

**Universidad ORT Uruguay**

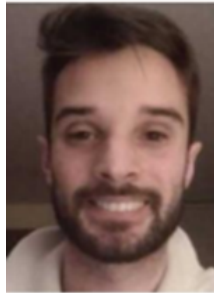
**Facultad de Ingeniería**

**Escuela de Tecnología**

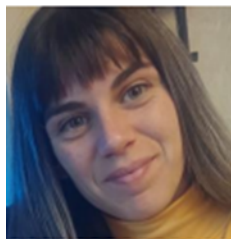
**OBLIGATORIO**

**METODOLOGÍAS DE DESARROLLO COLABORATIVO**

**Pablo Larnaudie – 340181**



**Natalia Rebella – 327283**



**Grupo: N2A**

**Docente: Gricelda Friss de Kereki**

**ATI**

**Fecha de entrega: 18-11-2024**

## INDICE

<b>METODOLOGIAS DE DESARROLLO COLABORATIVO.....</b>	<b>1</b>
<b>INDICE.....</b>	<b>2</b>
<b>INTRODUCCIÓN.....</b>	<b>3</b>
<b>DESCIPCIÓN DEL CLIENTE.....</b>	<b>3</b>
<b>PRESENTACIÓN DEL PROBLEMA.....</b>	<b>3</b>
<b>OBJETIVOS:.....</b>	<b>4</b>
<b>LISTADO DE NECESIDADES.....</b>	<b>5</b>
<b>REQUERIMIENTOS.....</b>	<b>5</b>
<b>PLAN PARA LA GESTIÓN DE INTERESADOS (STAKEHOLDERS).....</b>	<b>13</b>
<b>Plan para la gestión de los riesgos:.....</b>	<b>17</b>
Identificación de riesgos.....	17
Análisis y priorización de riesgos.....	18
Plan de respuesta a los riesgos.....	19
Estrategias de mitigación.....	19
Plan de Contingencia.....	20
Monitoreo y control de riesgos.....	21
<b>METODOLOGÍA PARA LA EJECUCIÓN DEL PROYECTO.....</b>	<b>21</b>
<b>PLANIFICACIÓN DEL PROYECTO.....</b>	<b>23</b>
Conceptos.....	23
Estimación.....	23
Presupuesto.....	23
Alcance.....	24
Esfuerzo estimado para cada fase de nuestro proyecto:.....	25
Estimaciones de tamaño.....	26
2. Estimación de afinidad (T-Shirt Sizing o Fibonacci).....	26
3. Estimación paramétrica.....	26
4. Estimación multipunto (promedio ponderado o estimación de 3 valores).....	27
5. Estimación de tamaño (puntos de función y puntos de caso de uso).....	27
Implementación seleccionada para nuestro proyecto.....	27
Dependencias.....	28
1. Dependencias obligatorias.....	28
2. Dependencias discrecionales.....	28
3. Dependencias externas.....	28
4. Dependencias Internas.....	29
<b>SOLUCIÓN PROPUESTA.....</b>	<b>29</b>
<b>SUPUESTOS.....</b>	<b>30</b>

## **INTRODUCCIÓN**

En el presente proyecto se buscará desarrollar e implementar una plataforma de software para la gestión de alquileres de casas de temporada en el departamento de Rocha. Esto incluye la gestión de reservas, pagos, calendarios y comunicación con los huéspedes.

Actualmente, la gestión de los alquileres se lleva a cabo de manera manual. La promoción de las propiedades se realiza a través de redes sociales, como Instagram o Marketplace, así como en plataformas externas como Airbnb. Los clientes interesados se ponen en contacto con los anfitriones mediante estos canales, y las reservas se registran manualmente utilizando hojas de cálculo. Este proceso requiere una intervención constante para mantener los datos actualizados y organizados.

El proyecto propone facilitar al cliente la administración de los alquileres (reduciendo los tiempos de gestión y los posibles errores en las reservas), mejorar la experiencia del usuario y optimizar el proceso de reservas, logrando así un manejo más efectivo, seguro y confiable del negocio, así como también un incremento en las reservas y calificaciones positivas.

## **DESCRIPCIÓN DEL CLIENTE**

El cliente es un emprendimiento familiar que gestiona un complejo de cuatro casas monoambiente, cada una con capacidad para cuatro personas, ubicado en la zona turística de Punta del Diablo, Rocha. Su actividad principal es el alquiler de estas viviendas a turistas durante la temporada alta de verano.

Actualmente, el negocio se enfoca en la gestión de reservas, la atención constante a los inquilinos durante toda su estadía, y el mantenimiento de las propiedades. Su objetivo principal es maximizar la ocupación y asegurar la satisfacción de los huéspedes.

Sin embargo, la gestión manual de reservas y pagos dificulta la logística y organización, aumentando la posibilidad de errores que impacta negativamente en la experiencia de los clientes.

## **PRESENTACIÓN DEL PROBLEMA**

El emprendimiento descrito se encuentra con grandes desafíos en torno a la gestión, comunicación y organización de sus tareas, todo lo cual depende de plataformas externas y procesos descentralizados. Como se menciona en los puntos anteriores, actualmente las reservas son registradas en Excel, debiendo ser actualizadas constantemente para evitar que se superpongan u otros posibles errores, ya que cada propietario maneja diferentes plataformas.

Por otro lado, la comunicación con múltiples personas en simultáneo genera incomodidad, malestar y pérdida de información, trayendo posibles errores en la gestión de las reservas y futuras peticiones que los huéspedes puedan solicitar a los propietarios. Otro punto a considerar es la administración de la gestión de finanzas, que al momento implica pérdida tanto de tiempo como de costos que conllevan estas tareas.

Dado este contexto, surge la necesidad de implementar una plataforma de software integral que centralice y automatice los procesos de reserva, pagos, y comunicación con los huéspedes. Esta solución no solo mejorará la eficiencia operativa del negocio, sino que también reducirá los tiempos de gestión, minimizando errores en la logística y elevando la calidad del servicio al cliente. Con una plataforma unificada, el negocio podrá maximizar la ocupación de las casas, optimizar la organización interna y aumentar la satisfacción de los huéspedes, logrando, en última instancia, un incremento en la cantidad de reservas y en las calificaciones positivas de los usuarios.

