**EUROPEAN UNIVERSITY OF LEFKE**

FACULTY OF ENGINEERING

Graduation Project 2

Employee attendance and manage control using QR Codes

### Manasbek Rakhmatulloev

### 184121

Attendance taking is one of the main priorities of the companies, because it affects productivity of the company, productivity of employees. There are many different ways to track employee attendance such as manually entering employee hours into a spreadsheet, card swiping, fingerprint and retina scans, tracking employees through a GPS system, a traditional attendance register and QR code scan. Among them QR code attendance becoming very popular, because of its simplicity, low cost, accessibility and fast attendance time.

There are two types of QR code attendance systems:

* First one is giving every employee a card with individual QR codes on it. Employees show this card to special device every time when they are attending. But it has many drawbacks. Among colleagues and acquaintances, these cards can be exchanged or copied to log in with their details and then mark their attendance. It is also depends on electricity and Wi-Fi.
* Second one is using employee’s smartphone for scanning one QR code that was provided on the screen, door or wall. Employees scan QR code in special app on their smartphones. There is no need for special devices. It works even when there is no electricity or Wi-Fi. Employees can’t cheat in it. Employees can see their attendance results at any time on their smartphones.

Employee attendance and manage control using QR Codes uses the second approach (type), because it is more efficient and effective method than the first one.

**Supervisor**

Ezgi Deniz Ulker

Publish Date

08.06.2022

**Table Of Contents**

[Manasbek Rakhmatulloev i](#_Toc106037723)

[184121 i](#_Toc106037724)

[Ezgi Deniz Ulker ii](#_Toc106037725)

[Publish Date ii](#_Toc106037726)

[08.06.2022 ii](#_Toc106037727)

[1. Introduction 1](#_Toc106037728)

[1.1 Problem definition 1](#_Toc106037729)

[1.2 Goals 1](#_Toc106037730)

[2. Literature Survey 2](#_Toc106037731)

[3. Background Information 3](#_Toc106037732)

[3.1 Required software 3](#_Toc106037733)

[3.2 Other software 3](#_Toc106037734)

[4. Design Documents 4](#_Toc106037735)

[4.1 Data flow diagram 4](#_Toc106037736)

[4.2 Your Context Diagram 6](#_Toc106037738)

[5. Methodology 6](#_Toc106037739)

[6. Conclusion 54](#_Toc106037740)

[6.1 Benefits 54](#_Toc106037741)

[a. Benefits to users: 54](#_Toc106037742)

[b. Benefits to me: 54](#_Toc106037743)

[6.2 Ethics 55](#_Toc106037744)

[6.3 Future Works 56](#_Toc106037745)

[7. References 56](#_Toc106037746)

# 

# 1. Introduction

Smartphones play an important part in our daily lives in this technological age. Smartphones can now handle the majority of problems fast and easily. With many social apps, business apps, problem-solving apps, educational apps, and marketing apps, it has made everyone's life simpler and easier. QR code attendance system for employee app is one of them. It uses employee’s smartphone for scanning QR code that was provided on the screen, door or wall. There is no need for special devices. It works even when there is no electricity or Wi-Fi. Employees can’t cheat in it. It uses GPS, unique device id and user registration by the authority of the administrator to make it secure. Employees can see their attendance results at any time on this app.

## Problem definition

There are many other employee attendance systems.

They have many drawbacks:

* Slow Attendance Taking
* More Paper Waste
* Special Equipment Required
* Depends on Electricity and Wi-Fi
* Assembling and Disassembling is hard
* Special Training for employees
* Not Secure

## Goals

The purpose of the project is to make attendance system simple, low cost, accessible, fast and secure.

