TASKCHECK
AN EMPLOYEE TASK TRACKING ANDROID
MOBILE APPLICATION

By
Jidnyasa K. Mantri
April 2019

TASKCHECK
AN EMPLOYEE TASK TRACKING ANDROID MOBILE APPLICATION
By
Jidnyasa K. Mantri

A project submitted in partial fulfillment of the requirements for the degree of

Master of Science in

Computer Information Systems

at

Grand Valley State University

April 2019

_____

Yonglei Tao   **04/25/2019**

Table of Contents

Abstract

Tracking of employee tasks is a very important part of the supervisor's job. Assigning all the employees their respective tasks and making sure they are performing them can be a very tedious process if it is performed manually. Therefore, this application can help the employer automate this process by simply downloading the app on their phones. With this application, the employer can create an account and make checklists for their employees. Similarly, the employees can sign up and use the checklists necessary. This application also provides several other functionalities to the employers such as searching the lists, reading the employees summary and editing the previously created checklists. It was developed using Java in Android Studio and is currently linked to SQLite database using APIs. By using this application, every small-scale organization employer can track the employee activities and make sure all the tasks get done on time, which in turn can help them ensure the quality of the products and the employee involvement.

Introduction

This project is essentially a simple mobile application which allows employers to keep a track of the tasks allotted to their employees. The idea for the application comes from the experience of working for dining services where the management handled all of the staff responsibilities through checklist. After noticing the multiple issues present with paper checklist such as misplacing or damaging the list, the need for the automation of this process was very clear. This application can definitely ensure more smooth and efficient functioning of the organization.

This problem can have different level of importance to different people but someone who has had the experience of working in small retail locations will definitely say that the severity of this problem cannot be ignored. Ensuring that all the employees are performing their tasks is a challenge every supervisor has regardless of the size of the organization. But this problem can seem significantly bigger to a supervisor who is also the sole owner of the business. They will have a lot of tasks to worry about, but the employee performance will certainly make the top three of their worries. Hence, this topic was selected for the project. The main goal of this project was to provide a candid task tracking application for the small company employees which was accomplished. Other smaller goals included making the application easy to use, lightweight and straightforward. There are several other ways which can be included to enhance this application which are mentioned in the future work section below.

Background and Related Work

A simple search on the Google Play Store will show that there are many employee tracking applications which focus on the employee hours, location etc. For example, an application called 'Timesheet' [1] can track the employees work time and location anywhere. Another application is called 'Labor Sync' [2].' and it tracks the employees work and status of their projects. Applications like these mainly focus on tracking the hours spend by the employees and the progress of their work. Although all these ideas are really useful, the idea behind this project is to create a no nonsense, straightforward application for allotting tasks to the employees and ensuring their completion.

To make sure of the originality of the idea, some time was spent to research the previous projects and it was found that most of these applications were tracking the employees' hours or they where just simple checklist applications. Also, as mentioned before, after working in retail for nearly two years, the need for this application was abundantly clear

Program Requirements

The basic requirement of this application is to create checklists and allow the employees to use them. But this application does a lot more than that. It starts with different login screens for both employees and employers and provides various other features which are as discussed below.

EMPLOYEE

For the employee, this application is pretty straightforward to use. It starts with needing the employee to either sign in or sign up for an account. The application will request very basic details such as company name, user name, email, and password. Once the user has signed in, the employee can only navigate through three screens which are shown in the Fig. 2 (Application Menus) below. The employee can then view and select the checklists and then check off the tasks completed by them. The employee screens on the right shows how the application will display those screens to the employees. Also, they can perform the 'Mark as completed' and 'Reset actions on these checklists as shown in the Fig. 1. When used, these menu options will mark all the tasks in a list and unmark all the tasks respectively.
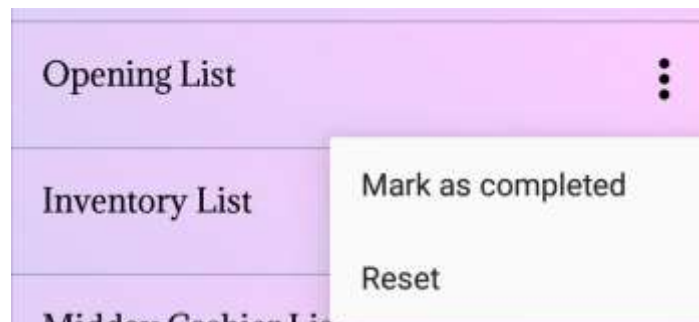

