**Confecciones Jomar**

**Plan de Pruebas**

**Preparado por**: Carolina Ceballos

**Historial de versiones**

|  |  |  |
| --- | --- | --- |
| Fecha | Autor | Comentario Versión |
| 29/09/2022 | Carolina Ceballos | Plan de pruebas 1.0 |

**Índice**

1. Introducción

2. Contextualización

3. Recursos

4. Alcance

5. Fuera de alcance

6. Estimación

7. Historias de usuario y criterios de aceptación

8. Estrategia

9. Criterios

10. Infraestructura

11. Entregables

12. Gestión de incidentes

13. Suposiciones

14. Riesgos

1. **Introducción**

El plan de prueba está diseñado como una línea de base para identificar lo que se considera dentro y fuera del alcance de las pruebas y cuáles son los riesgos, suposiciones y su estimación.

En este plan de pruebas se van a evaluar las distintas funcionalidades de la aplicación que se va a desarrollar para la compañía Confecciones Jomar y que tendrá como finalidad asignar citas en jornadas de RRHH. Lo anterior con el fin de verificar y validar que el producto satisfaga las expectativas del cliente y de los usuarios. Así como también reducir la probabilidad de que aparezcan defectos en producción.

1. **Contextualización**

La empresa Confecciones Jomar quiere construir un portal en el que se manejen dos tipos de usuarios: un usuario transaccional y otro administrador.

El usuario administrador será el encargado de crear las jornadas de salud en las diferentes regionales para que los usuarios se registren en el sistema y agenden su cita, éstas son creadas de manera simultánea para cada regional. Las jornadas disponibles son: Vacunación, Yoga,

Optometría y Nutrición.

Los usuarios deben registrarse con sus datos personales (la primera vez), iniciar sesión y agendar la cita. En el momento no se tiene una restricción por regionales, de manera que un empleado en Cali puede ver las jornadas activas en las demás regionales. Una vez realizada la cita, el sistema no debe mostrar al usuario los demás horarios disponibles, ya que tiene agendada una cita.

1. **Recursos**

|  |  |
| --- | --- |
| **Tester** | **% Participación** |
| Carolina Ceballos (QA manual y QA automation) | 100% |

1. **Alcance**

Las pruebas incluyen todas las funcionalidades que se describieron en el apartado de contexto**.**  Se realizarán pruebas funcionales y no funcionales para los distintos niveles de prueba y además en cuanto a pruebas no funcionales se aplicarán pruebas para validar el rendimiento y usabilidad del portal web.

Las pruebas funcionales y no funcionales serán realizadas por el QA manual y automation, sin embargo, las pruebas de usabilidad serán desarrolladas por el QA manual solamente.

1. **Fuera del Alcance**

No se encuentra dentro del alcance de este plan de pruebas la realización de pruebas de performance.

Por otro lado, no se aplicarán pruebas de seguridad debido a que las mismas no son necesarias por el momento.

1. **Estimación**

Se estima que la realización de las pruebas tendrá una duración de 12 semanas Aproximandamente

Por otro lado, para poder hacer esta estimación se utilizó un método de Planificación Póker con el fin de que los participantes del equipo llegaran a un consenso a través de las cards.

1. **Historias de usuario y criterios de aceptación:**

**HU1**

**Título:** Crear jornada de salud

**Descripción:** Como usuario administrador quiero ser capaz de crear jornadas de salud en las diferentes regionales a nivel nacional para que los usuarios se registren en el sistema y agenden su cita.

**Criterios de aceptación:**

* Cuando el usuario administrador cree una jornada de salud la misma se creará de manera simultánea en cada regional a nivel nacional.

**HU2**

**Título:** Registro de usuarios

**Descripción:** Como usuario transaccional quiero registrarme la primera vez con mis datos para poder iniciar sesión y agendar la cita.

**Criterios de aceptación:**

* Para registrar al usuario se le solicitará nombre completo, edad, email, teléfono y dirección de residencia.

**HU3**

**Título:** Consulta de jornadas activas

**Descripción:** Como empleado de una de las regionales a nivel nacional puedo consultar las jornadas activas y la información que se ha registrado.

**Criterios de aceptación:**

* Las jornadas activas podrán ser consultadas por cualquier empleado de cualquier regional a nivel nacional.

1. **Estrategia**

El equipo de desarrollo implementará pruebas unitarias y de integración como parte de su pipeline.

El equipo de testing realizará pruebas funcionales y no funcionales de forma manual y automatizada. Cada vez que una nueva versión llegue a testing se hará una prueba de regresión sobre aquellos escenarios que se consideren de prioridad alta.

Dentro de las pruebas no funcionales se ejecutarán pruebas de portabilidad, compatibilidad y usabilidad.