### **OBJETIVOS:**

- **Centralizar la gestión de reservas:** Desarrollar una plataforma web que permita registrar y visualizar todas las reservas en un solo lugar, evitando solapamientos y errores de duplicación. El objetivo es reducir las reservas conflictivas en un 100% desde la implementación del sistema.
- **Optimizar la experiencia del usuario:** Proveer a los clientes la posibilidad de consultar disponibilidad y realizar reservas de manera directa a través de la plataforma.
- **Automatizar el sistema de pagos:** Implementar una plataforma que permita procesar pagos de manera automática y generar los recibos correspondientes para cada reserva, facilitando la contabilidad y reduciendo el tiempo destinado a la gestión de pagos.
- **Centralizar el canal de comunicación con los huéspedes:** Implementar en la plataforma un único punto de contacto para los huéspedes, que permita centralizar la comunicación sin importar cuál de los propietarios esté a cargo en un momento determinado. De esta forma, los clientes pueden comunicarse de manera fluida a través de un solo número o medio, eliminando la necesidad de cambiar de contacto durante la estadía y asegurando una atención continua y consistente.
- **Aumentar la visibilidad y accesibilidad:** Desarrollar una integración del sistema de reservas con redes, tanto sociales como de alquiler de

propiedades (como Facebook, Instagram o Airbnb), que permitan redireccionar y centralizar todas las reservas o consultas dentro de la plataforma a desarrollar. Este punto permitiría maximizar la promoción, visibilidad y accesibilidad al complejo, permitiendo utilizar diversas estrategias de marketing digital, a través de los medios masivos de comunicación, cuidando finalmente el objetivo principal de dirigir al cliente hacia un punto único donde finalizará su proceso de alquiler.

## **LISTADO DE NECESIDADES**

Partiendo de lo anteriormente expuesto, procederemos a presentar un análisis para desglosar una lista de necesidades en base a lo que el cliente está solicitando.

- Centralización de información y de reservas.
- Simplificación en los procesos de reservas.
- Automatización y seguridad de los pagos.
- Capacidad de reporte y monitoreo en tiempo real.
- Control financiero al cerrar cada quincena.
- Unificación de la comunicación.
- Incremento de la visibilidad (integración con redes sociales y otros medios de comunicación masiva).
- Optimización del tiempo de gestión.
- Escalabilidad y flexibilidad en la plataforma.

## **REQUERIMIENTOS**

Partiendo de la lista de necesidades que obtuvimos posterior al análisis realizado de los detalles obtenidos en la entrevista con el cliente, detallaremos a continuación algunos de los requerimientos que el sistema de software debe cumplir para satisfacerlas.

La escala de prioridad seleccionada por el equipo para detallar el posible impacto en la solución del problema que brinda cada requerimiento es del 1 al 6, siendo 1 la prioridad más alta y 6 la más baja.

## REQUERIMIENTOS FUNCIONALES

<b>ID:</b>	RF1		
<b>Título</b>	Gestión centralizada de reservas	<b>Req Relacionados</b>	RF1, RF2, RF3, RF4, RF5, RF6, RF7, RF8.
<b>Descripción</b>	El sistema debe permitir registrar, consultar y actualizar todas las reservas en un espacio centralizado, evitando duplicaciones y conflictos en el calendario.		
<b>Prioridad</b>	1	<b>Origen</b>	Entrevista 23/10/24, Patricia Martínez (Propietaria)
<b>Precondiciones</b>	La plataforma debe estar operativa, y el usuario debe tener permisos para visualizar y gestionar reservas.		
<b>Entradas</b>	Datos de reserva (nombre del huésped, fechas de estadía, tipo de unidad reservada, contacto, etc.).		
<b>Salidas</b>	Reserva confirmada, calendario actualizado, y estado de la reserva (nueva, modificada o consultada).		
<b>Postcondiciones</b>	La reserva se almacena correctamente, sin solaparse con otras		

<b>ID:</b>	RF2		
<b>Título</b>	Automatización de pagos	<b>Req Relacionados</b>	RF1, RNF1
<b>Descripción</b>	El sistema debe permitir realizar y procesar pagos en línea de manera automática, generando información en la base de datos que permita a futuro agilizar la contabilidad al finalizar un determinado período (quincena)		
<b>Prioridad</b>	1	<b>Origen</b>	Entrevista 23/10/24, Patricia Martínez (Propietaria)
<b>Precondiciones</b>	El usuario, ya logueado debe haber seleccionado una reserva, la debe de haber confirmado y debe estar listo para proceder con el pago.		
<b>Entradas</b>	Información del pago (monto, métodos de pago, datos del huésped y administrador)		
<b>Salidas</b>	Confirmación de pago, recibos generados y notificaciones vía mail, tanto para el huésped como para el anfitrión.		
<b>Postcondiciones</b>	El pago se procesa, se guarda el registro en el sistema y se genera el recibo.		

<b>ID:</b>	RF3		
<b>Título</b>	Canal de comunicación unificado	<b>Req Relacionados</b>	RF1, RF2
<b>Descripción</b>	El sistema debe ofrecer un módulo de mensajería centralizado (mensajería asincrónica) que permita a los huéspedes comunicarse con el anfitrión de turno a través de un único canal, evitando la necesidad de cambiar de contacto o de plataforma.		
<b>Prioridad</b>	1	<b>Origen</b>	Entrevista 23/10/24, Patricia Martínez (Propietaria)
<b>Precondiciones</b>	El huésped debe estar registrado y logueado.		
<b>Entradas</b>	Mensajes enviados por el huésped o el anfitrión.		
<b>Salidas</b>	Mensajes enviados y recibidos en respuesta a las consultas, generando un hilo de conversación. En ambos casos se recibirán notificaciones tanto en la plataforma como en el mail (si así el usuario lo determina)		
<b>Postcondiciones</b>	La comunicación se guarda en el sistema y se almacena el historial de la conversación en un menú desplegable con un sistema de mensajería.		

