

CERTIFIED TECH DEVELOPER

Proyecto Final Integrador

Equipo 7

Aurrecoechea, Ignacio.

Faccini, Trinidad.

Medina, Sofía.

Murga, Luciana.

Nolan, Brenda.

Reporte final de Testing



COMO EN CASA

Contenido

Introducción	3
Resumen de las actividades de prueba	3
Alcance	3
Dentro del Alcance	3
Fuera de Alcance	3
Tipos de Pruebas Ejecutadas	3
Enfoque de la Prueba	4
Exit Criteria	4
Resumen de Resultados	5
Diseño de Pruebas	5
Ejecución de Pruebas	6
Ejecución Manual	6
Ejecución Automática	6
Reporte de Defectos	6
Todos los defectos	6
Defectos por prioridad	7
Defectos por Severidad	7
Defecto por Estado	7
Defectos Creados vs Resueltos	8
Defectos Abiertos	9
Lecciones Aprendidas / Conclusión	9

Introducción

Este documento es el Informe Final de Pruebas del sistema “Como en casa”. El propósito de este documento es proveer evidencia de que el Exit Criteria para el proceso de Testing se cumplió y por lo tanto, se concluye la fase de pruebas y puede cerrarse. Se demuestra que los Issues de GitLab relacionados con testing fueron implementados desde los Sprint 1 al 4. Este documento va a ser utilizado como entrada para la revisión general de las actividades de prueba y para tomar la decisión si el sistema cumple con las expectativas.

Resumen de las actividades de prueba

“Como en casa” es un sitio web de reservas de alojamientos como casas, cabañas y/o departamentos tanto en Argentina como en Colombia.

Los visitantes del sitio podrán navegar por las diferentes secciones, buscar alojamientos tanto por categorías como por ciudades y fechas disponibles. También conocer detalles de los mismos, ver galerías de fotos, puntuaciones otorgadas por usuarios que ya visitaron el establecimiento, la dirección, características y comodidades, políticas de cancelación, seguridad e higiene y especificaciones legales.

En el caso de que quieran realizar una reserva, se deben registrar como usuario y allí poseen más beneficios como consultar el historial, en el caso de que ya hayan reservado en nuestra página y los establecimientos que guardaron como favoritos. Además, aquel usuario que cuente con permisos de administrador podrá registrar, actualizar o eliminar alojamientos.

La url para poder ingresar es: <http://52.67.178.177/>

El estado de las actividades de prueba desde el sprint 1 hasta el 4 es cerrado, es decir que se completaron todas las tareas.

Confeccionar el reporte final de testing #76 · created 1 week ago by Ezequiel Cortés · Sprint 4 · Obligatoria · Testing / QA	CLOSED · 0 updated 7 hours ago
Implementar testing manual #75 · created 1 week ago by Ezequiel Cortés · Sprint 4 · Obligatoria · Testing / QA	CLOSED · 0 updated 7 hours ago
Implementar testing automatizado #74 · created 1 week ago by Ezequiel Cortés · Sprint 4 · Obligatoria · Testing / QA	CLOSED · 0 updated 7 hours ago
Implementar tests automatizados #59 · created 3 weeks ago by Ezequiel Cortés · Sprint 3 · Obligatoria · Testing / QA	CLOSED · 0 updated 1 week ago
Testing Manual #36 · created 1 month ago by Daniel Tejerina · Sprint 2 · Obligatoria · Testing / QA	CLOSED · 1 updated 4 weeks ago

Alcance

Dentro del Alcance

Las funcionalidades del sistema que fueron probadas fueron en base a los requerimientos tanto obligatorios como electivos entre ellos:

- Operaciones CRUD de:
 - Categorías
 - Ciudades
 - Características
 - Imágenes
 - Productos
 - Usuarios
 - Roles
- La creación de las páginas de:
 - Crear cuenta
 - Login
 - Home
 - Detalle del producto
 - Reserva del producto
 - Reserva exitosa del producto
 - Registrar producto
 - Creación exitosa del producto.
- Dentro de las páginas funcionalidades como:
 - Filtrar alojamientos por ciudad, por fechas disponibles y/o por categorías.
 - Poder seleccionar un producto, ver en detalle sus características, ver fechas disponibles y no disponibles de reserva, poder realizar una reserva del alojamiento seleccionado.
 - En el caso de estar logueado, poder chequear reservas e historial de reservas, solo si ya se ha efectuado una.
 - Y en el caso de estar logueado y tener permisos de administrador poder registrar, modificar o eliminar un producto.

