

# PROYECTO 03 - PRUEBAS DE APLICACIONES WEB

¡Es hora de comenzar el proyecto! En este proyecto, probarás la funcionalidad de compartir un automóvil de Urban Routes.

## Cómo entregar el proyecto

Hay dos documentos en los que tendrás que trabajar:

1. La plantilla de Google Sheets: completa todas las tareas de esta plantilla.
2. La plantilla de Google Docs: adjunta aquí un enlace a la tabla de Google Sheets después de completar las tareas, así como todos los demás enlaces requeridos.

En los títulos de los documentos escribe tu nombre, tu(s) apellido(s), el número del grupo y el número del sprint para el que estás haciendo esta tarea. Por ejemplo: "Anna Smith, 2do. grupo, 3er. sprint".

Permite el acceso a los comentarios a cualquier persona que tenga el enlace y adjúntalo a la plataforma en la pestaña "Supervisión". La pestaña aparecerá en la lección "Proyecto para el tercer sprint: Entrega".

## Lo que necesitas probar

En este proyecto, probarás la funcionalidad de **compartir un automóvil** en dos entornos:

- Google Chrome, resolución de pantalla 800x600
- Firefox, resolución de pantalla 1920x1080

Analiza los **diseños y requisitos** de compartir un automóvil. Si los requisitos y los diseños no coinciden, concéntrate en los requisitos.

Explora la aplicación en el banco de pruebas:

Servidor

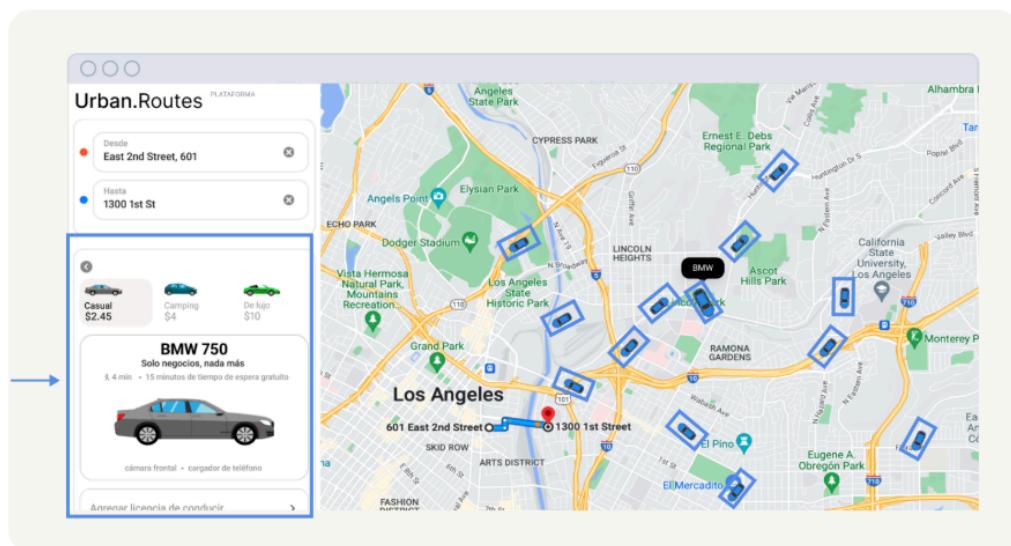


Primero, debes preparar la documentación de prueba para el diseño y la funcionalidad del formulario de reserva y luego realizar pruebas.

Antes de comenzar, estudia y analiza detenidamente los diseños de compartir un automóvil en Figma, en la pestaña "Rutas para compartir un automóvil".

# Bootcamp Ingeniero de calidad de software - TripleTen

## QA Engineer - Blanca Sánchez Gutiérrez



El proyecto tiene 5 pasos obligatorios y 1 paso extra adicional.

### 1. Crea una lista de comprobación para el diseño del formulario de reserva

- Elige solo una opción de viaje, por ejemplo, "Lujo".
- Presta atención a la ortografía, la disposición de los elementos y su apariencia. No olvides el mapa de navegación que se encuentra en la parte central de la interfaz.
- No olvides las ventanas emergentes: "Automóvil reservado"; "¿Seguro que quieras cancelar el viaje?"; y "Viaje cancelado".