* **Faster Attendance Taking:** It takes only a few seconds per employee for checking in.
* **Least Paper Waste:** There is no need for papers, because the whole attendance data will be stored in the online cloud. This data can be accessed at any time.
* **No Special Equipment Required**: No need for special equipment. It only needs smartphone and generated QR code for attending.
* **No Electricity:** It works even when there is no electricity or Wi-Fi.
* **Easy Assembling and Disassembling:** There is no need to worry about arranging laptops, staff members and stationery. It is easy to use a smartphone as a QR code scanner via mobile apps.
* **Simple Pre-preparation:** There is no need for special training for the employees, because smartphone QR code scanners are simple and easy to use.
* **Secure:** It is very hard cheat or hack system. It uses GPS, unique device id and user registration by the authority of the administrator for that.

# 2. Literature Survey

Using punch cards, log books, fingerprint systems, barcodes and QR codes among the different types of attendance systems that have been created, nevertheless causes several issues, such as supplying inaccurate information to users. The goal of a smartphone-based QR code attendance system is to computerize the conventional method of recording attendance and give an easy and efficient approach to track attendance at institutions today. Smartphones are the most prevalent device used in marketing and business.

* **Compare1:** It does not provide individual QR codes to its users like many other QR code attendance systems, instead it uses one QR code for all users for attending.
* **Compare2:** It uses smartphone for scanning the QR code that was provided, unlike many other systems which use individual cards and special equipment for attending.
* **Compare3:** Employees can see their attendance results through application on their smartphones, unlike many other systems in which employees don’t know anything about their attendance.
* **Compare4:** It works even when there is no electricity, unlike many other attendance systems which depend on electricity and Wi-Fi.
* **Compare5:** It is more secure than other QR code attendance systems. It uses GPS, unique device id and user registration by the authority of the administrator.

# 3. Background Information

## 3.1 Required software

* **Flutter :**

Flutter is the best SDK for mobile application development.

* **Android Studio :**

Android Studio is the best IDE for mobile application development.

* **Django :**

Django for backend.

* **PyCharm:**

PyCharm is the best IDE for python projects.

## 3.2 Other software

* **Adobe Photoshop and Flaticon :**

For designing icons.

* **Git :**

Used for repository.