<b>ID:</b>	RF4		
<b>Título</b>	Interfaz de reserva	<b>Req Relacionados</b>	RF1, RF2, RF3, RF8
<b>Descripción</b>	La plataforma debe ofrecer a los clientes una interfaz intuitiva para consultar la disponibilidad y realizar reservas de forma sencilla.		
<b>Prioridad</b>	2	<b>Origen</b>	Entrevista 23/10/24, Patricia Martínez (Propietaria)



<b>ID:</b>	RF5		
<b>Título</b>	Informes de reserva y ocupación	<b>Req Relacionados</b>	RF1, RF7, RF3, RF6
<b>Descripción</b>	El sistema debe poder generar informes automáticos que muestren estadísticas de ocupación, reservaciones realizadas y cancelaciones.		
<b>Prioridad</b>	2	<b>Origen</b>	Entrevista 23/10/24, Patricia Martínez (Propietaria)

<b>ID:</b>	RF6		
<b>Título</b>	Recordatorios y notificaciones	<b>Req Relacionados</b>	RF1, RF5
<b>Descripción</b>	La plataforma debe enviar notificaciones automáticas a los huéspedes y administradores sobre la aproximación de fechas de reserva.		
<b>Prioridad</b>	2	<b>Origen</b>	Entrevista 23/10/24, Patricia Martínez (Propietaria)

<b>ID:</b>	RF7		
<b>Título</b>	Registro de Check-in y Check-out	<b>Req Relacionados</b>	RF1, RF5, RF6
<b>Descripción</b>	El sistema debe incluir un módulo para registrar el momento en que los huéspedes hacen check-in y check-out, facilitando el control de ocupación.		
<b>Prioridad</b>	3	<b>Origen</b>	Entrevista 23/10/24, Patricia Martínez (Propietaria)

<b>ID:</b>	RF8		
<b>Título</b>	Gestión de opiniones y calificaciones de huéspedes	<b>Req Relacionados</b>	RF1, RF4
<b>Descripción</b>	La plataforma debe permitir que los huéspedes dejen comentarios y calificaciones sobre su experiencia, facilitando a los administradores la recopilación de feedback para mejorar el servicio.		
<b>Prioridad</b>	3	<b>Origen</b>	Entrevista 23/10/24, Patricia Martínez (Propietaria)

## REQUERIMIENTOS NO FUNCIONALES

<b>ID:</b>	RNF1		
<b>Título</b>	Seguridad de los datos de usuario	<b>Req Relacionados</b>	RF1, RF2
<b>Descripción</b>	La plataforma debe asegurar que los datos personales de los huéspedes y las transacciones financieras estén protegidos mediante encriptación y políticas de acceso restringido.		
<b>Prioridad</b>	1	<b>Origen</b>	Entrevista 23/10/24, Patricia Martínez (Propietaria)

<b>ID:</b>	RNF2		
<b>Título</b>	Escalabilidad del sistema	<b>Req Relacionados</b>	N/A
<b>Descripción</b>	La plataforma debe estar diseñada para permitir la incorporación de nuevas funcionalidades sin degradar el rendimiento del sistema.		
<b>Prioridad</b>	1	<b>Origen</b>	Entrevista 23/10/24, Patricia Martínez (Propietaria)

<b>ID:</b>	RNF3		
<b>Título</b>	Compatibilidad multiplataforma	<b>Req Relacionados</b>	Aplica a todos.
<b>Descripción</b>	La plataforma debe ser completamente accesible desde diferentes dispositivos (computadoras, celulares), asegurando una experiencia coherente en todos los dispositivos.		
<b>Prioridad</b>	2	<b>Origen</b>	Entrevista 23/10/24, Patricia Martínez (Propietaria)

<b>ID:</b>	RNF4		
<b>Título</b>	Políticas de cancelación o reembolso	<b>Req Relacionados</b>	RF1, RF2, RF7
<b>Descripción</b>	El sistema debe aplicar políticas de cancelación específicas que permitan al huésped cancelar dentro de un plazo de 15 días previos a la reserva, perdiendo la señal. Las cancelaciones cercanas a la fecha de entrada podrán ser penalizadas, perdiendo el 50% del total del pago.		
<b>Prioridad</b>	1	<b>Origen</b>	Entrevista 23/10/24, Patricia Martínez (Propietaria)

<b>ID:</b>	RNF5		
<b>Título</b>	Restricción de edad mínima para reservas	<b>Req Relacionados</b>	RF1
<b>Descripción</b>	El sistema debe verificar que el usuario que realiza la reserva sea mayor de 18 años. De lo contrario la reserva debe ser rechazada.		
<b>Prioridad</b>	3	<b>Origen</b>	Entrevista 23/10/24, Patricia Martínez (Propietaria)

## **PLAN PARA LA GESTIÓN DE INTERESADOS (STAKEHOLDERS)**

Los principales interesados en el proyecto son Patricia y Marco. Ambos van a presentar un papel fundamental en el transcurso de la implementación de la plataforma para la gestión de los alquileres de su negocio.

Al contar con la participación de ambos, vamos a requerir una estrategia de gestión que permita un enfoque eficiente y colaborativo para alcanzar los objetivos establecidos, que serán detallados a continuación.

### **PATRICIA**

- **Posicionamiento:** Patricia será la principal responsable de la gestión de los requerimientos del sistema. Ella es quien define las características y funcionalidades clave que la plataforma debe tener, basándose en las necesidades del negocio y las expectativas de los usuarios. Además, Patricia se encargará de proporcionar los recursos necesarios para el desarrollo del proyecto, como el apoyo de personal, financiamiento, materiales y herramientas. Ella tendrá la última palabra sobre las decisiones estratégicas relacionadas con el diseño y los alcances del sistema, asegurando coherencia con los objetivos generales del emprendimiento.
- **Relevancia:** Su relevancia en el proyecto es crucial, ya que su capacidad para identificar y definir los requerimientos correctos impacta directamente en el éxito de la implementación. Si los requerimientos no son claros o no se priorizan correctamente, el equipo de desarrollo podría generar soluciones que no se ajusten a las necesidades del negocio o que resulten en una plataforma poco funcional o ineficaz. Además, su responsabilidad de asignar recursos es clave para asegurar que el proyecto cuente con todo lo necesario para ejecutarse correctamente.
- **Estrategia de gestión:**
- **Acción 1:** Organizar reuniones periódicas con el equipo de desarrollo para definir y aclarar los requerimientos del sistema, asegurándose de que todas las partes involucradas comprendan las expectativas y objetivos.

