



# **SWE30010 - Managing IT Projects**

## **PORTFOLIO**

***Individual proposal and Research Essay***

Nguyen Dinh Nhat Minh (103802490)

***Human Resource Management With Attendance System***

Group 2

***Swinburne Vietnam, Ho chi Minh Campus - Spring 2024***

Lecturer: Thomas Hang

## GROUP 2 INFORMATION

Name	ID	Tutor	Class
Le Hoang Hai	103542974	Thomas Hang <a href="mailto:Nsam@swin.edu.au">Nsam@swin.edu.au</a>	Saturday 7:00 AM
Nguyen Dinh Nhat Minh	103802490		
Nguyen Nhat Huy	103802911		
Nguyen Ngoc Minh Thy	103802791		

### Table of Content

<b>Individual Tasks:</b>	<b>2</b>
TASK 01P: Scope and Product Backlog	5
TASK 03P: Solution Direction and Design	11
TASK 05P: Software Quality and Definition of Done	12
TASK 061C: Estimation Method	19
TASK 062C: Estimation Accuracy	<b>22</b>
TASK 071D: Quality Definition	28
TASK 072D: Quality Planning	36
TASK 073D: Quality Review	36
TASK 081HD: Software Project Planning, Design and Quality Management	66
TASK 082D: Software Development Methodology Comparison	71
TASK 15P: Peer Review	77
<b>Group Tasks:</b>	<b>79</b>
TASK 02P: Scope and Product Backlog	79
TASK 04P: Solution Direction and Design	82
TASK 06P: Software Quality and Definition of Done	92
TASK 07P: Project Proposal	101
TASK 08P: Sprint Planning Meeting	114
TASK 09P: Project Set Up	120
TASK 10P: Sprint 1 Mid-Point Progress Reporting	124
TASK 11P: Sprint 1 End-of-Sprint Progress Reporting	146
TASK 12P: Sprint Review	168
TASK 13P: Sprint Retrospective	171
TASK 14P: Software Design	173

# **Individual Tasks:**

## **TASK 01P: Scope and Product Backlog**

**Project Proposal: Human Resource Management system for Gemadept**

### **Synopsis**

Gemadept Corporation is a leading logistics and shipping company headquartered in Ho Chi Minh City, Vietnam. With a strong commitment to excellence, we are seeking a comprehensive Human Resource Management Software System to enhance our HR operations.

We would like to have a complete HR Management Software System that meets our specific requirements and integrates seamlessly with our existing systems. This system will be expected to streamline HR processes, increase efficiency, and improve employee management.

### **1. Background**

Gemadept, a leading logistics and shipping company headquartered in Ho Chi Minh City, Vietnam, is requesting a HRM software to streamline HR processes, enhance employee management, and improve overall organizational efficiency. The primary objective of this proposal is to introduce a robust Human Resource Management (HRM) system tailored specifically for Gemadept. This proposal aims to establish the creation of a comprehensive HRM infrastructure, spanning the stages of design, development, and maintenance.

### **2. Scope**

#### **2.1 Objectives:**

This system will be expected to streamline HR processes (recruitment, onboarding, payroll, and performance management), increase efficiency, and improve employee management by allowing employers to track and manage employee's data.

#### **2.2 Solutions:**

To achieve the objectives listed, this software will include but not limited to these key features:

- **Employee Information Management:** Centralized database for storing and managing employee data, such as personal details, employment history, and contact information.

- **Recruitment and Applicant Tracking:** Tracking of applied applications and managing the recruitment pipeline.
- **Onboarding and Offboarding:** Structured procedures for integrating new hires into the organization and managing the exit process for departing employees.
- **Payroll Management:** Adjustable automated calculation of employee salaries, deductions, and taxes.
- **Time and Attendance Tracking:** Tools for monitoring employee attendance, tracking work hours, and managing leave requests.
- **Reporting and Analytics:** Generation of customizable reports and dashboards to provide insights into employee's information and status.
- **Performance Management:** System for setting performance goals, conducting evaluations, and providing feedback to employees.
- **Learning and Development:** Scheduling training sessions and tracking employee progress in skill development.
- **Integration with Existing Systems:** Integration with other business systems such as ERP and accounting software.

### **3. Deliverables and schedule**

#### **3.1 Documentation:**

For the main product, instruction on how to operate and use the software will be provided via the following documentations:

- User Manuals
- System Architecture Documentation
- Training Materials

#### **3.2 Training Plan:**

A comprehensive training plan will be developed to ensure that all relevant users become proficient in utilizing the new HRM system. Our team will leverage this plan to facilitate and conduct training sessions, either through online platforms or in-person sessions, guiding users through the functionalities of the software.

#### **3.3. Schedule:**

This project is expected to take around 20 weeks to completed, while the maintenance support will be an on-going process. The timeline for each stage (from designing to implementation) of the project is outlined as follow:

- System Design and Prototyping: 4 weeks.
- Development and Testing: 12 weeks.

- Implementation and Training: 4 weeks.
- Post-implementation support and maintenance: On-going from go live date.

### **Initial Release Schedule of the Product backlog items**

No.	Item	Dependencies	Business Value (1 least – 10 most)	Release Schedule (Sprint 1   2   3   ...)
F1	Product UI/UX Design	None	7	Sprint 1
F2	HR database schema design	None	9	Sprint 2
F3	Employee information form implementation	F2	8	Sprint 3
F4	Recruitment module development	F2	7	Sprint 4
F5	Payroll calculation module development	F2	7	Sprint 5
F6	On, Offboarding checklist template implementation	F2, F4	6	Sprint 6
F7	Leave request feature implementation	F2, F3	6	Sprint 7
F8	Performance evaluation module development	F2, F3	6	Sprint 8
F9	Integrate with Gemadept's ERP system	After all modules are tested	9	Sprint 9
F10	Implement reporting dashboard	F9	8	Sprint 9
F11	Gemadept's HR staff training	After successful UAT	7	Sprint 10
F12	Post-implementation maintenance	Once product goes live	5	Sprint 10

# **TASK 03P: Solution Direction and Design**

## **1. SOLUTION DIRECTION**

### **1.1. PROBLEM DOMAIN**

- The problem domain of this project is **Human Resource** and **Online Management**.
- It involves developing a comprehensive and user-friendly HRM application that enables effective HR management for Gemadept. Key features of the project include:
  - **Employee Information Management:** Centralized database for storing and managing employee data, such as personal details, employment history, and contact information.
  - **Recruitment and Applicant Tracking:** Tracking of applied applications and managing the recruitment pipeline.
  - **Onboarding and Offboarding:** Structured procedures for integrating new hires into the organization and managing the exit process for departing employees.
  - **Payroll Management:** Adjustable automated calculation of employee salaries, deductions, and taxes.
  - **Time and Attendance Tracking:** Tools for monitoring employee attendance, tracking work hours, and managing leave requests.
  - **Reporting and Analytics:** Generation of customizable reports and dashboards to provide insights into employee's information and status.
  - **Performance Management:** System for setting performance goals, conducting evaluations, and providing feedback to employees.
  - **Learning and Development:** Scheduling training sessions and tracking employee progress in skill development.
  - **Integration with Existing Systems:** Integration with other business systems such as ERP and accounting software.