# Design Documents

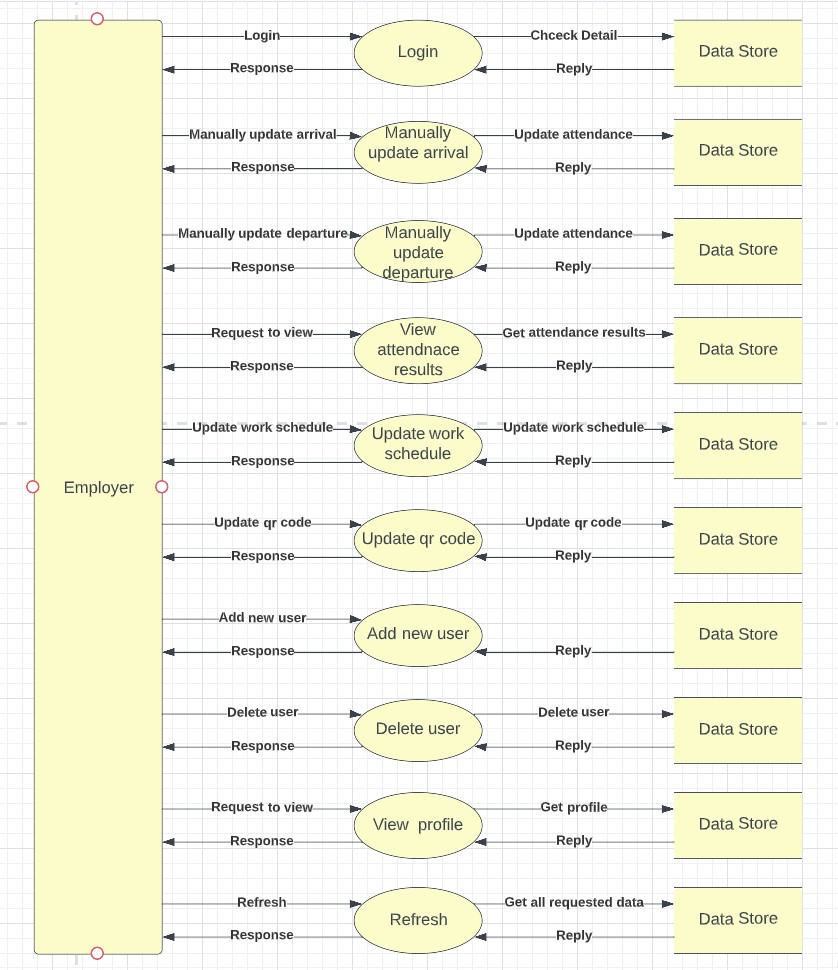
## Data flow diagram

* **Employee data flow diagram:**

**A picture containing diagram

Description automatically generated**

* Employer data flow diagram:

****

## Your Context Diagram

Diagram

Description automatically generated

# Methodology

This project has both frontend and backend. I wrote frontend in Flutter SDK and backend in Django Rest Framework. I connected them using Flutter’s http packages. I have not deployed backend to the online server, I used it as a local server. Phone can connect to the server using WI-FI or mobile hotspot.

One of the most important part of the project is that it can work in multiple platforms, like iOS, Android, Web, Windows, macOS and Linux.

**Flutter codes and Algorithms:**

**Pages:**

**There are in 8 screen pages in total:**

* **Login page**

**Graphical user interface, application

Description automatically generated**

* **QR code scanner page**

**Text

Description automatically generated**

* **Employee home page**

**Graphical user interface, application

Description automatically generated**

* **Employee attendance list page**

**Graphical user interface, text, application

Description automatically generated**

* **Employee Profile page**

**Graphical user interface, text, application, chat or text message

Description automatically generated**

* **Employer home page**

**Graphical user interface, application

Description automatically generated**

* **Employer attendance list page**

**Graphical user interface, text, application

Description automatically generated**

* **Employer add user page**

**Graphical user interface, text, application, chat or text message

Description automatically generated**

Registering pages to the router:

Text

Description automatically generated

**Dependencies:**

Text

Description automatically generated

* **Flutter:** For flutter apps we need this SDK package
* **Pie\_chart:** It is used for creating pie charts, that show daily and monthly attendance results of employees.

*Code:*

*Text

Description automatically generated*

* **Lottie:** It is used for showing lottie files, in other words animated files. Almost every page contains one lottie file.

*Code:*

*A picture containing text

Description automatically generated*

* **syncfusion\_flutter\_datepicker:** It is used for date and time picking in the app.

*Code:*

*Text

Description automatically generated*

* **Mobile\_scanner:** It used for scanning qr codes through camera.

*Code:*

*Text

Description automatically generated*

* **Http:** It is used for making http requests to the database.

*Code:*

*Text

Description automatically generated*

* **Flutter\_riverpod:** It is used as an provider for storing user preferences.

*Code:*

*Text

Description automatically generated*

* **Shared\_preferences:** It is used for saving user preferences like

Text

Description automatically generated

* **Intl:** It used for formatting the date

*Code:*

*Text

Description automatically generated*

* **Qr\_flutter:** It is used for generating QR code.

*Code:*

*A picture containing background pattern

Description automatically generated*

* **Path\_provider:** It is used for getting specific path from mobile device.

*Code:*

*Graphical user interface, text, application

Description automatically generated*

* **Gallery\_saver:** It used for saving images to the gallery.

*Code:*

*Text

Description automatically generated*

**HTTP requests:**

* **getToken:** This http request used for logging in and getting authorization token.

*Code:*

*Text

Description automatically generated*

* **getUserType:** This http request used for getting user type information, like if it is employee or employer.

*Code:*

*Text

Description automatically generated*

* **checkQRCode:** This http request used for comparing the scanned QR code with the QR code in the database.

*Code:*

*Text

Description automatically generated*

* **arrivedAt:** This http request used for saving arrival time and departure time in the database.

