Faculty of Computer & Information Technology

“Intelligent Payroll System (I.P.S)”

2022

Under Supervision of:

Dr. Kamal Hamza

Eng. Safinaz allam

National Egyptian E-Learning University (EELU)

Faculty of Computer & Information Technology

"Intelligent Payroll System (I.P.S)"

Submitted By:

No	Name	ID
1	Nashaat Nabil Ramzy (T.L)	18-00206
2	Froeid Samer Youssef	18-00039
3	Karl Bassem Kamal	18-00076
4	Abanob Hesham Gamil	18-00130
5	Abanob Sabet Sabet	12-00061

Under Supervision of:

Dr. Kamal Hamza	Eng. Safinaz allam
- Professor of Information Systems at Egyptian E-Learning University - Computers and Information Technology program Director	- Teaching Assistant in Computer and Information Technology faculty At Egyptian E Learning University

" This document is presented as partial fulfillment of the requirements for a Bachelor of Information Technology degree "

EELU- Assiut Center 2022

Acknowledgement

In the beginning, all praise and thanks be to God for the success and care that He granted us and helped us to complete our project.

We also owe all thanks and appreciation to Prof. **Dr. Hisham Mohamed Abdel Salam** (President of the Egyptian University for E-Learning),

Dr. Mahmoud Shoman (Dean of the Faculty of Computers and Information Technology at the Egyptian University for E-Learning) **and Dr. Mostafa Hashem** (Director of the Study Center in Assiut), for their endless support, guidance and encouragement in throughout the research process, and in a special letter of thanks.

We extend our sincere thanks and appreciation to the (computer and information technology program director) and our supervisor **Dr. Kamal Hamza**, for his continuous motivation and support during our work on implementing this project, we appreciate her kind guidance and encouragement to us from the beginning of this idea to its end.

We also extend our sincere thanks and appreciation to our supervisor, who was considered one of the team members, **Eng. Safinaz Allam**, for her tremendous efforts and the exceptional support she provided us during our work on the project and for providing the means of assistance to us during the implementation of our graduation project, whether assistance from people or training courses,

We owe it to her because she helped us shape the problem and provide insights into the solution, we would not have it done in a timely manner without her support.

Abstract

Through this document, we look forward to present our system and explaining the idea of our graduation project in a semi-detailed manner, as we seek to explain all the components of our system through 9 chapters, which begin with the Introduction in which we tried to explain the problem briefly and then presented the idea of our project and How do we seek to develop a system capable of solving these problems, then in the second chapter we described the main points that face the market to solve the problem in 2 factors, and in the third chapter reviewing the literature, we presented all the ideas on the ground that combine some qualities with our system, and we tried to show the positives of those ideas and take advantage of them and search for their negatives and problems and try to find solutions to make our system more attractive to customers, and through the fourth chapter we presented the methodology that we followed during the implementation of the project, which is the (Agile) methodology and how those steps were carried out, we also tried in the fifth chapter to present some of the challenges and difficulties that we faced during the implementation of those steps and clarifying the programs and applications that we used in implementing our system, in the sixth chapter we have drawn up schemes for the system requirements, which is a description of what the system and service we provide should do, in the seventh chapter we have put pictures of all the components of the system software, and pictures of the mobile application and Dashboard Website for our system, so that the reader of the document can get to know the system as much as possible, through the eighth chapter we provided some screen shots for some of our source code functions and, in the last chapter we put a summary of the project with some recommendations and some future work that we seek to implement and develop in the future.

This project is about to achieve a qualitative leap in the human resources management systems market, as we seek to introduce our Intelligent Payroll System.

Contents

CHAPTER (1)

Introduction

Introduction:

In general, many problems occur because of attendance and salary calculation for each employee in companies.

Recording absence and attendance manually is inaccurate and unfair.

- The objective of the proposed system is to record attendance accurately, fair, fast and very convenient for employees, the employer or the company, and calculate the employee's salary correctly and quickly, and it will be easy for the employee to know the reason for any deduction and to know the days of his attendance and absence easily.

- The idea of the project is a program when the employee arrives at the company, he will take a selfie using a phone camera, and the program will take his geographical location using the (GPS) to find out whether he is in the company or not, and based on that the employee's attendance is recorded today.

The employee takes the same steps upon completion of work to leave the company.

Chapter(2)

Problem Statement

2.1 For Manager:

- Knowing each employee when he arrived at the company and when he left the company or if the employee ran away or escape from work
- The problem of calculating each employee's salary according to the days of attendance and absence, and if there is any deduction from the salary or bonus
- Knowing the excuse of each employee and the difficulty of collecting all acceptable or rejected excuses
- The cost of any biometric system is expensive and needs an insurance system with a high additional cost

2.2 For Employee:

- The method of registering for attendance is difficult and takes time and is sometimes inaccurate
- Difficulty knowing arranged details of deductions or bonuses
- Difficulty sending the manager an excuse for being late or absent

CHAPTER (3)

Previous or similar solutions and their advantages and disadvantages

3.1 Attendance by Fingerprint

Method: The employee delivers and puts his fingerprint on the device and records the attendance.

Features : Absence is taken with the fingerprint and it is difficult to play with it or counterfeiting it.

Disadvantages:

1- The device is expensive and requires Infrastructure connection.
2- Paralyzing the movement of employees if their number is large and waiting for their turn to reach the device, and therefore the employee can record a delay in his waiting.
3- Recording attendance and departure only and has no other benefit.

3.2 Attendance by APP

Method: The employee takes a picture of himself when he arrives and when he leaves.

Features: Taking attendance in an easy, simple and fast way

Disadvantages:

1- The employee takes pictures of himself in different clothes and sends them at the time of attendance even if he does not arrive.
2- The APP recording attendance and departure only and has no other benefit.

3.3 Our System Features and Solution:

1- The way to register attendance is easy, simple and fast.

2- Make sure that the employee is already in the company to exclude any problem in attendance.

3- Inside the App, there is a personal page for the employee with all his personal information and his salary.

4- The ability to know the salary details of any deducts or bonuses.

5- There is no cost from specialized devices and security because already secured.

Chapter (4)

PROPOSED METHOD& APPROACH

4.1 Project implementation methodology:

In the beginning, a project is defined as a set of activities with specific beginning and end times that aim to manufacture a product or provide a service or any result of interest.

There are many types of projects, including small projects that employ a small number of workers, and their financing is from one person, or a limited number of people, medium enterprises, which employ a larger number of workers, and large enterprises, which employ a large number of workers, and depend on a large capital.

The projects are classified according to the subject they are working on, as there are commercial, industrial, cultural and political projects. The implementation of a project is subject to several developments and many modifications, depending on the surrounding circumstances and the available possibilities of money, time and human resources.

The project implementation steps are the cornerstone for the success of any project, as they include the effective use of available resources throughout the duration of the project, and constant communication with project staff, suppliers and customers. That the project manager has a schedule of all the activities that need to be implemented and what are the needs of each activity in terms of resources and implementation time.

4.2 (Agile) methodologies:

As we mentioned previously, every approach to project implementation has its pros and cons, and the waterfall approach is one of the most obvious work methodologies, but there is a negative point that represents a major obstacle in implementing projects, which is that it is a process of progress in one direction to

a large extent ("down" like a waterfall) and going back is very difficult and you need to start the implementation steps from the beginning.

That is why we turned to the (Agile) methodology specifically, for two reasons: The first reason is that it is similar to the waterfall methodology in terms of clarity and ease of understanding.

The other reason is that it solves the main problem of the waterfall methodology represented in the need to follow steps, as the (Agile) methodology is one of the gradual models of software development based on principles that focus more on people, results, cooperation and flexible responses to change. Instead of planning the entire project, it breaks down development work in small increments, completed in iterations, or short time frames. Each iteration includes all stages of the SDLC so that a working product is finally delivered. After several iterations, a new or updated product is launched.

The following is a simplified explanation of the (Agile) methodology .

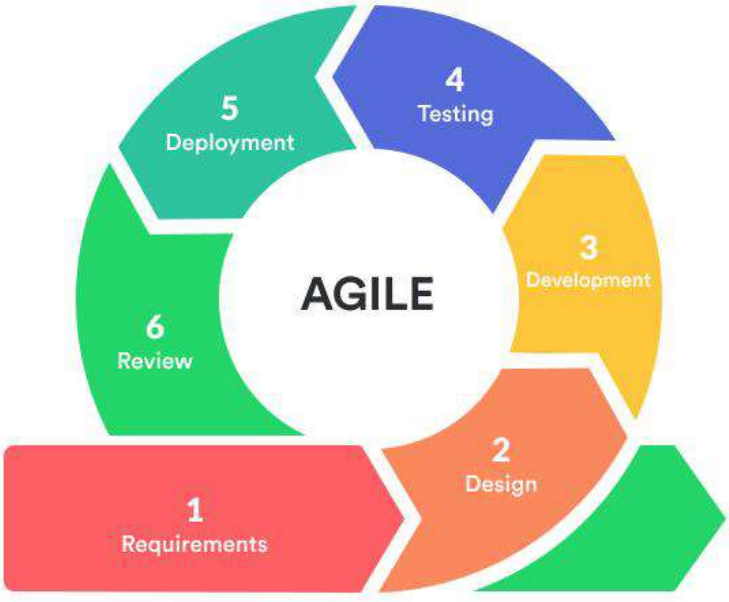


Figure 1: Agile Model Steps

❖ So what is the Agile methodology?

Agile is a methodology or principles in the software development process by which software is created and developed through collaborative efforts within a self-organised, cross-functional team.

These principles advocate adaptive planning, incremental evolution, early delivery, and continuous improvement, and encourage rapid and flexible response to change. These principles underpin the definition and continued development of many methodologies/software development approaches and the term was coined in the year 2001 when the Agile General Statement was formulated.

Agile's methods generally promote a project management process that encourages frequent inspection and adjustment, a leadership philosophy that encourages the teamwork of a self-organization, a set of best practices engineering that allows for the delivery of high-quality, agile software to the customer, and a business outlook that demands development. and company objectives. The conceptual foundations for this framework are found in modern approaches to management and operations analysis.

They are better ways to develop software by applying it and helping others to do so.

