**INTERNATIONAL UNIVERSITY – HO CHI MINH NATIONAL UNIVERSITY**

**SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**

**PROJECT FINAL REPORT**

**Course: Software Engineering – IT076IU**

**Lecturer: Mrs. Nguyen Thi Thuy Loan**

**ADVANCED EMPLOYEE MANAGEMENT SYSTEM**

--------------------------------------------------------

|  |  |  |
| --- | --- | --- |
| **Group 10** | | |
| **Name** | **ID** | **Contribution** |
| Nguyễn Hồng Quân | ITITIU18297 | 21% |
| Trần Nhất Đạt | ITITIU18295 | 13% |
| Đỗ Bình Duy | ITITWE18033 | 21% |
| Hoàng Tuấn Kiệt | ITITIU19065 | 15% |
| Trần Thành Công | ITITIU19089 | 15% |
| Hoàng Trung Phong | ITITIU19179 | 15% |

December 31st, 2021

**Table of Contents**

[CHAPTER 1. INTRODUCTION 1](#_Toc91866675)

[1.1. Motivation 1](#_Toc91866676)

[1.2. Problem Statement 1](#_Toc91866677)

[1.3. Scope 2](#_Toc91866678)

[CHAPTER 2. LITERATURE REVIEW 4](#_Toc91866679)

[2.1. Similar Applications / Systems 4](#_Toc91866680)

[2.2. Platform and Tools Review 5](#_Toc91866681)

[CHAPTER 3. SYSTEM DESIGN 7](#_Toc91866682)

[3.1. System Requirement Specification 7](#_Toc91866683)

[3.1.1. Functional Requirements 7](#_Toc91866684)

[3.1.2. Non-functional Requirements 13](#_Toc91866685)

[3.2. System Design Specification 14](#_Toc91866686)

[3.2.1. Use – Case Diagram 14](#_Toc91866687)

[3.2.2. Sequence Diagrams 15](#_Toc91866688)

[3.2.3. Entity-Relationship Diagram 20](#_Toc91866689)

[3.2.4. Database Relational Schema 21](#_Toc91866690)

[CHAPTER 4. SYSTEM IMPLEMENTATION 22](#_Toc91866691)

[4.1. Back-end 22](#_Toc91866692)

[4.2. Front-end 24](#_Toc91866693)

[CHAPTER 5. CONCLUSION 37](#_Toc91866694)

[5.1. Accomplished Works 37](#_Toc91866695)

[5.2. Strengths and Weaknesses 37](#_Toc91866696)

[5.3. Future Work 38](#_Toc91866697)

[REFERENCES 39](#_Toc91866698)

**Table of Figures**

[Figure 1. Advanced Employee Management System's Use-Case Diagram 14](#_Toc91866707)

[Figure 2. Sequence Diagram 1 – A User Logging into An Account 15](#_Toc91866708)

[Figure 3. Sequence Diagram 2 – A User Viewing Their Personal Records 16](#_Toc91866709)

[Figure 4. Sequence Diagram 3 – A Manager Viewing Their Employees' Profile 16](#_Toc91866710)

[Figure 5. Sequence Diagram 4 – A Manager Managing Their Employees' Profile 17](#_Toc91866711)

[Figure 6. Sequence Diagram 5 – A Manager Assigning Projects/Tasks to Their Employees 18](#_Toc91866712)

[Figure 7. Sequence Diagram 6 – An Admin Performing System Maintenance 19](#_Toc91866713)

[Figure 8. Advanced Employee Management System’s Entity-Relationship Diagram 20](#_Toc91866714)

[Figure 9. Advanced Employee Management System's Database Relational Schema 21](#_Toc91866715)

[Figure 10. Firebase configuration in /web/index.html 22](#_Toc91866716)

[Figure 11. Create, Update, and Delete Employee in /lib/service/employee\_service.dart 23](#_Toc91866717)

[Figure 12. Create Notification and Create New Task in /lib/service/notification\_service.dart 24](#_Toc91866718)

[Figure 13. Create Login page in /lib/authentication/sign\_in\_page.dart 25](#_Toc91866719)

[Figure 14. Login page on the browser 26](#_Toc91866720)

[Figure 15. Create recover password in /lib/authentication/recover\_pass\_page.dart 27](#_Toc91866721)

[Figure 16. Recover password on the browser 28](#_Toc91866722)

[Figure 17. List of Employee (Employee Table) on the browser 28](#_Toc91866723)

[Figure 18. Deleting an Employee from the Employee Table 29](#_Toc91866724)

[Figure 19. Viewing an employee's profile on the browser 29](#_Toc91866725)

[Figure 20. Modifying an employee's profile on the browser 30](#_Toc91866726)

[Figure 21. Adding a new Employee to the Employee Table 30](#_Toc91866727)

[Figure 22. Department Table on the browser 31](#_Toc91866728)

[Figure 23. Add a new Department to the Department Table 31](#_Toc91866729)

[Figure 24. Employee viewing their personal profile 32](#_Toc91866730)

[Figure 25. Employee viewing their current projects 32](#_Toc91866731)

[Figure 26. Employee viewing their current project and their tasks 33](#_Toc91866732)

[Figure 27. Employee viewing the notifications 33](#_Toc91866733)

[Figure 28. Manager viewing their department's employees 34](#_Toc91866734)

[Figure 29. Manager viewing all available projects (Project Table) 34](#_Toc91866735)

[Figure 30. Manager adding a new project to the Project Table 35](#_Toc91866736)

[Figure 31. Manager assigning tasks to their employees 35](#_Toc91866737)

[Figure 32. Manager adding a member to the project via their ID 36](#_Toc91866738)

[Figure 33. Manager modifying project's info 36](#_Toc91866739)

**List of Tables**

[Table 1. "User Login to an Account" Use Case 7](#_Toc91866740)

[Table 2. "User View Personal Records" Use Case 8](#_Toc91866741)

[Table 3. "Manager View Their Employees' Profile" Use Case 9](#_Toc91866742)

[Table 4. "Manager Assign Projects/Tasks to Their Employee" Use Case 9](#_Toc91866743)

[Table 5. "Manager Manages Their Employees' Profile" Use Case 11](#_Toc91866744)

[Table 6. "Admin Performs System Maintenance" Use Case 12](#_Toc91866745)

[Table 7. Project's Non-Functional Requirements 13](#_Toc91866746)

[Table 8. Project's Strengths and Weaknesses 37](#_Toc91866747)

# **INTRODUCTION:**

## **Motivation:**

The world is being nurtured and developed as each second goes by at a seemingly neck-breaking pace. And in this rapidly developing world, business opportunities never cease to appear. Many people, especially businesspeople, have great awareness of how valuable these opportunities are and hence, are eager to dive straight in and establish their own companies to either make a fortune for themselves or to contribute themselves to the development of this world.

For a company to function properly and, possibly, gain the most profit, employees are an utmost requirement. Of course, it is an indisputable fact that effectively managing a company’s employees is not an easy feat at all, harder still if that company has a somewhat “generous” number of employees. In reality, there exists a high number of companies that do not possess a suitable tool for managing their employees. As a result, they struggle to make progress, and their employees can hardly make ends meet.

Because of that, our team has come up with a **web-based** project for an Advanced Employee Management System. This system supports the management process between a manager of a department and that department’s employees. It will aid the manager in managing their employees’ profiles and records, as well as their respective assigned projects and tasks. Aiming to create the best user experience for the people working in a company, our team’s goal is to build a website where the flow of information is clear enough for proper internal management while still keeping the company’s data secured from external forces.

## **Problem Statement:**

Handling personnel information manually presents a variety of issues. This is visible in procedures like leave management, where an employee must fill out a document that may take weeks or months to be authorized. The usage of paperwork in some of these procedures may result in human mistakes, papers may wind up in the wrong hands, and not to mention that it is time demanding.

Several existing systems lack employee self-service, which means that employees cannot access and manage their personal information directly without going through their HR departments or supervisors. Another concern is that multinational firms maintain all employee information at the company's headquarters office, making it difficult to access employee data from various locations rapidly.

The aforementioned issues can be addressed by building and deploying a web-based human resource management system. This solution will store employee information in a database while maintaining complete privacy and authorized access. The project's goal is to put up an employee information system that includes the employee's status, educational background, and job experience to assist in monitoring the employee's performance and achievements via a password-protected system.

## **Scope:**

Though employee management encompasses many facets of human resource management, the project will be focused on the creation of the Advanced Employee Management System.

