

Clock Work Time Estimation

**Anthony Menendez
Christian Ott
Peter Stelzer
Pierson Hendricks**

Advisor: Dr. David Luginbuhl

Goal and Motivation

Goal: Assist in the estimation of task time cost

- Students often have difficulty with time management
- Poor ability to estimate how much time a certain task will take
- We want to alleviate this by tracking the time a student spends on their tasks + give feedback
- As students use the app, both they and the app will learn their work habits and improve the accuracy of the future time estimations

Features

1. User can refine their ability to estimate the time they require to complete tasks
2. User can view time-cost predictions from the app for registered tasks
3. User can create schedules using time blocks provided by app data

1. User Can Refine Their Estimation Ability

- User can register tasks to track
 - i. Users can choose from preloaded task profiles like “homework”, “test”, “study”, etc...
 - ii. Users can define their own custom task profiles
- User can begin sessions of their registered tasks where they provide a time–cost prediction and time themselves as they complete the task
- User can complete sessions and compare their predicted time with their actual time
- User can track the accuracy of their estimations over time for all of their registered tasks
- User can specify task parameters, such as difficulty or category, to link similar tasks together

Hypothetical Use

- The user is a CS student and has weekly programming homework.
- They would register a new task named “Weekly Programming HW” at the beginning of their semester.
- Each week, they would begin a session of “Weekly Programming HW” and indicate how long they think it will take.
- The user can pause the session when they take breaks and end when they complete their assignment.
- On completion, the app will provide feedback on their estimation and provide estimate how long future sessions will take.

2. User Can View App's Time-Cost Predictions

- User can view pessimistic, optimistic, and most likely time-cost predictions
- User can view time-cost predictions for tasks without time measurements by manipulating measurements for analogous tasks based on user-specified parameters
 - i. Parameters such as category and difficulty
- User can view time-cost predictions for tasks without time measurements by decomposing tasks into reusable subtasks

3. User Can Create Schedules

- User can place tasks on a calendar where each calendar day has a timeline.
 - i. Tasks appear as blocks on the timeline
 - ii. The length of the blocks are provided by app predictions
- User can visualize the range of predictions for applicable tasks
 - i. The region from optimistic to pessimistic is indicated

Novel Features

- Main Novel Feature:
 - The app attempts to gauge and improve a user's time-estimating ability.
- Performance Features:
 - The timing of task completion is used to monitor user performance speed.
 - Initial time-cost is provided by user and measured against actual task performance.
 - App forecasted task completions.

Tools

1. Kotlin
2. Android Studio
3. XCode
4. Apple ActivityKit

Challenges

1. Kotlin
2. Mobile App
3. Multiplatform Development

Milestone 1 (Feb 24)

- Compare and select technical tools for app development
- Provide small mobile demos to begin development process
- Begin further R&D into kotlin and multiplatform deployment
- Compare and select collaboration tools for software development, documents/presentations, communication, task calendar
- Create Requirement Document
- Create Design Document
- Create Test Plan

Task Matrix

Task	Anthony	Christian	Peter	Pierson
Compare and select Technical Tools	10%	10%	40%	40%
Mobile Demos	20%	20%	20%	40%
Kotlin R&D	25%	25%	25%	25%
Compare and select Collaboration Tools	40%	40%	10%	10%
Requirement Document	25%	25%	25%	25%
Design Document	25%	25%	25%	25%
Test Plan	25%	25%	25%	25%

Milestone 2 (Mar 26)

- Begin more evolved design for UI
- Implement, test, demo initial app navigation
- Implement, test, demo initial task list
- Implement, test, demo initial task time
- Implement, test, demo initial user time estimations

Milestone 3 (Apr 21)

- Begin designing extraneous menus
- Implement, test, and demo task parameters
- Implement, test, and demo settings menu
- Implement, test, and demo statistics displays
- Implement, test, and demo initial app time predictions
- Improve implementation of user time estimation feedback
- Improve menu design to be more intuitive