

UNIVERSIDAD TECNOLÓGICA NACIONAL
Facultad Regional Córdoba
Ingeniería en Sistemas de Información



Proyecto Final

Control de acceso para Countries y Barrios Privados

Sprint 0

Profesoras:

Ing. Ortiz, María Cecilia
Ing. Mac William, María Irene
Ing. Barale, Lorena

Alumnos:

Braicovich, Ezequiel	N° 70278
Cholod, Estrella	N° 71528
Guidobaldi, Fabián	N° 70770
Pagura, Alexis	N° 69805
Quiroga, Luis	N° 71358

Curso: **5K1**
Turno: **Mañana**
Ciclo lectivo: **2019**
Fecha de entrega: **15/05/2019**



ÍNDICE

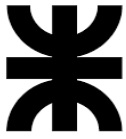
ÍNDICE	1
HISTORIAL DE VERSIONES.....	2
INTRODUCCIÓN.....	3
DEFINICIÓN DEL PRODUCT OWNER.....	4
DEFINICIÓN DEL EQUIPO Y SCRUM MASTER	4
DEFINICIÓN DEL PRODUCT BACKLOG INICIAL.....	4
STORY MAP	8
DEFINICIÓN DE LA HERRAMIENTA DE SOFTWARE PARA GESTIÓN DEL PROYECTO A UTILIZAR	8
<i>NOMBRE Y BREVE DESCRIPCIÓN DE LA HERRAMIENTA.....</i>	<i>8</i>
<i>FORMA DE COORDINACIÓN DEL TRABAJO</i>	<i>9</i>
<i>CICLO DE VIDA DE LOS ARTEFACTOS A UTILIZAR EN EL DESARROLLO DEL PROYECTO.....</i>	<i>9</i>
DEFINICIÓN DE LA ESTRUCTURA DE LA DOCUMENTACIÓN DEL PRODUCTO	10
USER STORIES.....	13
TÉCNICA DE ESTIMACIÓN A UTILIZAR.....	14
DEFINICIÓN DE LA TECNOLOGÍA A UTILIZAR EN EL DESARROLLO DEL PRODUCTO	15
MÉTRICAS DEL PROYECTO	15
PAUTAS DE CODIFICACIÓN Y TESTING	16
<i>BUENAS PRÁCTICAS DE CODIFICACIÓN</i>	<i>16</i>
<i>PLAN DE TESTING</i>	<i>17</i>
GESTIÓN DE CONFIGURACIÓN DEL PROYECTO	18
GESTIÓN DE DESPLIEGUE DEL PROYECTO.....	19
ANEXO.....	20



HISTORIAL DE VERSIONES

Información general del documento			
Título	Entrega N° 2: Sprint 0		Versión 1.1.0
Autor/es	Nombre	Alias	email
	Ezequiel Braicovich	EB	ezevalverde96@gmail.com
	Estrella Cholod	EC	cholod.estrella96@gmail.com
	Fabián Guidobaldi	FG	guidobaldifabian@gmail.com
	Alexis Pagura	AP	alexis_pagura_80@hotmail.com
	Luis Quiroga	LQ	luis268quiroga@gmail.com

Control de versiones			
Fecha	Autor/es	Versión	Resumen
03/05/2019	EB, EC, FG, AP, LQ	1.0.0	Versión base del documento.
15/05/2019	EB, EC, FG, AP, LQ	1.1.0	Se agregó la Gestión de Despliegue del Proyecto , y se realizaron modificaciones en los apartados Definición de la Tecnología a utilizar , Definición de la herramienta de Software para gestión del proyecto a utilizar y User Stories .



INTRODUCCIÓN

En esta entrega, correspondiente al Sprint 0, definiremos diferentes aspectos del proyecto, de acuerdo con la metodología seleccionada por el equipo para el desarrollo del producto.

Siguiendo con los lineamientos de Scrum, en el desarrollo de este trabajo:

- Se definirá quiénes cumplirán los roles de Product Owner, Scrum Master y Scrum Team;
- Se realizará una primera aproximación al Product Backlog, con las User Stories y Epics identificadas hasta el momento.
- Se establecerán las métricas del proyecto, y la técnica de estimación a utilizar. También se definirán las herramientas de software para la gestión de proyecto y la gestión de configuración.