In addition to the core functionalities, this development also focuses on:

- User-friendly interface for easy navigation.
- Customizable user roles and permissions.
- Automated workflows for HR processes.
- Mobile accessibility for remote workforce management.

### **1.2. SOLUTION DOMAIN**

#### **1.2.1. The Discussion of Alternative Solutions**

The HRM solution entails expertise in HR management, employee information systems, recruitment, applicant tracking, onboarding, offboarding processes, payroll management, reporting, analytics, integration with existing systems, etc. It also encompasses application programming, user interface design, and hosting servers.

The medium in which this solution can be implemented are plenty, from website, OS app, to mobile. However, we have chosen to develop a web-based application due to its adaptability and accessibility. Other options were ruled out due to their limitations in meeting the comprehensive requirements of Gemadept's HRM needs, the reason for why the alternatives were discarded is as follow:

<b>Alternative</b>	<b>Discard reasons</b>
Desktop-based application	<ul style="list-style-type: none"> <li>• It can take up unnecessary storage space on the user's device.</li> <li>• Limiting accessibility for users who require remote access or use multiple devices.</li> <li>• Maintenance of different version base on their OS can be tedious.</li> <li>• User needs to manually install updates.</li> </ul>
Mobile application	<ul style="list-style-type: none"> <li>• Elevated time and resources required to develop compatible versions for each mobile platform.</li> <li>• Restricted availability on mobile platforms such as iOS and Android.</li> <li>• Incompatibility or insufficient hardware capabilities of mobile devices may hinder the execution of the application's various tasks.</li> <li>• Elaborate approval procedures for mobile applications on App Store and Google Play, potentially causing project delays incapable of handling the app's various processes.</li> </ul>

Web-based applications provide greater accessibility, cross-platform compatibility, and ease of updates, which makes it the best solution out of all the options, for example, the user can access this app anywhere and does not need specific devices to be able to use this app.

The main platform where this application will be hosted is a hybrid of dedicated servers on AWS instances. This solution will guarantee the ease of customization and reliability of dedicated servers, while also incorporating the flexibility and scalability of the AWS cloud.

<b>Criteria</b>	<b>AWS and Dedicated hybrid</b>	<b>Dedicated server</b>	<b>On-premises</b>
-----------------	---------------------------------	-------------------------	--------------------

Scalability	Auto-scaling services facilitate handling increased traffic efficiently.	Limited scalability; manual action and downtime involved in scaling.	Limited scalability; additional hardware and resources required for scaling.
Reliability	Multiple availability zones and redundancy features minimize downtime.	Reliability dependent on hardware and network maintenance.	Reliability dependent on infrastructure maintenance.
Infrastructure	Reduced latency with data centers in various regions, enhancing proximity to customers.	Limited by physical server location; latency reduction requires servers in different regions.	Restricted by physical infrastructure location; latency reduction requires additional premises environments.
Security	The web will be protected by Laravel's various security modules, and AWS can provide the added security of ACLs and security group.	Security dependent on settings implemented on the dedicated server, which may require significant time and effort for maintenance and modifications.	Security dependent on infrastructure, necessitating more resources for comprehensive protection.
Cost	Pay-as-you-go model enables scaling resources as needed, cost-efficient for varying business sizes.	Costly hardware maintenance, less cost-effective for small businesses.	Costly setup, maintenance, and resources required.
Integration with other tools	Seamless integration with all Amazon services, and other framework that can help to integrate with Gemadept current technologies.	Limited integration, manual setup required.	Limited integration, manual setup required.

### **1.2.2. The KoST Analysis of My Knowledge**

#### **a) Knowledge (*what you know*)**

Required knowledge:

- **Web-based application:** Knowledge of HTML/CSS, PHP, Javascript and frameworks like Laravel and React.
- **Cloud services:** Knowledge of AWS platform.
- **Hosting server:** Expertise in server management, networking, databases, and understanding of hosting services.
- **Human Resource Management:** Knowledge of corporate human resource management, and management processes.

My knowledge:

- **Web Development Courses:** Enrolled in two web development courses and acquired foundational knowledge of HTML, CSS, PHP, and JavaScript.
- **Database Management System Course:** Enrolled in a Database Management System course and gained a basic understanding.
- **AWS Cloud Architecture Course:** Completed an AWS cloud architecture course and acquired a foundational understanding of its services.
- **HR Management Basics:** Possess basic knowledge of HR management through engagement in volunteer operations involving personnel allocation and administration.

#### **b) Skills (*your experience*)**

Required skills

Technical Skills:

- **Framework/Tools Proficiency:** Practically experienced in utilizing chosen frameworks/tools to facilitate smooth platform development.
- **Application Development and UI Design:** Proficient in application development projects and user interface design.
- **Server Management and Deployment:** Skilled in inspecting hosting servers and deploying platforms into Production environments.
- **Infrastructure Acquisition:** Knowledgeable in hiring or acquiring necessary infrastructure/resources.

Domain knowledge:

- **HRM Applications:** Experienced with HRM applications and their connections.
- **Human Resource Operations:** Understanding the core operations of human resources.
- **Third-Party Integration:** Proficient in third-party integration to ensure compliance with security regulations.

### My skills

Technical Skills:

- **Web Development:** Built three small web projects using HTML/CSS, Javascript, and PHP.
- **Database Management:** Proficient in creating and managing databases using SQL Developer by Oracle.
- **UI/UX Design:** Skilled in web user interface design utilizing Figma and Adobe XD.
- **Cloud Hosting:** Experienced in hosting websites on the AWS cloud.

Domain knowledge

- **Employee Data Auditing:** Experienced in auditing employee data and observing recruitment pipelines.
- **HRM System Feature:** Familiar with using features within a larger Human Resource Management (HRM) system.

### c) Technology (*existing solution*)

Web-based HRM systems can be built using a variety of current technologies and solutions. Among the essential tools and technologies are:

- **Proprietary HRM Software:** Companies like SAP (SuccessFactors), Oracle (Oracle HCM Cloud), and Workday offer comprehensive HRM suites that cover a wide range of HR functions. These solutions often provide modules for core HR, talent management, workforce planning, payroll, benefits administration, and analytics.
- **Cloud-based HRM Platforms:** Cloud-based HRM platforms like BambooHR, Namely, Zenefits, and Gusto offer scalable and customizable solutions for managing HR processes. These platforms typically provide features such as employee self-service portals, onboarding/offboarding tools, performance management, time and attendance tracking, and compliance management.
- **Open-source HRM Solutions:** Open-source HRM solutions provide flexibility and customization options for building web-based HRM applications. Popular open-source HRM frameworks and platforms include OrangeHRM, Sentrifugo, IceHrm, and Odoo HR. These solutions offer modules for managing employee information, leave management, attendance tracking, and more.

## **Rationale**

Following a comparative examination of several options and a gap analysis utilizing the KsoT framework, the following factors indicate that developing a web application and hosting it on AWS cloud services is the best course of action for this project:

- Web-based applications offer a user-friendly interface and are easily accessible from various devices, allowing them to seamlessly integrate and blend in with the current system.
- Scalability capabilities provided by AWS enable the platform to accommodate growing traffic and customer needs. By growing resources as needed, the pay-as-you-go model also guarantees cost-effectiveness.
- AWS provides features like Application Load Balancing, EC2 hosting, and comprehensive security options that are essential for running an e-commerce platform.