4.2.1 Agile Software Development Values:

In 2001, these 17 developers met at a resort in Snowbird, Utah to discuss these lightweight development avenues: Kent Beck, Ward Cunningham, Dave Thomas, Jeff Sutherland, Ken Schwaber, Jim Highsmith, Alistair Cockburn, Robert C. Martin, and Mike Biddle Ari Van Bennicum, Martin Fowler, James Greening, Andrew Hunt, Ron Jeffries, John Kern, Brian Marek and Steve Mellor. Building on their shared experience in software development and helping others to do so, they have jointly published the Manifesto on Agile Software Development containing four values:

A - Working software over comprehensive documentation:

That is, in general, in the Agile method, we care and focus on producing a software (product) that works with certain basic characteristics, and this is our vision of our insistence on providing a good product that meets the needs of the customer with an explanation of how the APP works and what it is The advantages which offers, and the desired characteristics that aim to help the Startups in the first place. It is not important to write the stage documentation with details, and this does not mean staying away from documentation, but only documenting important things such as the method of operating the product, and it is usually done after making sure that the product works with the required characteristics.

b- Individuals and interactions over processes and tools:

That is, attention and focus on team members and interaction among them instead of focusing on work tools and the flow of processes used to build the product, and this is one of the most important features of a successful work team, meaning that all members of the team work to make the project success and provide an app for the startups that meets all his needs and helps him to do Its tasks are more accurate, faster and less expensive, as the tools will not do anything without people who are able to use them correctly to accomplish the thing required of them.

C - Cooperation and participation with the customer in building the product is more important than negotiating contracts (Customer collaboration over contract negotiation):

Contracts are generally inflexible, especially in emerging projects, when the customer (the customer) explains his product and how he perceives it. The product is often not clear-cut, and therefore it is necessary to sit with the HR or employee or the specialist and hear the most important problems he faces and try to solve them and inquire about the most important advantages and the additions he seeks to have in our system, that is, the customer does not give the necessary details to build the product correctly, so when signing a contract at the beginning of the work (tasks, cost, time) we have determined what we will do and what will be produced, and therefore The modification becomes very difficult because the modification requires time, effort and cost, and often results

in problems with the customer. In Agile, the customer is involved in every step, and this gives the customer more understanding of what he wants because he sees the particular product, and therefore the modification or addition the product is easier and more flexible, and it gives him a full picture of the cost he will pay. This value in particular is one of the most important values in our project, especially in the future.

D - Responding to change is more important than following the plan set in the workflow (Responding to change over following a plan):

That is, attention to the implementation of the changes requested by the customer when the product is presented to him, his observations have priority over what is planned for the product, that is, it is also necessary to sit with the customer and take care of implementing the changes he requests, because he is the most knowledgeable of the problems he faces, and therefore its requirements should not be neglected because it is one of the most important factors for the success of the project, and as we mentioned earlier that the customer cannot fully describe the product from the beginning of the project, and this does not mean that there is no plan, the plans in (Agile) are general and level plans High and not a plan in detail.

This does not mean that the points that have been preferred (less important) are of no value, but here the prioritization is in favor of the agile values.

4.2.2 Principles of Agile Software Development:

Based on the previous values, several principles were reached that these seventeen software developers agreed on, and they are 12 principles of software development, and they are:

A - Gaining customer (client) satisfaction by providing an effective and usable (product) program, and adhering to delivery times.

B- Accepting and welcoming changes in requirements by the customer (client), even if at an advanced stage of development.

C- Deliver usable software in the shortest possible time and at regular intervals.

D - Programmers and technicians must work closely and with each other on a daily basis throughout the duration of the project.

E - Face-to-face conversations between team members are the best and fastest way to transfer information between the team.

F - Building projects by relying on motivated individuals, giving them the environment and the support they need, trust, appreciation and empowerment, is the right environment to motivate them more to make the project succeed

G- Software that works (usable product), is the main measure of progress.

H - Agile encourages the continuous development of individuals, their skills and knowledge. Sponsors and developers should always be able to maintain a constant rate of progress.

I - Continuous attention to excellence and quality in technical development and design.

J - Simplicity is an essential, vital and important part of Agile, i.e. reducing unnecessary and unimportant work.

K - Self-organizing work teams, providing the best requirements, structure and design.

L - The work teams evaluate and monitor their work to become more effective, and then control errors and behavior at regular intervals.

Chapter (5)

General steps for project implementation:

In fact, there is a great similarity in the basic and recognized steps for implementing any project in most methodologies, and these steps are activities that aim to ensure that implementation is carried out according to the objective plan, and it is a process of full assessment of the situation.

But what distinguishes one methodology from the other is the flexibility in implementing these steps to reach the desired product, and therefore we chose the (Agile) methodology in implementing the steps in our project.

The following steps are the basic general steps for the implementation of any project, whether commercial, humanitarian or service:

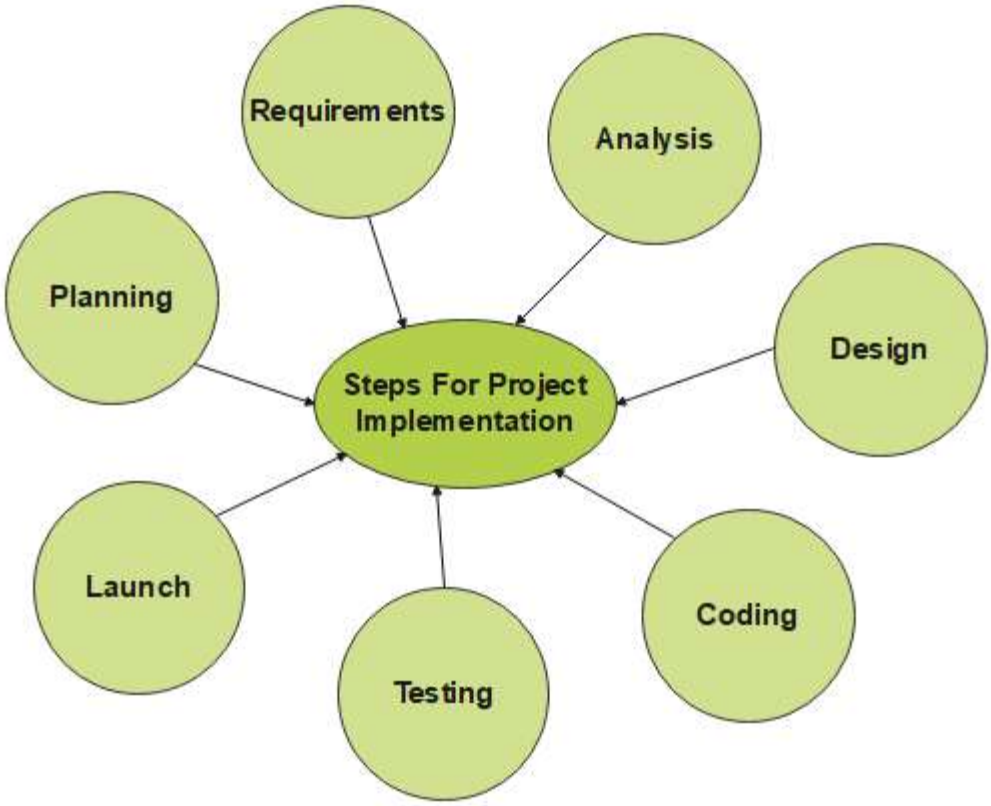


Figure 2: steps for project implementation

5.1 Planning:

In which a structure is prepared that includes the technical aspects, where we have planned the initial form that we seek to reach during our work on the implementation of the project, and the financial aspects by setting an initial cost for the project. It also includes the human resources represented in the division of the work team and define each individual has his role in the project, the scope of the project and the goals that we seek to achieve during the implementation period of the project to achieve the greatest benefit for the Staff management, and most importantly how these goals will be achieved, the expected final output, work to organize the progress of the projects, and evaluate the expected success of the project.

5.2 Requirements:

Determining the project requirements is one of the basic and very important elements necessary for the successful implementation of the project. We have identified the elements that we will use in our project, whether these elements are tools for designing the hardware part of the project or even tools for implementation or programs, applications and software to implement the software part of the project. After the project determines the appropriate method of production, it must determine the basic requirements of the production elements used in the production process and according to the requirements of the production method. Selected by the selected project.

5.3 Analysis:

At this stage, these data are analyzed and transformed into indicators that represent information about the time, accuracy, quality and cost of implementation, as this step is one of the most difficult and complex steps during the implementation of the project, as it was necessary to collect the largest amount of information about the systems The smart and non-smart ones that provide a service similar to the services provided by our system and analyze them to take advantage of the positive points in them, as well as identify the negative points and problems and try to solve them, and try to make solving these problems one of the factors that attract Owners of companies to our system, which will achieve sales for us , after that dividing the project into smaller steps and dealing with each of them separately, and from here we have chosen the (Agile) methodology to be our methodology for implementing the project in order to gain a better and easier understanding of the problem and objectives.

5.4 Design:

Here, solutions are designed for the requirements we have collected and the tasks and responsibilities for all stages are implemented. This stage is the actual beginning of the implementation of the project, and an attempt to benefit from every information collected in the previous stages, and based on the technical and financial detail Prepared in advance, and in which certain activities such as formulating and arranging, achieving goals. and at this stage, we actually started designing the database schema of the project in the best possible way and with all the tools and features available to us.

5.5 Coding:

Coding or writing the code, this stage is one of the longest stages in the implementation of the project and the most important because it is during which the codes responsible for doing most of the work in the project are written, and also this stage was the most difficult stage of the project without a dispute as it have to search in many sources and learn a lot of things and training courses to reach the appropriate codes that are able to carry out all the tasks that we have planned since the beginning of the project and try to collect all these codes together to reach the final form of the system.

5.6 Testing:

It is the process of testing the product if it is similar to the requirements that we set previously or not, where the system is tested and the accuracy of the results is known and whether the system achieves the functions and features that we planned for during the project, and whether it needs to be modified or added.

5.7 Launch:

The last steps of project implementation, in which reports and achievements of the project are presented, and the document for the project is written, and each part of the project is clarified in detail, then recommendations are made, with the required tasks delivered the project is closed by evaluating the outputs and staff performance, delivering requirements, terminating contracts for employees and service providers working for the project, identifying ways to communicate with the work team if the user encounters problems or if he has some additions or suggestions.

Chapter (6)

Diagrams

6.1 Flowchart Diagram:

A flowchart is a type of diagram that represents a workflow or process. A flowchart can also be defined as a diagrammatic representation of an algorithm, a step-by-step approach to solving a task.