The survey-based initiative stays rigorous; it does not expand to other institutions, but it can be mimicked.

This project's scope will be limited to the following:

* Employees' personal profiles: Employees will have access to their profiles and will be able to update their information.
* Digital absence application: By allowing employees to apply for and track their leave status through the system, all paperwork associated with leave management is eliminated. The HR manager will also be able to accept or reject leave requests using the system.
* Project management includes assigning tasks and projects to personnel, forming a project team, and tracking progress.
* Report generation: The HR manager will be able to create timely reports to monitor personnel, which may then be utilized for performance evaluations. The reports will include information on an employee's educational history, training, projects completed, and technical abilities.
* The admin will add an employee, and an employee ID and a default password will be produced and emailed to the new employee's email address. The HR manager will then be able to enter personnel information into the database.

# **LITERATURE REVIEW:**

## **Similar Applications / Systems:**

There are multiple websites and applications built for the same purpose (managing employees) as our team’s website. Some of the popular ones are:

* Weekdone: <https://weekdone.com/>
  + Strengths:
* Daily work connecting to personal, team, department, and organizational long-term goals.
* A modern approach for employee engagement with 360-degree staff/manager feedback, complete with emoji.
* Connects status reporting with existing systems of record.
* Brings the OKR (Objectives and Key Results) discipline to an organization.
  + Weaknesses:
* User setup and management and team/department creation requires some deliberate planning.
* The deadlines for plans do not connect to the due dates for tasks in systems of record.
* Connect team: <https://connecteam.com/>
  + Strengths:
* Stores that want a strong employee communications and HR software.
* The flexibility to assign a geofence to restrict where employees clock in and out was critical for one of our clients.
* The customer service is unparalleled.
  + Weaknesses:
* No auto scheduling or forecasting.
* Live support via email only.
* Limited reports.
* Gusto: <https://gusto.com/>
  + Strengths:
* Simple, clean, and accessible layout for users.
* Multiple other services and features offered including early pay, benefits, vacation, and owners draw options.
* Fast direct deposit options even for a small business.
* Extremely friendly and prompt customer service.
  + Weaknesses:
* The fees are on the higher end.
* Actual business name does not show up on bank statements making it hard to verify small.
* Workday: <https://www.workday.com/>
  + Strengths:
* Allows user to give feedback to others.
* A great storage of payroll and paystub information.
* Allow companies to upload mandatory training modules.
  + Weaknesses:
* Some of its interface is not very intuitive.
* The layout is outdated compared to other internal websites.
* And many, many more…

## **Platform and Tools Review:**

* Programming languages used for developing the website:
  + Dart.
  + HTML.
  + JavaScript.
* Tools used for development:
  + Visual Studio Code.
  + Firebase.
* Framework:
  + Flutter Framework.

# **SYSTEM DESIGN:**

## **System Requirement Specification:**

### **Functional Requirements:**

* Use-case 1:

Table 1. "User Login to an Account" Use Case

|  |
| --- |
| **User Login to an Account** |
| **Brief Description:**  The User Login to an Account use-case describes how a user logs into their account. |
| **Step-by-step Description:**   1. A user (Employee or Manager or Admin) goes onto the website. 2. The user clicks on the “Login” button. 3. The website shows the “Login” page. 4. The user proceeds to input the required fields (Username and Password) and clicks on the “Login” button. 5. The website checks if any of the required fields are empty or not. 6. When the fields are filled in, the website checks the database to verify the user’s inputs. 7. When the inputs are correct, the website shows the user’s personal page. |
| **Possible Alternatives:**   1. If any of the user’s input is empty, the website sends an alert message back to them. 2. If the user inputs the wrong Username or Password, the website sends an alert message back to them. 3. If the user inputs the wrong Username or Password for the 3rd time, the website locks the user’s access to their page for 24 hours, as well as sending an alert message back to them. 4. If the user is an Employee, they will be logged into their employee account. 5. If the user is a manager, they will be logged into their manager account. 6. If the user is an Admin, they will be logged into their Admin account. |

* Use-case 2:

Table 2. "User View Personal Records" Use Case

|  |
| --- |
| **User View Personal Records** |
| **Brief Description:**  The User View Personal Records use-case describes how a user view their personal information and records. |
| **Step-by-step Description:**   1. A user (Employee or Manager) logs into their respective account. 2. The website shows their personal account page. 3. The user clicks on the “View Personal Profile” button. 4. The website contacts the database to retrieve the data. 5. The website shows the user’s personal profile page. |
| **Possible Alternatives:** None |

* Use-case 3:

Table 3. "Manager View Their Employees' Profile" Use Case

|  |
| --- |
| **Manager View Their Employees’ Profile** |
| **Brief Description:**  The Manager View Their Employees’ Profile use-case describes how a manager view their department’s Employees’ profiles. |
| **Step-by-step Description:**   1. A Manager logs into their account. 2. The website shows the Manager’s personal page. 3. The Manager clicks on the “View Employees’ Profiles” button. 4. The website accesses the database to retrieve the data. 5. The website shows the queried Employees’ profiles to the Manager. |
| **Possible Alternatives:** None |

* Use-case 4:

Table 4. "Manager Assign Projects/Tasks to Their Employee" Use Case

|  |
| --- |
| **Manager Assign Projects/Tasks to Their Employee** |
| **Brief Description:**  The Manager Assign Projects/Tasks to Their Employee use-case describes how a manager assigns specific projects and tasks to their Employees. |
| **Step-by-step Description:**   1. A Manager logs into their account. 2. The website shows the Manager’s personal page. 3. The Manager chooses to assign Projects or Tasks to an Employee and clicks on the “Assign” button. 4. The website saves the assigned Projects or Tasks to the Employee’s profile in the database. 5. The website sends a confirmation message back to the Manager (successful). |
| **Possible Alternatives:**   1. If the Manager chooses the “Assign Projects” feature:    1. The website shows the “Project Assignment” page to the Manager.    2. The Manager fills in the Employee’s data and the Project to be assigned to them, then clicks on the “Assign” button.    3. The website checks if there are any empty fields:       1. If yes, the website sends an alert message back to the Manager (failed).       2. If no, the website contacts the database and saves the data of the Project to the Employee’s profile. 2. If the Manager chooses the “Assign Tasks” feature:    1. The website shows the “Task Assignment” page to the Manager.    2. The Manager fills in the Employee’s data, the Project they are working on, and the Task to be assigned to them, then clicks on the “Assign” button.    3. The website checks if there are any empty fields:       1. If yes, the website sends an alert message back to the Manager (failed).       2. If no, the website contacts the database and saves the data of the Project to the Employee’s profile. |

* Use-case 5:

Table 5. "Manager Manages Their Employees' Profile" Use Case

|  |
| --- |
| **Manager Manages Their Employees’ Profile** |
| **Brief Description:**  The Manager Manages Their Employees’ Profile use-case describes the three different actions a manager can take on their employees’ profile. |
| **Step-by-step Description:**   1. A Manager logs into their account. 2. The website shows the Manager’s personal page. 3. The Manager clicks on the “Manage Employees’ Profiles” button. 4. The website shows the “Employees’ Profiles Management” page. 5. The Manager chooses a feature to use out of three (Add/Modify/Remove). 6. The website carries out the chosen function and sends a confirmation message upon the successful completion of the process. |
| **Possible Alternatives:**   1. The Manager chooses the Add feature:    1. The website shows the “New Employee’s Profile” page.    2. The Manager fills in the to-be-added Employee’s details and clicks on the “Add” button.    3. The website checks if there are any empty fields:       1. If yes, the website sends an alert message back to the Manager (failed).       2. If no, the website contacts the database and save the data of the new Employee to their newly created profile. 2. The Manager chooses the Modify feature:    1. The website shows the “Employee’s Profile Modification” page.    2. The Manager alters the Employee’s details and clicks on the “Modify” button.    3. The website checks if there are any empty fields:       1. If yes, the website sends an alert message back to the Manager (failed).       2. If no, the website contacts the database and saves the altered data of the Employee.    4. The Manager clicks on the “Confirm Modification” button. 3. The Manager chooses the Remove feature:    1. The website shows the “Employee Profile(s) Deletion” page.    2. The Manager chooses which Employees’ profiles to remove and clicks on the “Delete” button and a “Confirm Deletion” button (safeguard).    3. The website contacts the database and removes the chosen Employee(s)’ profiles from the system. |