- **Acción 2:** Establecer un plan claro de recursos, incluyendo asignación de personal, presupuesto y herramientas, para garantizar que el equipo de desarrollo tenga lo necesario para avanzar.
- **Acción 3:** Establecer un sistema de comunicación constante entre Patricia y el equipo de desarrollo para asegurar que cualquier cambio o ajuste en los requerimientos se comuniquen de manera correcta,
- **Acción 4:** Validar y aprobar las funcionalidades y avances del proyecto en etapas, para asegurarse de que las soluciones propuestas sean coherentes con las necesidades del negocio.
- **Acción 5:** Gestionar la relación con los otros interesados y con los usuarios finales para obtener retroalimentación continua y asegurar que el sistema se ajuste a sus expectativas.

## **MARCO**

- **Posicionamiento:** Marco, por su parte, será el principal contacto del equipo de desarrollo para el seguimiento de la evolución del proyecto. Su rol será el de supervisar el progreso del desarrollo, asegurando que los plazos se cumplan, que las tareas estén bien distribuidas y que el proyecto avance sin desviaciones. Además, jugará un papel fundamental en la comunicación entre el equipo de desarrollo y los otros interesados, proporcionando retroalimentación.
- **Relevancia:** La relevancia de Marco radica en su capacidad para coordinar la ejecución del proyecto y asegurar que todo el equipo de desarrollo trabaje de manera coordinada con los objetivos establecidos. Será responsable de identificar y resolver posibles problemas durante el proceso de implementación, actuar rápidamente en la gestión de riesgos y asegurar que los entregables cumplan con los estándares de calidad esperados.
- **Estrategia de gestión:**
  - **Acción 1:** Establecer reuniones de actualización semanales con el equipo de desarrollo, para revisar el avance del proyecto, discutir problemas o desafíos y tomar decisiones rápidas para resolver cualquier obstáculo.
  - **Acción 2:** Implementar herramientas de seguimiento de proyectos para visualizar el progreso de las tareas y asegurarse de que el equipo esté trabajando en las prioridades correctas.

- **Acción 3:** Mantener una comunicación constante con los desarrolladores para asegurar que comprendan las expectativas del proyecto y puedan cumplir con los requisitos establecidos por Patricia y los demás interesados.
- **Acción 4:** Monitorear la calidad del trabajo entregado, evaluando si se cumplen los estándares establecidos y si se están alcanzando los objetivos de funcionalidad y usabilidad. Realizar pruebas continuas del sistema y asegurar que se implementen los ajustes necesarios.
- **Acción 5:** Gestionar los riesgos del proyecto, identificando posibles retrasos y trabajando en soluciones para mitigarlos antes de que se conviertan en problemas.
- **Acción 6:** Evaluar el avance del proyecto y la eficiencia del equipo, y realizar ajustes cuando sea necesario para asegurar el cumplimiento de los plazos y objetivos.

### **Matriz de Interés/poder**

Esta matriz posibilita la clasificación de los stakeholders según su nivel de interés en el proyecto y su capacidad de influencia en el mismo, permitiendo una adecuación de las estrategias a utilizar para la gestión a cada uno de ellos.

STAKEHOLDER	PODER	INTERÉS	POSICIÓN EN LA MATRIZ
Patricia	Alto	Alto	Gestionar de cerca.
Marco	Alto	Medio	Mantener informado

### **Matriz compromiso**

Interesado	U	R	N	S	L
Patricia				C	D
Marco			C	D	C

## REFERENCIAS

**C:** Nivel adecuado

**D:** Nivel deseado

**U:** Desconoce el proyecto y sus potenciales impactos

**R:** Conoce el proyecto y sus potenciales impactos, y se resiste a los cambios que pueden ocurrir.

**N:** Está al tanto del proyecto.

**S:** Está al tanto del proyecto y de sus impactos potenciales y lo apoya

**L:** Está al tanto del proyecto y de sus impactos potenciales y participa activamente contribuyendo al éxito del proyecto

## **ESTRATEGIAS DE GESTIÓN Y COMUNICACIÓN CON STAKEHOLDER**

Se detallarán algunas estrategias que busquen asegurar que los involucrados participen activamente en el proyecto, en relación a sus respectivos roles y/o responsabilidades, permitiendo y facilitando al equipo de desarrollo un trabajo alineado con los objetivos y expectativas del negocio, así como también una vía de comunicación ágil que permita responder a posibles ajustes o requerimientos.

### PATRICIA:

- Mantener a Patricia involucrada en la definición de requerimientos y asegurar su participación activa en la aprobación de recursos.
- Programar reuniones de planificación y validación para asegurarse de que se están cumpliendo las especificaciones y requerimientos, así como también para asegurar que los recursos se encuentren disponibles en tiempo y forma.
- Aprobar cambios de requerimientos, proporcionar recursos y dar feedback sobre el cumplimiento de los objetivos del negocio.

### MARCO:

- Facilitar la supervisión del avance del proyecto y asegurar que se mantenga informado al equipo de desarrollo sobre las expectativas y los objetivos del negocio, ofreciendo retroalimentación continua.
- Que identifique y actúe sobre posibles riesgos.

- Que mantenga una comunicación fluida con el equipo de desarrollo para resolver rápidamente cualquier inconveniente que pueda afectar los plazos.

#### CANALES DE COMUNICACIÓN:

- Reuniones semanales: Estas reuniones incluirán tanto a Patricia como a Marco y se revisarán el progreso y los próximos pasos.
- Mensajería Instantánea: Creación de un grupo de chat para consultas rápidas y actualizaciones instantáneas.

#### **Plan para la gestión de los riesgos:**

El objetivo de este plan es actuar con anticipación mediante la identificación, el análisis y proponiendo estrategias para gestionar los riesgos que podrían afectar la calidad del proyecto en el desarrollo de la plataforma. Este plan nos puede ayudar a minimizar el impacto de los riesgos potenciales y asegurar la entrega exitosa de la plataforma, cumpliendo con los requerimientos y expectativas del cliente.