I have the necessary abilities to construct a modest project like this one.

### **1.2.3. Final Solution**

The HRM Application will be built as a **web-based application** using **React** for front-end, and **Laravel** for the back-end.

- The HRM Application will be built on **Nginx Web Servers** accompanied by **Cloudflare** and hosted on an **EC2** instance, this combination offers a powerful solution for web application development, providing benefits in terms of performance, security, scalability, and reliability.
- **Oracle Database** will be used to store user data because it can handle a large volume of data efficiently, moreover, Oracle security and compliance are top-notch which can help with protecting user's sensitive data.

## **1. ARCHITECTURE DESIGN**

### **1.1. 3-TIER ARCHITECTURE DESIGN**

#### **TIER 1: PRESENTATION**

The front end interface of the application will be in charge for housing the UI of the application, it is composed of the following component:

- Cloudflare: For optimizing and securing the delivery of web applications.
- React: A JavaScript library for building user interfaces.
- Bootstrap: A front-end framework for developing responsive websites.

## TIER 2: APPLICATION

This layer handles the business logic and core functionalities of the platform, such as displaying reports, processing calculations, and analyze the data it collects. Tier 2 connects the user interface (tier 1) and the database (tier 3).

The framework used for this layer is Laravel, and it has the following component:

- Laravel: A PHP web application framework for building web applications following the model-view-controller (MVC) architectural pattern.
- Oracle DB connection: Allows Laravel to connect and interact with an Oracle database.
- Business Logic Modules: Custom modules developed within the Laravel framework to handle specific business logic and functionalities of the application.
- API Integration: Components responsible for integrating with external APIs to fetch or send data as required by the application.
- Authentication and Authorization: Components responsible for managing user authentication and authorization within the application, ensuring secure access to resources based on user roles and permissions, this part will be the backbone for the authentication feature in the front end.

## TIER 3: DATABASE

Data administration, data query, and storage are handled by this layer. Databases and a database management system are part of Tier 3. It is in charge of guaranteeing the availability, security, and confidentiality of data. It enables the execution of insert, update, and delete queries. In order to deliver the data that the user or application layer request, this layer communicates with Tier 2.

Oracle Database is used for this layer, its components are:

- Data Models: Structures and relationships defined within the Oracle database to organize and store data efficiently.
- Data Access Layer: Components responsible for accessing and manipulating data stored in the Oracle database, including querying, inserting, updating, and deleting records.
- Database Security: Measures implemented within the Oracle database to ensure data confidentiality, integrity, and availability, including user access controls, encryption, and auditing.
- Database Administration Tools: Tools used for monitoring, managing, and optimizing the performance of the Oracle database, such as Oracle Enterprise Manager and SQL Develop.

## 1.2. UML CLASS DIAGRAM

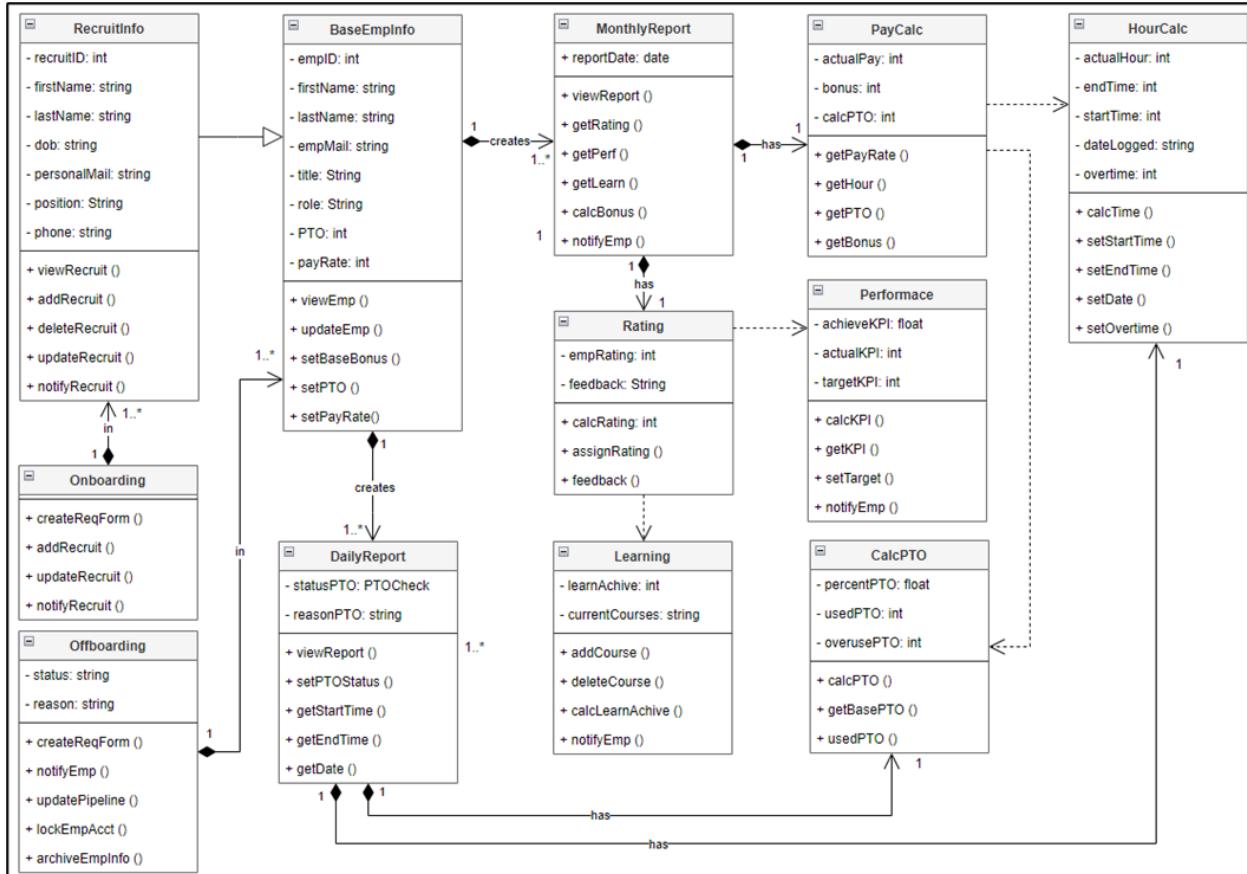


Figure 2: App Class UML

## TASK 05P: Software Quality and Definition of Done

### What is Quality?

With ISO's definition of quality, we can rule out that quality the existence of characteristics of a product, which can be assigned to requirements, and we can also determine the quality of a system by measuring the degree to which it satisfies the stated and implied requirements of its various stakeholders, ultimately providing value.

To determine the requirement needed for this HRM application, we can follow the simple framework outlined here:

#### Technical requirements

- Appropriate framework and the hosting server.
- Responsive user interface.
- User accessibility.
- Data security.

#### Functional Requirements

- User registration and account management.
- Employee management and tracking.
- Time, pay, performance calculations.
- Review/Rating system.
- Feedback mechanism.
- Notification and Alert settings.