* Use-case 6:

Table 6. "Admin Performs System Maintenance" Use Case

|  |
| --- |
| **Admin Performs System Maintenance** |
| **Brief Description:**  The Admin Performs System Maintenance use-case describes how an Admin performs maintenance on the system. |
| **Step-by-step Description:**   1. An Admin logs into their account. 2. The website shows the admin’s personal page. 3. The admin performs multiple checking on the website’s functions. 4. The admin performs multiple checking on the website’s database. 5. The website sends a message to other users (Employees and Managers) to let them know the system maintenance is in progress. 6. The website will make changes after the maintenance process is finished. |
| **Possible Alternatives:** None |

### **Non-functional Requirements:**

Table 7. Project's Non-Functional Requirements

|  |  |
| --- | --- |
| **Name** | **Description** |
| Performance Requirements | The website should run fast and smoothly even though many clients browsing the website at the same time. |
| Usability Requirements | The UI is easy to learn and navigate; buttons, headings, and help/error notifications are simple to understand |
| Compatibility Requirements | The website should be eye-catching and be functionally usable on many types of devices of varied sizes. |
| Security Requirements | Every transmission between the system’s database and  users must be encrypted |
| Attempt log into the website failed five times or more will lead users take a ban from 5 minutes to hours |
| Reliability | To make a system's reliability trial, keep track of the number of critical mistakes it has all over testing. |

## **System Design Specification:**

### **Use – Case Diagram:**

Diagram

Description automatically generated

Figure 1. Advanced Employee Management System's Use-Case Diagram

### **Sequence Diagrams:**

Diagram

Description automatically generated

Figure 2. Sequence Diagram 1 – A User Logging into An Account

A picture containing graphical user interface

Description automatically generated

Figure 3. Sequence Diagram 2 – A User Viewing Their Personal Records

Diagram

Description automatically generated with medium confidence

Figure 4. Sequence Diagram 3 – A Manager Viewing Their Employees' Profile

A picture containing graphical user interface

Description automatically generated

Figure 5. Sequence Diagram 4 – A Manager Managing Their Employees' Profile

Graphical user interface, application, Teams

Description automatically generated

Figure 6. Sequence Diagram 5 – A Manager Assigning Projects/Tasks to Their Employees

Diagram

Description automatically generated with medium confidence

Figure 7. Sequence Diagram 6 – An Admin Performing System Maintenance

### **Entity-Relationship Diagram:**

Diagram

Description automatically generated

Figure 8. Advanced Employee Management System’s Entity-Relationship Diagram

### **Database Relational Schema:**

Graphical user interface, website

Description automatically generated

Figure 9. Advanced Employee Management System's Database Relational Schema

# **SYSTEM IMPLEMENTATION:**

## **Back-end:**

* The /web/index.html file where we connect with the database.

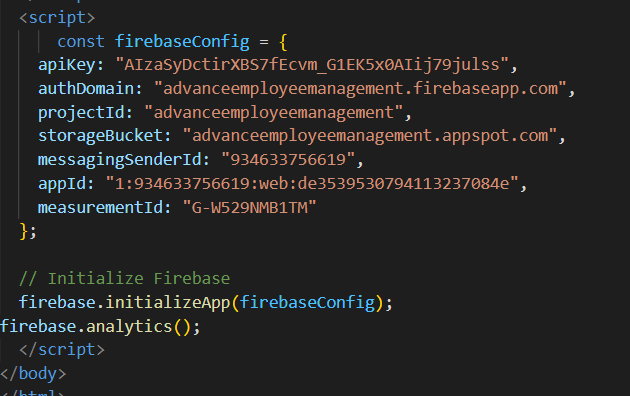


Figure 10. Firebase configuration in /web/index.html

* In /lib/service folder, we can perform queries for the data from database such as: create employee, delete employee, ...

*Example*: employee\_service.dart and notification\_service.dart classes.



Figure 11. Create, Update, and Delete Employee in /lib/service/employee\_service.dart

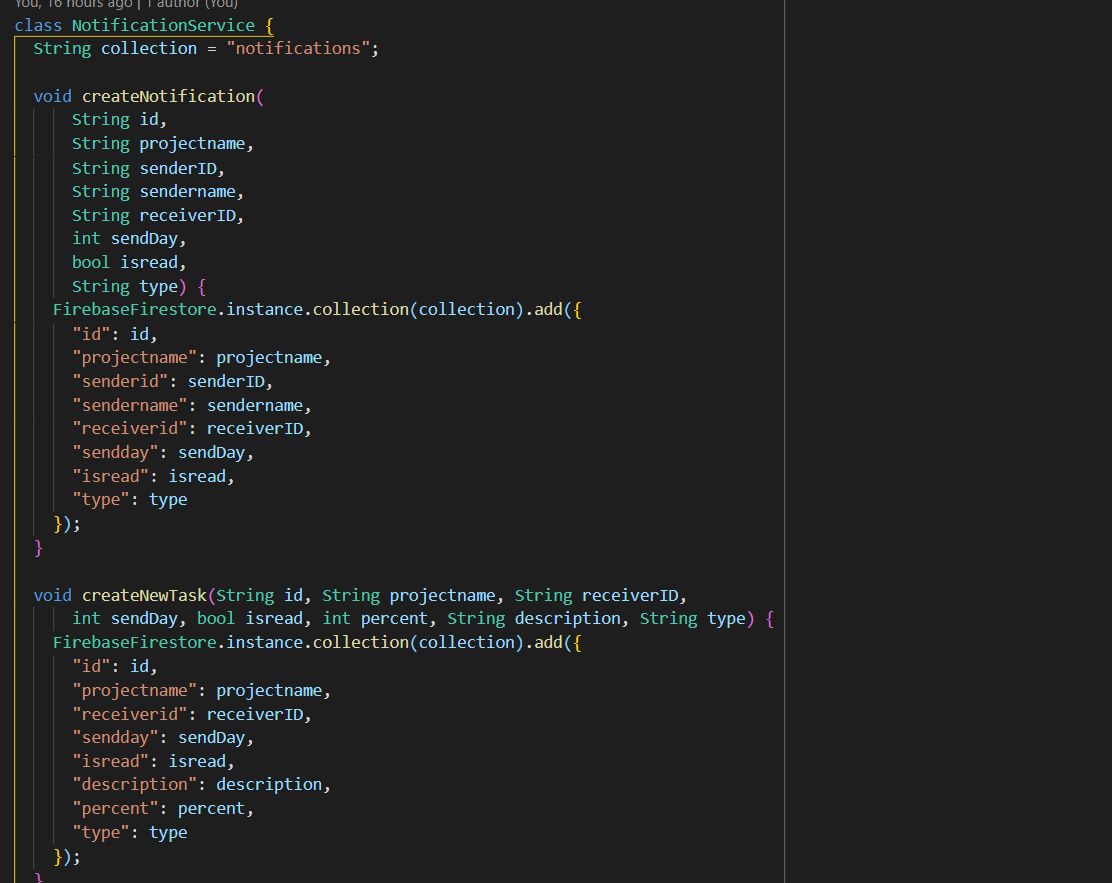


Figure 12. Create Notification and Create New Task in /lib/service/notification\_service.dart

## **Front-end:**

* Login:

Text

Description automatically generated

Figure 13. Create Login page in /lib/authentication/sign\_in\_page.dart

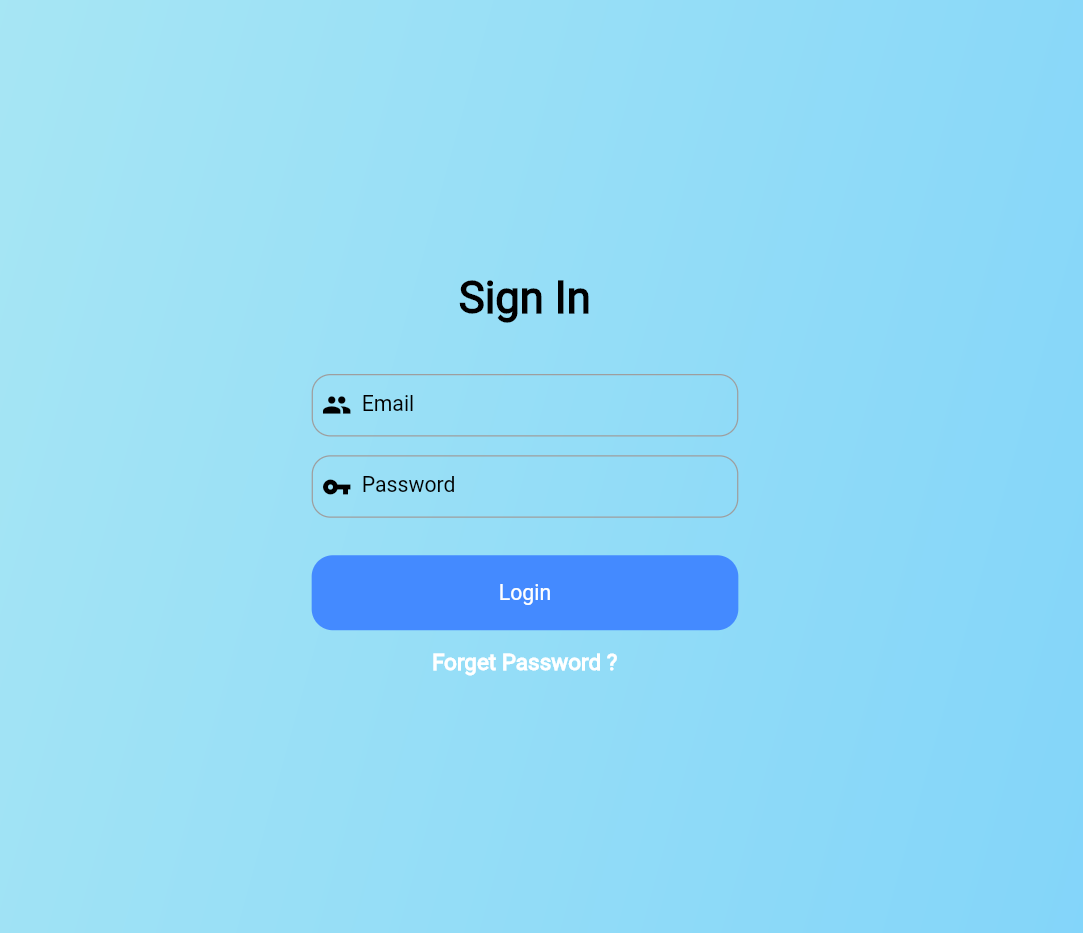


Figure 14. Login page on the browser

* Recover Password:

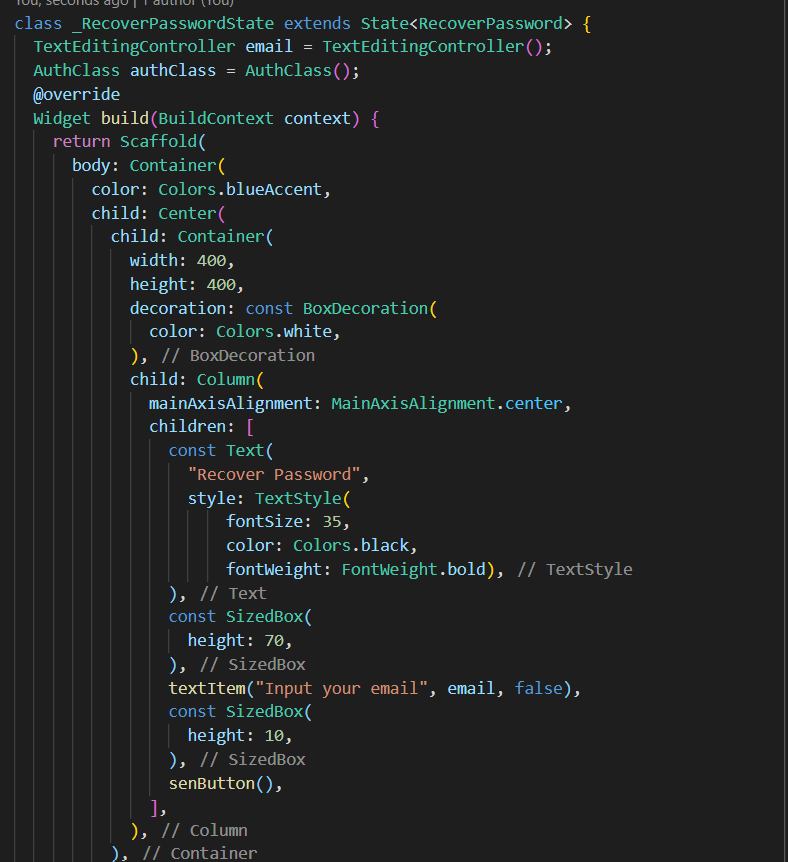


Figure 15. Create recover password in /lib/authentication/recover\_pass\_page.dart

Graphical user interface, application, website

Description automatically generated

Figure 16. Recover password on the browser

* Admin GUI:

1. Employee Table:

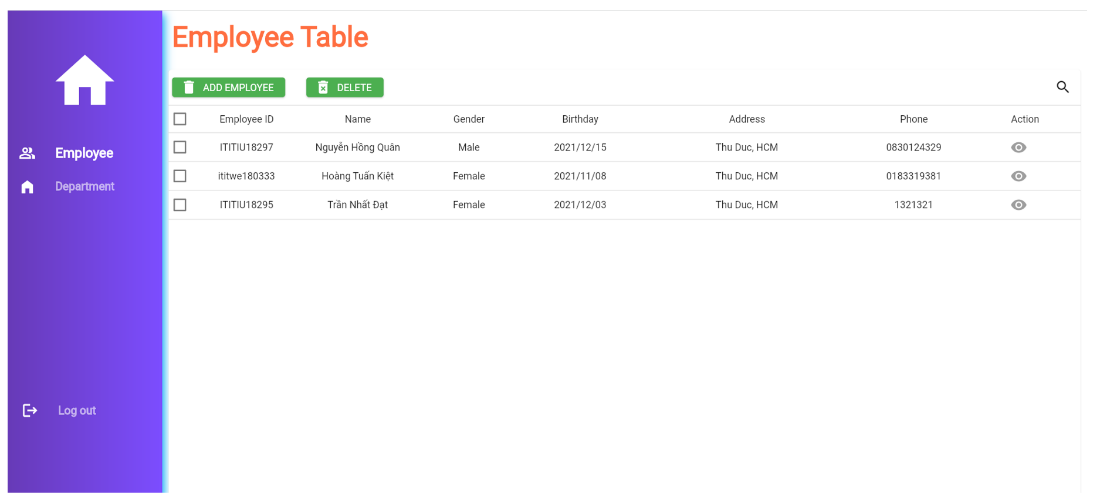


Figure 17. List of Employee (Employee Table) on the browser

1. Delete Employee:

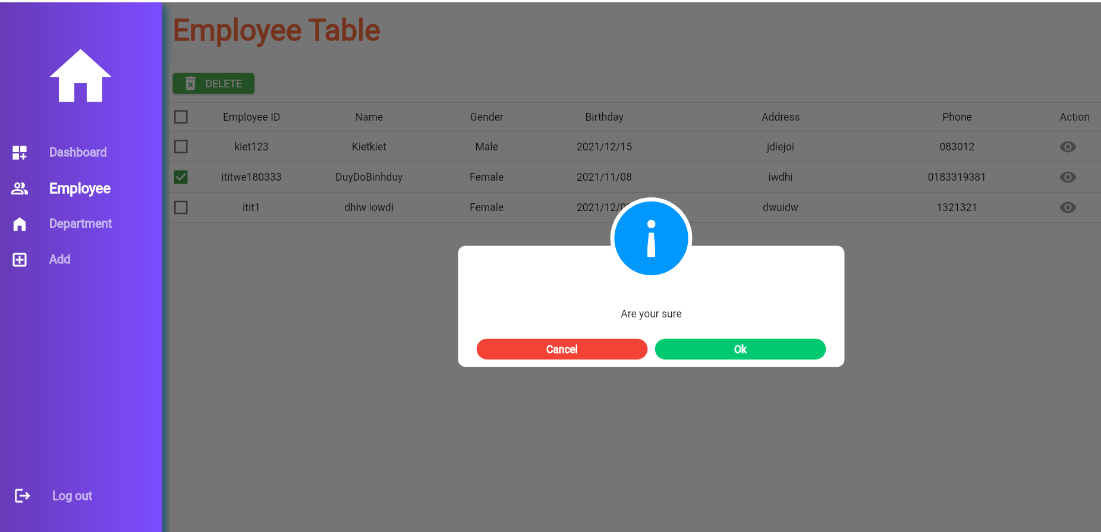


Figure 18. Deleting an Employee from the Employee Table

1. View Employee’s Profile:

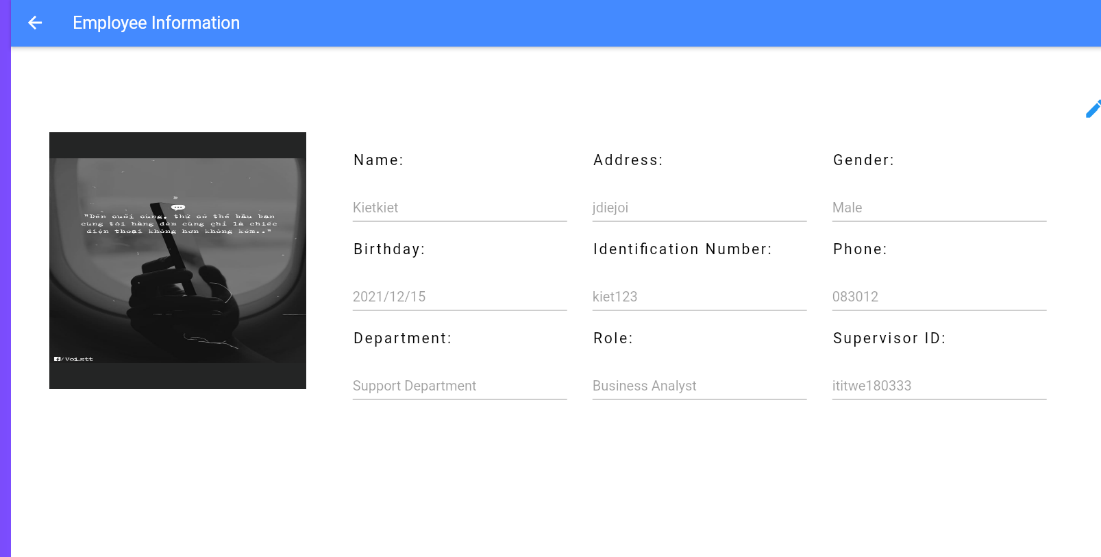


Figure 19. Viewing an employee's profile on the browser

1. Modify Employee’s Profile:

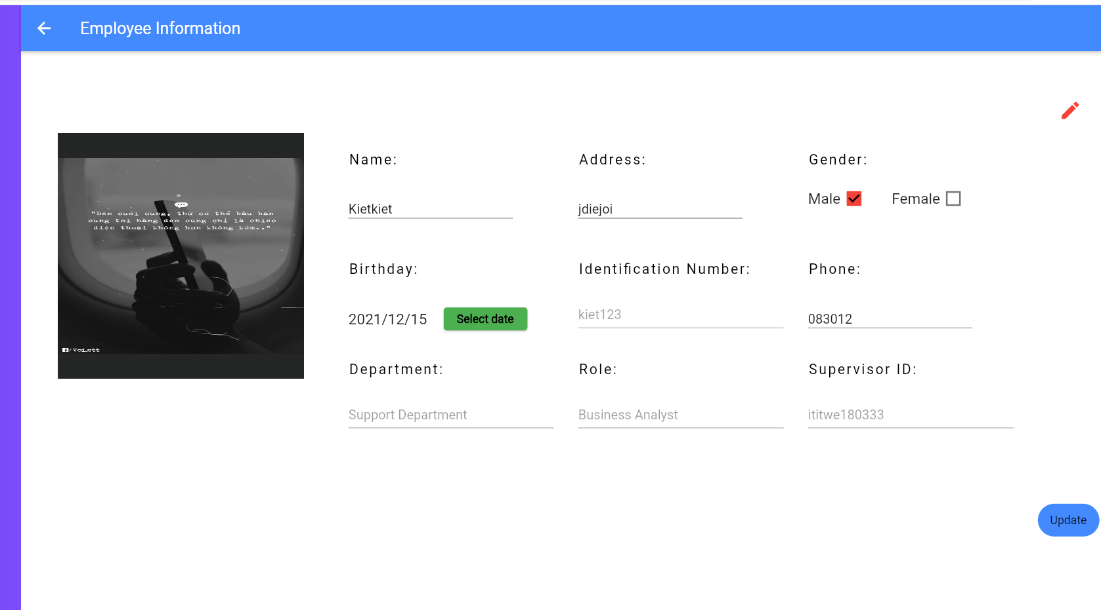