#### **Identificación de riesgos**

A continuación estaremos identificando los riesgos que creemos que son lo más relevantes, clasificados en cuatro categorías principales:

<b>Categoría</b>	<b>Riesgo</b>	<b>Descripción</b>
<b>Riesgos mismos del Proyecto</b>	Demoras y retrasos en el cronograma definido	Posibles demoras en la entrega de módulos de la plataforma debido a una mala estimación de tareas, posibles dificultades técnicas, o una disponibilidad inapropiada del equipo.
	Alcance no detallado correctamente	Ambigüedades o cambios frecuentes en los requerimientos, lo que puede llevar a un aumento del alcance o entregables incompletos mediante retrabajos.



<b>Riesgos Técnicos</b>	Problemas en la integración de sistemas externos	Dificultad para integrar las plataformas de redes sociales y sistemas de pago, lo que impacta en la funcionalidad de promoción y cobros de la plataforma.
<b>Riesgos de Seguridad</b>	Baja seguridad del manejo de los datos de los usuarios	Posibilidad de brechas de seguridad que expongan datos personales de huéspedes o información financiera sensible.
	Falta de cumplimiento de políticas de seguridad	No cumplir con las normativas de seguridad y privacidad (como protección de datos), perdiendo la confianza y reputación del trabajo realizado con el cliente en la plataforma.
<b>Riesgos Organizacionales</b>	Poca disponibilidad de recursos y materiales	Problemas de asignación u obtención de recursos humanos, materiales o financieros que podrían retrasar el desarrollo y afectaría la entrega de ciertas funcionalidades.
	Capacitación insuficiente del cliente	Los propietarios y el personal de gestión podrían tener dificultades para adaptarse a la plataforma si no reciben una capacitación adecuada.

### **Análisis y priorización de riesgos**

Cada riesgo se evalúa y prioriza de acuerdo con su probabilidad de ocurrencia y el impacto que tendría en el proyecto, utilizando una escala de 1 al 6, procedimos a hacer la siguiente tabla y a colocar las siguientes posibles probabilidades, impactos y prioridades que puede tener cada riesgo.

<b>Riesgo</b>	<b>Probabilidad (1-6)</b>	<b>Impacto (1-6)</b>	<b>Prioridad</b>
Retrasos en el cronograma	5	5	Alta

Alcance no definido claramente	4	5	Alta
Fallos en la integración externa	4	5	Alta
Vulnerabilidad en los datos	5	6	Alta
Falta de cumplimiento de seguridad	5	6	Alta
Poca disponibilidad de recursos	3	3	Media
Capacitación insuficiente del cliente	2	3	Media

### Plan de respuesta a los riesgos

Para cada riesgo que consideramos anteriormente procedimos a trabajar en una tabla sobre cada uno proponiendo una posible estrategia de mitigación viable en cada caso, por lo cual pudimos detallar a continuación la siguiente tabla;

### Estrategias de mitigación

Riesgo	Estrategia de Mitigación
Demoras y retrasos en el cronograma definido	Una posible solución puede ser dividir el proyecto en entregas iterativas, estableciendo metas semanales y realizando evaluaciones constantes del progreso.
Alcance no detallado correctamente	Asegurar revisiones diarias con el cliente para validar los requerimientos y documentar los cambios solicitados.
Problemas en la integración de sistemas externos	Realizar pruebas tempranas de integración para las plataformas externas, estas pruebas de integración permiten prever posibles errores en la unión de una implementación con un sistema externo, también permite establecer contacto directo con soporte técnico de cada plataforma.

Baja seguridad del manejo de los datos de los usuarios	Implementar métodos que permitan proteger datos, mediante métodos de autenticación seguros y también realizar pruebas de seguridad frecuentes que aseguren esta calidad.
Falta de cumplimiento de políticas de seguridad	Asegurarse de que todos los datos sensibles se almacenen y manejen según las regulaciones locales de privacidad, posiblemente fuera de repositorios globales para evitar hackeos.
Poca disponibilidad de recursos y materiales	Priorizar tareas que sean críticas y considerar la incorporación por periodos de tiempos reducidos de recursos adicionales si es necesario.
Capacitación insuficiente del cliente	Ofrecer una capacitación detallada con materiales de apoyo y soporte para consultas y asistencia.

### Plan de Contingencia

En el Plan de contingencia para cada riesgo decidimos seguir la misma lógica mediante tablas para poder asignar un plan de contingencia para cada riesgo.

Una vez que el riesgo sucedió y debemos tomar un accionar en el asunto debemos tener bien claro qué acción corresponde para cada riesgo permitiendo tener claro los pasos a seguir y evitar que el tiempo se expanda y el riesgo se siga incrementando con el pasar del tiempo, por ello, para tener conciencia de cómo accionar en cada caso, decidimos generar la siguiente tabla:

Riesgo	Plan de Contingencia
Demoras y retrasos en el cronograma definido	Ajustar el cronograma y reorganizar recursos en función de las tareas críticas; renegociar plazos si es necesario.
Alcance no detallado correctamente	Implementar un proceso de control de cambios, con revisiones de alcance antes de cada iteración importante.

Problemas en la integración de sistemas externos	Si la integración no es posible, definir alternativas provisionales o realizar ajustes en los flujos de trabajo del cliente.
Baja seguridad del manejo de los datos de los usuarios	Desarrollar un plan de respuesta a incidentes de seguridad que incluya la posibilidad de aislar y corregir rápidamente vulnerabilidades.
Falta de cumplimiento de políticas de seguridad	Realizar auditorías de seguridad periódicas para garantizar el cumplimiento de los estándares y ajustar la plataforma.
Poca disponibilidad de recursos y materiales	Establecer un acuerdo de respaldo con proveedores externos para servicios específicos en caso de imprevistos.
Capacitación insuficiente del cliente	Proporcionar un equipo de soporte o un punto de contacto para resolver dudas y problemas post-lanzamiento.

## **Monitoreo y control de riesgos**

Para el monitoreo y control de riesgos se constató que la mejor solución para llevar a cabo estas tareas de forma más eficaz será a través de que se realizarán reuniones semanales para revisar el estado de los riesgos y la efectividad de las estrategias de mitigación implementadas. Lo ideal sería documentar para tener un registro de incidentes, así en cualquier situación de riesgo que ocurra, evaluar la eficiencia de los planes de contingencia.

