Productivity Application

Department of Computer Science and Engineering

California State University, Fullerton

Project Advisor: Lidia Morrison

Submitted By: John Michael Lott

20 December 2025

Table of Contents:

Figures

Abstract

Within many organizations, the ability to track employee data is achieved by utilizing various applications to track time, tasks, shifts, and other metrics. In this project, the goal is to create a software application for work productivity to combine much of this functionality and expand on it. For daily use, that includes functions that allow users to log in to see a list of tasks for the day, allowing the manager to embed videos to be watched or even create a checklist task with a wide variety of options for completing them. This project also aims to create a modern UI/UX for employees and supervisors alike, allowing easy, intuitive, and fun interactions with the software. This project will also develop secure login access with user-specific data storage. On top of these features, another important goal is to have a simple UI that can be used for multiple platforms and operating systems to reach a broad user base.

1. Introduction

1.1 Background

Within various industries, many employees are exposed to a variety of productivity applications used for things like time clock tracking, scheduling, task management, and various other work-related functions. Typically, employees will have to use two or more different applications to allow employers to track all of this data. When interacting with this software, it is never the most intuitive to work with from both the employee and the administrative side, and it often misses functionality that can streamline workflow within organizations or facilities. On top of this, the UI for many of these applications is dated and difficult to parse through when specific access is needed.

1.2 Motivation

These problems have occurred within many organizations. This project aims to fix these problems for the various organizations that employ these large groups of minimum-wage workers. These workers do much of their work offline, so this project will work to create a seamless meshing of the various functionalities, allowing users to minimize time spent online accessing these services. With this, more time can be dedicated to important tasks while also making life easier for employees so they will not have to learn to use multiple applications to manage information in and out of the workplace. On top of this, if administrators have the ability to access all of the employee information in a central location, they can reduce costs by minimizing the needed software subscriptions.

1.2 Related Work

1.2.1 Category-Based Administrative Access Control Policies

The first article is titled Category-Based Administrative Access Control Policies. This article outlines Admin-CBAC, outlining the process of setting up access control in this manner. It uses the basic idea of entities, categories, relationships, and rules. This method is simplified by introducing the Admin Categories, which allow various admin role types to access separate rules or axioms. This method simplifies the setup of new users and allows for more access customization at the higher levels. Security is a massive factor in any company or system composed of administrators and base users. This security includes limiting access for those who can view, add, remove, or edit data. This method is also used to ensure that the proper functionality is provided to the corresponding users. The primary purpose of this application is to assist administrators in creating tasks and simplifying work systems they will be using to keep track of employee information, which makes this security essential to consider. This highlights the need for access control management with techniques mentioned in this Research article.

1.2.2 VideoSticker: A tool for Active Viewing and Visual Note-taking from Videos

The article "VideoSticker: A tool for Active Viewing and Visual Note-taking from Videos". This software assists in clipping notes from video content. The tool's goal is to automatically detect and track objects in a video and recognize text on the screen. These objects will be stored as "stickers", which can be easily accessed in theory to make note-taking easier. A difficult task for anyone who tries to create interactive tasks with videos is often to integrate tasks within the video. Task integration is necessary to keep the user engaged in tasks and to ensure proper staff training. The use case for this tool is to help create interactive media tasks within the application.

1.2.3 UIClip: A data-driven Model for Assessing User Interface Design

The article titled "UIClip: A data-driven Model for Assessing User Interface Design". This article details a machine learning model called UIClip, which will assess a UI's design quality and visual relevance based on images and a text description. The model is trained based on reference UI designs with grades based on established standards for UI design. Within any user experience, a developer knows that the UI should be attractive to users to get them to enjoy their interactions with it. The better the users can navigate and interact with the UI, the more time they have to focus on more important pressing tasks. The various software services currently used in the market have often been outdated and not the easiest to use, so the goal here would be to create a good UI that would help draw in more users.

2. Problem Statement

Within many workspaces such as fitness centers, retail, or custodial positions, amongst many others, the worker's primary focus is never software. In minimum-wage jobs such as these, the worker's primary focus should be on completing tasks within their allotted shift. Within these positions, the importance of software has risen as companies seek to track employee data electronically rather than to maintain hundreds of thousands of physical files. The use of digital data storage allows data to be backed up securely in case of an emergency and allows the necessary data to be easily accessible remotely. This shift to digital data collection and storage has led companies to work with various software providers to accomplish these tasks. The software currently does not accomplish many of the tasks needed, so it becomes necessary for administrators and employees to use multiple programs on a daily basis to track all of the necessary metrics. The lack of features and the need to learn to use multiple software services for employees makes both the administrator's and the employee's lives harder, taking time away from more pressing tasks.