Fuera de Alcance

Principalmente lo que no se testeó fue:

- Pruebas de integración de parte del Backend pero desde el IntelliJ porque desde Postman si se realizaron.
- De parte de las pruebas automatizadas con Jest no se realizaron pruebas en el componente DatePicker que es el utilizado en los calendarios ya que es un componente externo que solo es llamado cuando se necesita, pero si se realizaron pruebas manuales, tampoco de los componentes useContext y UsuarioContext ya que no los utilizamos; todos los componentes de Formik que son para validaciones solo se testearon de forma manual.

- Métodos agregados para mandar un email en el momento que un usuario crea una cuenta (actividad opcional).



Confirmación de cuenta

Recibidos



comoencasa.g7@gmail.com Ayer
para mí ▾



Hola María Luciana! Te damos la bienvenida a Como En Casa..

Hi María Luciana! Welcome to Como En Casa.

Tipos de Pruebas Ejecutadas

	Sprint 1	Sprint 2	Sprint 3	Sprint 4
Pruebas dinámicas	x	x	x	x
Pruebas estáticas	x	x	x	x
Pruebas unitarias	x	x	x	x
Pruebas de integración	x	x	x	x
Pruebas funcionales	x	x	x	x
Pruebas de casos de uso		x	x	x
Pruebas de usabilidad				x
Pruebas de regresión		x	x	x
Pruebas de humo		x	x	x
Pruebas de confirmación		x	x	x
Pruebas exploratorias		x	x	x

Enfoque de la Prueba

Desde el primero al cuarto sprint, se desarrollaron casos de prueba, tantos positivos como negativos y de caja negra como de caja blanca, contemplando en su mayoría las funcionalidades del sistema, para luego, cerca de finalizar cada sprint, se pudieran ejecutar anotando los defectos en la correspondiente planilla de defectos. A partir del segundo sprint esos casos de prueba que íbamos realizando ya los íbamos clasificando en test de regresión o de humo de acuerdo a las nuevas funcionalidades que se iban agregando, tratando de comprobar que lo que funcionaba antes siga haciéndolo. A su vez, se realizaron pruebas basadas en la experiencia, pruebas exploratorias, con el objetivo de probar el correcto funcionamiento de algunas funcionalidades y de las historias de usuario donde si ocurría un defecto también se lo anotaba en la planilla de defectos con el objetivo de corregirlo.

Además, en todos los sprints se realizaron pruebas estáticas referidas a la documentación, si notábamos que faltaba ser más específico en una historia de usuario nos dirigíamos al Product Owner en busca de una solución.

Aparte de realizar test manuales también realizamos test automatizados con Jest para cubrir el Front End y con Postman el Back End enfocándonos en los test unitarios y de integración. Como estas pruebas se realizaron desde el inicio del proyecto, cada vez que se generaba un nuevo build se volvían a correr las anteriores.

Con Jest realizamos 35 conjuntos de pruebas y 56 tests logrando una cobertura del 42% en statements. De parte de Postman creamos test para crear, listar, actualizar y borrar para las tablas de categorías, ciudades, características, productos, reservaciones y usuarios. Ambos cumpliendo con el exit criteria preestablecido con el grupo.

Link Planilla de Casos de Prueba:

<https://docs.google.com/spreadsheets/d/1j9Hvtwo6OWUo0lgwYt0rpk1fZwErXvWd/edit#gid=1960225858>

Link Planilla de Defectos:

https://docs.google.com/spreadsheets/d/1hggVR2_WNqhCU5cdrgo937ft39SumQj8/edit#gid=1920107956

Link del documento de Pruebas Exploratorias:

https://docs.google.com/document/d/1vw4yyqBtcRESzYpsPU73kVy10cr94CQqVCpalzZMZ_F4/edit?usp=sharing

Link del documento de Jest:

<https://docs.google.com/document/d/1622gbXlmRIU3qlmmEH4YJkC57acMkLgNuUoV9CcHmb8/edit?usp=sharing>

Exit Criteria

Se definió los siguientes criterios de aceptación para finalizar las pruebas:

- No se debe tener defectos en estado abierto de severidad crítica y/o bloqueante.
- La ejecución de los test realizados en Postman deben tener un 100% de pass rate.
- La ejecución de los test realizados por Jest deben alcanzar al menos el 40% de cobertura total en statements y lines (también está planteado así en nuestro package.json).