## **METODOLOGÍA PARA LA EJECUCIÓN DEL PROYECTO**

Para ejecutar el proyecto desarrollado en este documento, que supone implementar un sistema de gestión de alquileres y que presenta las características ya descritas, consideramos que sería adecuado utilizar una metodología ágil. Entre las existentes decidimos seleccionar Scrum, dado que brinda espacio a la flexibilidad, el trabajo en equipo y la adaptabilidad a cambios. Dicha metodología permite una alta interacción con los clientes, que se verán involucrados en distintas etapas del desarrollo del proyecto, por medio de sus diferentes roles y responsabilidades.

Por otro lado, la adaptabilidad al cambio que caracteriza a Scrum, permite el surgimiento de modificaciones o nuevas necesidades en cada sprint. Como cada sprint dura entre 2 a 4 semanas, la entrega de productos funcionales se lleva a cabo en períodos de tiempo muy cortos, facilitando un monitoreo constante del trabajo y desarrollo, una revisión y una devolución, asegurando la mejora continua en cada fase del proyecto.

## FASES DE LA METODOLOGÍA

1. **Formación del equipo Scrum:** compuesto por Scrum Master, Product Owner y Equipo de Desarrollo.
2. **Definición y priorización del backlog de producto** (lista priorizada de los requerimientos, mejoras y correcciones que el producto necesita para alcanzar los objetivos, actuando como un plan de trabajo para el equipo): En este caso, Patricia que es quien gestiona los requerimientos, trabajará con el equipo para desglosar los requisitos en historias de usuario, que reflejan funcionalidades para el sistema (como gestión de reservas, comunicación centralizada, control de disponibilidad). Cada historia incluirá detalles sobre el propósito y los resultados esperados.
3. **Planificación de los sprints:** se eligen las historias de usuario de mayor prioridad y se definen los objetivos específicos para cada iteración.
4. **Reuniones diarias (15 min):** se llevarán a cabo reuniones cortas todos los días para compartir avances, obstáculos y posibles pasos siguientes.
5. **Revisión del Sprint y retroalimentación:** cuando se finaliza un sprint, el equipo presentará el incremento del producto a los clientes para obtener retroalimentación (feedback), que permitirá ajustes inmediatos, asegurando una evolución acorde a las necesidades.
6. **Retrospectiva de sprint:** en esta etapa el equipo auto- evaluará su proceso de trabajo, analizando los aciertos y las áreas de mejora. Todas las conclusiones derivadas de esta etapa se aplicarán en el siguiente sprint.

## PLANIFICACIÓN DEL PROYECTO

### Conceptos

Según lo visto en el curso, algunos parámetros a manejar dentro de los conceptos para la planificación del proyecto son la Estimación y el Presupuesto.

## Estimación

La estimación es el proceso de hacer una evaluación cuantitativa de variables clave como costos, recursos, esfuerzo y duración, lo que nos va a ayudar a planificar de manera efectiva el desarrollo del proyecto.

- Costos a considerar

Los costos a considerar en el proyecto van a ser los necesarios para costear cada una de las diferentes herramientas que el proyecto requiera para su implementación, como licencias de software, servidores, diseñadores y testers, herramientas y servicios adicionales como API externas en el caso del uso por el equipo de Backend.

- Recursos necesarios

En tanto los recursos necesarios vamos a hacer referencia por un lado a los recursos humanos: desarrolladores, analistas, testers como también por otro lado a los recursos tecnológicos como pueden ser servidores, herramientas de desarrollo y de infraestructura.

- Esfuerzo

En este punto, vamos a estimar el tiempo y las horas de trabajo para cada fase del proyecto, como la planificación, el diseño, la codificación, las pruebas y la implementación, para poder organizar el proyecto

- Duración

En esta parte vamos a prever cuánto tiempo tomará cada fase del proyecto.

## Presupuesto

El presupuesto del proyecto nacerá del resultado obtenido y sobre todo, aprobado en la etapa de estimación, considerando cada uno de los puntos dentro de la estimación, generando un acuerdo entre un presupuesto ofrecido y el presupuesto aceptado.

## Alcance

Podemos encontrar dos tipos: **alcance del producto** (involucra las características y las funciones del mismo) y **alcance del proyecto** (refiere al trabajo necesario para cumplir con los requerimientos).

### **Alcance del producto:**

- El sistema debe permitir a los usuarios gestionar las propiedades para alquiler, mostrar detalles de las propiedades, permitir reservas, pagos y comunicación entre los inquilinos y propietarios.
- Interfaz amigable, sistema de pago en línea, panel de administración para los propietarios, funciones de búsqueda y filtrado.

### **Alcance del proyecto:**

**Fase de planificación:** Es importante entender el negocio y definir los requerimientos iniciales para consolidar las bases estructurales del proyecto.

**Desarrollo del sistema:** Aquí se hace la implementación de las funciones descritas.

**Pruebas:** Asegurar que todas las funciones trabajan correctamente asegurando la calidad del sistema.

**Despliegue:** Lanzamiento del sistema en producción, al alcance del usuario final.

**Mantenimiento:** Soporte continuo con actualizaciones necesarias.

### **Duración estimada para cada fase de nuestro proyecto:**

- **Fase de planificación y análisis de requerimientos:**
  - Tiempo estimado: 2-3 semanas
  - Objetivo: Definir y documentar los requisitos del sistema, incluyendo las necesidades de los usuarios y la estructura de la plataforma.
- **Diseño de la plataforma:**
  - Tiempo estimado: 3-4 semanas
  - Objetivo: Crear un diseño de la interfaz de usuario y definir la arquitectura de la base de datos y los flujos de trabajo.
- **Desarrollo:**
  - Tiempo estimado: 8-10 semanas
  - Objetivo: Implementar las funcionalidades principales de la plataforma, como el sistema de reservas, pagos, gestión de usuarios, y la interfaz de usuario.
- **Pruebas y control de calidad:**
  - Tiempo estimado: 3-4 semanas

- Objetivo: Ejecutar pruebas unitarias, pruebas de integración y pruebas de aceptación para asegurar que la plataforma funcione correctamente y esté libre de errores.
- **Despliegue y lanzamiento:**
  - Tiempo estimado: 1-2 semanas
  - Objetivo: Implementar el sistema en producción, realizar una verificación final y asegurarse de que todos los elementos funcionan correctamente en el entorno de usuario.
- **Mantenimiento y actualización:**
  - Tiempo estimado: Continuo
  - Objetivo: Brindar soporte técnico, corregir problemas de rendimiento y añadir mejoras en función del feedback de los usuarios.