The following is a general Flow chart for the Web App or the Android App

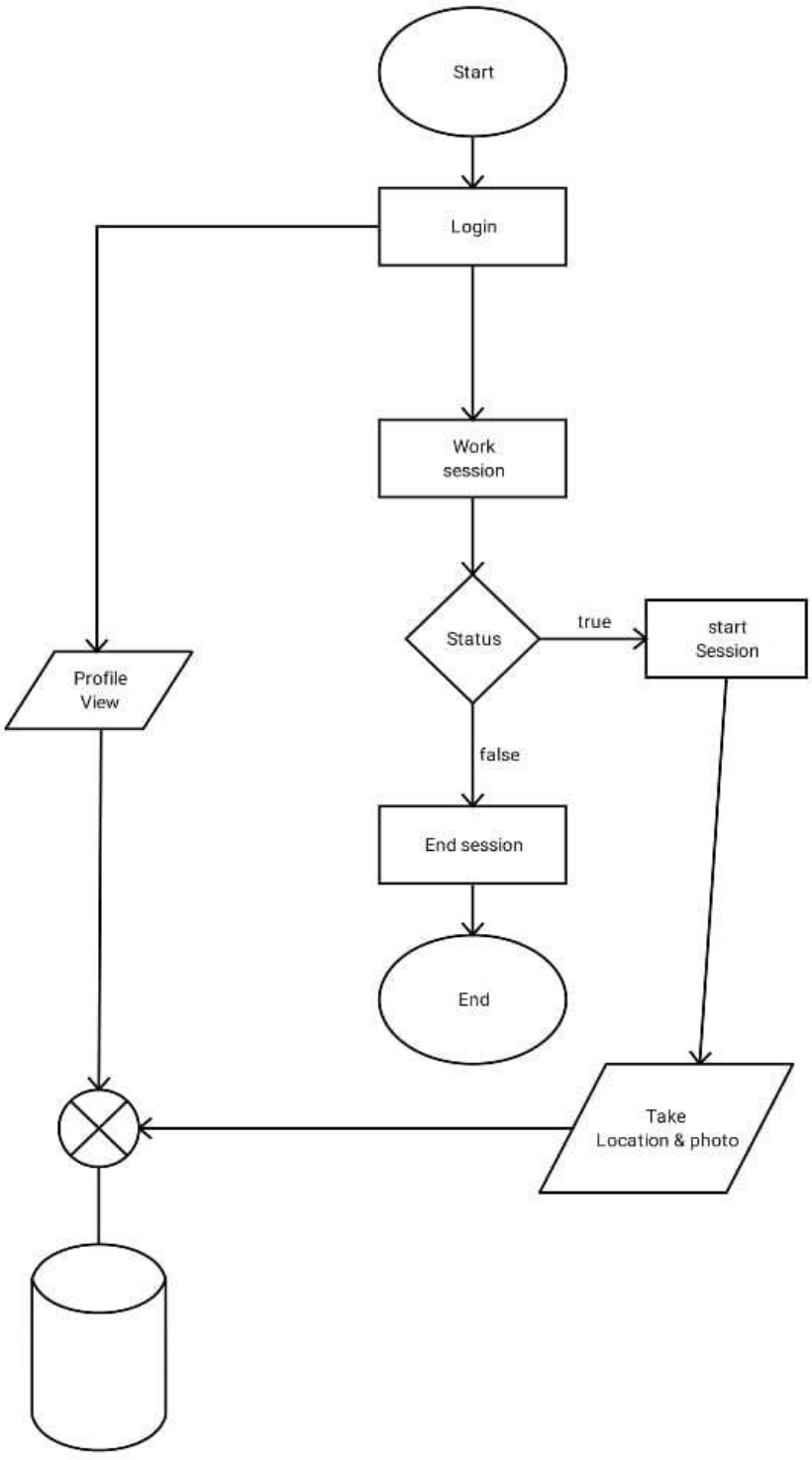


Figure 3: APP Flow chart

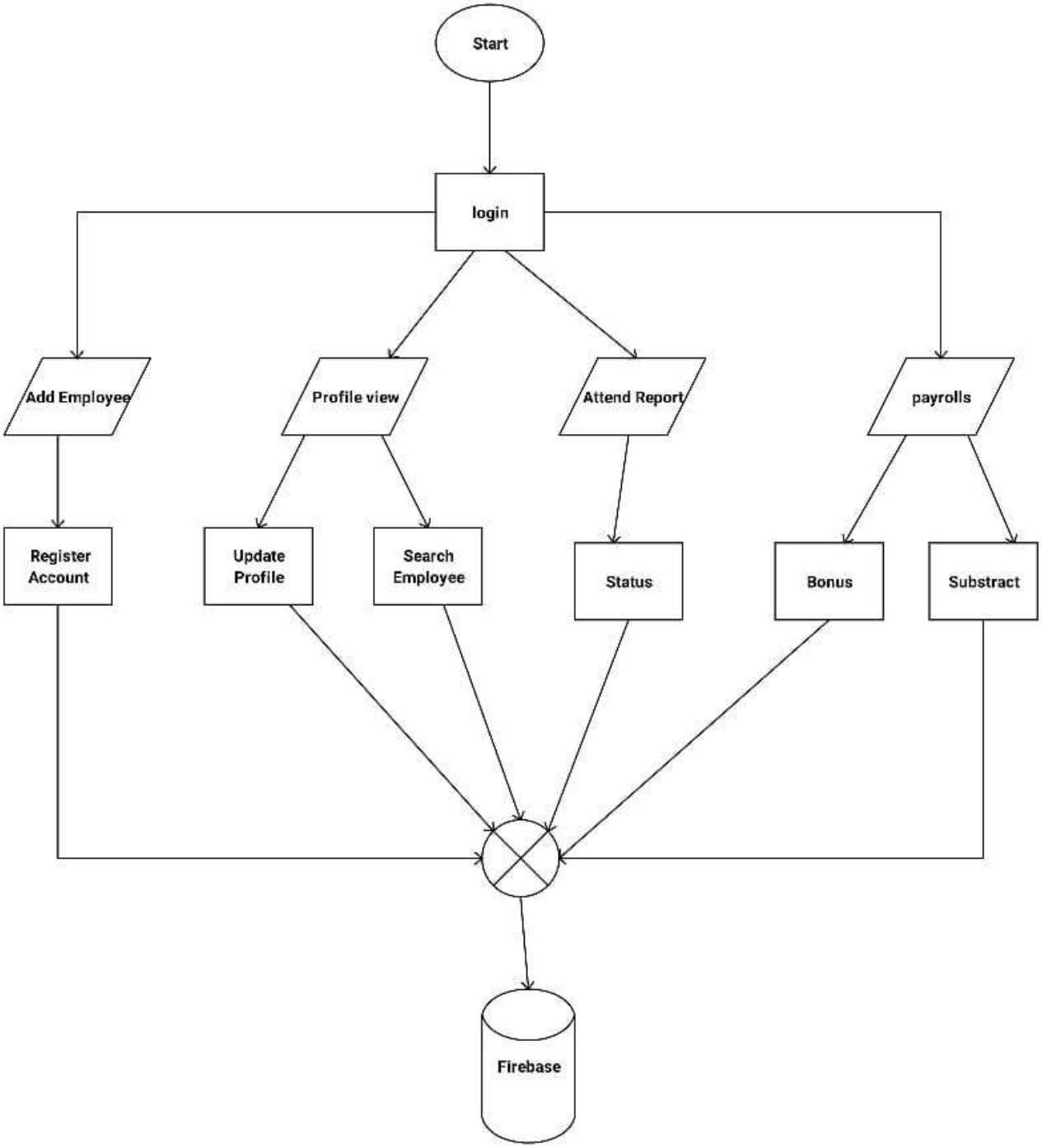


Figure 4: Web Page Flow chart

6.2 Use Case Diagram:

A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well.

The use cases are represented by either circles or ellipses. The actors are often shown as stick figures.

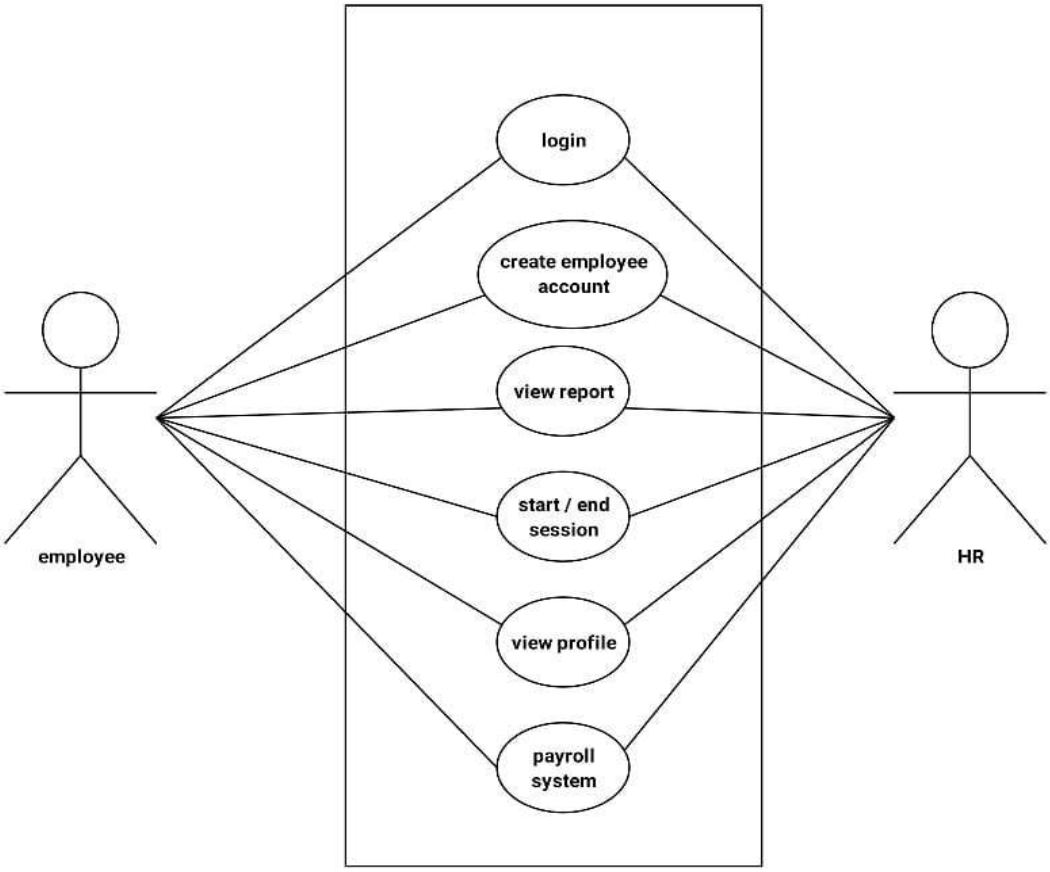


Figure 5: APP User Case

6.3 ERD Diagram:

For the ERD we worked with the Firebase store, Authentication, and storage to manage and store the data as it is known that the Firebase is nonrelation database so mange the collections and documents by primary key

6.4 Defining System Performance:

With regard to the development of our system, many improvements were made and many features were added, the purpose of which is to improve the accuracy of the system, reduce cost and speed of performance, allowing saving the time of the HR or the specialist, and other improvements and additions that help the specialist in the process of monitoring Payrolls and take care of its accurate and accessible. in the beginning, this system will help HR and specialists in taking care of their payrolls by handling the proses of sigh in and out to the work session and calculate the exact salary including the bonus and subtract and take care of managing employees' profiles.

