

AÑO: 2022

REPORTE FINAL DE TESTING  
EQUIPO 5: DIGITAL BOOKING

## INTEGRANTES:

MONTIVERO, TOMAS

CALICANTON, GABRIELA

STIERLI, MAXIMILIANO

MAC, EMELY

DELGADO, DANIEL

MARTINEZ, DANIELA

## Introducción

El presente documento corresponde al informe final de testing, del proyecto integrador, del equipo 5 primer tracko de la carrera Certified Tech Developer de Digital House.

Este tiene como objetivo dar a conocer los resultados del sitio que creamos para reservas de hoteles.

Desarrollamos de un MVP trabajando con la metodología SCRUM, planteada en 4 sprints que contenían un número limitado de issues, que plantearon los requerimientos técnicos para llegar al producto esperado

Se implementaron casos de prueba, cuya ejecución se realizó con técnicas manuales y automatizadas; lo que permite al equipo 5 recopilar las evidencias sobre el avance del proyecto

## Link al proyecto

<http://frontend-c7-grupo-05.s3-website.us-east-2.amazonaws.com/>

## Estado de los Issues

|  |  |  |  |
| --- | --- | --- | --- |
| N° De Issue | Sprint | Contenido | Status |
| #12 | 1 | Planear casos de test basados en las historias de usuarios que existen en el sprint, y luego ejecutarlos.  Realizar también una prueba exploratoria sobre lo que se desarrolló durante el Sprint. | Closed |
| #12 | 1 | Validar el funcionamiento de la API Rest creada por el Back End durante el Sprint.  Utilicen Postman para las pruebas y creen los scripts de prueba necesarios. | Closed |
| #41 | 2 | Utilizando Selenium IDE (o Selenium Webdriver), crear las pruebas automatizadas necesarias para cubrir lo desarrollado durante los sprints I y II. Subir evidencia de la ejecución de los test a la tarea de GitLab (captura de pantalla). | In progress |
| #42 | 2 | ELECTIVA - Opcional: Crear test funcionales utilizando JEST Framework. | Opend |
| #43 | 2 | Crear test unitarios y definir el porcentaje de cobertura que quiere alcanzar el equipo (como mínimo un 40 %). | Open |
| #44 | 2 | Desarrollar los scripts de pruebas automatizados utilizando Postman para las API desarrolladas en este sprint. Agregar las API desarrolladas en este sprint a la colección generada en el sprint I. Ejecutar los scripts con el test runner de Postman y subir a GitLab el archivo JSON con los resultados de la ejecución (debe contener los scripts de sprint I y II). | Closed |
| #45 | 2 | Testing exploratorio  Realizar test exploratorio sobre lo agregado en el sprint. Entregar un documento con notas y forma de la organización del test exploratorio (sesiones, tours, escenarios, workflows, etc.).  Casos de prueba manuales  Agregar a la planilla de casos de prueba sobre las funcionalidades del sprint II.  Realizar mantenimiento de los casos de prueba manuales generados en el sprint I, en caso de ser necesario.  Clasificar los casos de prueba en suite de smoke y regression test.  Generar la suite de prueba con los casos que se puedan ejecutar, luego ejecutar esta suite de prueba.  Subir a GitLab la planilla de casos de prueba adjunta o el link al archivo.  Testing estático  Revisar cada una de las historias de usuario del sprint II y reportar defectos en la planilla correspondiente (por ejemplo, falta de definición de algún componente). | Closed |
| #67 | 3 | Realizar pruebas exploratorias en las páginas de reserva y de confirmación de reserva. Verificar y validar las funcionalidades y la interfaz de usuario.  Agregar test cases manuales sobre las funcionalidades a entregar en este sprint y ejecutar un ciclo de prueba seleccionando tests cases necesarios, teniendo en cuenta de no duplicar esfuerzo (evitando de ejecutar los que ya se encuentran automatizados). | Closed |
| #67 | 3 | Añadir las pruebas necesarias para los endpoints de productos, reservas y usuarios a la colección creada previamente en Postman (scripts de pruebas automatizados).  (ELECTIVA): Añadir / Crear un script relacionado al proceso de reserva en las pruebas de regresión automatizadas creadas con Selenium IDE (o Selenium Web Driver).  (ELECTIVA): Crear y ejecutar las pruebas unitarias (unit testing) de las páginas de reserva y confirmación de reserva a través de JEST framework. | Closed (obligatorio), Open (opcional) |
| #75 | 4 | Agregar a la colección de Postman script automatizados para probar los métodos agregados en este sprint.  OPCIONAL: Crear y ejecutar test unitarios utilizando Selenium sobre las funcionalidades de la página de mis reservas (las realizadas por usuario) y sobre la página formulario de producto (la que se ingresa con rol administrador). | Closed |
| #76 | 4 | Realizar y registrar el testing exploratorio sobre las funcionalidades de la página de mis reservas y sobre la página formulario de producto.  Agregar casos de prueba Manuales para ambas páginas y clasificar los mismos en Smoke y Regression Test. | Closed |
| #77 | 4 | Ejecutar los casos de prueba que pertenecen a la suite de Regresión y generar un Reporte Final de Prueba que contenga el siguiente resumen (pueden ser gráficos o tablas agregadas en el template) :  Cantidad de casos de prueba creados por funcionalidad/historia de usuario tanto manuales como automáticos.  Cantidad de casos de prueba ejecutados, pasados y fallados.  Cantidad de defectos por estado y prioridad/severidad  Como exit criteria (criterio para finalizar el testing) no se debería tener defectos en estado abierto de severidad crítica y/o bloqueante. | Closed |