### **Non-functional Requirements**

- Have user-friendly UX/UI
- Integrate appropriate security techniques and comply with data protection regulations.
- Integrate with existing system, i.e., ERP, database, timesheet, etc.
- High scalability

### **Software Requirements**

Using the S.M.A.R.T guidelines that indicate that goals and achievement should be Specific, Measurable, Achievable, Realistic, and Time-bound. It is logical to pursue and align the HRM's application's quality definition on the ISO 25010 standard, which indicates the following objective to ensure the quality of a given software:

- **Functional Suitability:** Ensuring that the HRM software provides all necessary functions to support HR processes such as recruitment, employee onboarding, performance management, and training.
- **Performance Efficiency:** Ensuring that the HRM software performs tasks such as data processing and report generation within acceptable time frames and with minimal resource consumption.
- **Compatibility:** Ensuring that the HRM software can integrate with existing systems such as payroll software, time-tracking systems, and enterprise resource planning (ERP) systems.
- **Usability:** Ensuring that the HRM software is user-friendly and intuitive, allowing HR professionals and employees to easily navigate and perform tasks within the system.
- **Reliability:** Ensuring that the HRM software operates reliably without frequent crashes or errors, especially during critical HR processes.
- **Security:** Ensuring that the HRM software protects sensitive HR data such as employee personal information, payroll details, and performance evaluations from unauthorized access or modification.
- **Maintainability:** Ensuring that the HRM software can be updated and modified efficiently to accommodate changes in HR policies, regulations, and business requirements without introducing bugs or issues.

- **Portability:** Ensuring that the HRM software can be deployed across different hardware and software environments, such as on-premises servers or cloud platforms, with minimal effort and modifications

With ISO 25010 as the reference, we create the detail requirement as the definition of done for our project as follow:

- **Core functionalities development:** The platform must be constructed with all of the essential features listed in the project proposal. Both automated test cases and manual tests can be used throughout the testing process.
- **Usability and Scalability:** Consider factors like reaction time and resource usage while assessing the software's performance under workloads. Verify that the system is capable of handling an unexpected increase in workload.
- **Code Quality and Conventions:** Verify that the code complies with the requirements for readability, maintainability, naming conventions, and code structure.
- **Security:** Verify if the platform followed the necessary data protection and authentication guidelines. Performing security testing, reviewing protocols, and reviewing IT policies and documentation are all part of this process.
- **Testing and Documentation:** Conduct unit testing to verify individual components, integration testing to validate interactions between modules, and user acceptance testing (UAT) involving end-users.
- **UX/UI Testing:** Make that the UI is responsive, easy to use, and visually consistent. It ought to satisfy every design criterion listed in the proposal.

### DEFINITION OF DONE CHECKLIST

The following checklist is prepared based on the ISO25010 Model with 8 main characteristics to measure the 'Quality' of the HRM application.

No	(Sub) Characteristics	Description	Y/N
<b>1. Functionality Suitability</b>			
1.1	Functional Appropriateness	Deploy the platform on the cloud-based hosting server	<input type="checkbox"/>
1.2	Functional Appropriateness	The platform allows user creation for 2 account types: business account and user account	<input type="checkbox"/>
1.3	Functional Appropriateness	The platform allows users to switch between different languages. All content, messages, and alternative text are translated correctly	<input type="checkbox"/>

1.4	Functional Appropriateness	The employee report module can generate reports with meaningful insights from the data.	<input type="checkbox"/>
1.5	Functional Appropriateness	The platform's core functionalities can integrate with external systems, such as ERP, database, timesheet.	<input type="checkbox"/>
1.6	Functional Correctness	The number of errors of each core functionality of the e-commerce platform should be less than 5% of total automated test cases	<input type="checkbox"/>
1.7	Functional Correctness	During the account registration process, important fields such as name, password, and email are required	<input type="checkbox"/>
1.8	Functional Correctness	Employee information (base and daily/monthly report) correctly appear on the screen	<input type="checkbox"/>
1.9	Functional Correctness	The searching and filtering options return the correct result	<input type="checkbox"/>
1.10	Functional Correctness	99% of the daily/monthly process are completed and recorded in the databases at the end of day or at the time of generating report.	<input type="checkbox"/>
1.11	Functional Completeness	The app can display well-organized daily employee report with time, leave check, name, title/dept.	<input type="checkbox"/>
1.12	Functional Completeness	The platform can send notifications via email to remind users to check the timesheet of their work time daily at 3 o' clock.	<input type="checkbox"/>
<b>2. Performance Efficiency</b>			
2.1	Time Behaviour	<p>The following actions have a response time of less than 2 seconds:</p> <ul style="list-style-type: none"> <li>§ Return searching results after users search the employee's name or functions of the app.</li> <li>§ Display a list of employees.</li> <li>§ Display the base employee's information when clicking on employee.</li> </ul>	<input type="checkbox"/>

		§ Record employees check in to time calculation module. § Login/Logout	
2.2	Time Behaviour	The loading time of pages is less than 5 seconds	<input type="checkbox"/>
2.3	Resource Utilization	The CPU utilization when running the platform is below 80% and memory consumption is below 70%	<input type="checkbox"/>
2.4	Capacity	A simulated workload of 1,000 employees is used for testing. Could the platform handle this simulated workload without reducing the performance by at least 30%	<input type="checkbox"/>
<b>3. Compatibility</b>			
3.1	Co-existence	The platform can operate simultaneously with at least 2 other software in an operating system without conflicting	<input type="checkbox"/>
3.2	Interoperability	95% of record (reports, module's calculation) shows up correctly on third-party system	<input type="checkbox"/>
3.3	Interoperability	The operational data of the platform can be displayed on the data analytics tool in real-time with a maximum delay of 1 minute.	<input type="checkbox"/>
<b>4. Usability</b>			
4.1	Appropriate recognizability	Users complete the following tasks in less than 3 minutes:  § Register a new account.  § Complete their timesheet. § Managers can search for employees.	<input type="checkbox"/>
4.2	Learnability	90% of users (employees and manager) can complete basic tasks to serve their needs on the first day of the testing period	<input type="checkbox"/>
4.3	Operability	The average number of errors that test users encountered during a 15- minute testing session is less than 2 errors	<input type="checkbox"/>

4.4	User interface aesthetics	The platform is responsive on desktop	<input type="checkbox"/>
4.5	User interface aesthetics	<p>This condition is evaluated based on feedback survey</p> <p>The minimum average satisfaction score of 20 users about the UX/UI of the platform, including webpage structure, text font, color palette, languages, navigation, information display, buttons, etc. is 8 out of 10</p>	<input type="checkbox"/>
4.6	User error protection	<p>90% of test users can understand the content of the confirmation dialog box displayed when they:</p> <ul style="list-style-type: none"> <li>Submit timesheet.</li> <li>Update their information.</li> <li>Change the HRM settings.</li> <li>Generate report generation</li> </ul>	<input type="checkbox"/>
4.7	Accessibility	80% of non-text content (video, images) have alternative text descriptions	<input type="checkbox"/>
<b>5. Reliability</b>			
5.1	Maturity	Mean Time Between Failures (MTBF) metrics of the platform is at least 30 days	<input type="checkbox"/>
5.2	Fault Tolerance	Mean Time to Recover (MTTR) takes less than 2 hours to recover platform data from errors or failures	<input type="checkbox"/>
5.3	Recoverability	All functionalities of the platform can be fully restored from the backup within 5 hours after the failure	<input type="checkbox"/>