**Duración total estimada:** 4-6 meses (excluyendo el mantenimiento continuo).

#### **Esfuerzo estimado para cada fase de nuestro proyecto:**

- **Fase de Planificación y Análisis de Requerimientos:**
  - Esfuerzo estimado: 120-150 horas
  - Actividades: Reuniones con stakeholders, análisis de requerimientos, documentación.
- **Diseño de la Plataforma:**
  - Esfuerzo estimado: 160-200 horas
  - Actividades: Diseño de la interfaz, arquitectura de la base de datos, creación de wireframes y prototipos.
- **Desarrollo:**
  - Esfuerzo estimado: 400-500 horas
  - Actividades: Programación del sistema de reservas, pagos, módulos de administración, desarrollo de la interfaz de usuario, pruebas unitarias.
- **Pruebas y control de calidad:**
  - Esfuerzo estimado: 120-160 horas
  - Actividades: Pruebas de funcionalidad, pruebas de carga, corrección de errores y optimización.
- **Despliegue:**
  - Esfuerzo estimado: 40-60 horas
  - Actividades: Configuración del entorno de producción, monitoreo inicial y verificación final.
- **Mantenimiento y actualización:**
  - Esfuerzo estimado: Variable, continuo.
  - Actividades: Resolución de incidencias, ajustes de sistema y adición de mejoras.



## **Estimaciones de tamaño**

Para este proyecto podemos aplicar varias técnicas de estimación que se ajusten a las diferentes áreas del proyecto. A continuación presentaremos las vistas en clase, para luego detallar cuáles serían las adecuadas para este proyecto.

### **1. Estimación análoga**

- **Uso:** Esta técnica será útil si contamos con datos históricos de proyectos similares. En el caso de nuestros clientes, ellos utilizan al momento airbnb como plataforma de alquiler en línea.
- **Implementación:** Podemos analizar el tiempo, esfuerzo y costo de proyectos de gestión de alquileres anteriores o de proyectos similares en complejidad y funcionalidad, y ajustar las estimaciones basándonos en los recursos disponibles y la complejidad del proyecto actual.

### **2. Estimación de afinidad (T-Shirt Sizing o Fibonacci)**

- **Uso:** Aplicable en la etapa inicial de diseño para clasificar y estimar los módulos del proyecto basándose en el esfuerzo y la complejidad de los mismos.
- **Implementación:** Podemos clasificar los módulos del proyecto utilizando categorías de tamaño como "Pequeño", "Mediano", "Grande" o un esquema numérico de la serie de Fibonacci (1, 2, 3, 5, 8, 13, etc.) para reflejar el esfuerzo necesario.

### **3. Estimación paramétrica**

- **Uso:** Esta técnica es adecuada para calcular costos y duración utilizando parámetros específicos
- **Implementación:** Podemos definir parámetros clave como el número estimado de usuarios conectados a la misma vez para medir la capacidad de respuesta de la aplicación

### **4. Estimación multipunto (promedio ponderado o estimación de 3 valores)**

- **Uso:** Este método es útil para obtener una estimación más equilibrada considerando tres escenarios diferentes:
  - **Estimación optimista (O):** El tiempo mínimo si todo funciona sin problemas.
  - **Estimación pesimista (P):** El tiempo máximo si surgen problemas inesperados.

- **Estimación más probable (M):** Una estimación en función de lo que prevemos como el escenario más realista. La fórmula sería:

$$t = \frac{t_{optimista} + 4t_{promedio} + t_{peor caso}}{6}$$

## 5. Estimación de tamaño (puntos de función y puntos de caso de uso)

- **Uso:** La estimación basada en puntos de función y puntos de caso de uso sirve para evaluar el tamaño del software basándose en sus funcionalidades
- **Implementación:** Podemos dividir el sistema en puntos de función, considerando aspectos como información ingresada por el usuario.

### Implementación seleccionada para nuestro proyecto

Dado el tipo de proyecto, consideramos tomar algunas de estas técnicas para obtener una estimación precisa:

1. **Estimación análoga:** Aplicar primero para definir una estimación general del proyecto en base a proyectos similares.
2. **Estimación de afinidad:** Para clasificar los módulos y dar prioridad a los componentes críticos.
3. **Estimación de tamaño (Puntos de Función):** Para obtener un resultado basado en las funcionalidades de nuestro proyecto.

### Dependencias

En este proyecto vamos a estar considerando los conceptos de dependencias obligatorias, discrecionales, externas e internas, podemos desglosar las posibles dependencias de la siguiente manera:

#### 1. Dependencias obligatorias

Aquí encontramos las restricciones necesarias para el desarrollo de la plataforma:

- **Integración de medios de pago:** No se pueden realizar reservas hasta que el sistema de pagos esté completamente integrado y probado
- **Desarrollo de funcionalidad de autenticación:** Antes de implementar el módulo de gestión de propiedades, es obligatorio establecer un

sistema de autenticación que permita a los usuarios acceder a sus perfiles.

- **Infraestructura de seguridad:** No se puede poner la plataforma en línea sin antes implementar y probar los sistemas de seguridad.
- **Base de datos operativa:** La base de datos debe estar configurada y en funcionamiento antes de comenzar a desarrollar la lógica de reservas

## 2. Dependencias discrecionales

En este punto las dependencias discrecionales dependen de las buenas prácticas.