Figure 1 - Employee Checklist Options
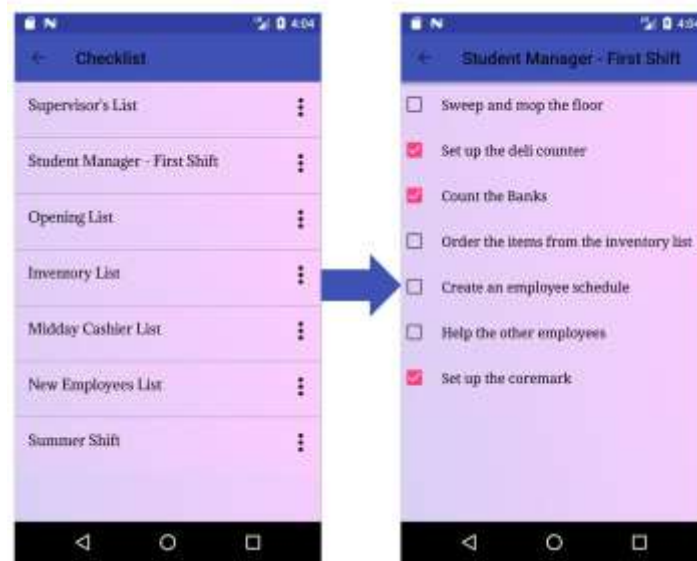
Figure 2 - Application Menus

Figure 3 - Employee Screens

EMPLOYER

This application provides majority of its functions to the employer. Just like the employee, the employer can too either sign up or sign into their account. Then they can navigate through five different screens mentioned above in the Fig. 1 which are Home, View Employee Summary, Search Checklists, Logout and Create and View Checklist. The Home screen is just a basic welcome page and the Logout screen is used to logout of the application. View Employee Summary page can be used to view the number of tasks completed by each employee as shown in Fig. 7. Search Checklists is an effective way of searching and

finding the needed checklists without having to scroll through all the lists present in the database. The Create and View Checklists is the most important screen in the application. With this screen, the employer can create new checklist and add new tasks for the employees. This functionality is essentially based on two screens which are the list screens and the task screens. The task screen displays the task which are present in certain lists and allows the employer the advantage to edit, delete or rearrange tasks. Meanwhile, the list screen provided the employer some functionalities which are available to the employees as well as some additional functions to edit or delete tasks as shown in the Fig. 4 below.
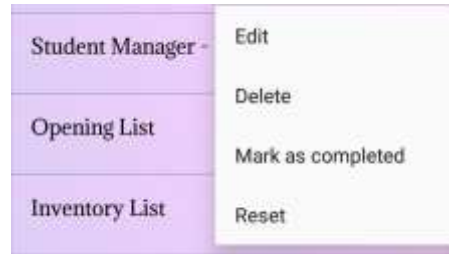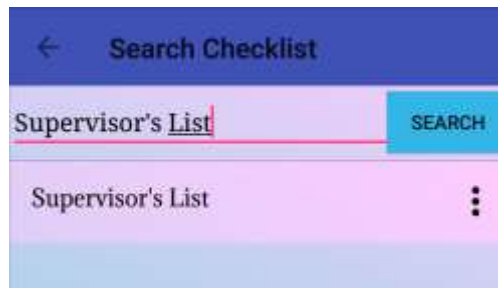