3. Proposed Project & Significance

This project aims to create a software application for work productivity that allows users to log in to see a list of tasks for the day, allowing the manager to embed videos to be viewed or even create a checklist task with a wide variety of options. With this application, the work would primarily focus on expanding the ability to create more types of tasks and forms. It will also create a more user-friendly interface and combine the functionality of applications so that only one software is needed for employees and administrators to complete their tasks. On top of this, the software will allow the managers to set interactive tasks with video plugins for on-shift training and create records of the staff in each department to track data on productivity. With this software, employees will only have to check one place for all their work-related tasks, and administrators will only need to check one location for all the information they need for employee management, improving workplace productivity so more gets done in workplaces.

4. Objectives (Step-by-step requirements and UX)

4.1 Objectives

In this project, there will be a robust and complete front-end user interface that will be easy for users to navigate from both the employee and administrative views. The program will limit certain functionality based on the user's credentials. The program will allow clock-in functionality, keeping a record of time. The program will also track user work schedules. The program will Implement a backend storage method to store various saved employee info, task forms, and other info related to each task type. The program will allow users to access it from both mobile and desktop interfaces.

4.2 UX Design

4.2.1 Getting Started

When the user gets the link, whether through an internet search or a link sent by system administrators, they will be directed to the site. In order to create an interface that can keep the users engaged, the site will be designed to be simple, modern, and spacious. The system will avoid cluttering the space the users will see with unnecessary information, dividing only the minimum amount of information needed between pages. The simplistic view will make it easy to navigate and avoid distractions for the users.

Users will have accounts created by system administrators, so they will not have to register themselves. The users will load into the main login page using a provided username and password. They will be able to select if they are using Admin access or if they need shift-only access to simplify the accessible information. Then the user will have the option to log in. (see Figure 4-1 Sign-in page)