**! No es necesario que pruebes la validación de campo ni el diseño de las ventanas "Agregar licencia de conducir", "Método de pago" y "Agregar tarjeta".**

Coloca la lista de comprobación en la primera pestaña de tu hoja de cálculo: "1. Lista de comprobación del diseño".

### 2. Crea una lista de comprobación para la funcionalidad de las ventanas "Método de pago" y "Agregar tarjeta"

Para hacerlo, necesitarás las secciones correspondientes de los requisitos: el campo "Método de pago" y la ventana "Agregar tarjeta".

- Para crear una lista de comprobación, utiliza la partición de clases de equivalencia y el análisis de valores límite.
- No olvides las pruebas positivas y negativas.

# Bootcamp Ingeniero de calidad de software - TripleTen

## QA Engineer - Blanca Sánchez Gutiérrez

Coloca la lista de comprobación en la segunda pestaña de tu hoja de cálculo: "2. Lista de comprobación de "Método de pago" y "Aregar tarjeta"".

### 3. Prepara casos de prueba para el botón "Reservar"

Consulta la sección "Botón 'Reservar'" en los requisitos. No olvides las pruebas positivas y negativas.

! Ten en cuenta que:

- La distancia y la duración del viaje se muestran en el botón "Reservar". Por ejemplo: "El recorrido será de [3] kilómetros y se hará en [4] minutos". No necesitas comprobar si estos datos se calcularon correctamente.
- Tampoco necesitas probar el temporizador que muestra el tiempo de espera gratuito.

Coloca los casos de prueba en la tercera pestaña de la hoja de cálculo: "3. Casos de prueba para el botón "Reservar"".

### 4. Prepara los casos de prueba para la función de reserva de automóviles

Consulta la sección "Reserva de automóvil" de los requisitos. No olvides las pruebas positivas y negativas.

Coloca los casos de prueba en la cuarta pestaña de la hoja de cálculo: "4. Casos de prueba para la reserva".

### 5. Prueba la aplicación y escribe informes de errores

Prueba la funcionalidad para compartir un automóvil utilizando las listas de comprobación y los casos de prueba que diseñaste previamente. Utiliza solo dos configuraciones de entorno:

# Bootcamp Ingeniero de calidad de software - TripleTen

## QA Engineer - Blanca Sánchez Gutiérrez

- Google Chrome, resolución de pantalla 800x600
- Firefox, resolución de pantalla 1920x1080

Las pruebas de diseño deben realizarse en **ambos** entornos. La lógica de la aplicación solo debe comprobarse en **un** entorno.

**!** Ten en cuenta que los requisitos dicen que después de llenar el campo "Agregar licencia de conducir", se activa un temporizador durante 30 segundos. La implementación actual de la aplicación **no** tiene esta función, por lo que no tienes que esperar cada vez que realizas una prueba.

Cuando ejecutes las pruebas, marca los resultados como APROBADO o NO APROBADO. Si la prueba tiene el estatus de NO APROBADO, escribe un informe de errores en Jira. No olvides introducir el ID en la hoja de cálculo correspondiente: hay una columna llamada "Enlace al informe de errores" en cada pestaña.

Sigue el [enlace](#) a la pantalla de registro de Atlassian (Jira) y mira los videos a continuación para crear una cuenta y aprender a enviar informes de errores.

### ► Videos: creación de una cuenta y compilación de un informe de errores en Jira

 Tendrás que dar al equipo de revisión acceso a tu espacio de trabajo de Jira. Abre las siguientes instrucciones para descubrir cómo hacerlo.

### ► Cómo dar acceso al equipo de supervisión de proyectos

## 6. Resumen (\*)

En el futuro, a menudo tendrás que resumir los resultados de tu trabajo. Para practicar, puedes hacer una tarea adicional. El equipo de supervisión de proyectos la revisará y te dará su feedback, pero esto no afectará a tu resultado.

Pon tus respuestas en la plantilla de Google Docs en la sección "Conclusiones".

- Como usuario, ¿cuál es tu impresión de Urban Routes?
- Cuéntanos en algunas frases qué lograste probar específicamente.
- Resume los resultados de las pruebas. Por ejemplo, lograste ejecutar todas las pruebas y encontrar algunos errores. Adjunta enlaces a los errores para que tus colegas puedan abrirlos y verlos rápidamente.
- Haz una conclusión: ¿Crees que es posible dar a los usuarios este tipo de producto? Recuerda que el equipo de testers hace recomendaciones, pero la gerencia toma la decisión final.