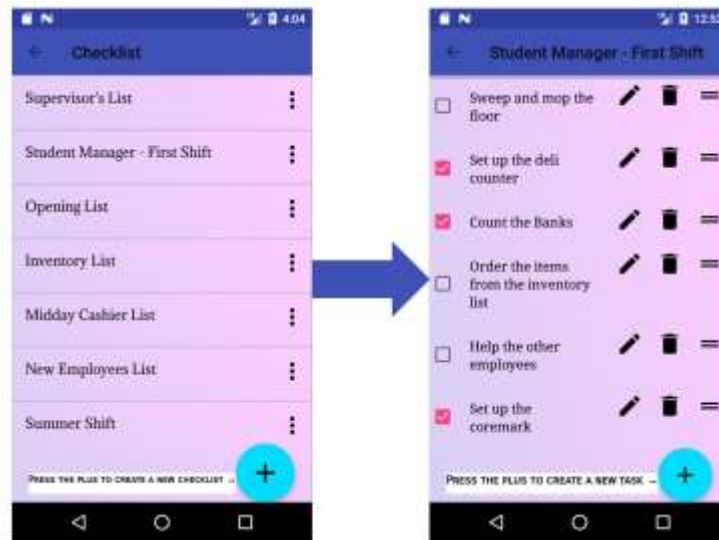**Figure 4 – Employer Checklist Options**

Figure 5 - Search Checklist
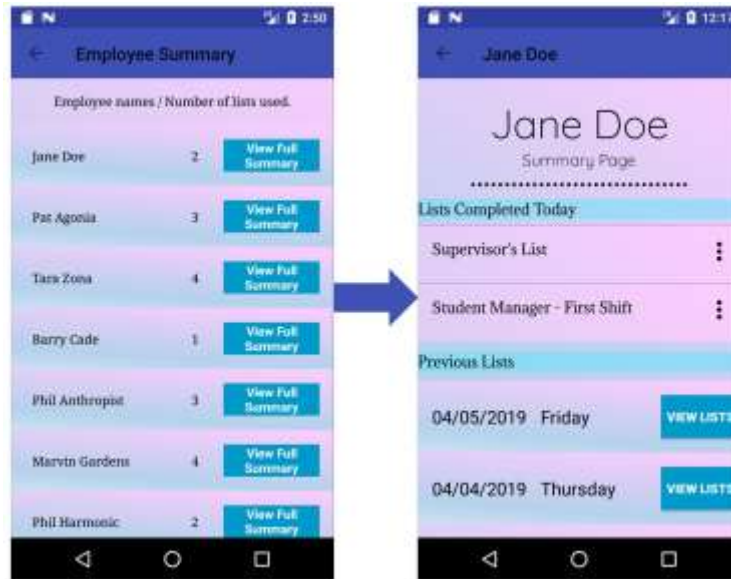

Figure 6 - Employer Screens

Figure 7 - Employee Summary

Implementation

As this project is a complex checklist application, the working is of the app itself is very simple to understand. A user logs in, creates or edits a list and an another can check tasks of this list. Therefore, every user has different read and write permissions depending on the user account type. Another thing that depends on the user's account are the pages that they are allowed to access. For example, an employee cannot view another employee's performance summary. So, the root idea of this app's functionality is the user's account level. The higher the account level, the more features you can access.

This project was developed on Android Studio [3] which is based on the IntelliJ IDE [4]. For the development, the programming language of Java [5] and XML [6] were used for the application logic and GUI respectively. The database currently used is SQLite [7] which helps store the data locally and makes the application functional. But in the future, the application will be linked to Google Firebase [8] to ensure that the data is stored online and hence can be accessible from anywhere.

Among all of the tools mentioned above, the author's experience mainly lies in the tools used for development purposes. Having used Java [5] and XML [6] before, the development part of the project was pretty straightforward. But the backend of the project was the area of conflicts. The first attempt for backend was through Google Firebase [8]. But after the multiple attempts, the connection seemed to be taking longer than expected. Therefore, the approach was changed to using SQLite [7]. There were several other languages considered such as SQL [9], JavaScript [10], PHP [11] and PL/SQL [12] but in the end SQLite was selected. Since, the author had no experience in SQLite [7], they had to start by studying the language for some time before starting the actual coding. Hence, this project was helpful in learning a new language which will be extremely beneficial in future projects.