Resumen de Resultados

Diseño de Pruebas

	Test manual	Test automatizado de integración	Test automatizado unitario	Test total
Crear cuenta	3	5	2	9
Login	4	5	1	9
Cerrar sesión	1	0	0	1
Validación de datos	4	4	1	9
CRUD categorías	1	16	2	13
CRUD ciudades	2	16	0	18
CRUD características	0	14	0	14
CRUD productos	4	16	0	20
CRUD reservas	4	15	3	22
Páginas crear cuenta y login	7	0	3	10
Filtrar alojamiento por categoría	2	1	0	3
Filtrar alojamiento por ciudad	3	1	0	4

Filtrar alojamientos por fechas	4	2	0	6
Filtrar por ciudad y fechas	1	1	0	2
Ícono en el header para volver atrás	1	0	0	1
Presionar logo te lleve al home	1	0	1	2
Marcar como favorito un producto	1	0	0	1
Página del Home	10	0	1	2
Presionar "Ver Más" te lleve a la página de detalle del producto seleccionado	2	0	1	2
Página del detalle del producto	3	0	0	3
Carrusel de imágenes	2	1	1	4
Fechas disponibles y no disponibles en el calendario	5	1	0	6
Botón "Iniciar Reserva"	3	0	1	4
Página detalle de la reserva	4	0	1	4
Formulario con campos disabled	2	0	1	3
Desplegable con horarios de entrada	0	0	1	1
Botón "Confirmar Reserva"	2	0	1	3

Página de reserva exitosa	2	0	2	4
Link volver al home	1	0	1	2
Crear usuario con rol de administrador	0	3	1	4
Página registrar producto	2	0	5	7
Responsive	1	0	3	4
Botón "Registrar alojamiento"	1	0	0	1
Ícono en el header que redireccione a la página de creación de producto	1	0	1	2

Ejecución de Pruebas

Ejecución Manual

	Test pasado	Test fallado	Test no ejecutado	Test total
Crear cuenta	3			3
Login	4			4
Cerrar sesión	1			1
Validación de datos	4			4
CRUD categorías			x	0
CRUD ciudades			x	0

CRUD características			x	0
CRUD productos			x	0
CRUD reservas			x	0
Página de crear cuenta y login	7			7
Filtrar alojamiento por categoría	2			2
Filtrar alojamiento por ciudad	3			3
Filtrar alojamientos por fechas	4			4
Filtrar por ciudad y fechas	1			1
Ícono en el header para volver atrás	1			1
Presionar logo te lleve al home	1			1
Marcar como favorito un producto	1			1
Página del Home	10			10
Presionar "Ver Más" te lleve a la página de detalle del producto seleccionado	2			2
Página del detalle del producto	3			3
Carrusel de imágenes	2			2
Fechas disponibles y no disponibles	5			5

en el calendario				
Botón "Iniciar Reserva"	3			3
Página detalle de la reserva	4			4
Formulario con campos disabled	2			2
Botón "Confirmar Reserva"				2
Página de reserva exitosa	2			2
Link volver al home	1			1
Página registrar producto	2			2
Ícono en el header que redireccione a la página de creación de producto	1			1
Botón "Registrar alojamiento"	1			1
Guardar fechas de los calendarios para no tener que agregarla en todos		1		1

Los métodos del CRUD no se ejecutaron manualmente porque esos métodos los probamos de forma automatizada por Postman. Y la funcionalidad de conservar las fechas de los calendarios con el fin de que el usuario no tenga que cargarlos en todos los calendarios no se logró por lo cual se encuentra en la planilla de defectos.

Ejecución Automática

Resultados de la ejecución de los test unitarios en Jest e informe de cobertura:

Test Suites:

35 passed, 35 total

Tests:

56 passed, 56 total

Snapshots:

0 total

Time:

35.316 s

Ran all test suites.