## Pruebas manuales y de regresión

El 95 % de los test manuales y de regresión fue aprobado en un comienzo, con las correcciones reportadas se llevó el valor máximo a un 98 % de efectividad en la ejecución de los test.

## Postman

Los test realizados en postman fueron efectivos en un 98 %, se continúan las mejoras para llevar la efectividad al 100%.

## Pruebas con Selenium

Quedan pendientes de corrección.

Porcentaje total de tests realizados y con aprobación.

En el proceso se crearon 55 casos de prueba a lo largo de los 4 sprint, teniendo la

ejecución del 100 % de los casos de pruebas planteados y fueron aprobadas

## Conclusión

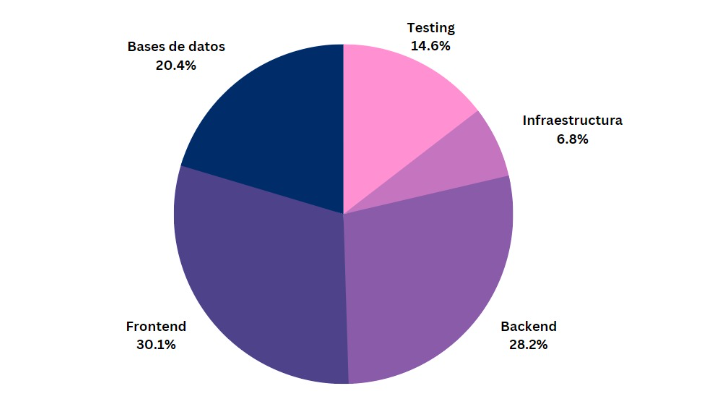
En el desarrollo del proyecto integrador se realizaron 80 issues, de los cuales 12 fueron requerimientos para el equipo QA. Las pruebas realizadas fueron sido secuenciales. Fue probada la Api creada por el equipo de Backend y que el consumo de la Api, por parte de la página web, funcione correctamente.

La página Web fue creada desde cero, y en un primer momento se desarrollaron las funcionalidades enfocadas a la estructura general del proyecto y los estilos, para luego anexar desde el backend la configuración requerida para almacenar los datos de los productos, y relacionar estos con las reservas del usuario o administrador.

Fue levantada una infraestructura dentro de AWS, que también fue testeada para que funcione en los horarios de 24/7.

Sin el rol QA no habría sido posible ser detallista en la exploración de errores y su pronta solución, incluso en el último minuto de la creación de nuestro Digital Booking fue puesto a prueba, pasando debidamente los test.

Los issues en general fueron asignados a un integrante específico de cada rol, pero algunos issues fueron compartidos entre especialidades logrando una distribución de tareas como se puede observar en el siguiente diagrama:



El equipo se mostró resolutor, de bueno clima de trabajo, con muchas ganas de aprender y seguir aprendiendo.