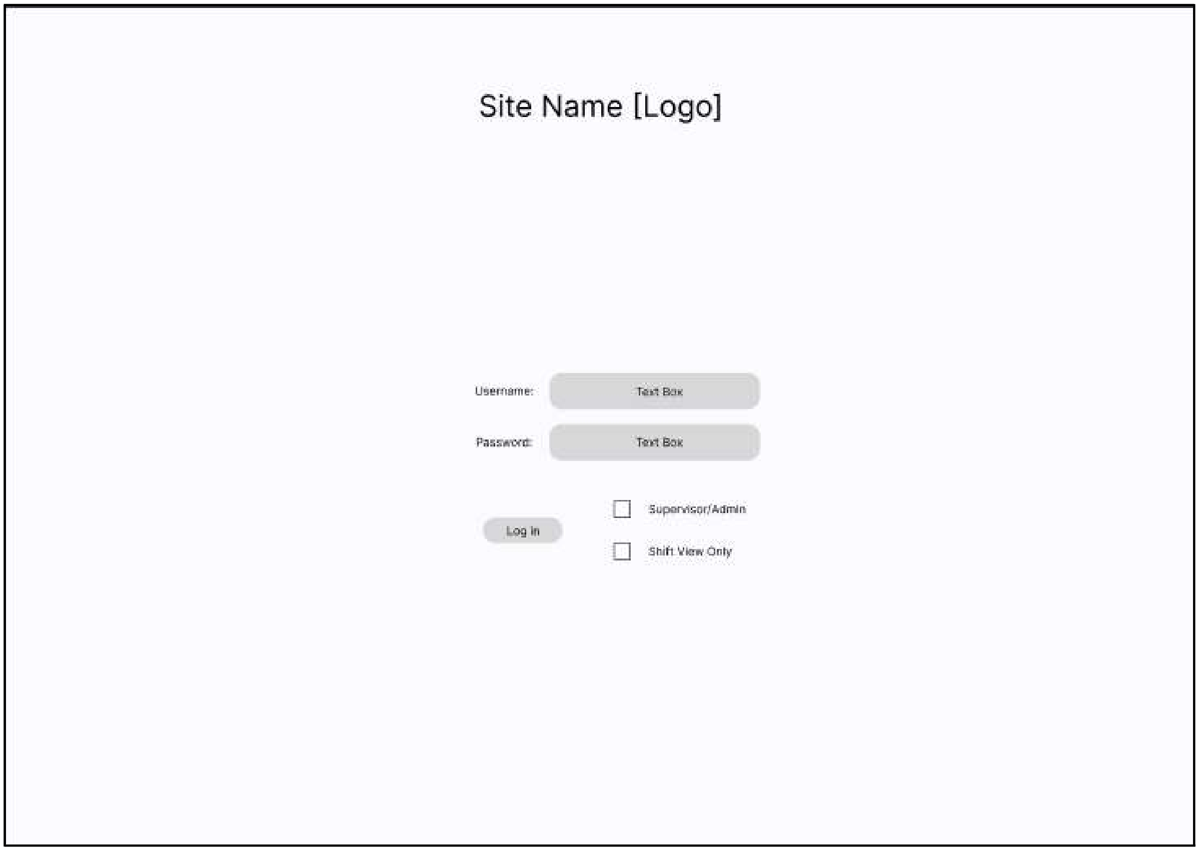

Figure 4-1 Sign-in page

4.2.2 Main Page Interface

Users will be greeted with a main welcome page when they log in. The user will see a header for the page containing a link to their profile and a logout button at the upper right side of the screen. The header will also have an option on the left end to open and collapse the side menu. The side menu, when open, will have four general links for employees to select. The first link will be to the schedule page showing the user's weekly shifts. The second link will take the user to the Clock In/Out page, where they can carry out those tasks. The following link will be the Timesheet/Time logs. This link will allow the users to view their past clockings within the system. The last link will direct the user to the shift-specific view, which will be used to display on-shift tasks and activities based on the user's current shift. The various sections of the page, such as the header, menu, and content section, will each be different colors to differentiate for users. (See Figure 4-2)