5.4	Availability	The platform is accessible for 99.9% of operating hours	<input type="checkbox"/>
<b>6. Security</b>			
6.1	Confidentiality	Block at least 95% of unauthorized access attempts and send alerts to users	<input type="checkbox"/>
6.2	Confidentiality	Data encryption and secure authentication protocols are integrated	<input type="checkbox"/>
6.3	Integrity	100% of data is stored and transmitted correctly. Data in the databases is reflected correctly on the user interface and data analytics tools	<input type="checkbox"/>
6.4	Integrity	The platform follows data protection regulations	<input type="checkbox"/>
6.5	Non-repudiation	Implement non-repudiation measures for at least 95% of user actions	<input type="checkbox"/>
6.6	Accountability	100% of user interactions on the platform and the platform events are recorded in the logbook	<input type="checkbox"/>
6.7	Authenticity	95% of user authentication processes are successful	<input type="checkbox"/>
<b>7. Maintainability</b>			
7.1	Analyzability	Reduce time taken to diagnose and resolve reported issues to less than 1 hour 30 minutes	<input type="checkbox"/>
7.2	Modifiability	New change/enhancement is integrated into the platform 1 week after the change request is made	<input type="checkbox"/>
7.3	Testability	Maintain a code coverage of at least 80% through automated test cases	<input type="checkbox"/>
7.4	Modularity	The platform architecture is designed with at least 5 modules	<input type="checkbox"/>
7.5	Reusability	All codes have naming conventions, and comments to modify when needed without affecting other parts (Low coupling – High cohesion)	<input type="checkbox"/>
7.6	Reusability	A minimum of 80% code is reusable	<input type="checkbox"/>
<b>8. Portability</b>			

8.1	Adaptability	The time to deploy the platform on a new hosting server (on-premises or cloud server) is less than 7 days	<input type="checkbox"/>
8.2	Installability	The time to complete the installation and configuration of all platform components in an environment is less than 8 hours	<input type="checkbox"/>
8.3	Replaceability	Could we replace a component of the platform with a migration time of less than 1 week without impact on other components?	<input type="checkbox"/>

## TASK 061C: Estimation Method

### BACKLOG

This table below shows a portion of our product backlog, specifically just that of sprint one.

No.	Item	Dependencies	Business Value (1 least – 10 most)	Release Schedule
F1	Product UI/UX Design	None	7	Sprint 1
F2	Website for HRM	None	9	Sprint 1
F3	HR database schema design	F2	8	Sprint 1
F4	Manager Portal	F1, F3	7	Sprint 1

### 1. Sprint Backlog Item to develop

Following discussions with my team members, we have decided to prioritize the development of the **Website for the HRM Application**. Our decision is based on utilizing the SMART framework, which revealed that our team possesses the most relevant skill sets for this task, including UI/UX design and web coding. Moreover, this will serve as the backbone of our project, enabling us to house further features and functionalities.

The decision not to prioritize Product UX/UI Design was driven by its cosmetic nature, which we determined would not significantly impact the website's functionality. Therefore, it can be deferred to a later stage.

Similarly, the schema design and manager portal will be contingent upon the completion of the website's backbone. As such, they will not be prioritized and will be developed once the website's core functionality is established.

## 2. Work Breakdown Structure

As a team, we leveraged our collective experiences and expertise to thoroughly discuss and outline the tasks required to complete the sprint backlog items. For the website, we identified five primary tasks that need to be accomplished. These tasks will be further delineated using the Work Breakdown Structure (WBS) model to ensure a detailed and comprehensive approach to completing this sprint item.

We utilize the Software Development Lifecycle's stages to determine how to break the task down into various sections:

Software Development Lifecycle	Work Breakdown Structure
Planning & Analysis	Requirement Collection
Design	Architecture Design
Implementation	Development
Testing & Integration	Testing & Deployment
Maintenance	No maintenance at this stage as the whole product has not been finalized.

*Figure 2.1: SDLC table to determine WBS.*

We adopt a hybrid approach, combining both activity-based and product-based methods, to construct our Work Breakdown Structure (WBS). We first concentrate on identifying the main requirements associated with each product, which serves as the foundation for outlining the general framework of our website. Subsequently, we delineate the first task within each product, breaking them down into sub-tasks.

This approach enables us to efficiently address tasks required for specific components within our build, facilitating prompt completion as they arise and ensuring comprehensive control over our product development process.

Through this method, we have devised the design of the WBS diagram as follows:

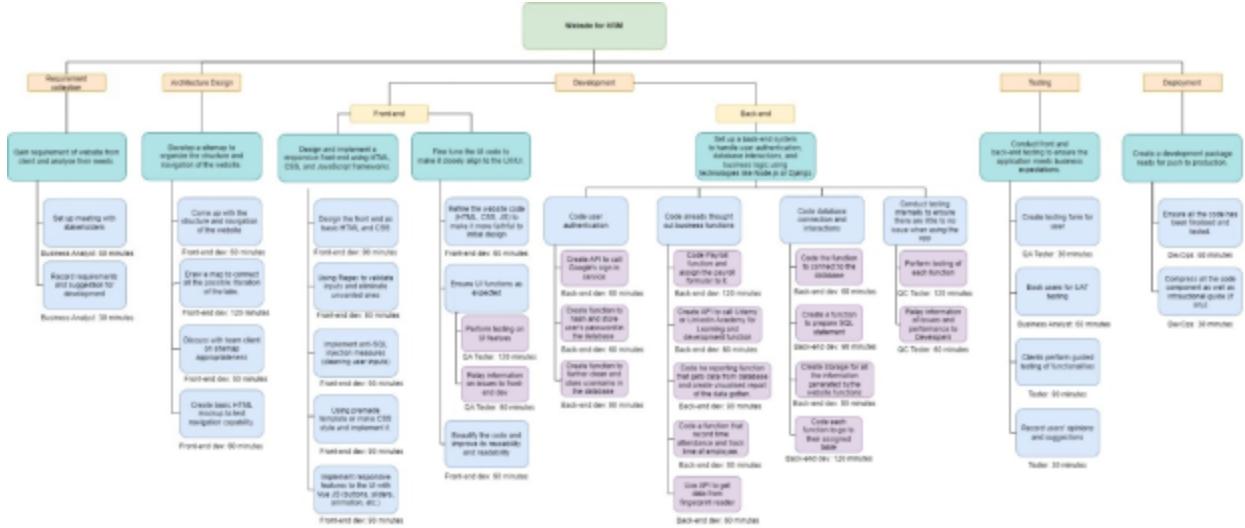


Figure 2.1: WBS Chart with Time Estimation for each task

The WBS structure aids in clarifying tasks by breaking them down into smaller, more manageable components. As a result, each main task that lacks clarity regarding its purpose will be complemented by two to three sub-tasks to provide greater clarity and detail.

### 3. Rationale and Analysis

The first step is gathering the client's requirements, after which business analysts will schedule a meeting to record the client's website specs. This step requires the least amount of time because it mostly focuses on the client's additional expectations for the website, as the core criteria were specified in the project proposal.