Results, Evaluation, and Reflection

Overall, this project can be classified as a successful project. As mentioned before the main goal of this project was too to create an employee task tracking application and that was accomplished. But there are several other objectives of this project which were not done as well as expected or they could have been done better. For example, the employee summary page shown in Fig. 7, it currently only shows the employee lists completed by the employees while the initial plan was to create a complete dashboard to show the employee's performance. But this plan required integrating the application with Google Firebase [8] so as to keep the data constant online. By integrating Firebase in the application, we could easily call the required values and show them in a dashboard including showing not just the checklists but also completed task. Unfortunately, this wasn't achieved with SQLite [7] and the employee summary page right now just presents the checklists completed by the employees. In the future versions of this application, the author hopes to make the required changes in order to have a clear representation of the employees' performance.

Conclusions and Future Work

In conclusion, this application can provide a simple solution to the problem faced by every employer regarding task management. This mobile application can help automate the tedious process of employee task tracking and save the employer's time. Also, since the employer won't have to print out papers to hand out tasks for the employee's, this application also will help save both the company's and nature's resources. As for the future work in the development of this application, the three ideas that stand out are: Connecting it to Firebase, Different permissions for different users and Random Photo Activity Generator. The reason for wanting to connect the application to Firebase or any similar web service is mentioned above several times. And allocating different permissions for different users can help establish a hierarchy in the user accounts and not just have the two that are present currently in the application. Another idea is to create a random photo activity generator which will pop up randomly any time an employee checks the task. To close the activity, the employee will have to take a picture of the task they just checked off which will be saved in the database as a proof of the task completion. Therefore, allowing the employer to be even more assured regarding the successful completion of the task. There can be several more beneficial additions which can be made to the project in the future versions, but the ideas suggested above will be given the first preference. Once the application is perfectly functional, it will be deployed on the play store for public download. Hopefully, this application is helpful for a lot of small business owners and entrepreneurs to manage at least this one aspect of their extremely difficult job.

Bibliography

[1] "Best Mobile Time Clock App - Mobile Employee Timesheets," *TSheets*. [Online]. Available: https://www.tsheets.com/pages/android-twopart. [Accessed: 25-Apr-2019].

[2] "Labor Sync," *Labor Sync*. [Online]. Available: https://www.laborsync.com/. [Accessed: 25-Apr-2019].

[3] "Download Android Studio and SDK tools," *Android Developers*. [Online]. Available: https://developer.android.com/studio. [Accessed: 25-Apr-2019].

[4] "IntelliJ IDEA: The Java IDE for Professional Developers by JetBrains," *JetBrains*. [Online]. Available: https://www.jetbrains.com/idea/. [Accessed: 25-Apr-2019].

[5] "java.com: Java + You." [Online]. Available: https://www.java.com/en/. [Accessed: 25-Apr-2019].

[6] "XML," *Wikipedia*. 12-Apr-2019.

[7] "SQLite Home Page." [Online]. Available: https://www.sqlite.org/index.html. [Accessed: 25-Apr-2019].

[8] "Firebase," *Firebase*. [Online]. Available: https://firebase.google.com/. [Accessed: 25-Apr-2019].

[9] "SQL - Wikipedia." [Online]. Available: https://en.wikipedia.org/wiki/SQL. [Accessed: 25-Apr-2019].

[10] "Free JavaScript training, resources and examples for the community." [Online]. Available: https://www.javascript.com/. [Accessed: 25-Apr-2019].

[11] "PHP: Hypertext Preprocessor." [Online]. Available: https://www.php.net/. [Accessed: 25-Apr-2019].

[12] "PL/SQL," *Wikipedia*. 12-Apr-2019.