All files

44.02% Statements

221/502

18.54% Branches

23/124

32.88% Functions

49/149

44.53% Lines

220/494

File		Statements		Branches		Functions		Lines	
src	<div></div>	62.5%	5/8	0%	0/2	50%	1/2	62.5%	5/8
src/Components	<div></div>	50%	5/10	100%	0/0	20%	1/5	50%	5/10
src/Components/AddImage	<div></div>	100%	2/2	100%	0/0	100%	1/1	100%	2/2
src/Components/Administration	<div></div>	53.57%	15/28	0%	0/6	30%	3/10	53.57%	15/28
src/Components/Authentication	<div></div>	57.14%	12/21	0%	0/4	36.36%	4/11	57.14%	12/21
src/Components/Block	<div></div>	50%	4/8	100%	0/0	50%	1/2	50%	4/8
src/Components/BookingPage	<div></div>	8.69%	2/23	0%	0/6	50%	1/2	8.69%	2/23
src/Components/Button	<div></div>	100%	2/2	100%	0/0	100%	1/1	100%	2/2
src/Components/Calendar	<div></div>	34.28%	12/35	25%	2/8	14.28%	1/7	36.36%	12/33
src/Components/Categories	<div></div>	54.54%	6/11	100%	0/0	25%	1/4	54.54%	6/11
src/Components/Confirmation	<div></div>	100%	3/3	100%	0/0	100%	1/1	100%	3/3
src/Components/Footer	<div></div>	100%	3/3	100%	0/0	100%	1/1	100%	3/3
src/Components/Gallery	<div></div>	54.54%	6/11	100%	0/0	50%	3/6	54.54%	6/11
src/Components/Header	<div></div>	44.44%	4/9	50%	1/2	20%	1/5	44.44%	4/9
src/Components/Heading	<div></div>	80%	4/5	100%	0/0	50%	1/2	80%	4/5
src/Components/Home	<div></div>	100%	4/4	100%	0/0	100%	1/1	100%	4/4
src/Components/InputForm	<div></div>	42.85%	3/7	25%	1/4	50%	1/2	42.85%	3/7
src/Components/LocationSearch	<div></div>	50%	4/8	100%	0/0	33.33%	2/6	66.66%	4/6
src/Components/Login	<div></div>	50%	3/6	0%	0/2	50%	1/2	50%	3/6
src/Components/LoginRegister	<div></div>	75%	6/8	50%	2/4	100%	1/1	75%	6/8
src/Components/Map	<div></div>	100%	3/3	100%	0/0	100%	1/1	100%	3/3
src/Components/MenuMobile	<div></div>	42.85%	9/21	35.71%	5/14	40%	2/5	42.85%	9/21
src/Components/MyReservations	<div></div>	88.88%	16/18	50%	2/4	71.42%	5/7	88.23%	15/17
src/Components/Policies	<div></div>	100%	3/3	100%	0/0	100%	1/1	100%	3/3
src/Components/ProductPage	<div></div>	7.89%	3/38	0%	0/12	12.5%	1/8	8.1%	3/37
src/Components/Recomendations	<div></div>	46.15%	6/13	0%	0/2	40%	2/5	46.15%	6/13
src/Components/Register	<div></div>	57.14%	4/7	25%	1/4	50%	1/2	57.14%	4/7
src/Components/ScrollToTop	<div></div>	46.66%	7/15	25%	1/4	16.66%	1/6	46.66%	7/15
src/Components/Searcher	<div></div>	31.39%	27/86	0%	0/26	6.25%	1/16	32.14%	27/84
src/Components/User	<div></div>	45.83%	11/24	50%	2/4	11.11%	1/9	45.83%	11/24
src/Components/UserFavorites	<div></div>	80%	4/5	100%	0/0	66.66%	2/3	80%	4/5
src/Components/UserHeader	<div></div>	36.66%	11/30	33.33%	4/12	20%	1/5	36.66%	11/30
src/Components/UserPreferences	<div></div>	100%	3/3	100%	0/0	100%	1/1	100%	3/3
src/Components/UserPreferences	<div></div>	100%	3/3	100%	0/0	100%	1/1	100%	3/3
src/Components/UserReservations	<div></div>	37.5%	9/24	50%	2/4	12.5%	1/8	37.5%	9/24

Resultados de la ejecución de los test de integración en Postman:

Tabla User:

All Tests	Passed (12)	Failed (0)
		1
▶ POST	ADD User [Sofia] SIGNUP	1 0
▶ POST	ADD User [Sofia] LOGIN	1 0
▶ POST	ADD UserAdmin [Luciana] SIGNUP	1 0
▶ POST	ADD UserAdmin [Luciana] LOGIN	1 0
▶ POST	ADD UserAdmin [Brenda] SIGNUP	1 0
▶ POST	ADD UserAdmin [Brenda] LOGIN	1 0
▶ POST	ADD UserAdmin [Trini] SIGNUP	1 0
▶ POST	ADD UserAdmin [Trini] LOGIN	1 0
▶ POST	ADD User [Iñaki] SIGNUP	1 0
▶ POST	ADD User [Iñaki] LOGIN	1 0
▶ GET	FIND User BY id [1]	1 0
▶ GET	FIND User BY email [sofiamedina02@gma...	1 0