- **Desarrollo de interfaz de usuario:** Es recomendable que el diseño de la interfaz se inicie después de definir claramente la estructura de datos y los requisitos funcionales para prevenir futuros errores o malentendidos.
- **Implementación de notificaciones:** La función de notificaciones (por ejemplo, de confirmación de reserva) debería desarrollarse después de que el flujo básico de reservas esté funcional, para garantizar que las notificaciones se envíen con los datos correctos.
- **Módulo de soporte al cliente:** La integración del módulo de atención al cliente se recomienda después de que el flujo de uso y reserva esté probado y funcionando, para que el soporte pueda enfocarse en las consultas reales de los usuarios.

## 3. Dependencias externas

Las Dependencias externas incluyen a factores fuera del control directo del equipo y pueden afectar el proyecto si no se cumplen:

- **Servicios en la nube:** La disponibilidad de servidores y almacenamiento en la nube
- **Medios de pago:** La integración con plataformas de pago externas

## 4. Dependencias Internas

Entendemos que las dependencias internas están bajo el control del equipo y constan de los siguientes puntos:

- **Pruebas de calidad y aseguramiento:** Las pruebas de calidad del sistema dependen de que todos los módulos del proyecto estén completados y funcionales
- **Capacitación del equipo:** La capacitación interna en el uso de nuevas herramientas y frameworks depende de la disponibilidad del equipo y debe programarse antes de comenzar cada fase

- **Estrategia de marketing:** Las campañas de lanzamiento dependen de que el proyecto esté en una etapa avanzada, se requiere coordinar los tiempos con el equipo de marketing y desarrollo.
- **Documentación del sistema:** La documentación final del sistema depende de que todas las funciones estén completamente implementadas

## **SOLUCIÓN PROPUESTA**

La solución consiste, como venimos desarrollando, en el desarrollo de un sistema de gestión centralizado para la administración de reservas y comunicaciones para un complejo de casas de alquiler vacacional. Este sistema será una aplicación web alojada en la nube, con acceso posible desde cualquier dispositivo con conexión a internet, tanto aplicaciones web como mobile.

A continuación detallaremos las tecnologías y herramientas que utilizaremos para el desarrollo de nuestra aplicación, en base a los conocimientos adquiridos hasta el momento en la carrera cursada.

### **Tecnologías seleccionadas**

- **Lenguaje de programación:** seleccionamos C# como lenguaje principal debido a su rendimiento y facilidad de integración con otros sistemas. Es muy efectivo para el desarrollo de aplicaciones web gracias a ASP.NET Core, que permite crear aplicaciones rápidas, escalables y seguras. Así mismo, al ser uno de los lenguajes más utilizados, asegura un soporte técnico y recursos constantemente actualizados.
- **Base de datos:** utilizaremos SQL Server que es totalmente integrable con C# y ASP.NET, además de ser una base de datos relacional confiable.

### **Herramientas seleccionadas**

- **Control de versiones:** para dicha tarea utilizaremos Git, que es una herramienta muy efectiva para dichas tareas, que permite a los desarrolladores rastrear cambios, colaborar de forma eficiente y gestionar diferentes versiones del proyecto. Lo utilizaremos junto con GitHub para alojar los diferentes repositorios en la nube, asegurando la disponibilidad y el respaldo de nuestro código.
- **Entorno de desarrollo:** seleccionamos Visual Studio Code como entorno ideal para trabajar con tecnologías web y bases de datos

(mediante MongoDB podemos interactuar con base de datos a través de Visual Studio Code), gracias a su amplia gama de extensiones, flexibilidad y compatibilidad con diferentes lenguajes. Su interfaz amigable e intuitiva facilita el flujo de trabajo.

- **Alojamiento en la nube** - AWS: Amazon Web Services es una buena opción de plataforma en la nube por su gran confiabilidad y escalabilidad. Permite el despliegue de aplicaciones web, garantizando la disponibilidad y posibles futuras expansiones, en caso de crecimiento del negocio.
- **Pruebas:** Para ejecutar las pruebas manuales elegimos la plataforma Jira mediante XRay permitiendo traquear todas las tareas de cada uno de los integrantes del equipo y así también permite asegurar la calidad en el proyecto y aplicar diferentes metodologías con las herramientas que encontramos dentro de la misma, como los dashboards o la gestión de Test Cases, Test Plans, Test Set, y las diferentes utilidades que ofrece Jira para optimizar el trabajo de los testers manual. Para las pruebas automatizadas vamos a estar utilizando la tecnología Javascript vamos a implementar WebdriverIO con Cucumber como BDD con Mocha y Chai para hacer aserciones y reducir el tiempo de trabajo de los testers manuales.

## **SUPUESTOS**

Para la solución presentada, se asumieron los siguientes supuestos:

1. **Requerimientos funcionales y no funcionales:**
  - Los clientes tendrán acceso a internet estable para interactuar con la aplicación web.
  - Los propietarios del complejo turístico estarán capacitados para usar la plataforma de gestión sin necesidad de formación técnica avanzada.
  - Los usuarios podrán realizar reservas y gestionar pagos en línea a través de la plataforma sin inconvenientes, utilizando métodos de pago digitales comunes (tarjetas de crédito, PayPal, etc.), lo que permitirá automatizar el proceso de reserva y cobro.
2. **Entorno de desarrollo y producción:**
  - La aplicación se ejecutará en un entorno en la nube AWS, asegurando alta disponibilidad.
  - Los navegadores web utilizados por los usuarios estarán actualizados y compatibles con las tecnologías empleadas (HTML, CSS3, JavaScript).
3. **Metodologías y desarrollo:**

- Se seguirá una metodología ágil (Scrum), asumiendo que las reuniones y la retroalimentación de los interesados serán frecuentes y efectivas.
- Los sprints tendrán una duración de dos semanas, y el equipo estará disponible para realizar las revisiones necesarias al final de cada iteración.
- Se parte de la premisa de que los interesados estarán disponibles para priorizar y validar funcionalidades durante el desarrollo.

**4. Tecnología y herramientas:**

- Se asume que el equipo de desarrollo tiene experiencia previa en el uso de las tecnologías seleccionadas (C#, SQL Server, AWS).
- Los entornos de desarrollo y producción estarán correctamente configurados con Git para control de versiones y Visual Studio Code como editor principal.

**5. Interacción con terceros:**

- Se considera que la integración con plataformas de terceros, como medios de pago o sistemas de correo electrónico, será viable y estable con las APIs disponibles actualmente.