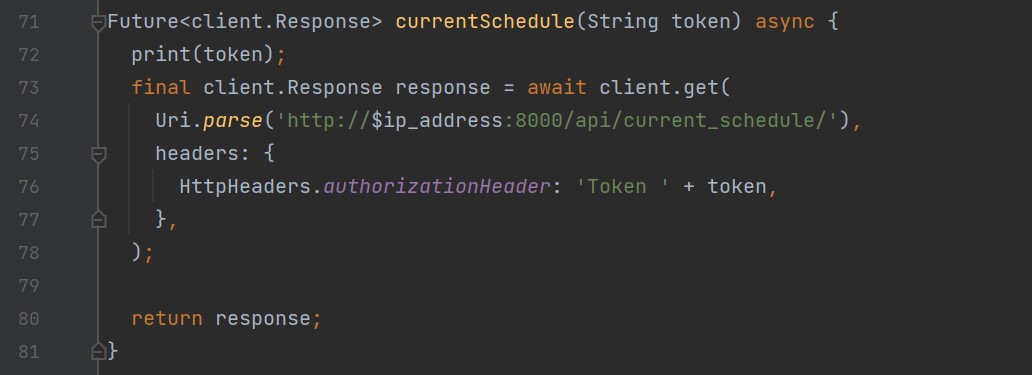
*Code:*

*Text

Description automatically generated*

* **currentSchedule:** This http request used for getting current work schedule.

*Code:*

**

* **workSchedule:** This http request used for getting weekly work schedule.

*Code:*

*Text

Description automatically generated*

* **getUserMonthlyAttendance:** This http request used for getting monthly attendance results of the specific user.

*Code:*

*Text

Description automatically generated*

* **updateQRCode:** This http request used for updating current QR code in the database.

*Code:*

*Text

Description automatically generated*

* **updateWorkSchedule:** This http request used for updating current working schedule in the database.

*Code:*

*Text

Description automatically generated*

* **updateIsDayOff:** This http request used for updating weekly working and not working days.

*Code:*

*Text

Description automatically generated*

* **getUsersDailyAttendance:** This http request used for getting all the employees’ daily attendance results.

*Code:*

*Text

Description automatically generated*

* **getUserAuth:** This http request used for getting all the authenticated users’ information from the database.

*Code:*

*Text

Description automatically generated*

* **AddUser:** This http request used for adding new employee or in other words user to the database.

*Code:*

*Text

Description automatically generated*

* **DeleteUser:** This http request used for deleting user from the database.

*Code:*

*Text

Description automatically generated*

* **AddStaff:** This http request used for adding new employer or in other words user to the database.

*Code:*

*Text

Description automatically generated*

* **GetUser:** This http request used for getting specific employer information form the database.

