# Usability Test Plan

## Testing Overview

This usability test plan evaluates the core functions and user experience of our iOS public transportation application. The goal of this is to observe how effectively real users can navigate key features

Participants will be asked to complete a set of realistic tasks that reflect common transport use cases such as finding a route, adjusting display settings for accessibility, and managing saved journeys. During each session, we will record observations about user behaviour, task success, error, and user feedback.

The test will be moderated and conducted in a controlled environment using Figma loaded with the prototype frames.

Our testing objectives are as follows:

- Evaluate how easily users can complete key tasks such as journey planning and changing accessibility settings.
- Identify usability issues or points of confusion in the current prototype.
- Measure overall user satisfaction and gather feedback for future design improvements.

This testing phase will serve as a key step in validating design decisions, identifying usability issues, and gathering key feedback to refine the prototype. The test will involve six participants from the Victoria University of Wellington's SWEN303 course who reflect the app's intended user base, including commuters, students, and individuals with varying levels of transit experience and accessibility needs.

## Methodology

Six participants will be acting as test users in usability testing. Users will carry out testing with a prototype version of the system created on Figma.

- Primary Approach - In-Person Testing:

Testing will be conducted in a controlled environment where participants and moderators are co-located, specifically in ECS computer labs CO238 or CO242.

- Alternative Approach - Remote Testing:

If circumstances require, testing can be conducted remotely through Discord voice channel with screen sharing and Figma prototype access.

**Participants:**

Participants in testing have been selected from SWEN303 class. Based on the prerequisites of the course, participants are likely 3rd year computer science or engineering major students, or someone studying programming at some level.

The age group is likely 20-30, representing a generation that has grown up with digital technology and is familiar with mobile applications and user interfaces. Additionally, all participants have studied transportation applications as part of this course curriculum, giving them prior exposure to common design patterns and user experience principles in this domain.

Participants will be asked to complete 3-6 tasks that provide a comprehensive overview of the system's functionality. During task completion, participants are encouraged to think aloud to verbalize their thought process and decision-making. We will gather feedback from participants regarding the usability of the user interface and overall system function, with particular attention to any usability issues or problems they encounter. Following task completion, participants will complete a Prototype Usability Survey based on the System Usability Scale (SUS).

We are assuming that participants' average skill level with the system will be relatively high for first-time users because they are software engineering students with prior knowledge of transportation app design. Therefore, we believe they will be able to complete all tasks set out for them in testing, although this skewed participant pool and their pre-existing knowledge need to be taken into account when reviewing test results.

**Procedure:**

- In-Person Testing:

Test participants will be asked to complete testing in a quiet corner of the lab with only one participant at a time. One facilitator and one note-taker will join them physically. If other team members want to observe, screen sharing will be set up so they can watch remotely without being physically present to avoid unnecessary pressure on the participant. The test will use lab computers or team member's laptops to open the Figma prototype after introduction and conduct the test.

- Remote Testing:

Test participants will be asked to complete testing in a quiet environment, preferably at their workstation or desk. The test facilitator and note-taker will join the participant in a Discord voice channel with screen sharing, with one participant at a time. The participant will be sent the system prototype through a Figma link which they can open in a browser without downloading additional software.

- Common Procedure for Both Approaches:

Depending on time constraints, the introduction and training may be conducted for all 6 participants at once, followed by individual testing sessions. Otherwise, the process will be conducted individually for each participant.

Prior to opening the prototype, the test facilitator will introduce the system and explain the participant's role in the testing process. To encourage honest feedback, participants will be reassured that they are helping to evaluate the system's design and that their performance is not being assessed. The facilitator will also explain that any personal information, including names, will only be used for organizing the testing session and analysing results. If any team members are observing the session, participants will be informed of their presence. The facilitator will then conduct the training as described in the Training section before accessing the prototype.

During individual testing sessions, the facilitator will present tasks one by one, ensuring participants understand each task before beginning. Participants will complete tasks using the think-aloud protocol as trained. If participants forget the task during completion, the facilitator may repeat the task or parts of it as needed. The facilitator will gather post-task feedback while the note-taker documents responses, behaviours, completion times, and any usability issues. After all tasks, participants will complete the SUS questionnaire independently.

# The Script

**Intro:**

This app focuses on improving the public transportation experience for urban users, making it easier for people with varying schedules, purposes, and preferences to navigate, plan, and use public transport more effectively.
The primary focus is around planning public transport routes. Users can tailor their route preferences, such as departure time, transport method choices, and accessibility modes.
Users may also save commonly used routes as a favourite for easy access, and share routes with others.