Al finalizar cada sprint se realizarán pruebas de aceptación de usuario. Al mismo tiempo, se aprovecharán esas reuniones para hacer pruebas de usabilidad.

Las pruebas de aceptación serán realizadas y coordinadas con las oficinas centrales. Los usuarios se escogerán según su nivel de experiencia en las áreas del sistema.

1. **Criterios**

Las pruebas del equipo de testing comenzarán cuando hayan pasado satisfactoriamente las pruebas unitarias.

El criterio de finalización de las pruebas será cuando se hayan ejecutado todas las pruebas planificadas o, si el tiempo apremia, al menos se hayan ejecutado los casos de prueba de prioridad alta y media.

El criterio para pasar a producción será que no haya errores bloqueantes o críticos sin resolver.

1. **Infraestructura**

Nuestros ambientes de prueba tendrán las siguientes configuraciones de Hardware:

* 1 portátil Toshiba
* 1 computador de escritorio HP con Windows 10
* 1 notebook Samsung con ubuntu 22.04 LTS

Por otro lado, los ambientes de prueba tendrán las siguientes configuraciones de Software:

* Maven con una versión 3.8.6: Es un gestor de dependencias.
* Git con una versión 2.36.1: Es un sistema de versionamiento
* Java con la versión 1.8.0\_333: Es un lenguaje de programación
* Aplicación Web de Pet Store que funciona en la siguiente dirección URL/pet.
* Postman con la versión 9.4: Es un probador de API’s
* Karate con la versión 2.2.3: Es un framework de automatización de pruebas
* Test Assured con la versión 3.0.2.3: Es un framework escrito en Java y diseñado para simplificar las pruebas sobre servicios basados en REST).

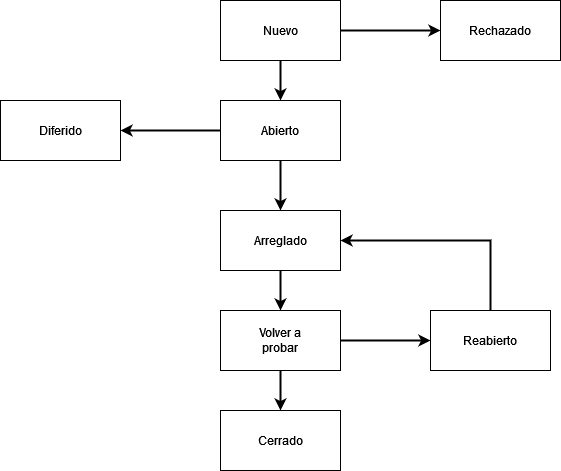
1. **Entregables**

Los entregables que se generarán son:

* Plan de pruebas
* Lista de escenarios de pruebas ejecutadas
* Informe de estado al finalizar cada sprint
* Reporte de defectos

1. **Gestion de incidentes**

Se utilizará el siguiente workflow para la gestión de defectos:



1. **Suposiciones**

La severidad será asignada por el tester y la prioridad por el product owner.

Las severidades que se utilizarán serán:

• ***Bloqueante:*** cuando una funcionalidad no se pueda utilizar y no haya otra forma de realizar esa misma acción.

***• Crítica:*** cuando una funcionalidad no se pueda utilizar pero exista un camino alternativo para realizar eso mismo.

***• Alta:*** cualquier defecto sobre una funcionalidad prioritaria del sistema o que tiene impacto secundario en una funcionalidad prioritaria del mismo.

***• Medio:*** cualquier defecto que impacte en una funcionalidad secundaria.

• ***Bajo:*** cualquier defecto cosmético o errores de ortografía.

1. **Riesgos**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No** | **Riesgos** | **Probabilidad**  **(1-5)** | **Impacto**  **(1-5)** | **Severidad**  **(Prob\*Impct)** | **Plan de Contingencia** |
| 1 | Retrasos en la implementación de las funcionalidades. | 2 | 5 | 10 | Evaluar el avance del desarrollo de las funcionalidades y re-planificar el avance del proyecto de ser necesario. |
| 2 | Los usuarios no están listos para las pruebas de aceptación (UAT) | 2 | 4 | 8 | Coordinar con las oficinas centrales la selección temprana de los usuarios. |
| 3 | Los datos ingresados por los usuarios no quedan registrados | 2 | 5 | 10 | Hacer pruebas con la base de datos para verificar que los datos se guardan correctamente. |
| 4 | Los empleados de las regionales no pueden traer los datos de los usuarios debido a fallas en las consultas. | 3 | 5 | 5 | Verificar que las consultas a la base de datos se escribieron correctamente. |
| 5 | Los datos ingresados por los usuarios tienen errores de ortografía o son escritos entre mayúsculas y minúsculas. | 4 | 3 | 15 | Crear un función que estandarice las letras mayúsculas y las minúsculas. |