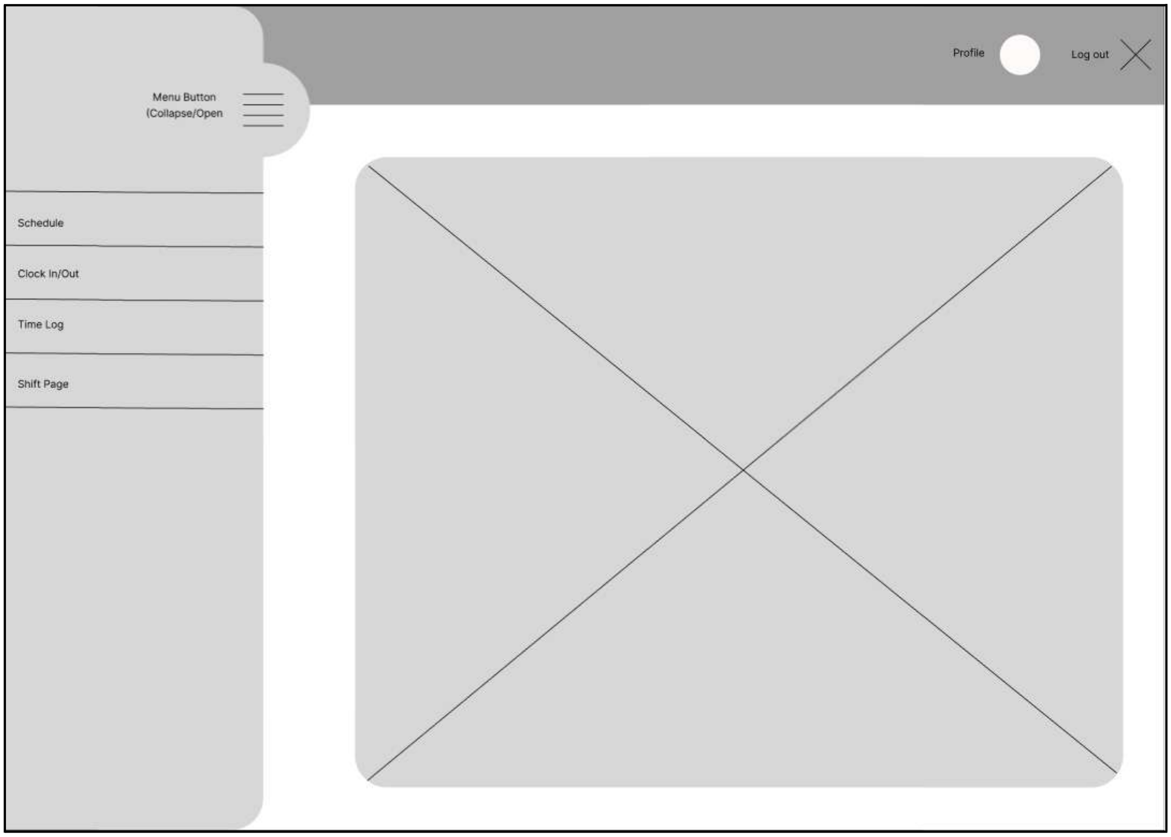

Figure 4-2 Main Page Interface

4.2.3 Task Views

The following pages the user accesses will generally follow the same format as the main page interface. However, the center main content box will have various content interfaces based on what is being done. This content view will be in the background when the menu is open but will be in full view when the menu is closed. An example of one of the different content views is the clock in/out view. This view will display the current time of day, including hours, minutes, and seconds in a 12-hour format. The clock will be seated in the upper half of the content view, leaving plenty of space for buttons in the lower section. The lower half will be dedicated to the buttons users will use to update their clockings. There will be a clock-in and clock-out button, which will add clocking to the system records. Some other buttons that can be included are the buttons for adding a break for longer shifts and meal breaks. A generous amount of space will be left in the bottom half to allow for added functionality in the future. (See Figure 4-3)

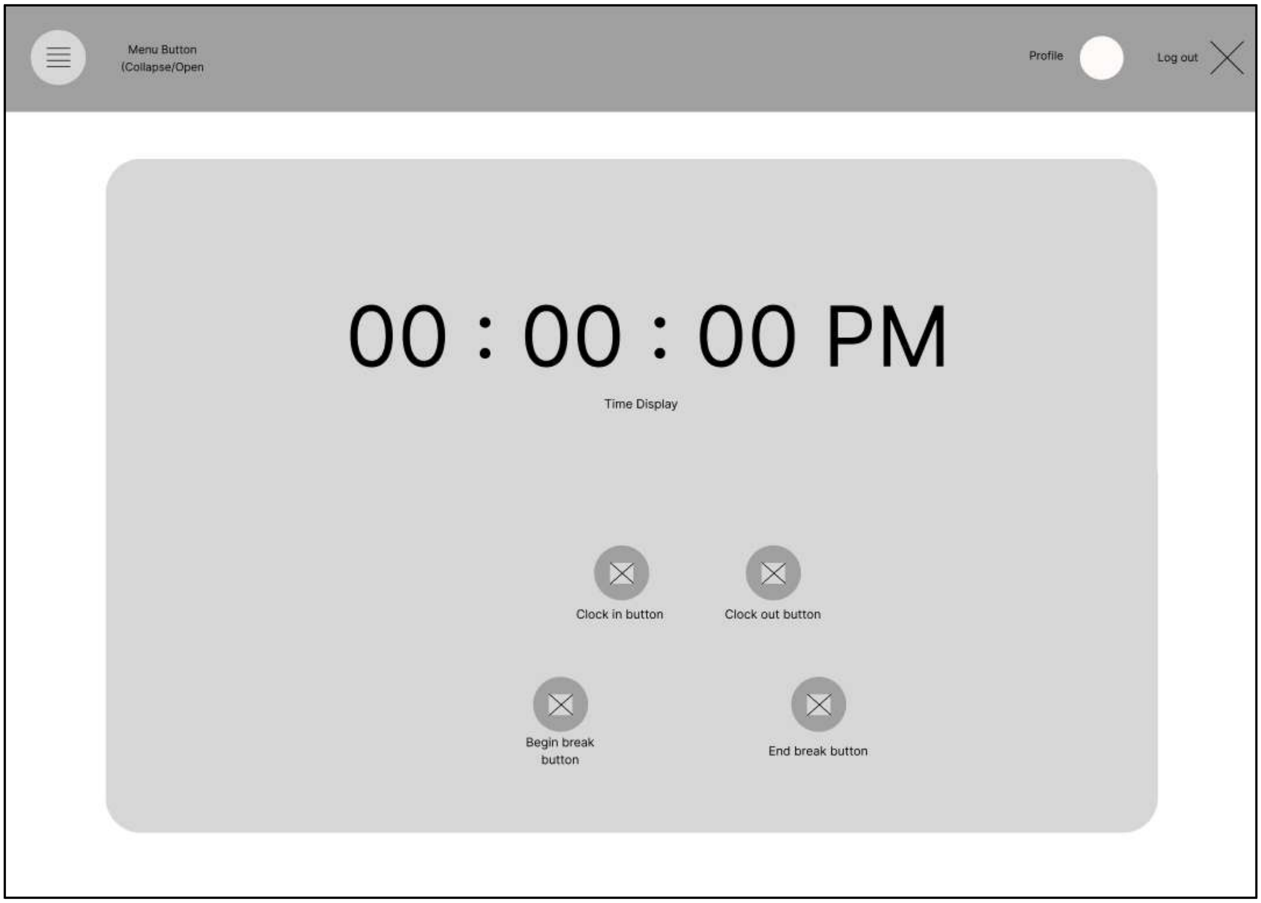

Figure 4-3 Clock In View

5. Activities

5.1 Functionality

The goal of any application is always for people to use it and find it helpful. Within this application, the goal is to solve problems that are prevalent in many organizations. One of those problems is the need for multiple applications for employee management. In order to remedy this problem, there is a need for vastly increased functionality within software. This issue will be remedied within this application by establishing a more intuitive UI that will be easy to navigate, allowing users to access various functionalities easily when needed.