**Your Tasks:**

We have designated six short tasks for each of you to complete, and a survey to fill. If you are unable to progress with any of the tasks please inform your facilitator, we will either give you a hint on how to move forward, or will move onto the next task.
Please verbalize your thoughts while completing the tasks. Keep in mind that any feedback you have can also be recorded in the survey. We want as much detail as possible.

**Facilitator will then introduce and guide the user through each task in sequence**

# Roles

**Testing Facilitator: (1 person):**

- Go through the script with test subjects.
- Provides task pre-made hints to subjects if they are in the hint group and need help.

**Note Taker (1-2 people):**

- Records relevant notes from the test subjects during testing, such as their spoken thought process during the task, points where the subject seemed stuck and areas which the subject navigated through well.
- Notes interactions that occur between the testing organisers and the test subjects.

**Analyst (1+ people):**

- Performs analysis on the data after testing has been completed
- Helps suggest improvements based on trends in subject answers

**Test Subject (6 people) :**

- Completes our 3 main tasks given by the Facilitator either with instructions or without instructions
- Gives input on test organisers questions so that they can be recorded

## Tasks

We initially planned to compare tasks with and without instructions to see how guidance affects user performance. However, with our limited number of participants, this comparison wouldn't give us meaningful results. We'd need more testers to see any real differences. Therefore, we switched to more specific tasks that are easier to measure and evaluate with our smaller group.

**Task 1.1:**

From the Home Screen, find a route to Victoria University that only includes buses and trains. During planning you should aim to find any details about the route you can. Finish the task by starting the route.

**Task 1.2:**

From the Route Information screen examine any alerts associated with your route and find what stops are impacted.

**Task 2.1:**

From the Home Screen, change the accessibility settings such that in app text-size is set to large.

**Task 2.2:**

From the Settings screen, change the in-app language to Te Reo Māori. Return to the home screen.

**Task 3.1:**

From the Home Screen, find the route information of a favourited route.

**Task 3.2:**

From the Favourite Screen, copy the share link of a favourited route.

## Metrics

After group discussion, our team decided to use the System Usability Scale (SUS) to collect user feedback during testing. We chose this method because it is straightforward to analyse and offers a clear, standardised scoring system on a 0 to 100 scale, making it easy to interpret and compare results.

## Usability Goals

Each user testing our prototype will be given a survey/questionnaire at the end of testing which contains 10 questions based on the “System Usability Scale (SUS)”.

To calculate usability scores, each questionnaire response is assigned a numerical value depending on the wording of the question. The scale is as follows:

**0 – Strongly Disagree, 1 – Disagree, 2 – Neutral, 3 – Agree, 4 – Strongly Agree.**

Responses from all ten SUS questions will be summed and multiplied by 2.5 to produce a score out of 100. While a SUS score of at least 68 is considered above average, we are aiming to have a score of 80 or higher. This would align with our group's goal of having a user-friendly functional app while still being able to provide insights into improvements our prototype could benefit from.

This process will allow us to gain a great understanding of what to keep in our prototype as well as highlight positive design choices our group made. Considerations in improving our design will be made based on user feedback and will be discussed as a group once testing as finalised.

## Problem Severity

A problem can refer to any issue, error, or exception that causes a user's experience with the program to be hindered, prevented or otherwise negatively affected (including personal frustration or annoyance!)

Problems will be classified according to their severity - this is fairly subjective, however important factors to consider are impact on the usability of the program, frequency of problem, as well as the importance of a task. Problems will be classified in 5 stages as shown below:

1 - Cosmetic
The issue causes a minimal impact on the program, and is purely a cosmetic issue that may be slightly distracting or feel uncoordinated.
E.g. corners not being rounded, typos, etc.

2 - Minor
The issue causes a small impact on the program, and may make certain outcomes slightly more cumbersome or act a bit unexpectedly.
E.g. missing text on buttons, slight colour variations, etc.

3 - Significant
The issue causes a non-negligible impact on the program, and causes a hindrance or otherwise unexpected outcome.
E.g. Buttons missing between pages, confusing layout, etc.

4 - Major
The issue is a severe problem that should be fixed quickly, and can completely impede a program or lead to a completely incorrect outcome.
E.g. Buttons don't work properly, components get stuck on screen, etc.

5 - Critical
The issue is an extremely problematic issue which completely prevents proper usage of the program and needs to be fixed as soon as possible.
E.g. No ability to navigate back or forward, pop-ups block the screen and cannot be closed, etc.