecto**](#_mp3pevdokebi) **10**

[**Priorización de Épicas**](#_bfy6yezibpty) **10**

[**Definición de Historias de Usuario**](#_3wox57a1ejo8) **11**

[**Product Backlog del Proyecto**](#_76w7zwmt4h24) **11**

[**Estimación de Puntos de Historia**](#_p8zplv6as1k0) **12**

[**Técnica de Estimación**](#_lz4gljuhl1fb) **12**

[**Sprint Planning**](#_b9ruj2y89bvl) **12**

[**Definición y Estimación de Tareas**](#_pu51hbmnqdvg) **13**

# **Introducción**

El presente informe corresponde al avance de la Fase 2 del proyecto Agenda Barber, desarrollado como parte del proceso de titulación de la carrera de Ingeniería en Informática. En esta etapa se detalla el plan de proyecto, la visión Scrum, la planificación de sprints, las épicas, el backlog y la estimación de tareas, con el objetivo de asegurar una ejecución ordenada y medible del desarrollo.

El proyecto busca entregar una solución digital funcional y escalable para la gestión integral de barberías, integrando reservas, pagos y un módulo informativo sobre estilos y cortes.

### 

# **Datos del Documento**

* Nombre del Proyecto: Agenda Barber – Sistema de Reservas, Servicios e Información para Barberías
* Etapa: Fase 2 – Planificación y Ejecución Inicial
* Integrantes:  
  + Danyel Pereira – Backend y lógica de reservas/pagos
  + Benjamín Durán – Frontend y diseño de interfaz
  + Javier Sereno – Análisis, pruebas y documentación
* Carrera: Ingeniería en Informática
* Institución: Duoc UC, Sede San Joaquín
* Periodo Académico: Segundo semestre 2025

### 

# **Información del Proyecto**

Agenda Barber es una aplicación web destinada a optimizar la gestión de barberías mediante una plataforma en línea que permita a los clientes agendar citas, explorar servicios, pagar en línea.  
 El sistema también proporciona un panel administrativo para los barberos o administradores, donde se podrá controlar disponibilidad, servicios, precios y generar reportes básicos.  
 Se busca digitalizar procesos manuales comunes en el rubro, mejorando la productividad y experiencia del cliente.

# **Propósito del Plan de Proyecto**

El propósito de este plan es establecer una ruta estructurada para el desarrollo del proyecto bajo la metodología Scrum, definiendo sus etapas, responsables, backlog y planificación.  
 Permite asegurar que los objetivos técnicos y funcionales se cumplan en los tiempos definidos, reduciendo riesgos y facilitando la colaboración del equipo en torno a entregables concretos en cada sprint.

# **Visión del Proyecto Scrum**

El proyecto Agenda Barber será desarrollado bajo la metodología ágil Scrum, utilizando iteraciones cortas denominadas sprints de tres semanas.  
 El enfoque se centra en entregar valor incremental: cada sprint generará un producto parcialmente funcional, testeable y mejorable.  
 La visión general del proyecto Scrum es construir un sistema web estable, intuitivo y útil, que profesionalice la gestión de barberías pequeñas y medianas.

### 

# **Alcance del Proyecto**

Incluye:

* Agenda en línea con gestión de horarios por barbero.
* Registro de reservas y notificaciones automáticas.
* Módulo de pagos en línea (Webpay o MercadoPago).
* Catálogo de servicios con precios y descripción.
* Módulo informativo con tipos de cortes, estilos y recomendaciones.
* Panel administrativo con control de barberos, clientes y reportes básicos.

Excluye:

* Aplicación móvil nativa.
* Integración con sistemas contables externos.
* Funcionalidades de marketing o fidelización.

### 

# **Metodología de Desarrollo**

Se aplicará Scrum como marco ágil de trabajo, con dos sprints principales:

* Sprint 1 (3 semanas): desarrollo del backend, API REST y lógica de reservas/pagos.
* Sprint 2 (3 semanas): desarrollo del frontend, integración con el backend, pruebas y módulo informativo.

Cada sprint incluye actividades de planning, daily scrum, review y retrospective, garantizando un flujo de mejora continua.

# **Definición de Roles y Responsabilidades**

| **Rol** | **Integrante** | **Responsabilidades** |
| --- | --- | --- |
| **Product Owner** | Javier Sereno | Definir requerimientos, validar avances y asegurar cumplimiento de objetivos. |
| **Scrum Master** | Danyel Pereira, Benjamín Durán | Coordinar al equipo, eliminar impedimentos y asegurar cumplimiento de Scrum. |
| **Development Team** | Benjamín Durán, Danyel Pereira, Javier Sereno | Desarrollar, probar y documentar las funcionalidades del sistema. |

# 

# **Costos (Estimación en CLP)**

| **Concepto** | **Descripción** | **\* Costo estimado (CLP)** |
| --- | --- | --- |
| **Horas de desarrollo** | 3 integrantes × 60 h aprox. × $2.000/h | $360.000 |
| **Hosting y dominio** | Servidor educativo y dominio .cl anual | $30.000 |
| **Recursos de diseño** | Figma, herramientas visuales y prototipado (free/plus) | $10.000 |
| **Pruebas y despliegue** | Servicios de test, logs o API de pago sandbox | $15.000 |
| **Total estimado** |  | **$415.000 CLP** |

\* Nota*:* los valores son simulados para fines académicos, reflejando un costo aproximado de desarrollo en contexto de egreso.

