Software Testing Specifications

ClockWork Time Estimation Tool

Team Members

Anthony Menendez   -   amenendezmen2022@my.fit.edu
Christian Ott   -   cott2020@my.fit.edu
Peter Stelzer   -   pstelzer2023@my.fit.edu
Pierson Hendricks   -   phendricks2023@my.fit.edu

Client & Faculty Advisor

Dr. David Luginbuhl   -   dluginbuhl@fit.edu

Table of Contents

1. Introduction

This document outlines the testing plan for various features of the project. Each of the core features is divided into various test cases and alternative paths containing a procedure and expected result to account for as much as possible.

2. Core Function 1: Task Completion Timing

2.1 - Purpose

The timer is the primary means by which the user will monitor how long they take to complete a task and compare that to their estimated time. Users should be able to start and stop the timer at any time and time data should be recorded by the app.

2.2 - Test Case 1: First time performing a type of task

2.2.1 - Procedure

1. User clicks on a button to open the task profile creator
2. User inputs a name and information for the task profile
3. User selects the newly created task profile
4. User inputs their time estimate
5. User starts the timed session

2.2.2 - Expected Output

The program displayed a running timer with clickable "Stop" and "Pause" buttons.

2.3 - Test Case 2: Starting a timed session for a chosen task

2.3.1 - Procedure

1. User selects a task profile
2. User creates a new session
3. User inputs their time estimate
4. User starts the timed session

2.3.2 - Expected Output

The program displayed a running timer with clickable "Stop" and "Pause" buttons.

2.3.3 - **Alternative:** *The desired task profile does not exist*

The application did not proceed with creating a session as a task was not selected. The program also displayed an option for the user to create a new task profile.

2.4 - Test Case 3: Pausing a timed session

2.4.1 - Procedure

1. User clicks the pause button

2.4.2 - Expected Output

The program displayed a paused timer with clickable "Stop" and "Resume" buttons.

2.5 - Test Case 4: Ending a timed session

2.5.1 - Procedure

1. User clicks the "Stop" button

2.5.2 - Expected Output

The program displayed a stopped timer with a series of questions regarding the task's difficulty as well as the time taken to complete it versus the user's estimated time.

3. Core Function 2: Future Task Prediction

3.1 - Purpose

The core principle of the application lies in helping users make better time estimations for completing tasks. The application should be able to predict with a reasonable degree of accuracy the time it should take to complete a given task based on prior information about that task.

3.2 - Test Case 1: Viewing a time estimate prediction

3.2.1 - Procedure

1. User selects a task profile
2. User clicks on the time estimate input area but does not confirm input

3.2.2 - Expected Output

The application displayed a message near the time estimate input area with a recommended time based on prior completion history for that task.

4. Core Function 3: Task Schedule Creation

4.1 - Purpose

The user should be able to organize upcoming tasks into a calendar based on the estimated time the tasks would take to make integrating such tasks into the user's schedule easier. For example, an assignment estimated to take one hour should be displayed as a one-hour block on a calendar day.

4.2 - Test Case 1: Adding a task to the schedule

4.2.1 - Procedure

1. User navigates to the calendar section
2. User clicks on the 'add task' button
3. User chooses a task session to add
4. User selects a start time for the task session
5. User clicks on the 'confirm' button

4.2.2 - Expected Output

The task was successfully added to the calendar and displayed in the calendar view.

4.2.3 - Alternative: *Chosen start time conflicts with an existing calendar entry*

The program notified the user of a time conflict with the time they have selected and the existing task on the calendar. The program did not allow the user to confirm that task until the time was changed or the other task was moved.

4.3 - Test Case 2: Removing a task from the schedule

4.3.1 - Procedure

1. User navigates to the calendar section
2. User clicks on a task from the calendar
3. Use clicks on the 'remove task' button
4. User clicks on the 'confirm' button

4.3.2 - Expected Output

The application successfully removed the chosen task from the calendar view.