Además, este trabajo también incluye definiciones con respecto al producto, como la tecnología a utilizar, las pautas de codificación y testing a seguir, la estructura de la documentación, y las herramientas de software a utilizar, tanto para la gestión de configuración del software como para la gestión del proyecto en su totalidad.



DEFINICIÓN DEL PRODUCT OWNER

Durante el desarrollo de este producto, el rol de Product Owner estará a cargo del plantel docente de la cátedra. La docente María Irene Mac William, debido a su experiencia y conocimientos en el tema, será quién llevará a cabo este rol principalmente, y nos acompañará durante todo el proceso de desarrollo del software.

DEFINICIÓN DEL EQUIPO Y SCRUM MASTER

Para la definición del Scrum Master se decidió tener un rol compartido, rotativo por sprint, designando a dos integrantes del grupo considerando sus aptitudes para dicho rol: los estudiantes Estrella Cholod y Alexis Pagura.

El equipo de desarrollo estará constituido por los cinco integrantes del grupo: Ezequiel Braicovich, Fabián Guidobaldi, Luis Quiroga, Estrella Cholod y Alexis Pagura, teniendo en cuenta los conocimientos técnicos de cada miembro y considerando para cada sprint el Scrum Master designado, en este caso se le fijará una cantidad menor de tareas del desarrollo, dado que tendrá que invertir tiempo en cubrir otro rol dentro de ese sprint.

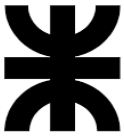
DEFINICIÓN DEL PRODUCT BACKLOG INICIAL

Roles

- Propietario/inquilino
- Encargado de servicios
- Visitante
- Encargado de Ingreso/Egreso

Listado de Épicas

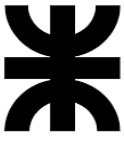
- Administrar propietarios
- Administrar el control de acceso
- Administrar barrios/countrys
- Administrar mapa del barrio/country
- Administrar reservas de instalaciones/servicios
- Administrar eventos
- Administrar Administrador
- Administrar Encargado de Ingreso/Egreso



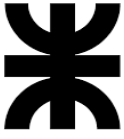
- Administrar Instalación/Servicio

Dado que este tipo de historias (Épicas) son muy grandes para estimar y construir, se analizó cada una para detallarla en User Story; aún así, no se quiso dejar de nombrarlas para tener una visión general del proyecto. A continuación, se especifica cada épica:

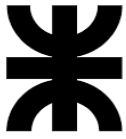
- Administrar propietarios
 - Registrar propietario
 - Registrar baja de propietario
 - Modificar propietario
 - Consultar propietario
 - Iniciar sesión como propietario
- Administrar el control de acceso
 - Registrar ingreso de propietario
 - Registrar egreso de propietario
 - Cancelar ingreso de propietario
 - Cancelar egreso de propietario
 - Modificar ingreso de propietario
 - Modificar egreso de propietario
 - Consultar ingreso de propietario
 - Consultar egreso de propietario
 - Registrar ingreso de visitante
 - Registrar egreso de visitante
 - Cancelar ingreso de visitante
 - Cancelar egreso de visitante
 - Modificar ingreso de visitante
 - Modificar egreso de visitante
 - Consultar ingreso de visitante
 - Consultar egreso de visitante



- Verificar identidad de visitante (por primera vez)
- Administrar barrios/countries
 - Registrar barrio/country
 - Registrar baja de barrio/country
 - Modificar barrio/country
 - Consultar barrio/country
- Administrar mapa del barrio/country
 - Agregar mapa
 - Eliminar mapa
 - Modificar mapa
 - Consultar mapa
 - Registrar ubicación de interés
 - Borrar ubicación de interés
 - Modificar ubicación de interés
 - Consultar ubicación de interés
- Administrar reservas de instalaciones/servicios
 - Registrar agenda de disponibilidad
 - Cancelar agenda de disponibilidad
 - Modificar agenda de disponibilidad
 - Consultar agenda de disponibilidad
 - Registrar solicitud de reserva
 - Cancelar solicitud de reserva
 - Modificar solicitud de reserva
 - Consultar solicitud de reserva
 - Confirmar solicitud de reserva
- Administrar eventos