*Code:*

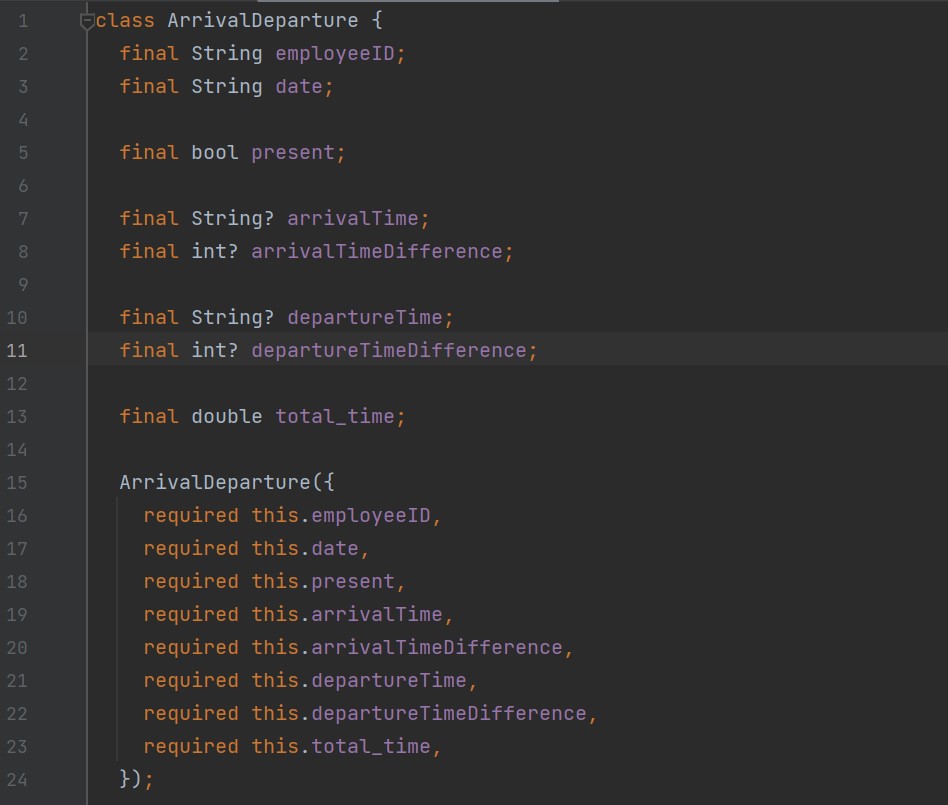
*Text

Description automatically generated*

**Models:**

* **ArrivalDeparture:** This model is used for converting arrival departure model information to flutter objects or from flutter objects to arrival departure model.

*Code:*

**

*Text

Description automatically generated*

* **Auth:** This model is used for converting user authentication model information to flutter objects or from flutter objects to user authentication model.

*Code:*

*Text

Description automatically generated*

* **User:** This model is used for converting user profile model information to flutter objects or from flutter objects to user profile model.

*Code:*

*Text

Description automatically generated*

*Text

Description automatically generated*

*Text

Description automatically generated*

*Text

Description automatically generated*

* **WorkSchedule:** This model is used for converting work schedule model information to flutter objects or from flutter objects to work schedule model.

