

Usability Test Plan

Testing Overview

The following document describes the plan for usability testing of the prototype of the Flat Finances application.

These are the objectives of usability testing:

- Determine a first-time user performance and satisfaction with the application; allowing comparison to be done on later usability testing.
- Identify potential errors in the application and design concerns to be altered in later versions.
- Determine user satisfaction with the application.

These objectives will be achieved through analysis of data from test participants, including participant feedback and interaction with the system. Specific metrics have been set out in this test plan to standardised data obtained for test participants.

The intended user base of this Flat Finances application are people from the general public who live in a group flatting situation, especially University students. Participants in testing consist of a user group of 6 (expected) students studying SWEN303 at Victoria University of Wellington.

Methodology

Six participants will be acting as test users in usability testing which will be carried out through screen-sharing remotely. Users will carry out testing with a prototype version of the system created on Figma.

Participants

Participants in testing have been selected through group pairs suggested by SWEN303 lecturer Simon McCallum. 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. Therefore, we believe they will be able to complete all tasks set out for them in testing, although this relatively high skill level should be taken into account when reviewing test results.

The participants will be asked to complete a number of tasks presented to them which give a comprehensive overview of the function of the system. We will ask for feedback from participants about the usability of the user interface and function of the system, including that of potential errors in the system. They will then be asked to fill out a response form containing more in depth questions of system usability/function.

Training

Participants will be shown a quick overview of the system via screen-sharing before testing begins. Not much emphasis will be put on navigation throughout the system so testing can be done as to see how intuitive the system layout is, although some system navigation will be seen by users as different displays are shown. An overview of the usability test procedure will be relayed also.

Procedure

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 then join a participant in a voice call and the participant will be sent the system prototype through a Figma link which they can open in a browser without downloading additional software.

The test facilitator will then introduce the system to the participant and their role in testing. To ensure honest answers are obtained from the participant it is important to ensure them that it is them that is assisting in evaluating the system and that they themselves are not being evaluated on their proficiency. Additionally, the facilitator will inform the user that personal information such as their name will not be used anywhere outside of identifying them for testing procedure and analysis. The facilitator will then give a quick overview of the purpose and function of the system with reference to the prototype.

The facilitator will then ask test participants to read aloud each task before completing it, voicing their thought process as they do so. After each task the facilitator will ask the participant for their general thoughts on the process. Another person present in the call will be the note taker who will record notes on the participants response to each task, as well as their thoughts during completion. Notes on time taken, any potential errors in the system and differences in expectation by the user and the actual system will be recorded also. After all tasks are completed the users will be asked to complete a Google form questionnaire independently. This questionnaire will ask users to evaluate system function through the function of each task and overall as well questions on general function and their interaction with it.

The Script

Intro

The purpose of our app is to assist in organising the costs in a flat group. Users create/join a flat group in which different payments can be created by any member of the group and shared with the other members. These can be repeated payments, ie. weekly. The split of payments (how much each flat member pays) can be set when creating the payment. When created, these payments will appear on the overview page so users can see upcoming payments. When a payment is due a user will be notified so they can see how much is to be paid for which payment. A user can mark a payment as paid when it is due. Users can see which other members of a flat group have marked a notification as paid.

Your Tasks

In order to make the testing process as simple as possible, we'll get you to perform a short set of tasks on our Figma prototype. At the end of each task, please note any difficulties you had with the task. When you've finished the task set, please fill out the Google form. We've tried to make the form as concise as possible, so that it won't take too long. That being said, we'd really appreciate as much detail in the feedback as you're comfortable giving us. We have included 3 tasks with instructions and 3 without instructions. We've done this to simulate the contrast in experience between that of an ideal user flow (with instructions) compared to that of a new user (without instructions). Please comment on any differences between the two.

The tester will then proceed to ask the user to complete the tasks detailed below.