- Registrar evento
 - Modificar evento
 - Consultar evento
 - Registrar baja de evento
 - Registrar lista de invitados
 - Modificar lista de invitados
 - Consultar lista de invitados
 - Registrar baja de lista de invitados
 - Generar invitaciones
- Administrar Administrador
 - Registrar Administrador
 - Registrar baja de Administrador
 - Consultar Administrador
 - Modificar Administrador
 - Iniciar sesión como Administrador
- Administrar Encargado de Ingreso/Egreso
 - Registrar Encargado de Ingreso/Egreso
 - Registrar baja de Encargado de Ingreso/Egreso
 - Consultar Encargado de Ingreso/Egreso
 - Modificar Encargado de Ingreso/Egreso
 - Iniciar sesión como Encargado de Ingreso/Egreso
- Administrar Instalación/Servicio
 - Registrar instalación/servicio
 - Registrar baja de instalación/servicio
 - Modificar instalación/servicio
 - Consultar instalación/servicio



- Administrar listados
 - Generar listados de propietarios por barrio
 - Generar listados de Encargado de Ingreso/Egreso por barrio
 - Generar listados de ingreso y egreso por período
 - Generar listados de ingreso y egreso por propietario
 - Generar listado de ingresos y egresos por visitante
 - Generar listado de personas dentro del predio en un período
 - Generar listado de instalaciones/servicios disponibles por barrio
 - Generar listado de ocupación de instalaciones/servicios
 - Generar informe de ingresos y egresos por hora, por día o por período
 - Generar listado de barrios

STORY MAP

Se lleva a cabo la planificación de cada sprint con sus User Story respectivas, considerando el tiempo que se dispone por semana del grupo de trabajo. se aclara que cada Sprint tendrá una duración de cuatro semanas la cual se tuvo en cuenta para la designación de tareas en la misma.

Por cuestiones de claridad, y para facilitar su lectura, el detalle de este apartado se incluye en el **ANEXO**.

DEFINICIÓN DE LA HERRAMIENTA DE SOFTWARE PARA GESTIÓN DEL PROYECTO A UTILIZAR

NOMBRE Y BREVE DESCRIPCIÓN DE LA HERRAMIENTA

Para llevar a cabo la gestión de nuestro proyecto hemos seleccionado la herramienta Taiga, la cual es considerada una de las herramientas de desarrollo de software líder de los equipos ágiles.

Taiga es una herramienta de software libre y código abierto, creada para gestionar y colaborar en proyectos ágiles, principalmente aquellos que utilizan metodología Scrum y kanban, además permite gestionar issues.

De igual manera Taiga posee otros módulos como wiki, videoconferencia (gracias a una solución de terceros), locaina, actualización de equipo y como si fuera poco gracias a su potente API permite la integración con servicios de terceros como Slack, GitHub, GitLab, Bitbucket, HipChat, Gogs, Hall entre otros.



Taiga es distribuida bajo la licencia de código abierto Affero GPL, está escrita en Django (backend) + AngularJS (frontend) y su código fuente está alojado en GitHub para que sea utilizado por la comunidad.

FORMA DE COORDINACIÓN DEL TRABAJO

Con respecto a la coordinación de trabajo, cada miembro del equipo tendrá una cuenta propia en Taiga y podrá realizar cambios en línea que se irán actualizando en tiempo real a los demás miembros.

Para llevar a cabo la distribución de tareas utilizaremos la herramienta ya mencionada Taiga, la cual nos permitirá llevar a cabo la administración del proyecto. Esta herramienta cuenta con su versión web y una versión mobile para dispositivos Android e IOS. A través de esta herramienta especificaremos las actividades que fueron realizadas en cada sprint y para cada user detallando en cuales participó cada miembro del equipo.

Con respecto a las horas de trabajo que serán destinadas por los integrantes del grupo, hemos estimado que se destinarán 5 horas semanales por cada integrante, lo cual nos da un total de 25 horas semanales.

Con respecto a la duración de los sprint, estos fueron definidos con una duración de 4 semanas. Para cada sprint serán destinadas 100 horas de trabajo aproximadamente.

CICLO DE VIDA DE LOS ARTEFACTOS A UTILIZAR EN EL DESARROLLO DEL PROYECTO

El principal artefacto que utilizaremos es el Product Backlog, el cual es un conjunto de todas las tareas que se pretenden desarrollar durante el desarrollo de un proyecto. El mismo está compuesto por las características del producto (user stories, temas y épicas), bugs que deben ser corregidos y trabajo técnico. Este artefacto persiste durante todo el desarrollo del proyecto y va sufriendo modificaciones en los ítems que lo componen durante la ceremonia del Product Backlog Refinement.