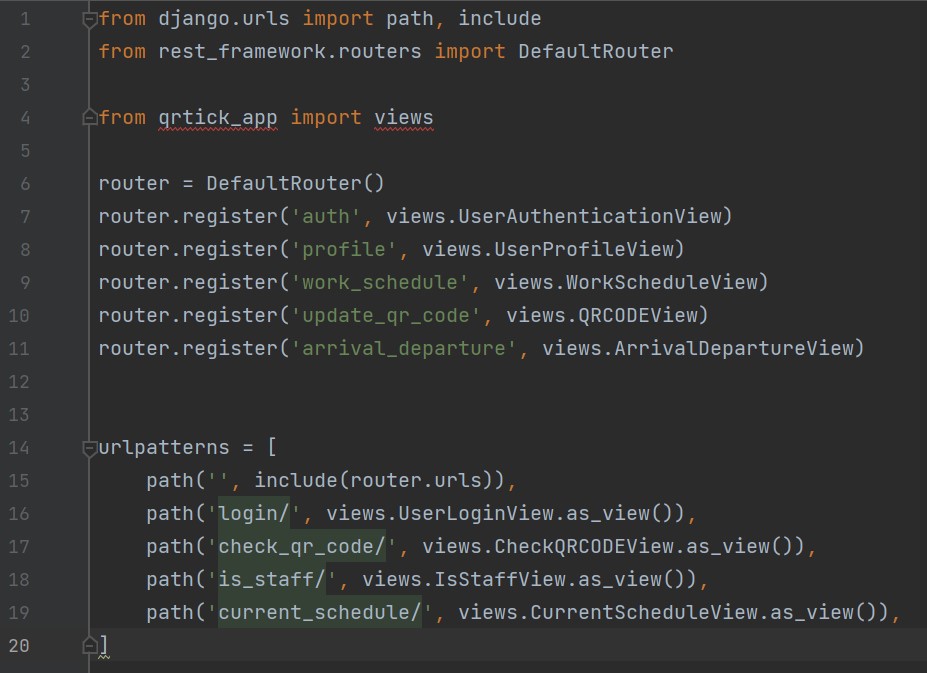
*Code:*

*Text

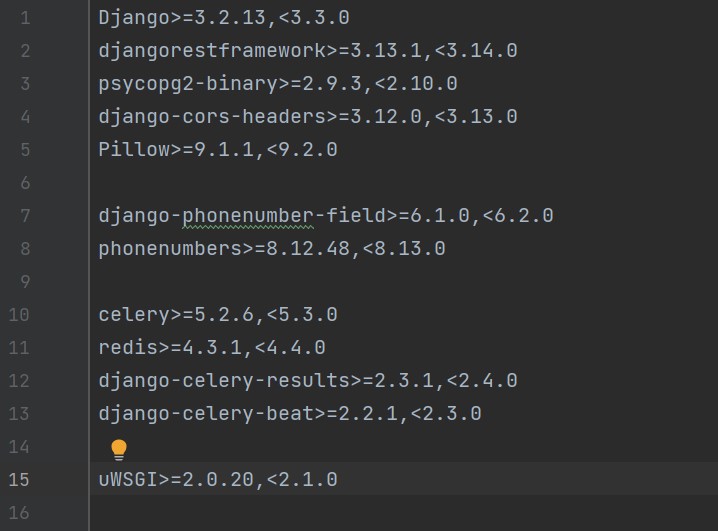
Description automatically generated*

**Django REST API codes and Algorithms:**

**Routes:**

****

**Dependencies:**

****

**Models:**

* **UserAuthentication:** It is used for storing user authentication information.

*Code:*

*Text

Description automatically generated*

* **UserProfile:** It is used for storing user information.

*Code:*

*Text

Description automatically generated*

* **ArrivalDeparture:** It used for storing user’s arrival and departure information.

*Code:*

*Text

Description automatically generated*

* **WorkSchedule:** It used for storing weekly work schedule.

*Code:*

*Text

Description automatically generated*

* **QRCODE:** It is used for storing QR code.

****

**Views:**

* **UserAuthenticationView:** It is used for receiving http requests for user authentication model.

*Code:*

*Text

Description automatically generated*

* **UserLoginView:** It is used for receiving http requests for logging in and getting authentication token.

*Code:*

*Text

Description automatically generated*

* **UserProfileView:** It is used for receiving http requests for user profile model.

*Code:*

*Graphical user interface, text

Description automatically generated*

* **WorkScheduleView:** It is used for receiving http requests for work schedule model.

*Code:*

*Text

Description automatically generated*

* **QRCODEView:** It is used for receiving http requests for QR code model.

*Code:*

*Text

Description automatically generated*

* **CheckQRCODEView:** It is used for receiving http requests and comparing QR codes.

*Code:*

*Text

Description automatically generated*

* **ArrivalDepartureView:** It is used for receiving http requests for arrival departure model.

*Code:*

*Text

Description automatically generated*

* **IsStaffView:** It is used for receiving http requests and checking if the user is staff user.

*Code:*

*Text

Description automatically generated*

* **CurrentScheduleView:** It is used for receiving http requests and returning current schedule back to the user.

*Code:]*

*Text

Description automatically generated*

**Serializers:**

* **UserAuthenticationSerializer:** It is used for serializing and deserializing user authentication model.

*Code:*

*Text

Description automatically generated*

* **UserProfileSerializer:** It is used for serializing and deserializing user profile model.

*Code:*

*Text

Description automatically generated*

* **WorkScheduleSerializer:** It is used for serializing and deserializing work schedule model.

*Code:*

*Text

Description automatically generated*

* **QRCODESerializer:** It is used for serializing and deserializing QR code model.

*Code:*

*Text

Description automatically generated*

* **CheckQRCODESerializer:** It is used for serializing and deserializing request data from check QR code view.

*Code:*

*Text

Description automatically generated*

* **ArrivalDepartureSerializer:** It is used for serializing and deserializing arrival departure model.

*Code:*

*Text

Description automatically generated*

* **IsStaffSerializer:** It is used for serializing and deserializing request data from is staff view.

*Code:*

*Text

Description automatically generated*

**Cronjobs:**

* **Every\_2\_hour\_00:00: 0 \*/2 \* \* \*:** This crontab executes some task every two hours every day.