Figure 20. Modifying an employee's profile on the browser

1. Add Employee:

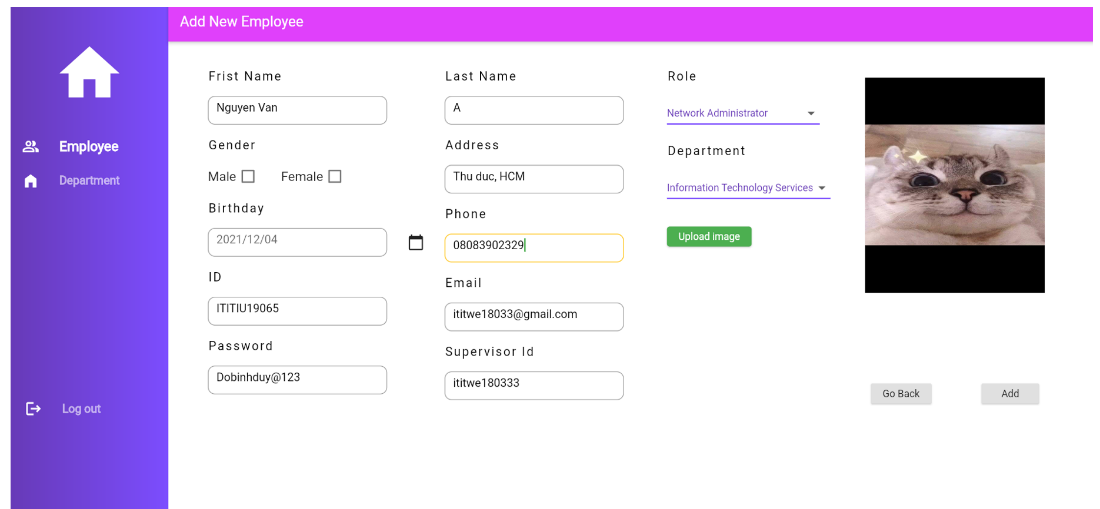


Figure 21. Adding a new Employee to the Employee Table

1. Department Table:

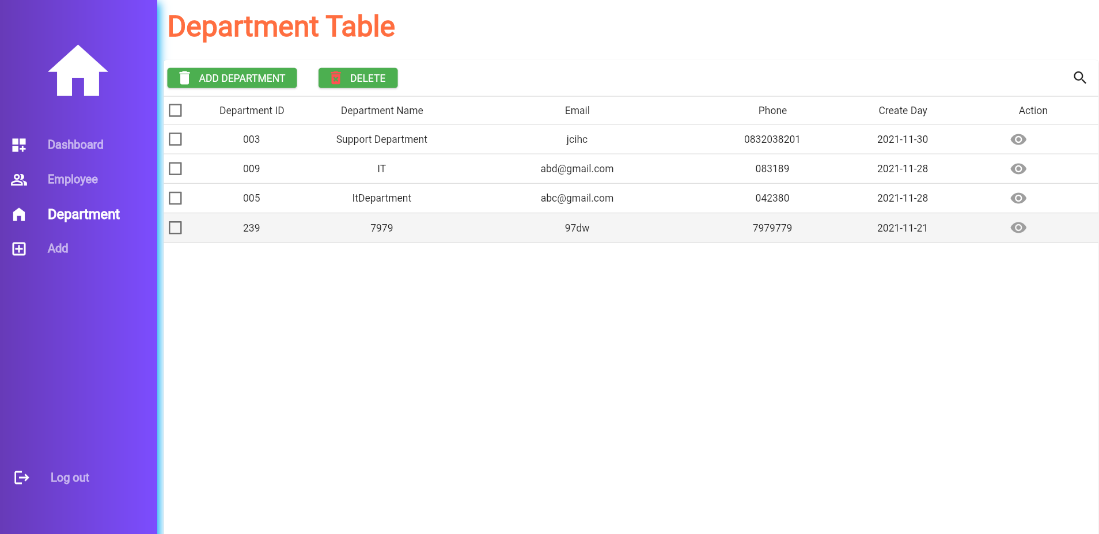


Figure 22. Department Table on the browser

1. Add Department:

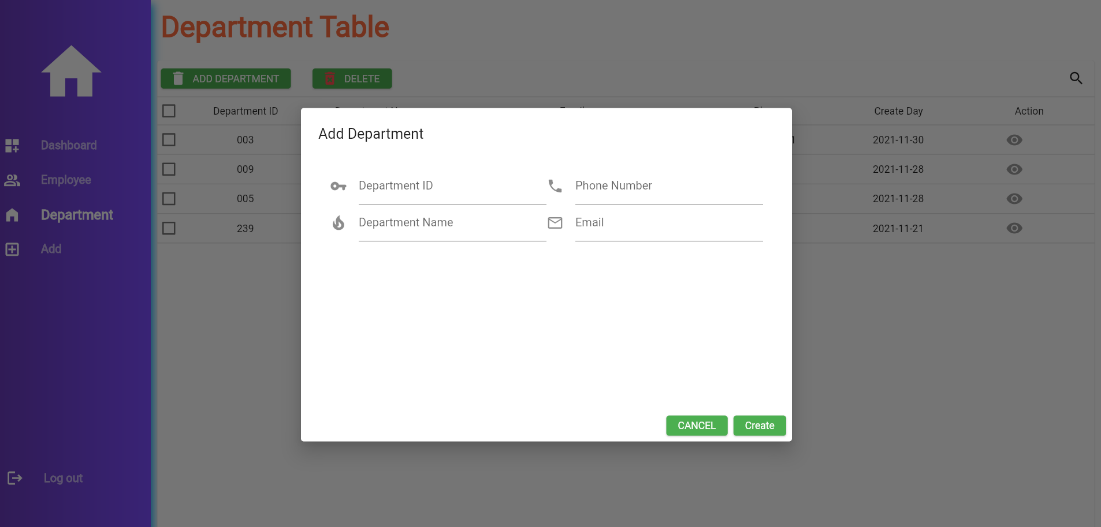


Figure 23. Add a new Department to the Department Table

* Employee GUI:

1. View Personal Profile:

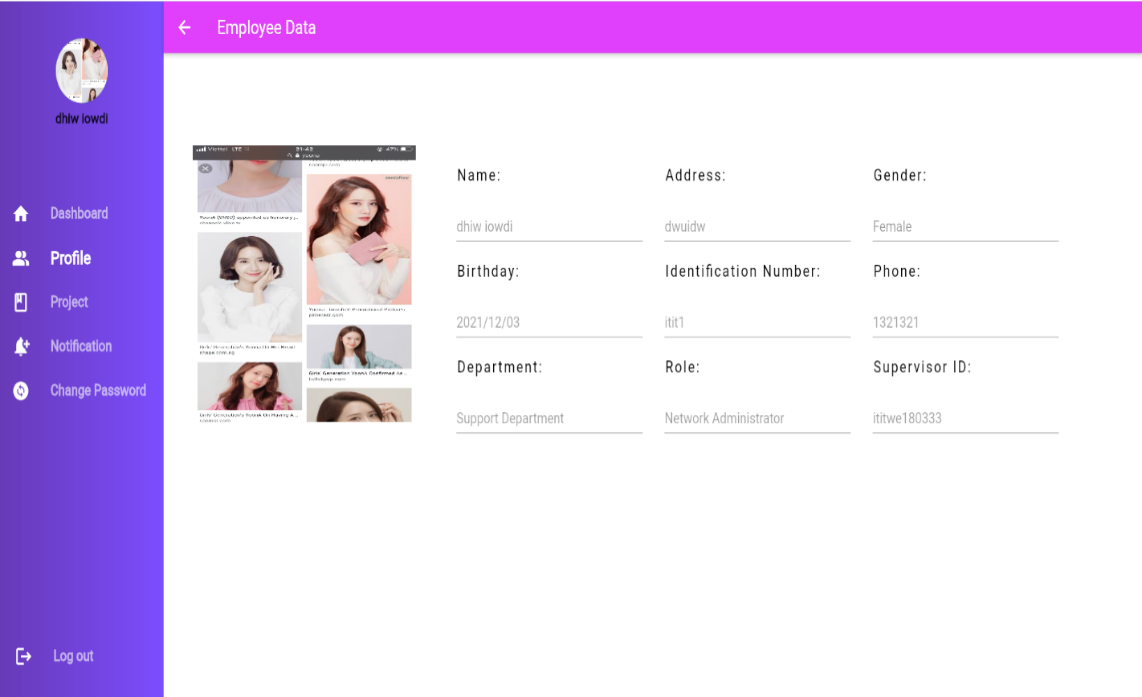


Figure 24. Employee viewing their personal profile

1. View Current Project:

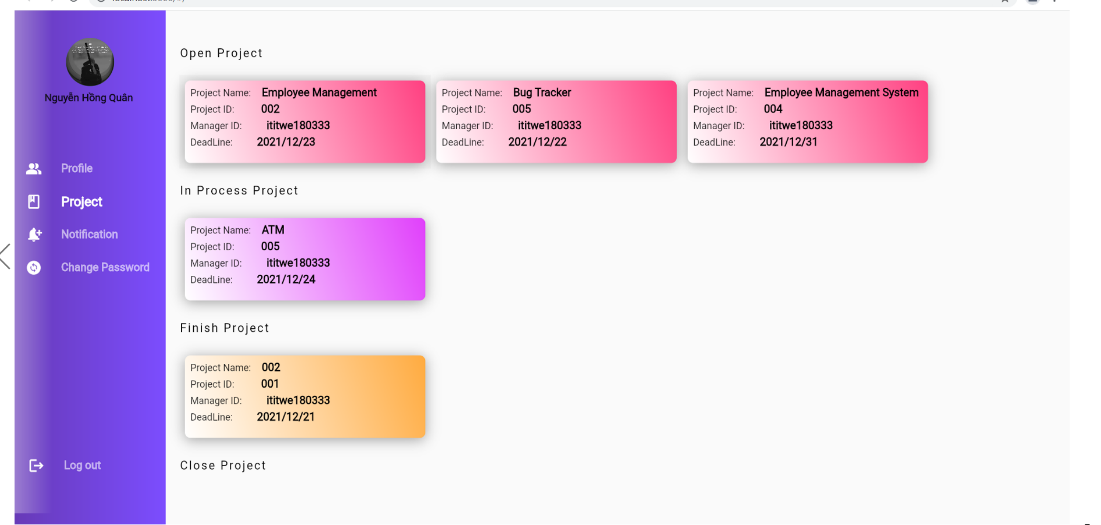


Figure 25. Employee viewing their current projects

1. View project information and tasks:

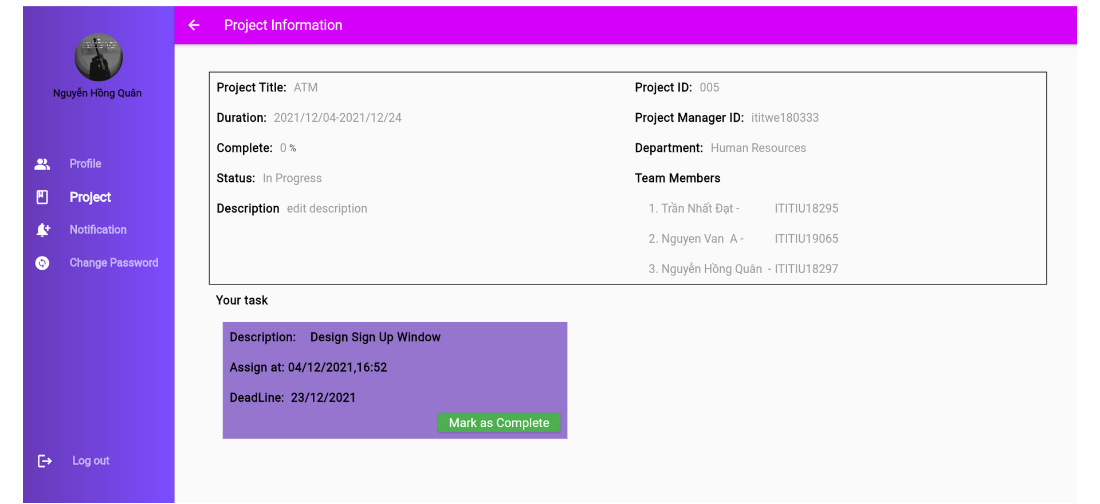


Figure 26. Employee viewing their current project and their tasks

1. View all notification:

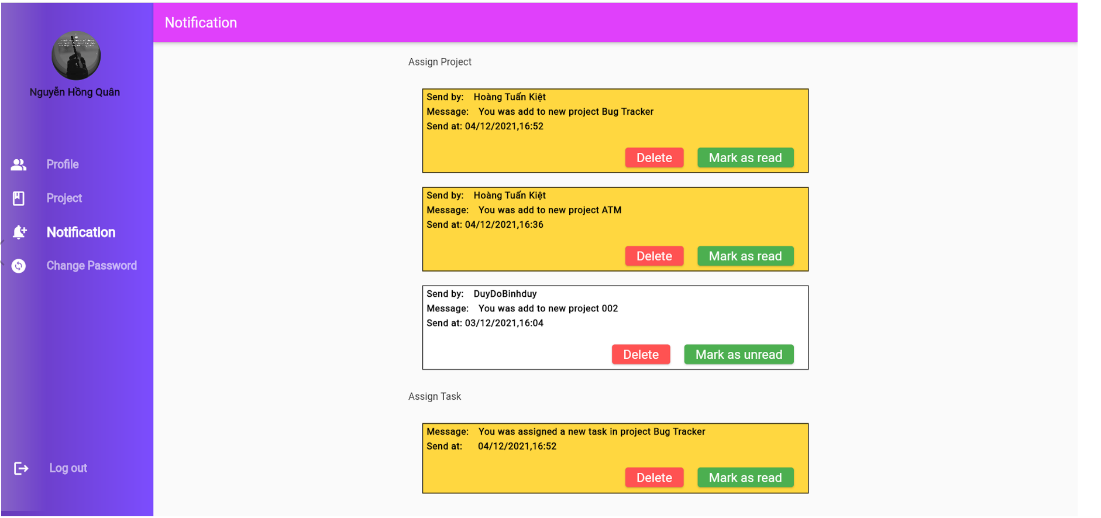


Figure 27. Employee viewing the notifications

* Manager GUI:

1. View all employees belonging to this manager:

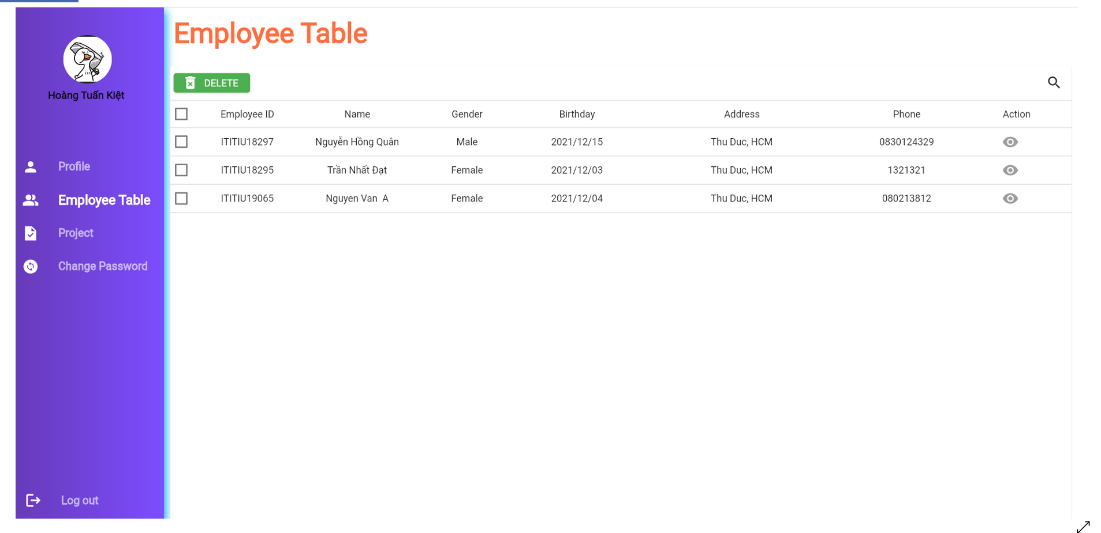


Figure 28. Manager viewing their department's employees

1. View all current projects:

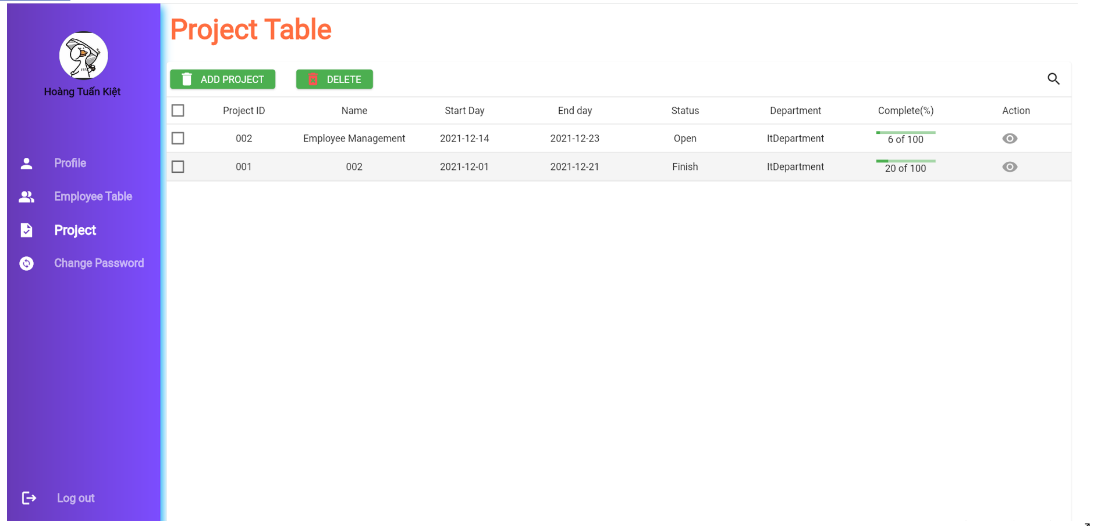


Figure 29. Manager viewing all available projects (Project Table)