Las user stories que fueron definidas en el product backlog no podrán cambiar mientras un sprint se encuentre en ejecución, los cambios deberán ser realizados durante la ceremonia previamente mencionada, en la cual podremos agregar, eliminar o refinar los ítems que componen el product backlog.

Luego de esta ceremonia debemos definir cuáles serán los ítems que serán desarrollados en el siguiente sprint, para ello llevaremos a cabo la ceremonia del Sprint Planning. Como resultante de esta ceremonia obtenemos el Sprint Backlog. El ciclo de vida del Sprint backlog comienza cuando son definidos los ítems a ser desarrollados en este sprint y termina cuando todos los ítems han sido desarrollados. Como paso intermedio, este artefacto es modificado cada vez que una user story ha sido completada.

Como resultante de cada uno de estos avances del producto obtenemos las distintas Versiones del producto y su ciclo de vida comienza cuando la primer funcionalidad implementada es aceptada por



el cliente y termina cuando el producto está completamente terminado. Las versiones se van modificando a medida que una nueva funcionalidad es desarrollada al concluir un sprint.

DEFINICIÓN DE LA ESTRUCTURA DE LA DOCUMENTACIÓN DEL PRODUCTO

La estructura de la documentación del producto que se utilizará, a partir de la convención entre los miembros del grupo, es la siguiente:

Planificación y Seguimiento por Sprint

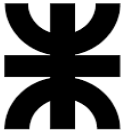
- Índice
- Introducción
- Definiciones Generales del Proyecto (Sprint 0)
 - Definición de Product Owner
 - Definición del Equipo y Scrum Manager
 - Definición del Product Backlog Inicial (Listado Epics, US)
 - Story Map (reunión de grooming)
 - Definición de la Herramienta de software para la gestión del proyecto a utilizar
 - Nombre y breve descripción de la herramienta
 - Forma de coordinación del Trabajo
 - Ciclo de Vida de los artefactos a utilizar en el desarrollo del proyecto: US, Epic, Caso de Uso, Caso de Prueba, Defecto, etc
 - Definición de la estructura de la documentación del producto (carpeta Definición de la Solución)
 - US (Template a utilizar)
 - Técnica de estimación a utilizar
 - Definición de la Tecnología a utilizar en el desarrollo del producto: Lenguaje de Programación, Base de Datos, etc.
 - Métricas del proyecto
 - Pautas de codificación y testing
 - Gestión de Configuración del Proyecto



- Gestión de Despliegue del Proyecto
- Gestión por Sprint (Sprint 1..n)
 - Introducción
 - Planificación
 - Product Backlog actualizado y priorizado
 - Story Map (reunión de grooming)
 - Sprint Backlog (Pila del Sprint)
 - Estimación del Sprint
 - Resultados
 - Resultado de la Review
 - Seguimiento del Sprint (Daily Meeting)
 - Métricas
- Resultado de la Retrospectiva (proyecto)
- Análisis del Impacto Ambiental

Definición de la Solución

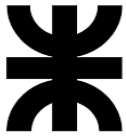
- Índice
- Introducción
- Listado de Epics y US (incrementar con cada Sprint)
- Definición de los temas agrupadores de las US (Paquetes)
- Para cada tema (paquete):
 - Descripción de US
 - Diagrama de Comunicación/Secuencia
 - Diagrama de clases de análisis
 - Identificación de Patrones de Diseño utilizados
 - Casos de Prueba
 - Diagrama de Clases



- Diagramas de Transición de Estados
 - DER
- Definición de la Arquitectura del Sistema
 - Diagramas de Despliegue
 - Ambiente de Implementación
- Glosario

Manuales

- Usuario
- Procedimientos
- Instalación



USER STORIES

El template a utilizar para las User Stories identificadas a lo largo del desarrollo del producto corresponde a la interfaz de creación de User Stories de la plataforma Taiga, que se muestra a continuación:

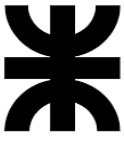
The screenshot shows the 'New User Story' interface in the Taiga platform. The main form has a 'Subject' field, a 'Description' text area with placeholder text 'Please add descriptive text to help others better understand this User Story', an 'Add tag' button, an 'Attachments' section with a 'Drop attachments here!' message, and a 'CREATE' button. On the right, there is a sidebar with a 'NEW' dropdown, a 'Status' dropdown, a 'Points' dropdown, and a list of tags including 'UX', 'Design', 'Front', 'Back', and 'total points'.