Tabla Categories:

All Tests	Passed (18)	Failed (0)
		1
▶ POST	ADD Category [Montaña]	2 0
▶ POST	ADD Category [Playa]	4 0
▶ POST	ADD Category [Campo]	2 0
▶ POST	ADD Category [Ciudad]	2 0
▶ POST	RE - ADD Category [Montaña]	1 0
▶ GET	LIST ALL Categories	1 0
▶ GET	FIND Category BY id [1]	2 0
▶ GET	FIND Category BY titulo [Playa]	2 0
▶ PUT	UPDATE Category [id:3-urlImagen]	2 0

Tabla Cities:

All Tests Passed (17) Failed (0)

RUN SUMMARY

			1
▶	POST	ADD City [BsAs]	2 0
▶	POST	ADD City [Rosario]	3 0
▶	POST	ADD City [VLA]	2 0
▶	POST	ADD City [P.Madryn]	2 0
▶	POST	RE - ADD City [BsAs]	1 0
▶	GET	LIST ALL Cities	1 0
▶	GET	FIND City BY id [1]	2 0
▶	GET	FIND City BY titulo [Rosario]	2 0
▶	PUT	UPDATE City [id:4 - name]	2 0

Tabla Features:

All Tests Passed (17) Failed (0)

			1
▶	POST	ADD Feature [Cocina]	2 0
▶	POST	ADD Feature [Aire acondicionado]	2 0
▶	POST	ADD Feature [Estacionamiento]	2 0
▶	POST	ADD Feature [Wifi]	2 0
▶	POST	ADD Feature [Pileta]	2 0
▶	POST	ADD Feature [Televisión]	2 0
▶	GET	LIST ALL Features	1 0
▶	GET	FIND Feature BY id [1]	2 0
▶	PUT	UPDATE Feature [id:4-icon]	2 0

Tabla Products:

All Tests Passed (18) Failed (0)

			1
▶	POST	ADD Product [Cabañas Peuma]	2 0
▶	POST	ADD Product [Cabañas del Sol]	2 0
▶	POST	ADD Product [Norway]	2 0
▶	POST	ADD Product [Las Lomitas]	2 0
▶	POST	RE - ADD Product [Cabañas Peuma]	1 0
▶	GET	LIST ALL Products	1 0
▶	GET	FIND Product BY id [1]	2 0
▶	GET	FIND Product BY name [Norway]	2 0
▶	GET	FIND Product BY date [2021-12-02,2021-1...	1 0
▶	GET	FIND Product BY availability	1 0
▶	PUT	UPDATE Product [id:3-latitude]	2 0

Tabla Reservations:

All Tests Passed (14) Failed (0)

			1
▶	POST	ADD Reservation [Cabañas Peuma]	2 0
▶	POST	ADD Reservation [Cabañas del Sol]	4 0
▶	POST	ADD Reservation [Norway]	1 0
▶	POST	ADD Reservation [Las Lomitas]	1 0
▶	GET	LIST ALL Reservations	1 0
▶	GET	FIND Reservation BY id [3]	1 0
▶	GET	FIND Reservation BY user id [1]	1 0
▶	GET	FIND Reservation BY id product [2]	1 0
▶	PUT	UPDATE Reservation [id:2-arrivalDate]	2 0

Delete Reservations:

All Tests Passed (6) Failed (0)

▶ DELETE DELETE Reservation [id:1]	1 0	1
▶ GET FIND Reservation BY id [1]	1 0	
▶ DELETE DELETE Reservation [id:2]	1 0	
▶ DELETE DELETE Reservation [id:3]	1 0	
▶ DELETE DELETE Reservation [id:4]	1 0	
▶ GET LIST ALL Reservations	1 0	

Delete Categories, Cities, Features, Products:

All Tests Passed (26) Failed (0)

▶ DELETE DELETE Product [id:3]	1 0	1
▶ GET FIND Product BY id [3]	1 0	
▶ DELETE DELETE Product [id:1]	1 0	
▶ DELETE DELETE Product [id:2]	1 0	
▶ DELETE DELETE Product [id:4]	1 0	
▶ GET LIST ALL Products	1 0	
▶ DELETE DELETE Feature [id:1]	1 0	
▶ GET FIND Feature BY id [1]	1 0	
▶ DELETE DELETE Feature [id:2]	1 0	
▶ DELETE DELETE Feature [id:3]	1 0	
▶ DELETE DELETE Feature [id:4]	1 0	
▶ DELETE DELETE Feature [id:5]	1 0	
▶ DELETE DELETE Feature [id:6]	1 0	
▶ GET LIST ALL Features	1 0	