# **Bootcamp Ingeniero de calidad de software - TripleTen**

## **QA Engineer - Blanca Sánchez Gutiérrez**

### **Cómo se evaluará tu trabajo**

Tu trabajo se evaluará de acuerdo con los siguientes criterios:

- Los casos de prueba están bien diseñados y las comprobaciones son exhaustivas.
- Las listas de comprobación están bien creadas y las pruebas están minuciosamente diseñadas.
- La prueba se realizó de manera eficiente y se completaron todas las pruebas planeadas.
- Qué errores se detectaron y si se informaron correctamente.

### **Espera la comprobación y los comentarios del equipo de supervisión de proyectos**

La supervisión del ejercicio se efectuará en un plazo de 48 horas. Si hay observaciones, se te devolverá el proyecto para que lo revises. Tienes 48 horas para corregirlo y volver a enviarlo para su supervisión.

## Entregable:

Blanca Sánchez, 45 Cohort - 3er. sprint

### 1. Enlace a la tabla de Google con las tareas:

- 1. [Lista de comprobación del diseño](#)
- 2. [Lista de comprobación del método de pago](#)
- 3. [Casos de prueba del "botón reservar"](#)
- 4. [Casos de prueba para la función de "Alquiler"](#)

### 2. Enlaces a los informes de errores en Jira:

✉ <https://blancasanchezgtz.atlassian.net/jira/software/c/projects/TRIP/list>

### 3. Conclusiones (\*)

- 1) Como usuario, ¿cuál es tu impresión de Urban Routes? Cuéntanos en algunas frases qué lograste probar específicamente.
- 2) Resume los resultados de las pruebas. Por ejemplo, lograste ejecutar todas las pruebas y encontrar algunos errores. Adjunta enlaces a los errores para que tus colegas puedan abrirllos y verlos rápidamente.
- 3) Haz una conclusión: ¿Crees que es posible dar a los usuarios este tipo de producto? Recuerda que el equipo de testers hace recomendaciones, pero la gerencia toma la decisión final.

#### 1. Impresión general de Urban.Routes

La aplicación ha alcanzado un desarrollo sólido y funcional en varias áreas, pero todavía se encuentra en etapa de mejora y validación antes de su versión final, pienso que el diseño de la aplicación es claro, intuitivo y facilita la navegación del usuario.

El módulo que presenta más incidencias es el “Método de pago”, aunque es funcional requiere de ajustes en la validación de entradas y manejo de errores antes de considerarse listo. Recomiendo priorizar la revisión de los campos de captura para garantizar integridad y consistencia en los datos financieros.

## **2. Resultados de las pruebas**

Tras la realización de las pruebas se detectaron algunas inconsistencias que se detallan a continuación:

### **Campo “Número de tarjeta”**

- El sistema permite ingresar más de 12 caracteres numéricos, cuando el límite debería ser exactamente 12 dígitos.
- Se debe restringir la entrada a caracteres exclusivamente numéricos, sin letras ni símbolos especiales.

### **Campo “Número de código”**

- Solo debería permitir hasta 2 caracteres numéricos, pero el sistema no valida correctamente este límite.
- No debe aceptar espacios ni caracteres especiales (por ejemplo: “-”, “#”, “@”).
- El campo no debe ser enviado sin datos.

### **Botón “Cancelar”**

- El sistema no responde al presionar el botón “Cancelar” el viaje lo que impide completar el flujo alternativo de la reserva y no fue posible validar los mensajes: ¿Seguro que quieres cancelar el viaje? y viaje cancelado.

### **Posibilidad de entregarlo a usuarios finales**

- La aplicación no está lista para pasar a la fase de su implementación en producción, debido a que aún requiere ajustes técnicos y de diseño para cumplir con los requerimientos establecidos.
- Entregarla ahora sería riesgoso, ya que el usuario final encontraría múltiples fallos que podrían comprometer su usabilidad y confiabilidad.
- Es fundamental realizar más pruebas y verificar que todos los requerimientos se cumplan por completo antes de su entrega.