Dentro de esta plantilla, podemos distinguir los datos obligatorios, de los datos que son opcionales. Si bien el único dato obligatorio dentro de la plataforma es el **Nombre o Asunto** (Subject), dentro del equipo se definió también como un elemento obligatorio la **Descripción** de la User Story, para facilitar la comprensión de todos los miembros de lo que cada User Story representa.

La descripción de la User Story, a partir de una convención entre los miembros del equipo, debe incluir:

1. La lógica de funcionamiento de esa User Story, para que cualquier miembro del equipo pueda ser capaz de entenderla y completarla de principio a fin.
2. Los criterios de aceptación que definen las pruebas a realizar para verificar si esa User Story está completa o no.

Además, para cada User Story se pueden agregar los siguientes datos:



- La **estimación**, que si bien no es obligatoria al momento de crear la User Story, una vez realizada la estimación por todo el grupo es necesario agregar esta información para poder realizar la priorización del Product Backlog. Para la estimación, se tienen en cuenta 4 aspectos:
 - Experiencia de Usuario (UX)
 - Diseño
 - Front
 - Back
- La **asignación**, para identificar a los miembros del equipo responsables de esa User Story.
- Los **archivos adjuntos**
- El **estado**, que puede ser:
 - Nuevo
 - Listo
 - En proceso
 - Listo para probar
 - Hecho
 - Archivado
- **Fecha de expiración**, o fecha límite.

TÉCNICA DE ESTIMACIÓN A UTILIZAR

La técnica de estimación con la que vamos a trabajar a lo largo de todo el desarrollo de este producto será **Poker Planning**. Esta técnica consiste en una dinámica ágil, en la que el equipo utilizará una baraja de Poker modificada para llevar a cabo rondas de estimación con ayuda de estas cartas.

Para cada sesión de estimación, los integrantes del equipo deberán tener instalada la app **Scrum Poker Cards (Agile)**, la cual nos permitirá contar con una baraja estándar para llevar a cabo las estimaciones.

En cada baraja hay una pseudo-secuencia de Fibonacci modificada. Nosotros vamos a trabajar sólo con las primeras cartas para evitar estimar tareas demasiado grandes. Así cada participante tendrá las siguientes cartas: 0, 1/2, 1, 2, 3, 5, ? e infinito. El cero significa que la historia ya está hecha o no requiere ningún esfuerzo, el interrogante significa que nos falta información para estimar esa historia o tarea, y el infinito es que es demasiado grande y hay que investigar un poco para descomponerla en tareas o historias más pequeñas.



La sesión de estimación se lleva a cabo de la siguiente manera:

- Todo el equipo se reúne en un lugar donde no haya interrupciones para estimar, y todos deben conocer lo que se va a estimar. Si hay gente que no está al tanto de lo que se va a estimar, la sesión debe comenzar con una explicación y una sesión de preguntas para despejar cualquier duda sobre las historias que se van a estimar.
- Una por una, se leen y discuten las user stories. Una vez que todos tienen claro en qué consiste, cada uno elige una carta en función del esfuerzo que prevé requerirá esa user story. No es posible seleccionar un valor no incluido en la baraja. Sólo estimarán los que después van a desarrollar (el Scrum Master no estima, sólo participa para resolver dudas).
- Si no hay consenso, se abre la discusión. El debate no debe ser muy extenso; por ejemplo, se puede hacer que los que tienen la estimación más alta y más baja justifiquen su elección. Se repite la estimación nuevamente en busca de consenso. Si no se consigue a la segunda se vuelve a discutir. A la tercera, si no hay consenso, se escoge o bien la media o bien el máximo (mejor el máximo).

Al final de la sesión el resultado es una estimación consensuada y validada por todo el equipo para cada una de las user story o tareas seleccionadas.

DEFINICIÓN DE LA TECNOLOGÍA A UTILIZAR EN EL DESARROLLO DEL PRODUCTO

En el desarrollo de este producto, utilizaremos diferentes tecnologías, teniendo en cuenta las distintas plataformas que involucra el alcance de este producto.