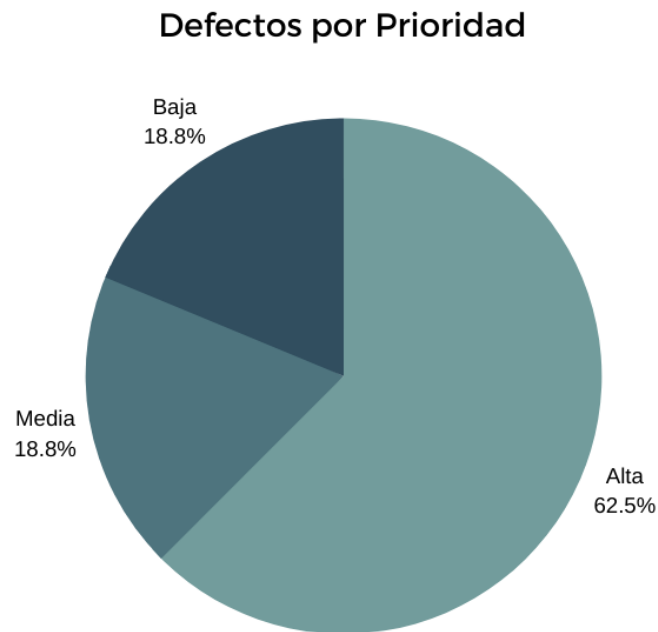
Tanto los test ejecutados en Postman como en Jest demuestran que se cumplió con el Exit Criteria establecido con el grupo.

Reporte de Defectos

Todos los defectos

La siguiente sección muestra información con respecto al número total de defectos que se han presentado durante la duración de la fase de prueba.

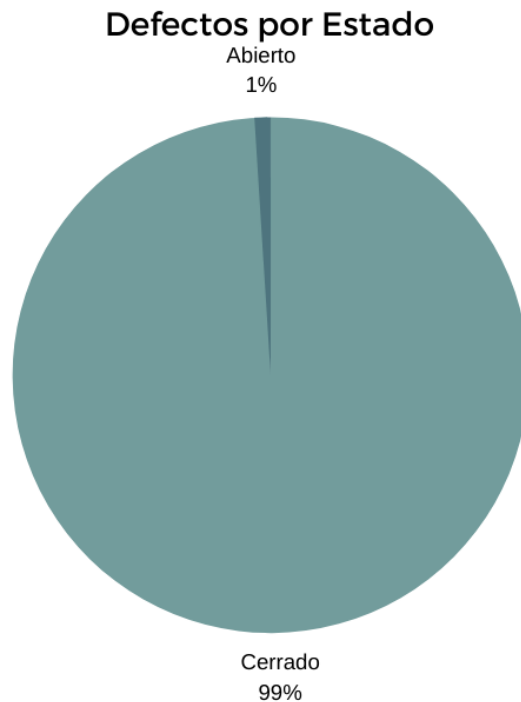
Defectos por prioridad



Defectos por Severidad



Defectos por Estado



Defectos Abiertos

La siguiente sección muestra información con respecto al número total de defectos que permanecen abiertos al final de la fase de pruebas.

Únicamente tenemos un defecto abierto pero no es de estado crítico ni bloqueante, tiene prioridad y severidad baja. Trata sobre guardar los datos del usuario para que no tenga que colocar en todos los calendarios la fecha que desea para reservar el alojamiento. Dentro del reporte de defectos se lo encuentra con el id BUG-15.

Lecciones Aprendidas / Conclusión

Testing es el proceso de verificación y validación de una aplicación. Dentro de nuestro proyecto nos resultó fundamental para poder detectar defectos y así poder resolverlos rápidamente ahorrándonos tiempo con el objetivo de entregarle a nuestro cliente un software de calidad.

Además, nos ayudó a comprobar que nuestro sistema cumple con todas las premisas preestablecidas ya que las pruebas se desarrollaban de forma temprana de acuerdo a los requerimientos del cliente. Y no solo se testeaba sobre las funcionalidades sino que sobre la documentación también buscando especificidad y no cometer errores en el código por malos entendidos.

En fin, el Testing es realmente importante para una experiencia satisfactoria del usuario y si no se realiza en paralelo con el código luego resultará más difícil resolver esos errores.