5.2 User Friendly

Within any organization, a pressing issue is the dated UI, which oftentimes makes the software more difficult to navigate. Some of this has to do with specific wording and jargon used, but as a developer, a primary goal is to simplify the user's experience so that the software becomes a facilitator of more efficient workflows rather than a hindrance. To improve on this idea of building a simple and effective interface, it will be necessary to continuously update the UI and be conscious of how much content is being displayed on the screen. The goal will be to minimize clutter and allow easy readability with a natural flow of information.

5.3 Security

Within any software development cycle, an essential pillar to consider is the tradeoff between time and security. Added levels of security will complicate each step of the development process, raising development costs in time and, in some cases, money. Within this application, security is an essential part of storing staff data such as private identification info, hours worked, things like sick pay, and overtime used, amongst other things. Regular employees would not want all of their personal information to be accessible to anyone in or out of the company, and administrators need to limit improper time log modifications by restricting edit access. To ensure a secure application, a secure login with role-based access will be implemented, ensuring the proper users have the proper access. The connection to the database and the data being sent will be encrypted. These and other security best practices will be implemented as needed.

6. Development Environment

6.1 Software Requirements

This project will be coded in JavaScript with React.js, potentially with some Typescript and Tailwind for formatting. An SQL database will be used for database storage. In order to promote scalability, the site will be hosted within AWS services to reach a broader user base.

Type	**Tools Used**
Programming Languages	JavaScript, TS, HTML
IDE	VS Code
Development System	Windows 11 Desktop

Table 6-1 Software Requirements

6.2 Hardware Requirements

The program will run as a web application to start and will not be hardware intensive. The program will run on any system with a browser and the Internet.

Type	**Hardware**
Processor	Intel core i3,5,7,9/AMD Ryzen processors
Processor Speed	Over 2.0 GHz
RAM	8 GB or more
Internet	25 Mbps or more

Table 6-2 Hardware Requirements

7. Reports and Products

The final application created in this project will be cross-platform and will assist in managing workplace productivity. This project will be presented and delivered for demonstration to the Computer Science Department and the advisor at the conclusion of the project. Additionally, materials will be provided, such as the source code and a step-by-step guide for using the application.

8. Schedule

Listed here is the planned timetable for the proposed project to allow it to be completed in a timely fashion. (See Table 8-1 below)

2025	January		February				March				April					Summary	
Week	1	2	1	2	3	4	1	2	3	4	1	2	3	4	5	Hours	Percent
Research	6	10	10	4												30	15.63%
Design		8	8	6	6											28	14.58%
Development				6	8	15	15	12	12	12	6					86	44.79%
Testing										4	2	2	2			10	5.21%
Modification											6	6	4			16	8.33%
Final Report												4	8	8		20	10.42%
Demonstration															2	2	1.04%
Hours	6	18	18	16	14	15	15	12	12	16	14	12	14	8	2	192	100%

Table 8-1 Schedule for development

9. References

Clara Bertolissi, Maribel Fernandez, and Bhavani Thuraisingham. 2024. Category-Based Administrative Access Control Policies. ACM Trans. Priv. Secur. Just Accepted (September 2024). https://doi-org.lib-proxy.fullerton.edu/10.1145/3698199

Yining Cao, Hariharan Subramonyam, and Eytan Adar. 2022. VideoSticker: A Tool for Active Viewing and Visual Note-taking from Videos. In Proceedings of the 27th International Conference on Intelligent User Interfaces (IUI '22). Association for Computing Machinery, New York, NY, USA, 672–690. https://doi-org.lib-proxy.fullerton.edu/10.1145/3490099.3511132

Jason Wu, Yi-Hao Peng, Xin Yue Amanda Li, Amanda Swearngin, Jeffrey P Bigham, and Jeffrey Nichols. 2024. UIClip: A Data-driven Model for Assessing User Interface Design. In Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology (UIST '24). Association for Computing Machinery, New York, NY, USA, Article 45, 1–16. https://doi-org.lib-proxy.fullerton.edu/10.1145/3654777.3676408