Para el diseño y desarrollo de la interfaz web, se utilizarán los lenguajes HTML para la estructura de la página web, CSS para el diseño visual de la página, y diferentes librerías de JavaScript para generar la funcionalidad y la interacción del sitio web con el usuario.

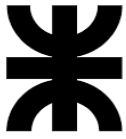
Por otro lado, para el desarrollo del backend y el frontend en la plataforma mobile, tanto Android como iOS, se utilizarán las librerías de React Native, desarrolladas por Facebook para el desarrollo de aplicaciones móviles que se comportan como aplicaciones nativas.

Finalmente, para lograr la persistencia de los datos que serán consultados por las distintas plataformas, utilizaremos el motor de base de datos MySQL, el cual estará alojado en un servidor web que se contratará para tal efecto.

MÉTRICAS DEL PROYECTO

Las métricas a emplear durante el desarrollo del proyecto serán el **Burndown Chart** y el **Velocity Chart**.

Burndown Chart:



Esta métrica se utiliza para hacer un monitoreo constante del progreso en cada sprint, y es una de las mejores herramientas de visibilidad de proyectos que asegura que el mismo se está desarrollando de manera fluida. Provee medición día a día de la cantidad de trabajo que queda en el Sprint. En el gráfico se puede ver la cantidad de trabajo restante que va quedando en el sprint; esta gráfica generalmente tiende a cero, por lo que, con la información histórica que provee, es fácil ver si el equipo va por el camino correcto. Usando esta métrica el equipo puede calcular la **Burndown Velocity**, que es la tasa de productividad promedio de cada día. Por ejemplo, un equipo puede tener alrededor de 50hs de trabajo; sabiendo eso, es posible calcular y estimar el día de terminación del sprint, incluso de la liberación basado en el modelo de trabajo restante. La ventaja del Burndown Chart es que se puede comparar la velocidad actual con la duración estimada del proyecto, y de esa manera saber qué necesita el equipo para terminar a tiempo.

Velocity Chart:

El propósito principal del Velocity Chart es tener una visión general de cuánto trabajo se ha entregado para cada sprint. Nos ayudará a tener una visión clara de las perspectivas futuras y de la carga de trabajo del equipo.

Los puntos de historias parcialmente completados o incompletos no deben contarse al calcular la velocidad. La velocidad debe rastrearse a lo largo del Sprint en el Sprint Burndown Chart y hacerse visible para todos los miembros del Equipo.

La velocidad es un mecanismo de retroalimentación clave para el equipo, ya que nos ayuda a medir si los cambios en el proceso están mejorando su productividad o empeorando. También facilita un pronóstico muy preciso de cuántas historias puede hacer el equipo en un Sprint. Para propósitos de pronóstico, se recomienda usar el promedio de la velocidad de los últimos tres Sprints.

PAUTAS DE CODIFICACIÓN Y TESTING

BUENAS PRÁCTICAS DE CODIFICACIÓN

Con respecto al desarrollo de la aplicación móvil hemos decidido trabajar en Android y estas son algunas de las buenas prácticas que se utilizan para optimizar Java para Android:

- Soportar el trabajo off-line siempre que sea posible persistiendo la información en el almacenamiento del dispositivo.
- Crear métodos con el menor número de parámetros posible. Esta es de primero de SOLID.
- Sacar fuera de los bucles las constantes y creación de nuevos objetos
- Siempre que vayamos a acceder a la misma posición de un array varias veces, es mejor guardar esa posición en una variable local y así evitar el acceso repetido al índice del array.
- Referenciar a null instancias de objetos que ya no se van a usar, para que el recolector de basura libere memoria.



- Usar excepciones únicamente cuando sea necesario ya que cada excepción lanza un nuevo objeto.

Con respecto al diseño Web de nuestra aplicación, estas son algunas buenas prácticas que se llevan a cabo para construir una página web atractiva:

- **Avance de la Animación:** El uso de HTML5, CSS y JQuery puede hacer las webs más atractivas y animadas, esto facilitará la interacción con el usuario.
- **Fotografías profesionales y de gran tamaño**
- **Videos presentan productos:** El vídeo adquiere importancia en los nuevos diseños web de grandes marcas.
- **Diseño web móvil primero y después escritorio:** Muchas marcas están comenzando con su diseño web en la versión móvil para después adaptarlo a la versión tablet, portátil y de escritorio.