The front-end designer or developer is responsible for drafting the website's layout during the process of building the sitemap. Based on my prior sitemap design experience from my web development degree, this work can be very time-consuming and tedious. It entails laying up every tab and feature that the website might have. Furthermore, extensive team discussions are required to guarantee fluid programming and seamless function linkage; as a result, this task may take longer than expected.

With regard to the principal duty of creating and executing the front-end framework, the design stage may proceed in a gradual manner. That could speed up the process, though, if a UI draft has been created using a platform like Adobe XD or Figma. On the other hand, due to the thorough nature of code development, writing the front-end with the selected framework will require significant resources and effort. Additionally, potential DDOS and SQL injection attacks must be dealt with on the front end, adding another level of complexity and requiring more work.

As a checkpoint to guarantee the promised usability and user-friendliness, the front-end must be modified to match the design precisely once it has been developed with responsive functionality and features. Additionally, testing the website to find and fix any security flaws is part of this step. It goes without saying that in order to fully guarantee security and efficient operation, this stage will also require a substantial time commitment.

As the back-end handles the majority of the website's functions, development inevitably takes a long time. Even though authentication code is time-consuming since it must follow security guidelines, it could nevertheless be completed faster than the website's many features. Creating a database connection is time-consuming because SQL queries must be prepared in order to query the data, and the connection must be kept open at all times so that clients can get the information they need without having to repeatedly shut down. Additionally, because of the nature of the coding needed, coding and building data tables might take a lot of time.

It will take a lot of time and work to carefully review the functions throughout the back-end development stage because QC testing is also necessary to guarantee functionality and fix vulnerabilities.

User testing and feedback gathering comprise the penultimate step, which can be finished really rapidly. The main task of this step is to arrange user testing sessions and gather input to improve the website. Other than scheduling users' spare time, it doesn't require any labor.

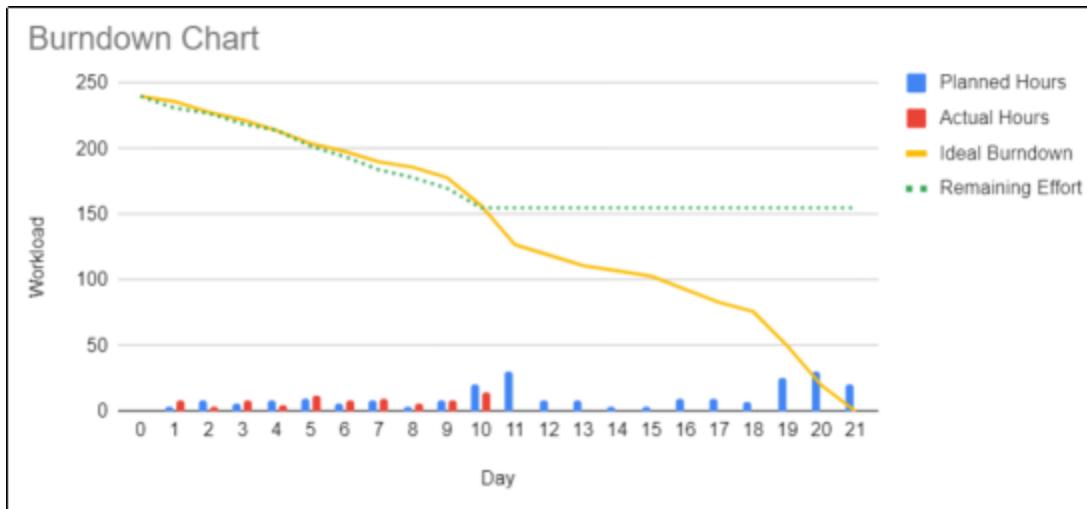
DevOps must be included in the deployment stage of the process in order to examine all pertinent product details prior to packaging the product for production. DevOps shouldn't have too much trouble determining the product's readiness if all testing proceeds as planned, making this task reasonably rapid and not unduly time-consuming.

## **TASK 062C: Estimation Accuracy**

### **Evidence**

#### **1. Work Burndown Chart**

A burndown chart to keep track of the days it should take in order for the sprint to be completed. It is important to note that this chart represent the total effort for all the sprint product in sprint one and not particularly any sprint product.



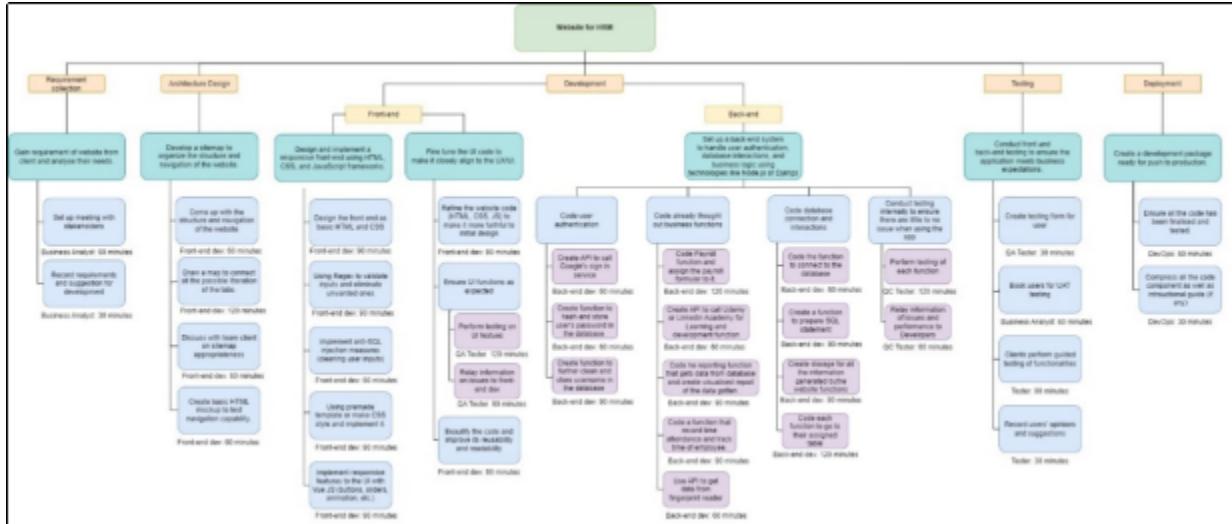
*Figure 1.1: Sprint 1 burndown chart*

Sprint 1						
Day	Burned Down		Balance		Daily Completed	
	Estimate	Actual	Estimate	Actual		
0	0	0	240	240	0	
1	4	9	236	231	9	
2	8	4	228	227	4	
3	6	8	222	219	8	
4	8	5	214	214	5	
5	10	12	204	202	12	
6	6	8	198	194	8	
7	8	10	190	184	10	
8	4	6	186	178	6	
9	8	8	178	170	8	
10	21	15	157	155	15	
11	30		127	155		
12	8		119	155		
13	8		111	155		
14	4		107	155		
15	4		103	155		
16	10		93	155		
17	10		83	155		
18	7		76	155		
19	26		50	155		
20	30		20	155		
21	20		0	155		

*Figure 1.2: Sprint 1 burndown table*

## 2. Work Breakdown Structure

We created a WBS chart to estimate the effort it would take to develop the base website for our project. Other sprint products are merely features or design for our website, hence we put our focus on analyzing this particular product.