6.5 Implementation Issues and Challenges:

Difficulties and challenges in implementing a project or idea always occur, especially if the work team is a beginner in the field. This is the case of our project. Over the course of implementing the project steps, we faced several challenges and difficulties, such as:

A - Skills of developers (work team):

As we mentioned earlier, we are still beginners in the field, and this is one of the biggest challenges for us. Whenever we intend to implement a part of the project, it is necessary to search and inquire about this part in terms of the software part represented by courses and Installing parts of the device.

B - Time:

We are all still students and studying, and we were also required at the same period to work on developing this system. This is also one of the most important challenges and difficulties of organizing time.

C - Reaching the specialists (HR):

there was some difficulty in reaching the HR to listen to their opinions on the idea of our project and to know their requirements and needs

Chapter (7)

Designs & Component

7.1 Participating Technology (Software):

7.1.1 Visual Studio Code:

Visual Studio Code is a script editor from Microsoft. The editor is open source and works on Windows, Mac and Linux operating systems and was launched by Microsoft in April 2015. Through it, we can write Dart language, which is the basic Flutter language

7.1.2 Android Studio:

Android Studio is a platform for writing applications that makes it easier for developers to write the source code for Android applications, and it also allows the developer to preview the appearance of his application on various screen sizes immediately during development, and it facilitates the development of applications and through which he can write with JAVA.

7.1.3 Firebase Dashboard:

Firebase provides detailed documentation and cross-platform SDKs to help you build and ship apps on Android, iOS, the web

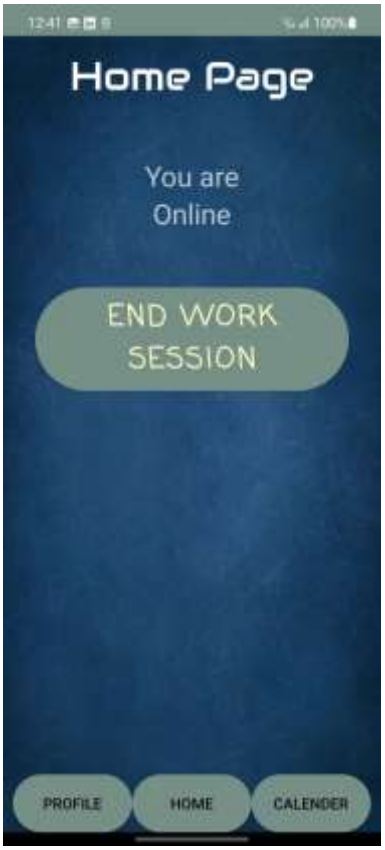
7.2 Mobile App

Method: The employee arrives at the company and takes a picture of himself, and then the app takes his location on the map using GPS.

7.2.1 The contents of the Mobile App:

1- Login Activity: Through it the user could Login or Reset it’s password

2- Home Page: Through it, the attendance session will start, take a picture of the employee, and transfer to the page of taking the employee's location on the map to make sure that he is actually inside the company. As well as recording the departure from the company at the end of the day.



Home page – online Statues

Home page – offline Statues

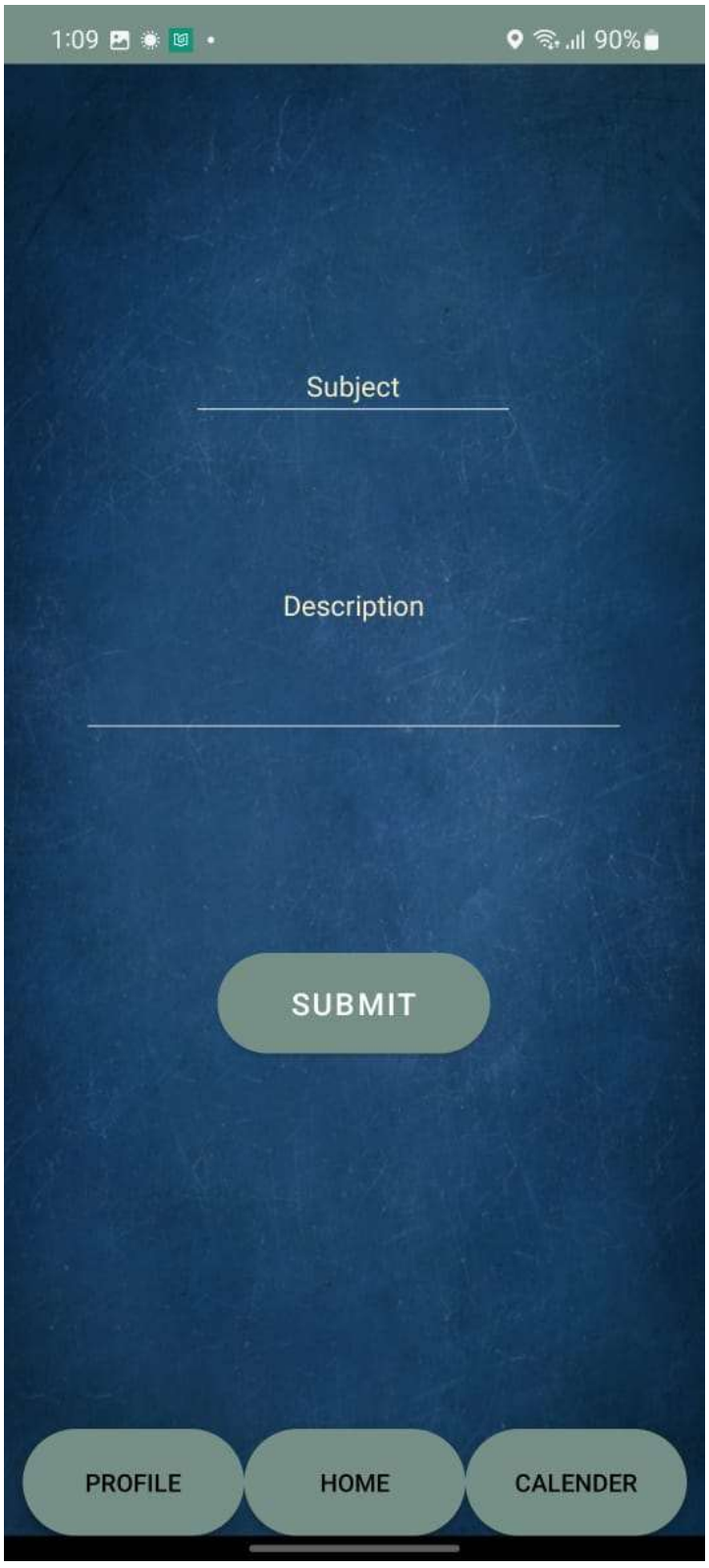
3- Profile: Through it, the employee begins to view his personal data and the fixed daily salary.

4- Salary Report:

5- Calendar:


PROFILE
HOME
CALENDER

6- Add Request: through it the employee to submit any request he need to let the Admin know his request to complete the process



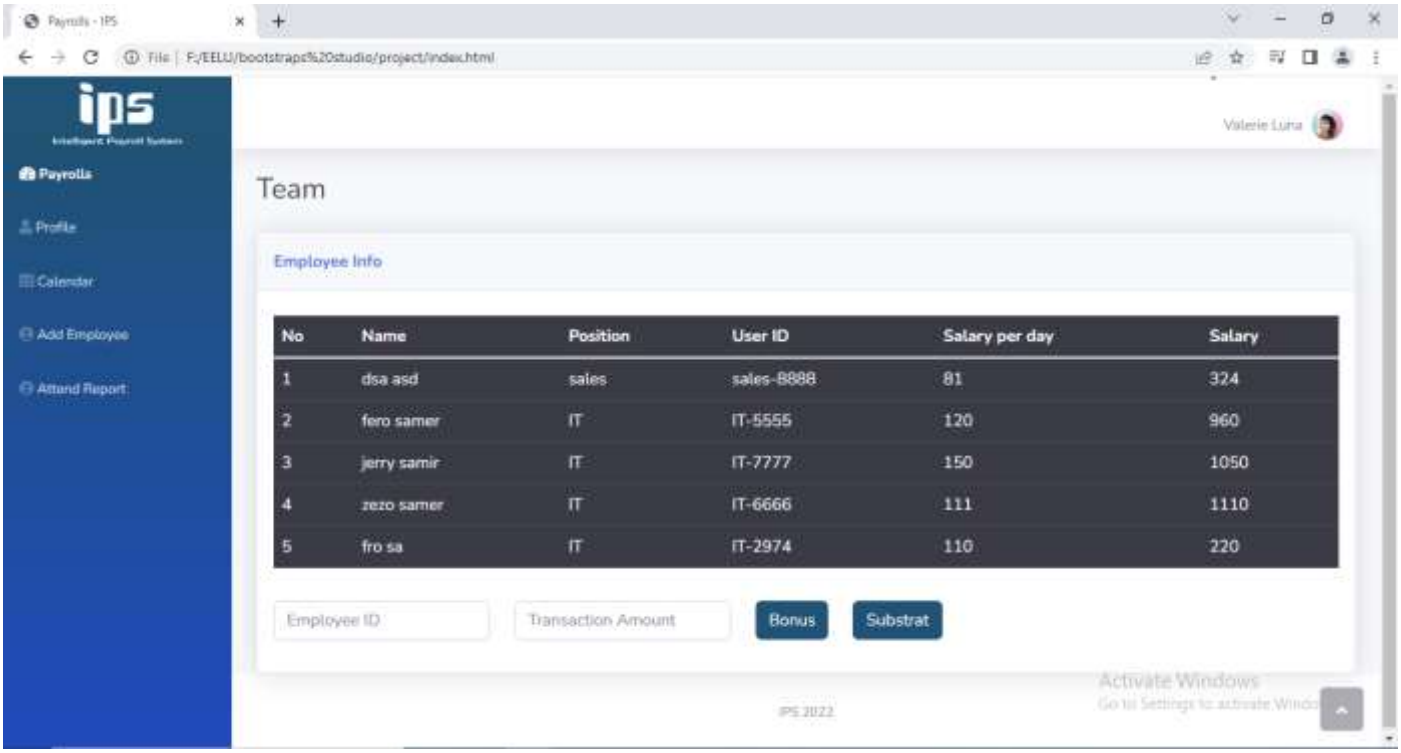
7.3 Dashboard Website

The web page is for the administrator.