1. Add new project:

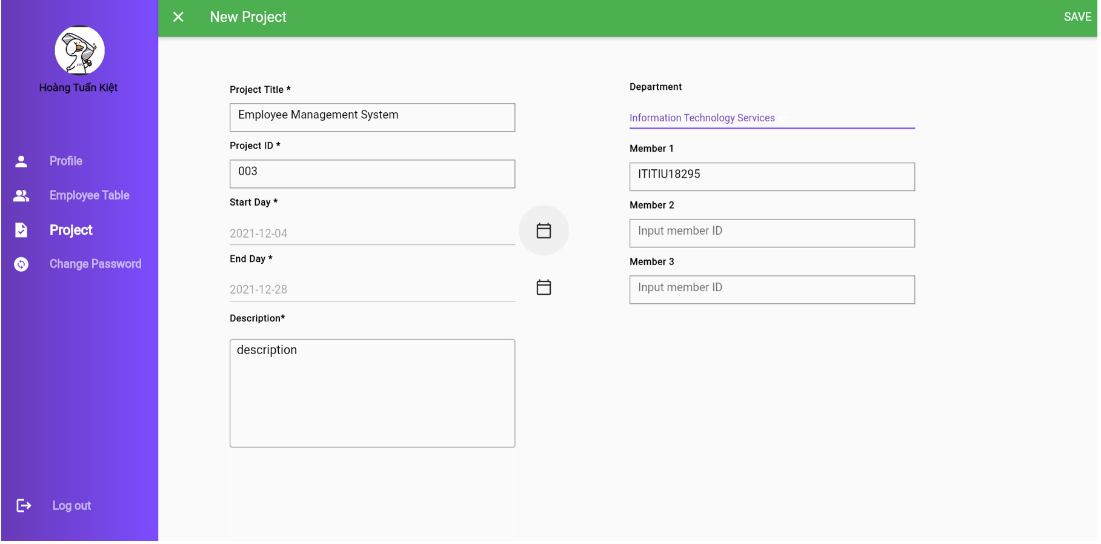


Figure 30. Manager adding a new project to the Project Table

1. Assign task for employees:

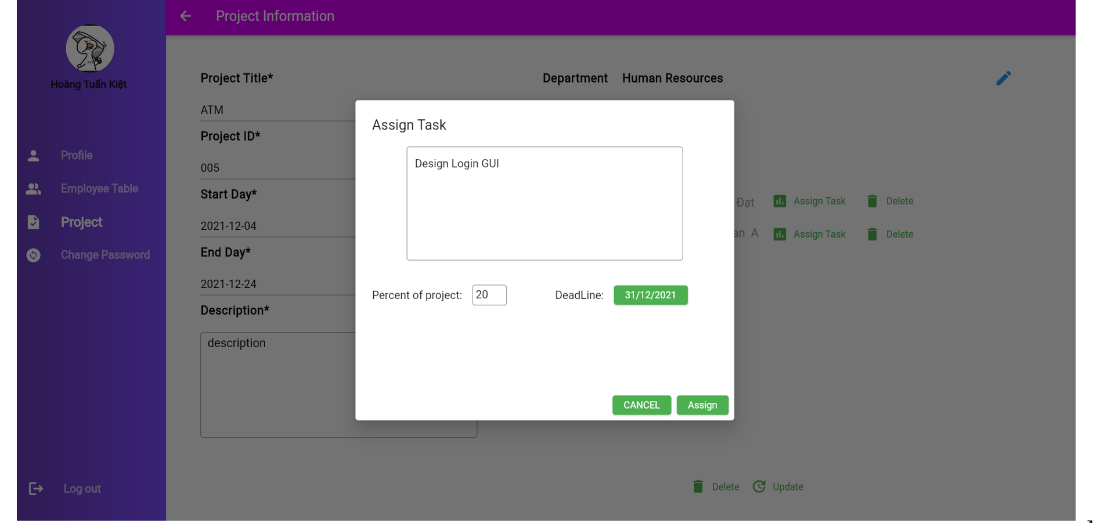


Figure 31. Manager assigning tasks to their employees

1. Add new members to the project via their id:

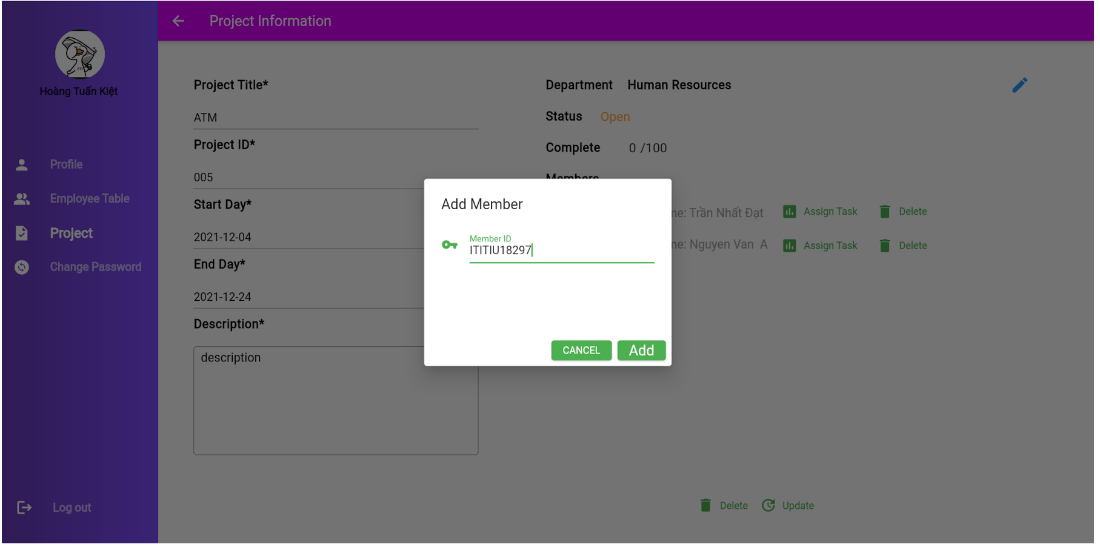


Figure 32. Manager adding a member to the project via their ID

1. Update project’s information (deadline, status...):

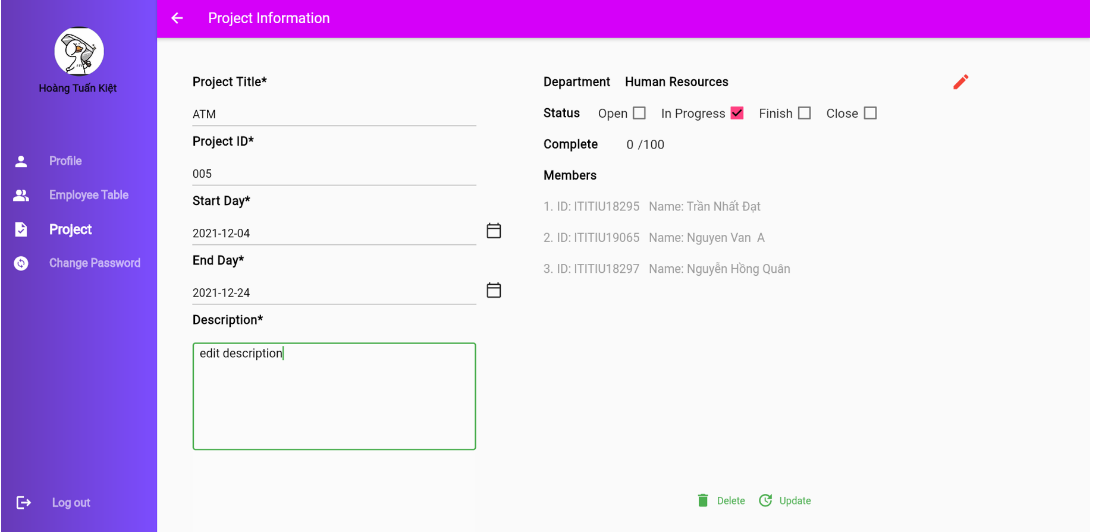


Figure 33. Manager modifying project's info

# **CONCLUSION:**

## **Accomplished Works:**

By the end of the project, the website can perform the following tasks:

* The system allows Users to authenticate accounts.
* The system allows Users to view their personal information.
* The system allows Managers to view every employee’s profile that is under their department.
* The system allows Managers to assign tasks/projects to employees.
* The system allows Managers to manage their employees’ records.
* The system allows Users to use password recovery.
* The system allows Admins to perform system (website and database) maintenance.

## **Strengths and Weaknesses:**

Table 8. Project's Strengths and Weaknesses

|  |  |
| --- | --- |
| **Strengths** | **Weaknesses** |
| The system allows users to recover their password of account. | UI (User Interfaces) is still decent, not very eye-catching |
| The system response quickly to the user’s request. | The system does not optimize the comment section. |
| The system has friendly GUI of the website, so it is easy to use | Media such as images or videos are not supported |
|  | Security is still weak |

## **Future Work:**

The fundamental features of the system were established in the prototype to display the concept of AEMS. Future work will entail building the remaining planned features in order to have a fully functional version of the web application. These features are:

* Allowing users to add media to their profiles allows them to further personalize their profiles.
* Implementing the sponsors' dashboard for managing their campaigns through room clustering and connecting.
* Improving AEMS speed, which allows it to load faster on slower connections.
* Rather than inputting an employee's or manager's name, we may scan their barcode for the first time, look them up, and find the individual using the scanned QR code or barcode. There are a few open-source products and APIs that aid in the implementation of this feature (e.g., Google custom search, but it only allows 1000 free queries per day).
* App development for other platforms, such as the Apple iPhone, iPad, Nexus tablet, and Windows Phone App distribution via digital distribution networks such as Google Play and the iOS App Store.
* The use of semantic web technologies to define relationships in a social network.

Other aspects of future work will involve finding possible sponsors who will use AEMS. This will entail interviewing them, polling them, determining their needs, creating the necessary features, and completing a usability assessment study focusing on our sponsors.

# **REFERENCES:**

1. <https://www.youtube.com/watch?v=vZaztF9Yvgc&t=11s>
2. <https://www.youtube.com/watch?v=p1NPtRdvQBU&t=5s>
3. <https://www.trustradius.com/products/workday-hcm/reviews?qs=pros-and-cons>
4. <https://www.trustradius.com/reviews/weekdone-2019-09-07-07-06-01>
5. <https://www.trustradius.com/products/gusto/reviews?qs=pros-and-cons>
6. <https://www.getapp.ie/reviews/105401/connecteam>