*Figure 2.1: WBS Chart with Time Estimation for each task*

### **3. Website building – Front end and Back end**

Per inquiry with our developer, he spent 40 hours (which amount to 4 hours per day) on the website's backend and front end, he also recorded all of his work on Github for ease of review.

## Front end Github:

- o- Commits on Mar 23, 2024
- Update EmployeeLayout.vue**  
HaileInnoTech committed 5 days ago · ✓ 1 / 1    dd98c7e  
- o- Commits on Mar 22, 2024
- Update Payroll.vue**  
HaileInnoTech committed last week · ✓ 1 / 1    baa1fb00  
- Add more bugs, and fix some bugs**  
HaileInnoTech committed last week · ✓ 1 / 1    c2c4285  
- Add more bug to fix later. Update Author, Authen for role admin, add form for create new account.**  
HaileInnoTech committed last week · ✓ 1 / 1    9e8754a  
- o- Commits on Mar 20, 2024
- Update payroll component**  
HaileInnoTech committed last week · ✓ 1 / 1    a528228  
- Update router Update design**  
HaileInnoTech committed last week · ✓ 1 / 1    995bbae  

Commits on Mar 19, 2024

- Create `vercel.json`  
HaileInnoTech committed last week 1 / 1 7fbe22 ⚡ ↗
- Update `Login.vue`  
HaileInnoTech committed last week 1 / 1 bfa1194 ⚡ ↗
- Production  
HaileInnoTech committed last week 1 / 1 b1bf6b4 ⚡ ↗
- Update chart, filter, backlog  
HaileInnoTech committed last week 640ee53 ⚡ ↗

Commits on Mar 18, 2024

- Update employee data view  
HaileInnoTech committed 2 weeks ago 616769d ⚡ ↗
- Update Pagination for employee list  
HaileInnoTech committed 2 weeks ago 09d0161 ⚡ ↗

Commits on Mar 17, 2024

- Update render list of employees  
HaileInnoTech committed 2 weeks ago ebacfe ⚡ ↗

Commits on Mar 13, 2024

- Update Mainpage based on sketch  
HaileInnoTech committed 2 weeks ago e7b5ec ⚡ ↗
- Add Horizontal navigation header  
HaileInnoTech committed 2 weeks ago 5a45147 ⚡ ↗

Commits on Mar 12, 2024

- Add login page template  
HaileInnoTech committed 2 weeks ago 384ed9e ⚡ ↗

Commits on Mar 6, 2024

- Initial commit  
HaileInnoTech committed 3 weeks ago 37dc27 ⚡ ↗

Figure 3.1, 3.2, 3.3: Front-end Github commits

## Back end Github:

Commits on Mar 22, 2024

- Update `request.rest`  
HaileInnoTech committed last week 118924f ⚡ ↗
- Update `index.js`  
HaileInnoTech committed last week 39e96c9 ⚡ ↗
- Fix and improve  
HaileInnoTech committed last week 0594c5c ⚡ ↗

The image displays two screenshots of GitHub commit histories, illustrating parallel development between front-end and back-end projects.

**Screenshot 1 (Top):**

- Commits on Mar 21, 2024:
  - Update author, authen** (HaileInnoTech committed last week) - Commit ID: afab6c6
- Commits on Mar 20, 2024:
  - Update index.js** (HaileInnoTech committed last week) - Commit ID: 56e93bf
- Commits on Mar 19, 2024:
  - Update index.js** (HaileInnoTech committed last week) - Commit ID: c4e187b
- Commits on Mar 16, 2024:
  - test** (HaileInnoTech committed last week) - Commit ID: 37dd269
  - Update api to set bonus by employee email** (HaileInnoTech committed 2 weeks ago) - Commit ID: 43c0429
  - Update API for employee Salary by Email** (HaileInnoTech committed 2 weeks ago) - Commit ID: 4f6144f

**Screenshot 2 (Bottom):**

- Commits on Mar 13, 2024:
  - Update api get employee data by email** (HaileInnoTech committed 2 weeks ago) - Commit ID: 9a0f1e4
  - Update api get attendance by email** (HaileInnoTech committed 2 weeks ago) - Commit ID: b378d0b
  - Update api get total attendance** (HaileInnoTech committed 2 weeks ago) - Commit ID: dcba9ae
  - Update api get Employee Data** (HaileInnoTech committed 2 weeks ago) - Commit ID: 069351b
- Commits on Mar 6, 2024:
  - Setup environment for API** (HaileInnoTech committed 2 weeks ago) - Commit ID: 2fb8c32
  - Initial commit** (HaileInnoTech committed 3 weeks ago) - Commit ID: edf1b74

Figure 3.4, 3.5, 3.6: Back-end Github commits

As can be seen, he was able to push a number of commits on the dates that is parallel to the 9 dates he was able to work on the project. There is a lack of 1 commit on both the front and the back end which equals to 1 date, but it is due to the fact that some of the packages that should have been a separate commit, were collectively pushed into 1 commit.

#### 4. Rationale and Analysis

We devise a product backlog that defined 4 sprints for our project, each sprint would be as long as 21 days, and the total work hour it would take for each of them is 240 hours (about 3 hours per day for 4 person). In our first project sprint, we have got four tasks to tackle, building the website is the main focus and will likely be the most lengthy.

As of the moment, our team had finished the first 10 days of the sprint, so we reached the halfway milestone of our project. According to our burn down table and chart, we should have worked on the entire project at least 83 hour at this point and looking at the work breakdown structure, building the website will take around 44.5 hours in total, and since we had defined the website as the utmost important product in our plan, we put a lot more effort in to finishing that product first.

Given the predicted 44.5 hours needed for it in total, we were able to work on this website for 40 hours, give or take. This is within the 10% error margin of our project, as we have not yet completed the additional testing duty. Thus, of the 85 hours that we really worked on the sprint, the website consumes almost half of our time, leaving us with 45 hours to concentrate on the other three products.

Completing the website we had envisioned seemed like a great accomplishment, especially considering the amount of time and work that goes into establishing a website or software. It resembles demolishing a massive wall. The foundation for future feature additions is established by the essential functionality of this website.

In order to get ready for the upcoming sprint, we took some time to evaluate our performance from the previous one and determine how we were able to achieve a narrow margin of error. We found a few key components that truly helped us accomplish our goal of finishing the website in the allocated amount of time:

- **Consulting with Professionals:** Although the term "professionals" can be a bit wide, we tried to interact with people who might offer insightful information on the intricacies of creating a website (such as YouTube teachers, peers who have done a lot of web development, etc.). We felt that by getting outside advice, we were able to increase the precision of our estimates and gain a deeper understanding of the work needed for each component.
- **Leveraging past experiences:** Since every member of our team had studied web development and was aware of its complexity, we were able to analyze past data to find trends and patterns in the amount of time it took to complete tasks. It proved to be quite beneficial for us in terms of project planning.
- **Prioritize Important Tasks:** We eat the frog and complete the most important functions of our website first (e.g., database, payroll, handling user's data, etc.), this allow us to do the smaller tasks later and not be too overwhelmed with the amount of work ahead, effectively make our processes more efficient.
- **Regular Meeting:** We held frequent refinement sessions prior to sprint planning because we believed in checking in with one another. This allowed us to review and reevaluate our initial estimations and gain insight into the work progress of other team members. By means of cooperative deliberations and input from

fellow team members, we can enhance our approximations, accommodate new insights or modifications in specifications, and ensure compliance with team capabilities..