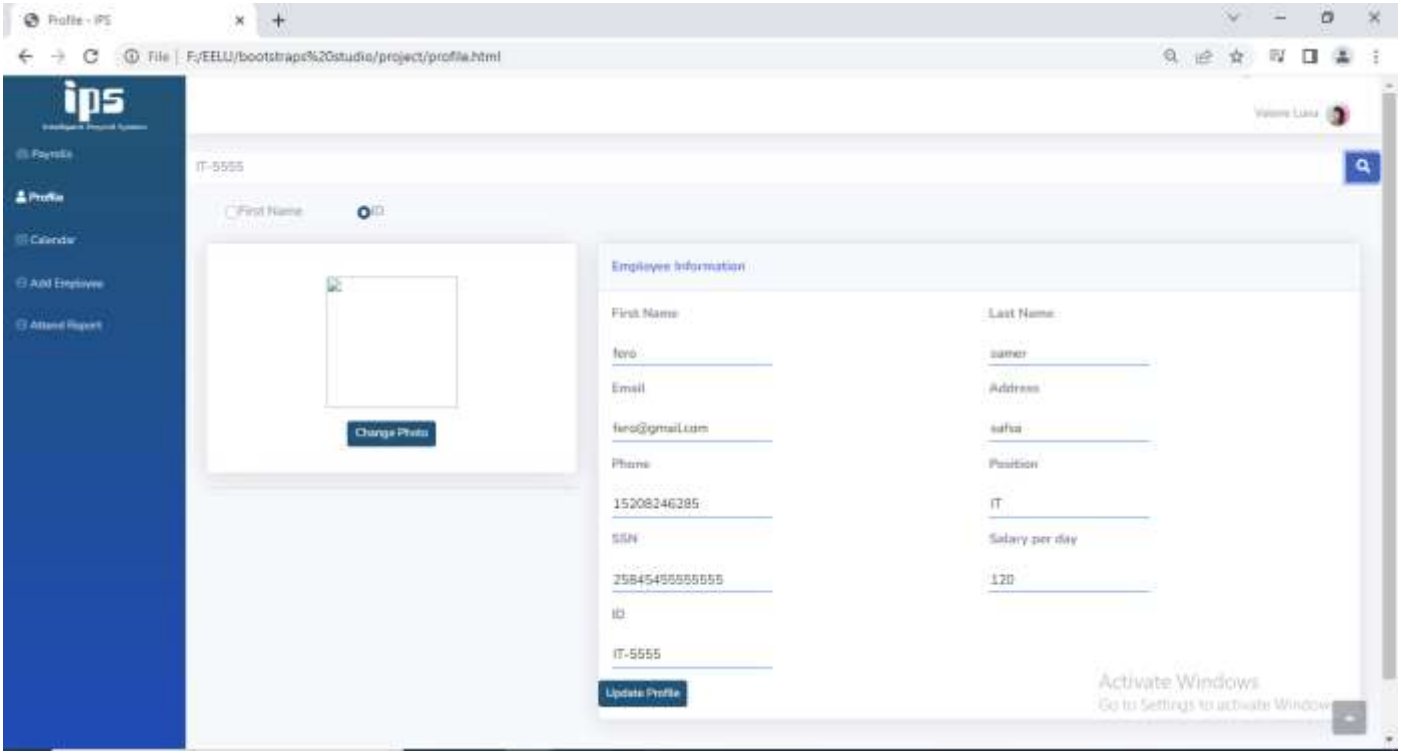
The administrator enters the page through the login page. The web page consists of five main pages

7.3.1 The contents of the Web Page:

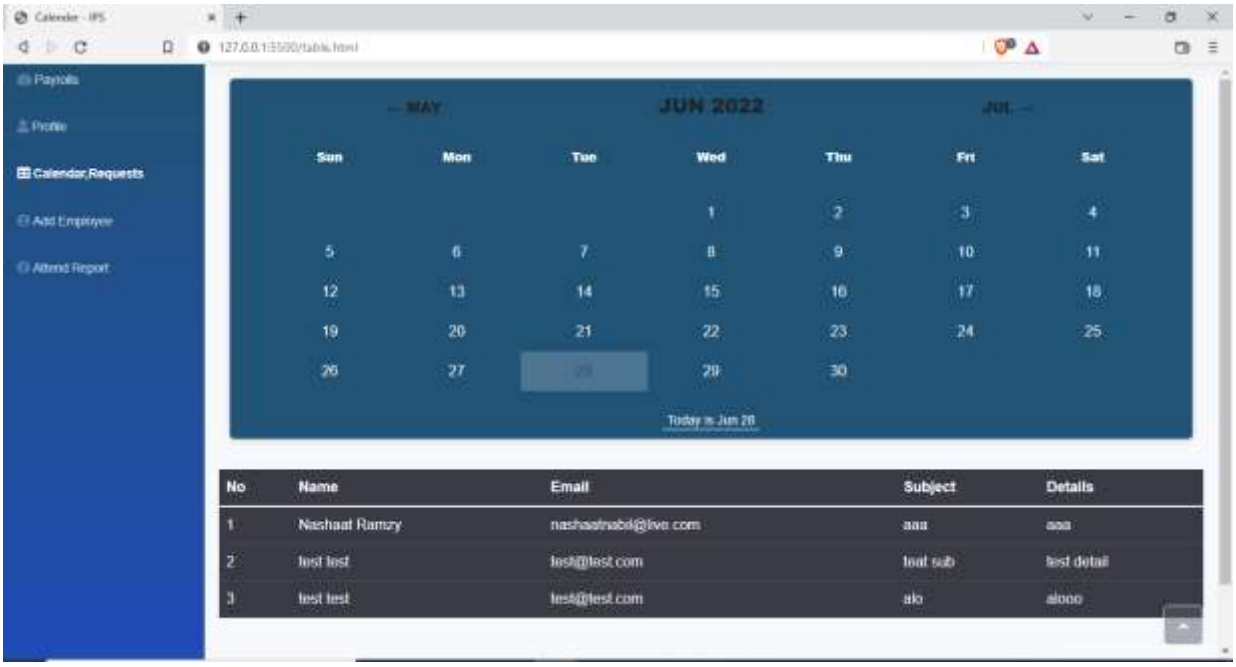
1- Payroll: Through it, it is possible to add a bonus or Deduction from salary of employees easily and quickly.



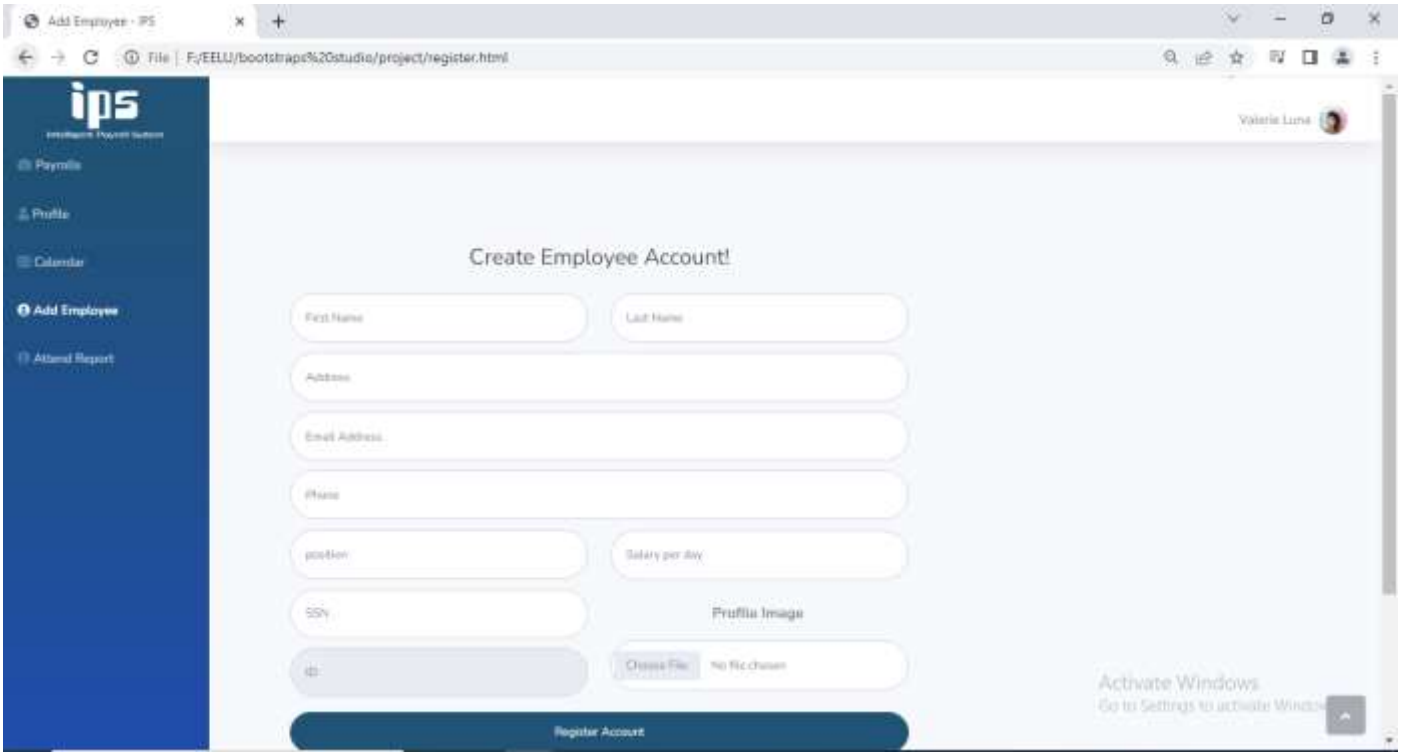
2- Employee Profile: Through it, it is estimated that the employee's data is accessed through the first name or ID. Everything about the employee personal data or the value of his fixed daily salary, and it can be modified or deleted as an employee account in case the employee is fired from the company.