PLAN DE TESTING

Existen muchas formas para realizar pruebas de software, pero entre las más importantes podemos destacar la prueba de **Caja blanca** y la prueba de **Caja negra**.

La prueba de caja blanca se centra en los detalles procedimentales del software, por lo que su diseño está fuertemente ligado al código fuente. Esta prueba permite verificar cada uno de los flujos de ejecución del programa analizando sus salidas ante la presencia de distintas entradas seleccionadas por el ingeniero de pruebas.

En el proyecto llevaremos a cabo los siguientes tipos de prueba de caja blanca, los cuáles serán combinados para lograr un testing efectivo:

- Cobertura de caminos básicos
- Cobertura de sentencias
- Cobertura de decisión
- Cobertura de condición
- Cobertura de condición/decisión
- Cobertura múltiple

Por otro lado, la prueba de Caja Negra nos enfocamos solamente en las entradas y salidas del sistema, sin preocuparnos en tener conocimiento de la estructura interna del programa de software. Para obtener el detalle de cuáles deben ser esas entradas y salidas, nos basamos en los requerimientos de software y especificaciones funcionales



Las pruebas mencionadas permiten probar cada una de las condiciones existentes en el programa, identificar claramente las entradas, salidas y estudiar las relaciones que existen entre ellas, permitiendo así maximizar la calidad de las pruebas y en dependencia del resultado se contará con un sistema más estable y confiable.

Se registra cada prueba de la siguiente manera:

Nombre de la prueba		Número de la prueba	
Tipo de prueba de testing			
Propósito			
Pre-requisitos			
Datos a utilizar	Entradas: Salidas:		
Acciones a realizar			
Resultado Final			
Criterios de aprobación	Errores Graves: Errores Medios: Errores Leves:		

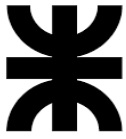
GESTIÓN DE CONFIGURACIÓN DEL PROYECTO

Para la gestión de configuración del proyecto, utilizaremos la plataforma GitHub, ya que es una de las más difundidas, y está integrada en la mayoría de los entornos de desarrollo.

En cuanto a la estructura del repositorio que utilizaremos, se empezará a definir al inicio del Sprint 1, a partir del análisis de la estructura de código fuente que se genera, a la cual también se le agregará un apartado para poder mantener también la documentación y todo el material asociado al desarrollo del producto.

La estructura del repositorio será la siguiente:

- Documentación
 - Carpeta 1 – Estudio Inicial
 - Carpeta 2 – Sprints



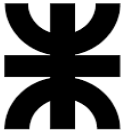
- Sprint 1
 - Sprint 2
 - Sprint 3
 - Sprint 4
 - Sprint 5
 - Sprint 6
 - Sprint 7
- Carpeta 3 – Descripción de la solución
 - Manuales
- Código Fuente
 - Mobile
 - Web
 - Base de Datos

GESTIÓN DE DESPLIEGUE DEL PROYECTO

El producto final constará de una aplicación Mobile, así como también una plataforma web. La aplicación Mobile se va a encontrar disponible tanto en App Store como en Play Store, y se podrá descargar de forma gratuita, lo que permitirá que cualquier persona que disponga de un dispositivo móvil pueda obtener acceso a la misma.

Por otra parte, el acceso y el uso de la plataforma web es muy diferente con respecto a la aplicación Mobile. En primer lugar, el acceso es restringido, y sólo los usuarios administrativos de ciertos countries, los cuales se hayan puesto en contacto con nuestro equipo de desarrollo y hayan solicitado la habilitación correspondiente, podrán trabajar con nuestra plataforma web, así como también los vecinos de dichos countries.

Luego de que un country se contacta con nosotros, y gestiona la implementación del servicio, procedemos a habilitar a los usuarios administrativos de dicho country; estos usuarios van a poder trabajar con la plataforma web, y desde la misma se le permitirá cargar todas las instalaciones que dispongan junto con sus horarios correspondientes. Otro requisito que se debe tener en cuenta es que debemos verificar que al momento de habilitar a un determinado country, el mismo cuente con un lector de la documentación de identificación y un lector de código QR.



ANEXO