*Code:*

**Text

Description automatically generated**

* **Create\_daily\_attendance\_list:** This task is used for creating daily attendance list for every working day.

*Code:*

*Text

Description automatically generated*

**Permissions:**

* **UpdateOwnAuthroization:** Employers can change or update employees’ authorization information. Employees can change only their own authorization information.

*Code:*

*Text

Description automatically generated*

* **UpdateOwnProfile:** Only Employees can change or update their own information in the database.

*Code:*

*Text

Description automatically generated*

* **UpdateWorkSchedule:** Only employers can change or update the work schedule. Employees can only get work schedule information.

*Code:*

*Text

Description automatically generated*

* **UpdateArrivalTime:** Only employees can update their arrival and departure time.

*Code:*

*Text

Description automatically generated*

**Docker:**

* **Dockerfile:** This file creates python container in virtual machine.

*Code:*

*Text

Description automatically generated*

**Text

Description automatically generated**

* **Docker-compose:** This file creates and executes all the containers in virtual machine.

*Code:*

*Text

Description automatically generated*

*Text

Description automatically generated*

*Text

Description automatically generated*

# Conclusion

The purpose of this project is to make attendance system simple, low cost, accessible, fast and secure.

## Benefits

### Benefits to users:

1. **Faster Attendance Taking:** It takes only a few seconds per employee for checking in.
2. **Least Paper Waste:** There is no need for papers, because the whole attendance data will be stored in the online cloud. This data can be accessed at any time.
3. **No Special Equipment Required**: No need for special equipment. It only needs smartphone and generated QR code for attending.
4. **No Electricity:** It works even when there is no electricity or Wi-Fi.
5. **Easy Assembling and Disassembling:** There is no need to worry about arranging laptops, staff members and stationery. It is easy to use a smartphone as a QR code scanner via mobile apps.
6. **Simple Pre-preparation:** There is no need for special training for the employees, because smartphone QR code scanners are simple and easy to use.

### Benefits to me:

1. Increase my knowledge in mobile and backend development
2. Increase my self-confidence as a Mobile developer
3. In-depth understanding of QR codes
4. Increase my project management and problem solving skills

1. Increase my creative and critical thinking skills

**Why did I choose this project?**

I chose this project because wanted to learn more about QR codes. I think nowadays every programmer should know about it. Recent years, I began to notice QR codes everywhere, from Wi-Fi passwords up to COVID-19 test results, so started to interest in this topic. I searched some QR code projects from the internet and found that there are only a few projects about the QR code attendance system. These projects had some drawbacks, so I wanted to do a new better one. I think there are also many fields in which QR codes can be used, but I started it with QR code attendance system.

## Ethics

* Employees must use smartphones which have Camera and GPS.
* Employees must bring their smartphones that were registered by the admin, to the work.
* Employees must use QR code attendance application for scanning QR codes that were provided by the employer.
* Employees must allow access to Camera and GPS for QR code attendance application on their smartphones.
* Employees must scan QR codes that were provided by the employer with their smartphones, while arriving and leaving.
* If there are problems with attending, employees must visit employer (admin) office for manual attendance.
* If employees change their phone to another one, they must inform about it to the employer (admin) for using QR code attendance application on their new phones.
* Attendance can be quantified and verified, and employers keep track of employee attendance and leave records.

## Future Works

For now, I have no idea, but if there is something I can add or improve I will do it.

# References

[1]: QR Attendance System => [https://www.attendezz.com/#](https://www.attendezz.com/)

[2]: Using QR codes for employee and task tracking => <https://www.allgeo.com/blog/employee-and-task-tracking-using-qr-codes>

[3]: QR Code Attendance for Faster Attendance Taking => <https://hrmlabs.com/qr-code-attendance-for-faster-attendance-taking/>

[4]: Как использовать QR-коды для отслеживания посещаемости и участия? => <https://pageloot.com/ru/qr-коды-для/отслеживание-посещаемости/>

[5]: Managing Employee Attendance => <https://www.shrm.org/resourcesandtools/tools-and-samples/toolkits/pages/managingemployeeattendance.aspx>