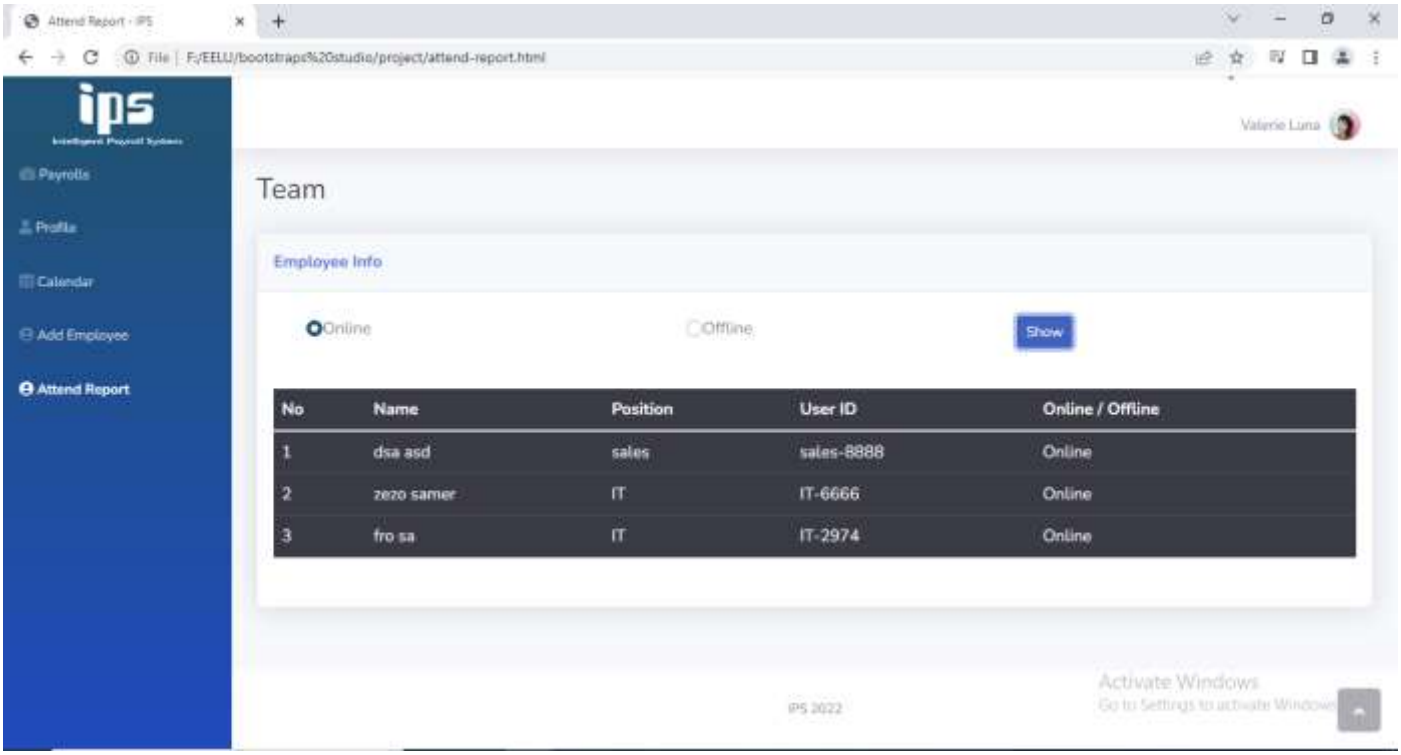
3- Calendar: Static calendar and in future will add Different features and through it the admin could see list of employees requests



4- Add Employee : Through it, he can add a new employee, and after adding the employee, he creates a user & pass for the employee's account, through which he can enter his account in the app.



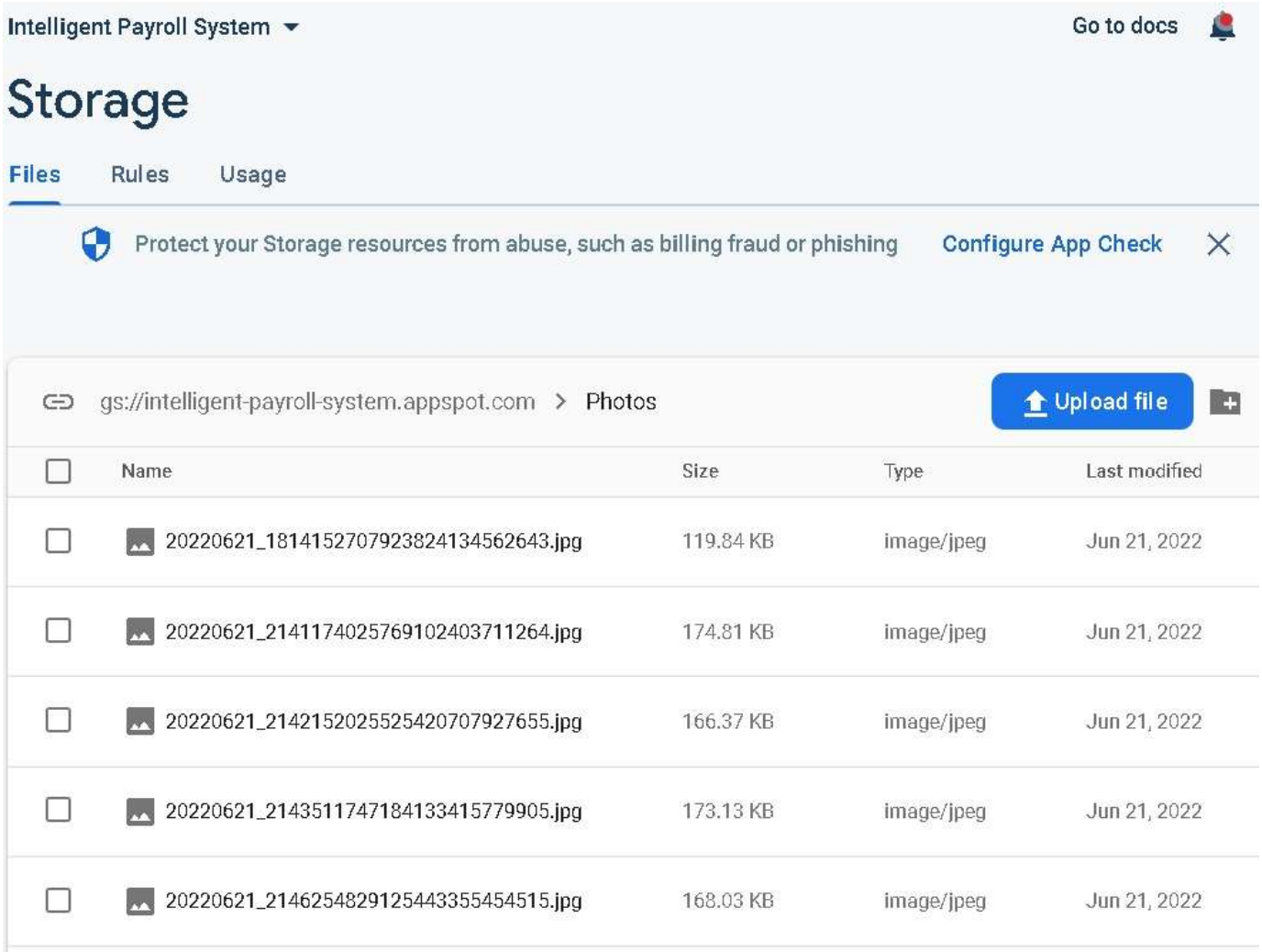
5- Attend Report : Through it, it is estimated to know all the employees present and absent, and the current salary is known by calculating the number of days of attendance and the employee's salary per day.