Roles

Facilitator

- Contacts the test group
- Schedules times and locations for testing
- Provides additional information on requests

Note Taker

- Records notes from the test subjects
- Records any interaction between the testing organisers and the test subjects

Analyst

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

Test Subject

- Completes the tasks given by the Facilitator
- Inputs answers to the questions asked

Tasks

Task 1:

First, starting from the sign in screen, sign in to the app and join an existing flat. Steps: 1. Press 'Sign In' on the initial screen 2. Press 'Sign In' again 3. Press 'Join' on the welcome page If at any point during this task you were lost on the next step to complete the task please describe it in the feedback survey.

Task 2:

From the overview page, I would like you to create a new payment for just your flat group members, split evenly among all members. Steps: 1. From the Home/Overview page, press 'Create' on the footer bar 2. Ensure all members of the flat have boxes ticked 3. Press 'Split Payment' button 4. Ensure the payment split is even 5. Press 'Complete Payment' button If at any point during this task you were lost on the next step to complete the task please describe it in the feedback survey.

Task 3:

I would like you to show me how you would invite a new user to the group you are currently in. Steps: 1. Navigate to the 'Flat Info' section 2. Press 'Copy Group Code' 3. Send code to member through an external messaging service If at any point during this task you were lost on the next step to complete the task please describe it in the feedback survey.

Task 4:

Check the payment status of each flatmate for an expense. If at any point during this task you were lost on the next step to complete the task please describe it in the feedback survey.
1. Select a payment 2. Locate flatmates payment status

Task 5:

Edit the details of an existing payment If at any point during this task you were lost on the next step to complete the task please describe it in the feedback survey. 1. Select a payment 2. Navigate to edit button 3. Save edit

Task 6:

Leave the flat group you are currently a part of. If at any point during this task you were lost on the next step to complete the task please describe it in the feedback survey 1. Navigate the system to find the leave button

Metrics

After researching effective testing methods, we decided to use a System Usability Scale. It's appropriate for our project as it can be used on small sample sizes with reliable results and it can easily identify the usability of a system. After following the advice on [this website](#) and considering how we'd get the most useful feedback for the tasks we chose, we created the following survey for testers to complete.

Usability Goals

The usability goals stated below are in adherence to the “System usability Scale (SUS)” detailed by usability.gov (<https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html>). Asking a broad spectrum of questions through our feedback questionnaire, in conjunction with the nature of the set of tasks to be performed by the participant, allow us to gain a significant understanding of their overall experience.

In order to calculate the scores obtained in our questionnaire, we will perform a simple mathematical function on each score to obtain a percentage of satisfaction. Each answer to the question will be given a numerical value as follows, and this would depend on the nature of the question. For example, Question 1 (I think that I would like to use this system frequently), would be scored as follows: 0 – Strongly Disagree, 1 – Disagree, 2 – Neutral, 3 – Agree, 4 – Strongly Agree. The summation of the numerical values of all ten questions will then be multiplied by 2.5, which will convert the answers from a range of 0-40 to 0-100.

As specified by usability.gov, research indicates that a **“SUS” score above 68 is average** and below this is below average. Taking into account the small number of participants in our user testing, it is likely that our results will not give a fully accurate representation of what a larger population would. That being said, it is important to gauge the efficacy of our design in terms of user experience, and since we are striving for an above average experience, we have decided that a **“SUS” score of 75 is desired for our design**. Whilst ambitious, we believe that this score will demonstrate the user experience of our application (based on the audience tested on) was an overall pleasant experience, whilst still allowing room for error. This will allow us to take feedback and strive to build on this score for future iterations of our application design.

Problem Severity

This section will aid in prioritising any problems experienced while testing. The classification is made by factors such as impact, frequency and, the importance of task said issue is associated with. This classification system will be used in order to identify the most important and immediate issues that need to be fixed.

What is considered a problem?

Certain features will not be available due to the product being a prototype, this test will only take into account problems resulting from tasks in the script that negatively affect the user experience.

Classification

Minor - Low impact problem generally very mynute difference to overall user experience and low frequency.

- Example: button alignment slightly off.

Moderate - The problem will have a significant effect on the overall user experience, but will not be something that completely renders the program unusable.

- Example: certain function does not work.

Major - This problem makes the app completely unusable or does not allow the user to complete a major function.

- Example: Cannot return to home page.