### 

# **Resumen de Riesgos**

| **Riesgo** | **Descripción** | **Probabilidad** | **Impacto** | **Estrategia de mitigación** |
| --- | --- | --- | --- | --- |
| **Integración de API de pago** | Errores o incompatibilidades con Webpay/MercadoPago. | Media | Alta | Usar entornos de prueba y documentación oficial antes de producción. |
| **Desfase en entregas** | Retrasos por dependencias entre backend y frontend. | Media | Media | Planificación temprana y reuniones de sincronización. |
| **Fallas en despliegue** | Problemas en hosting o configuración de servidor. | Baja | Media | Documentar entorno e implementar ambiente local espejo. |
| **Baja usabilidad** | Interfaz confusa o no adaptada al usuario final. | Media | Media | Pruebas de usabilidad con usuarios de prueba y revisión UX. |
| **Dependencia de librerías externas** | Cambios o bugs en librerías (Bootstrap, Django REST). | Baja | Alta | Usar versiones estables y documentadas. |

# **Épicas para el Proyecto**

1. **Gestión de reservas:** creación, modificación y cancelación de citas.
2. **Módulo de pagos en línea:** integración con pasarela segura.
3. **Gestión de servicios y precios:** administración del catálogo.
4. **Panel administrativo:** control de usuarios, barberos y reportes.
5. **Módulo informativo de cortes:** sección educativa con estilos y recomendaciones.
6. **Notificaciones automáticas:** confirmación y recordatorio al cliente.

# **Priorización de Épicas**

| **Épica** | **Prioridad** | **Justificación** |
| --- | --- | --- |
| Gestión de reservas | Alta | Núcleo del sistema y primera funcionalidad entregable. |
| Pagos en línea | Alta | Aumenta compromiso del cliente y reduce inasistencias. |
| Servicios y precios | Media | Complementa el flujo de reserva. |
| Panel administrativo | Media | Necesario para la gestión interna. |
| Módulo informativo | Baja | Agrega valor, pero no es esencial para el MVP. |
| Notificaciones automáticas | Media | Mejora la experiencia del cliente. |

# **Definición de Historias de Usuario**

Ejemplos representativos:

* **HU1:** Como cliente, quiero visualizar los barberos disponibles para elegir el horario que más me acomode.
* **HU2:** Como cliente, quiero pagar mi reserva online para confirmar mi cita.
* **HU3:** Como administrador, quiero gestionar los servicios y precios disponibles.
* **HU4:** Como barbero, quiero visualizar mis citas programadas por día.
* **HU5:** Como cliente, quiero recibir una notificación que me recuerde mi cita.
* **HU6:** Como usuario, quiero acceder a información sobre tipos de cortes según mi tipo de cabello.

# **Product Backlog del Proyecto**

| **ID** | **Historia de Usuario** | **Prioridad** | **Épica Asociada** |
| --- | --- | --- | --- |
| HU1 | Visualizar barberos y horarios | Alta | Gestión de reservas |
| HU2 | Pago de reserva online | Alta | Módulo de pagos |
| HU3 | Gestión de servicios y precios | Media | Gestión de servicios |
| HU4 | Panel de administración de citas | Media | Panel administrativo |
| HU5 | Recordatorios automáticos | Media | Notificaciones |
| HU6 | Información de tipos de cortes | Baja | Módulo informativo |

# **Estimación de Puntos de Historia**

| **Historia de Usuario** | **Descripción breve** | **Puntos** |
| --- | --- | --- |
| HU1 | Ver barberos y horarios | 8 (es el núcleo, implica lógica de agenda, consultas, validaciones) |
| HU2 | Pagar reserva online | 13 (alta complejidad, requiere integración con API externa) |
| HU3 | Gestionar servicios y precios | 5 (moderado, CRUD simple) |
| HU4 | Panel administrativo | 8 (múltiples vistas y roles) |
| HU5 | Recordatorios automáticos | 3 (baja complejidad, evento simple) |
| HU6 | Módulo informativo de cortes | 5 (moderado, pero con diseño visual) |

# **Técnica de Estimación**

Se utiliza la técnica **Planning Poker**, basada en la secuencia de Fibonacci (1, 2, 3, 5, 8, 13).  
 Cada integrante propone una estimación de esfuerzo y se llega a consenso.  
 Esto permite evaluar la complejidad de cada historia sin asociarla directamente a tiempo, sino al esfuerzo relativo de desarrollo.

# **Sprint Planning**

| **Sprint** | **Duración** | **Objetivos principales** | **Entregables** |
| --- | --- | --- | --- |
| **Sprint 1** | 3 semanas | Desarrollo del backend, API REST, módulo de reservas y pagos, base de datos y pruebas iniciales. | API funcional, BD configurada y endpoints testeados. |
| **Sprint 2** | 3 semanas | Desarrollo del frontend, integración total, módulo informativo, panel admin y pruebas finales. | Sistema integrado y funcional con UI. |

# **Definición y Estimación de Tareas**

**Sprint 1**

* Configuración del entorno Django REST (3 ptos)
* Diseño del modelo de datos y migraciones (5 ptos)
* Implementación del módulo de reservas (8 ptos)
* Integración API de pagos (8 ptos)
* Pruebas iniciales de API (3 ptos)

**Sprint 2**

* Diseño de interfaz con Bootstrap/Tailwind (5 ptos)
* Integración frontend-backend (8 ptos)
* Desarrollo módulo informativo (5 ptos)
* Implementación de notificaciones (3 ptos)
* Pruebas de sistema y documentación final (4 ptos)