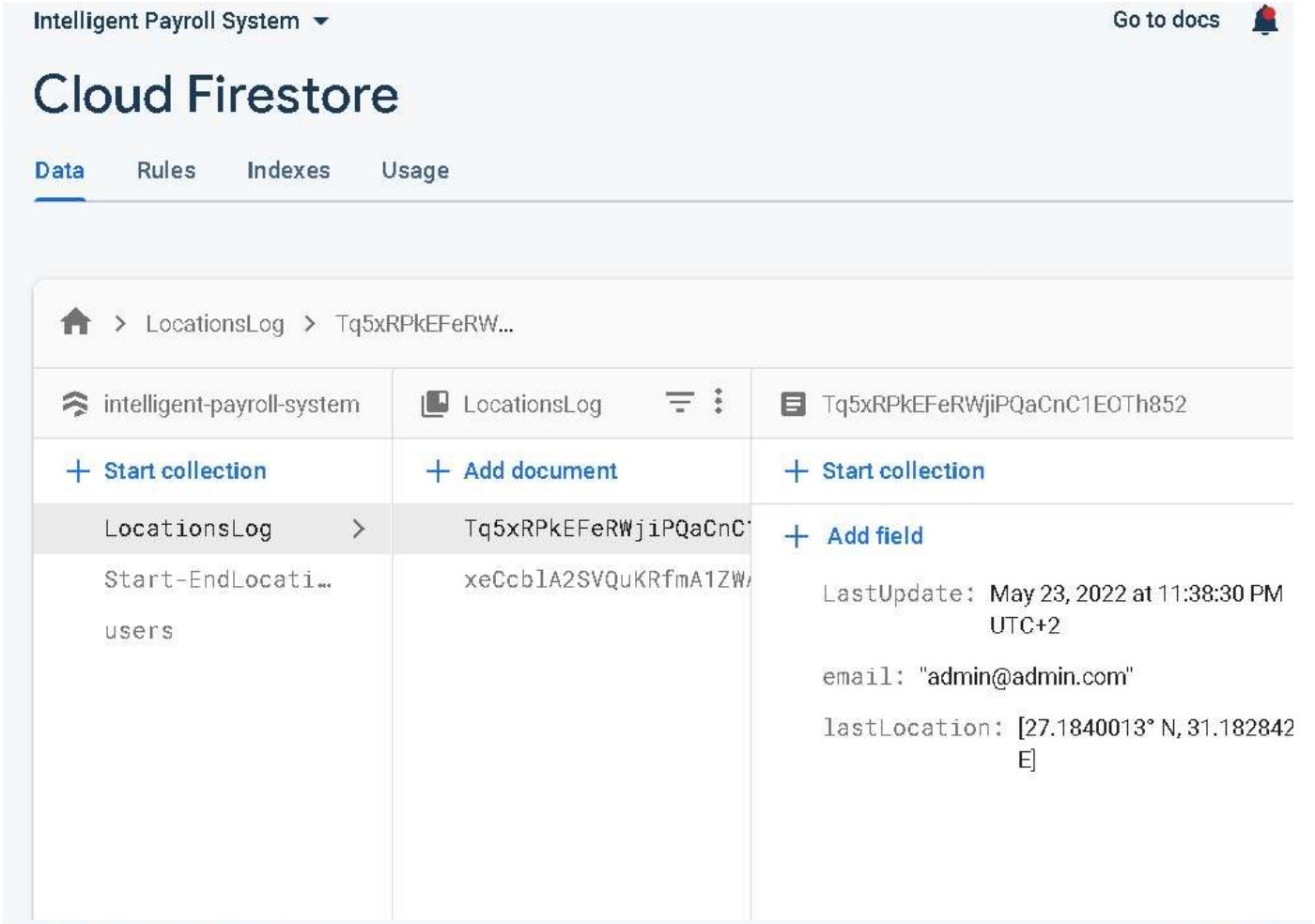


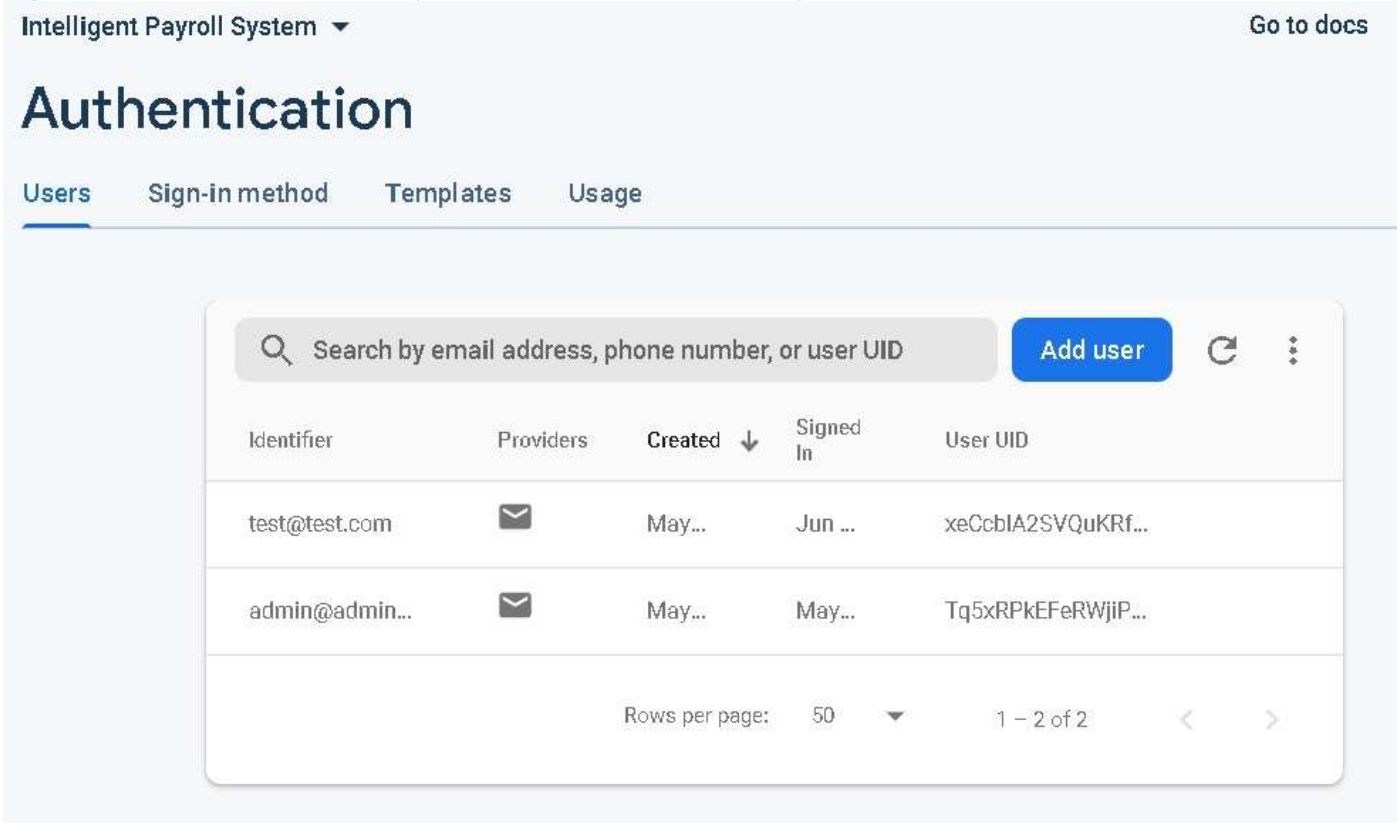
7.4 Fire base

The firebase takes care of many operations from the server side, and we chose to use it because the idea was based on solving the problem without resorting to costly solutions in storing and securing data and taking care of authentication operations

Below are different pictures of firebase Dashboard:


Intelligent Payroll System
Go to docs
Storage
Files
Rules
Usage
Protect your Storage resources from abuse, such as billing fraud or phishing
Configure App Check
gs://intelligent-payroll-system.appspot.com > Photos
Upload file
Name
Size
Type
Last modified
20220621_181415270792382413456264 3.jpg
119.84 KB
image/jpeg
Jun 21, 2022
20220621_214117402576910240371126 4.jpg
174.81 KB
image/jpeg
Jun 21, 2022
20220621_214215202552542070792765 5.jpg
166.37 KB
image/jpeg
Jun 21, 2022
20220621_214351174718413341577990 5.jpg
173.13 KB
image/jpeg
Jun 21, 2022
20220621_214625482912544335545451 5.jpg
168.03 KB
image/jpeg
Jun 21, 2022


Intelligent Payroll System
Go to docs
Cloud Firestore
Data
Rules
Indexes
Usage
LocationsLog > Tq5xRPkEFeRW...
intelligent-payroll-system
LocationsLog
Tq5xRPkEFeRWjiPQaCnC1EOTh852
Start collection
Add document
Start collection
LocationsLog
Start-EndLocati...
users
Add field
LastUpdate: May 23, 2022 at 11:38:30 PM UTC+2
email: "admin@admin.com"
lastLocation: [27.1840013° N, 31.182842 E]


Intelligent Payroll System
Go to docs
Authentication
Users
Sign-in method
Templates
Usage
Search by email address, phone number, or user UID
Add user
Identifier
Providers
Created
Signed In
User UID
test@test.com
May...
Jun ...
xeCcblA2SVQuKRf...
admin@admin...
May...
May...
Tq5xRPkEFeRWjiP...
Rows per page: 50
1 – 2 of 2

Chapter (8)

System Implementation, Coding And Testing

8.1 Mobile App Functions description

8.1.1 Check if location is connected

```
public void CheckLocationEnabling() {

    LocationManager lm = (LocationManager)
            getSystemService(Context.LOCATION_SERVICE);
    boolean gps_enabled = false;
    try {
        gps_enabled = lm.isProviderEnabled(LocationManager.GPS_PROVIDER);
    } catch (Exception e) {
        e.printStackTrace();
    }

    if (!gps_enabled) {
        new AlertDialog.Builder( context: MainActivity.this)
                .setMessage("GPS Enable")
                .setCancelable(false)
                .setPositiveButton( text: "Settings",
                        new DialogInterface.OnClickListener() {
                            @Override
```

8.1.2 Check if Internet is connected

```
private boolean CheckInternet() {
    ConnectivityManager connectivityManager = (ConnectivityManager) getSystem
    NetworkInfo wifiInfo = connectivityManager.getNetworkInfo(ConnectivityMana
    NetworkInfo mobileInfo = connectivityManager.getNetworkInfo(ConnectivityMa

    if ((wifiInfo != null && wifiInfo.isConnected()) || (mobileInfo != null &&
        return true;
    } else {
        showDialog();
        return false;
    }
}
```

8.1.3 Taking user photo and compress it to get the best performance

```
private void dispatchTakePictureIntent() {
    Intent takePictureIntent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
    // Ensure that there's a camera activity to handle the intent
    if (takePictureIntent.resolveActivity(getPackageManager()) != null) {
        // Create the File where the photo should go

        try {
            photoFile = createImageFile();
        } catch (IOException e) {
            Log.d( tag: "my log", msg: "Excep2 : " + e);
        }

        // Continue only if the File was successfully created
        if (photoFile != null) {
            pathToFile = photoFile.getAbsolutePath();
            Uri photoURI = FileProvider.getUriForFile( context: HomepageActivity.
                    authority: "com.example.aps.fileprovider",
                    photoFile);
            takePictureIntent.putExtra(MediaStore.EXTRA_OUTPUT, photoURI);
```

Taking the photo

```
try {
    mProgress.setTitle("Uploading");
    mProgress.show();

    Uri uri = FileProvider.getUriForFile( context: HomepageActivity.this,
            authority: "com.example.aps.fileprovider",
            photoFile);
    Bitmap bmp = null;
    try {
        bmp = MediaStore.Images.Media.getBitmap(getContentResolver(), uri);
    } catch (IOException e) {
        e.printStackTrace();
    }
    ByteArrayOutputStream baos = new ByteArrayOutputStream();

    //here you can choose quality factor in third parameter(ex. i choosen 25)
    bmp.compress(Bitmap.CompressFormat.JPEG, quality: 25, baos);
    byte[] fileInBytes = baos.toByteArray();
```

Compress the photo

8.1.4 Uploading user photo

```
StorageReference filepath = mStorage.child("Photos").child(uri.getLastPathSe
filepath.putBytes(fileInBytes).addOnSuccessListener(new OnSuccessListener<Up
    @Override
    public void onSuccess(UploadTask.TaskSnapshot taskSnapshot) {
        Toast.makeText( context: HomepageActivity.this, text: "Upload Successful!
        mProgress.dismiss();
        Intent map = new Intent( packageContext: HomepageActivity.this,
                com.example.aps.MapsActivity.class);
        map.putExtra( name: "email", email);
        map.putExtra( name: "UID", UID);
        map.putExtra( name: "Statues",isActive);
        map.putExtra( name: "docID", docID);
        startActivity(map);
    }
```

8.1.5 Getting user Location and store it

```
private Location getLastKnownLocation() {

    mLocationManager = (LocationManager) getApplicationContext().getSystemSer
    List<String> providers = mLocationManager.getProviders( enabledOnly: true);
    Location bestLocation = null;
    if (ContextCompat.checkSelfPermission(getApplicationContext(), Manifest.p
            PackageManager.PERMISSION_GRANTED || ContextCompat.checkSelfPermi
            PackageManager.PERMISSION_GRANTED) {
        ActivityCompat.requestPermissions( activity: MapsActivity.this, new Strin
    } else {
        for (String provider : providers) {
            l = mLocationManager.getLastKnownLocation(provider);

            if (l == null) {
                continue;
            }
            if (bestLocation == null || l.getAccuracy() < bestLocation.getAcc
                // Found best last known location: %s", l);
                bestLocation = l;
```

8.1.6 Submitting Request:

```
RequestData.put("email", email);
RequestData.put("subject", etSubject.getText().toString());
RequestData.put("detail", etDetail.getText().toString());
RequestData.put("name", name);

CollectionReference StartRef = db.collection( collectionPath: "EmpR
Task StartQuery = StartRef.add(RequestData);
StartQuery.addOnCompleteListener(new OnCompleteListener<Void>(
    @Override
    public void onComplete(@NonNull Task<Void> task) {
        if (task.isSuccessful()) {
            Toast.makeText( context: Attendace.this, text: "Success
            Intent profile = new Intent( packageContext: Attendace
                    com.example.aps.myprofile.class);
            profile.putExtra( name: "email", email);
            profile.putExtra( name: "UID", UID);
            startActivity(profile);
            finish();
```

8.2 Dashboard Website Functions description

8.2.1 Attend Report JS code

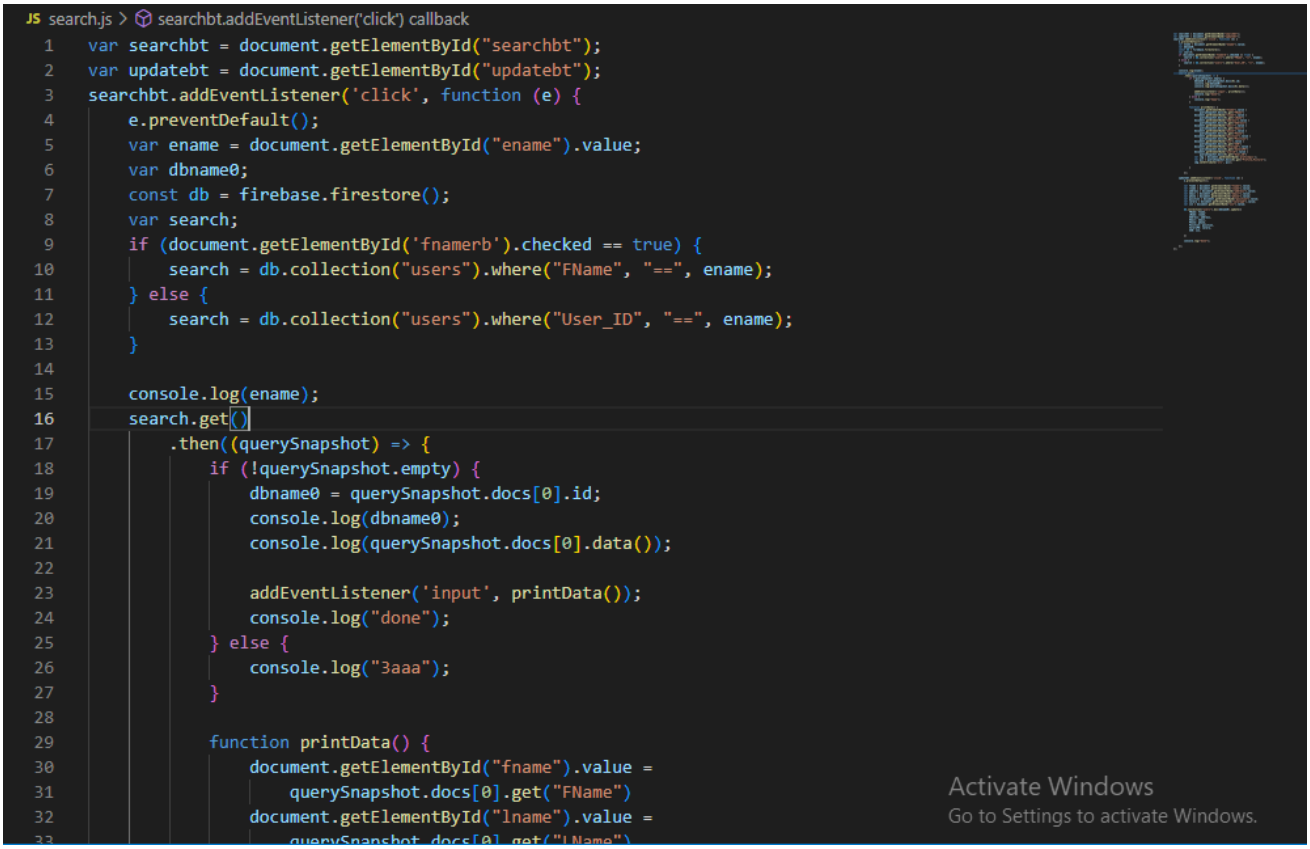
```
            tbody.innerHTML = "";
            users1.forEach(element => {
                AddItemToTable(element.FName + " " + element.LName, element.Position, element.User_ID, element.
            });
        }
        async function GetAllDataOnce() {
            if (document.getElementById('onb').checked == true) {
                const q = query(collection(db, "users"), where("Active", "==", "Online"));
                const querySnapshot = await getDocs(q);
                var users1 = [];
                querySnapshot.forEach(doc => {
                    users1.push(doc.data());
                });

            } else if (document.getElementById('offb').checked == true) {
                const q = query(collection(db, "users"), where("Active", "==", "Offline"));
                const querySnapshot = await getDocs(q);
                var users1 = [];
                querySnapshot.forEach(doc => {
                    users1.push(doc.data());
                });

            }
            AddAllItemsToTable(users1);
        }

        window.addEventListener('DOMContentLoaded', function () {
            document.getElementById('activebt').addEventListener("click", () => {
                window.onLoad = GetAllDataOnce();
                console.log('clicked');
            });
        });
```


8.2.2 Profile JS Code



search.js > searchbt.addEventListener('click') callback

```
var searchbt = document.getElementById("searchbt");
var updatebt = document.getElementById("updatebt");
searchbt.addEventListener('click', function (e) {
    e.preventDefault();
    var ename = document.getElementById("ename").value;
    var dbname0;
    const db = firebase.firestore();
    var search;
    if (document.getElementById('fnamerb').checked == true) {
        search = db.collection("users").where("FName", "==", ename);
    } else {
        search = db.collection("users").where("User_ID", "==", ename);
    }

    console.log(ename);
    search.get()
        .then((querySnapshot) => {
            if (!querySnapshot.empty) {
                dbname0 = querySnapshot.docs[0].id;
                console.log(dbname0);
                console.log(querySnapshot.docs[0].data());

                addEventListener('input', printData());
                console.log("done");
            } else {
                console.log("3aaa");
            }

            function printData() {
                document.getElementById("fname").value =
                    querySnapshot.docs[0].get("FName")
                document.getElementById("lname").value =
```

8.2.3 Adding Employee JS Code

```
var registerbt = document.getElementById("registerbt");

registerbt.addEventListener('click', function (e) {
    e.preventDefault();

    var fname = document.getElementById("fname").value;
    var lname = document.getElementById("lname").value;
    var address = document.getElementById("address").value;
    var email = document.getElementById("email").value;
    var phone = document.getElementById("phone").value;
    var position = document.getElementById("position").value;
    var salary = document.getElementById("salary").value;
    var ssn = document.getElementById("ssn").value;
    var img = document.getElementById("img").value;
    var password = lname + ssn.substr(ssn.length - 4);
    var userid = position + "-" + ssn.substr(ssn.length - 4);

    firebase.auth().createUserWithEmailAndPassword(email, password)
        .then((userCredential) => {

            const user = userCredential.user;

        })

    const booksRef = firebase.firestore().collection('users').add({
        FName: fname,
        LName: lname,
        Address: address,
        Email: email,
        Phone: phone,
        Position: position,
        SalaryPD: salary,
```


8.2.4 Payroll JS Code

```
        function substract(employeeid, amount) {
            var currentdate = new Date();
            var datetime = "Last Sync: " + currentdate.getDate() + "/"
                + (currentdate.getMonth() + 1) + "/"
                + currentdate.getFullYear();
            db1.collection('salary').add({
                user_id: employeeid,
                type: "substract",
                amount: amount,
                date: datetime
            });
            window.onload = GetAllDataOnce;
            return -amount;
        }
        async function getdata(employeeid, salary) {
            const querySnapshot1 = await getDocs(collection(db, "salary"));
            var salary1 = [];
            querySnapshot1.forEach(doc => {
                salary1.push(doc.data());
            });
            return Promise.resolve(sum(salary1, employeeid, salary));
        }
        function sum(salary1, employeeid, salary) {
            var s = salary;
            salary1.forEach(element => {
                if (element.user_id == employeeid) {
                    if (element.type == "bonus") {
                        s = s + element.amount;
                    } else if (element.type == "substract") {
                        s = s - element.amount;
                    }
                } else {
```

8.2.5 Calendar and Request JS Code

table.html > { } "table.html" > html > body#page-top > script > GetAllDataOnce

```
            td1.innerHTML = ++stdNo;
            td2.innerHTML = name;
            td3.innerHTML = email;
            td4.innerHTML = userid;
            td5.innerHTML = active;

            trow.appendChild(td1);
            trow.appendChild(td2);
            trow.appendChild(td3);
            trow.appendChild(td4);
            trow.appendChild(td5);

            tbody.appendChild(trow);
        }
        function AddAllItemsToTable(users1) {
            stdNo = 0;
            tbody.innerHTML = "";
            users1.forEach(element => {
                AddItemToTable(element.name, element.email, element.subject, element.detail);
            });
        }
        async function GetAllDataOnce() {
            const querySnapshot = await getDocs(collection(db, "EmpRequest"));
            var users1 = [];
            querySnapshot.forEach(doc => {
                users1.push(doc.data());
            });
            AddAllItemsToTable(users1);
        }
        window.onLoad = GetAllDataOnce();

    </script>
```

Summary

Our project facilitates the process of registering attendance and departure in an easy, simple and secure manner without Forgery, so that each employee get paid in a fair way by taking a personal picture of him from his mobile phone and taking his geographical location via GPS.

In addition to the ease of calculating the current salary and its details of any increases or decreases.

Future Work:

Our dream for the future:

1- Enable different companies to use the system separately by subscription

2- A device in the company to take attendance in case the registration from the employee's phone is not possible for any reason

3- Transferring salaries from the platform through the payment Gateway

4- The possibility of extracting reports in different formats such as pdf

6- A way to verify the employee's face image using artificial intelligence

7- Enable IOS Devices version