We believed that accuracy achieved in our project's progress stems from a combination of planning, prioritization of critical tasks, and adaptability to changing circumstances. By leveraging these strengths and remaining agile in our approach, we are well-positioned for continued success in future sprints and beyond.

## **TASK 071D: Quality Definition**

### **SPRINT BACKLOG**

This table below shows a portion of our sprint backlog, specifically that of sprint one.

No.	Item	Business Value (1 least – 10 most)
F1	Product UI/UX Design	7
F2	Website for HRM	9
F3	HR database schema design	8
F4	Manager Portal	7

#### **1. Sprint Backlog Item to develop**

The *Website for HRM* item will be the product with the highest prioritization and will be the first thing to be produced in the meantime. Its steadfast development is due to its business value and importance i.e., being the backbone for other functions to be developed upon.

In SDLC, quality separates the different between a still-developing product and a fully functional one. Hence, it is important to define a quality checklist to ensure the sprint product can perform well in the real world.

#### **2. Software Quality**

##### **Definition of Done**

For a product to be shippable or done, it must satisfy the stated and implied needs of its various stakeholders, and thus provides value. In this sprint, for the website to be considered done, its functional requirement should be fulfilled. All core functionalities of the website can work efficiently and do not have any severe errors that corrupt the system. Core functionalities include:

- Biometric Attendance Tracking
- Employee information Management
- Payroll Integration
- Reporting and Analytics
- Time checks and attendance
- Performance and Development tracking
- Integration with third party services
- Basic website capability
- These definitions should be the base to build the quality checklist on.

### ISO25010 Software Quality Model

Using the ISO25010 model, and the desired core functionalities of the website listed above, we can select the characteristics which can be the most significant to our product, and in this case, we will implement the Functional Suitability characteristic and its Functional Correctness sub-characteristic to build our quality checklist on.

To create a checklist that can present how well the functional correctness of the actual build can be, we will be applying the SMART framework again to help us measure our own capabilities and devise specific metrics and thresholds to assist in designing the quality checklist.

Applying the elements, we can finalize a quality checklist for our website as follow:

Functional Suitability – Functional Correctness		
No.	Metrics	Thresholds
1	Individually, each calculation module (e.g., payroll, time worked, rating) should have little to no calculation errors.	< 5% calculation error
2	When a calculation is made in any one of the modules, the related attribute in other modules must update correctly.	< 5% calculation error

3	Employee's information updates should correctly appear on the screen.	100% accuracy
4	Employee's information should be recorded correctly within the database	100% accuracy
5	The dashboard chart should be used properly to show clear details of employee's information.	Clear visualization with < 5% deviation
6	The database searching and filtering options should return the correct result.	95% accuracy
7	Biometrics token must be correctly verified and apply for the correct personnel	95% accuracy
8	Data conversion of the fingerprint to be stored in the database must be accurate for each of the employees.	< 5% fingerprint data conversion error
9	During the account registration process, important fields such as name, password, and email are required.	100% of the information is fulfilled
10	Most authentication attempts must be successful	90% success rate
11	The website must pass many test cases that prevent XSS and exploiting URLs.	98% test case passed
12	Proper input sanitization should be checked to prevent code injection or unintended characters. Additionally, handling of special characters, whitespace, and input encoding should be validated.	100% input fields are protected
13	Form submission and data transmission should be tested to ensure that data is correctly processed and sent.	95% test case passed
14	Error messages should be displayed when invalid data is entered, providing clear instructions on how to correct them.	100% accuracy
15	Links to different functionalities should be correct.	100% accuracy
16	API connection to other services (i.e., existing ERP, Google sign in, Google sheet, etc.) should have little error	< 5% API errors
17	The website should display and run as expected on various browsers.	98% of browser tested must passed

Figure 2.1: Functionality correctness checklist

### 3. Metric and Threshold values rationale

We based our checklist around the functional correctness; hence, it will be heavily focused on the accuracy and correctness of each of the HRM website's core capabilities in addition to every website's basic functionality.

Justification for each of the items mentioned in the checklist are as follow:

Functional Suitability – Functional Correctness	
No.	Rationale
1	As a reporting platform, calculation must be as precise as possible, hence calculation errors or formula errors must be minimal.
2	Like the previous metric, this one is concerned with calculation taken from various models. And the result of this calculation must not have calculation error larger than 5%
3	Functions that record data from the main website must update the database correctly 100% of the time to aid in various other functions.
4	On the webpage, information of the employees must be displayed in full and without errors. This will not only help with visualization but also aid in modular calculations.
5	Functions that visualize or report data of employee
6	The function that queries the data from database to the website should be able to accurately get the correct result not only for displaying but for other modules calculation as well.
7	The fingerprint token of each employee should record correctly as set up in the database.
8	To track the employees' time and attendance, it is important that their fingerprint data is collected correctly with little to no errors.
9	For employee's data to not be missed or not recorded, all the important field are required.
10	The authentication functions must work 90% of the time in order for people to access their information.
11	XSS and URL exploitation are serious and pose significant threat to the data and privacy of the website, hence, this feature must be thoroughly tested.
12	All the input fields of the website must be able to deal with wrongful data to prevent SQL injections and other breaches.
13	Data transmission from one layer (front, back, database) must be checked and ensure the accuracy of data.
14	Should error arise when using the website, error handling prompt must pop up for users or developer to fix the issue.
15	Ensure all the link to different function points exactly to the specified function.

16	Integration of other services should be available most of the time and should only be down (intentionally or unintentionally) when traffic is low.
17	To ensure smooth operation on different choice of browsers, this website and its functionalities should be tested on various browsers.

Figure 3.1: Functionality correctness checklist

By checking the quality checklist after the sprint or product is done, we can have a total view of the progress of our project, enabling us to make more informed decisions and improving the product as needed.

## **TASK 072D: Quality Planning**

### **HRM WEBSITE QUALITY CHECKLIST**

<b>Functional Suitability – Functional Correctness</b>		
No.	Metrics	Thresholds
1	Individually, each calculation module (e.g., payroll, time worked, rating) should have little to no calculation errors.	< 5% calculation error
2	When a calculation is made in any one of the modules, the related attribute in other modules must update correctly.	< 5% calculation error
3	Employee's information updates should correctly appear on the screen.	100% accuracy
4	Employee's information should be recorded correctly within the database	100% accuracy
5	The dashboard chart should be used properly to show clear details of employee's information.	Clear visualization with < 5% deviation
6	The database searching and filtering options should return the correct result.	95% accuracy
7	Biometrics token must be correctly verified and apply for the correct personnel	95% accuracy
8	Data conversion of the fingerprint to be stored in the database must be accurate for each of the employees.	< 5% fingerprint data conversion error

9	During the account registration process, important fields such as name, password, and email are required.	100% of the information is fulfilled
10	Most authentication attempts must be successful	90% success rate
11	The website must pass many test cases that prevent XSS and exploiting URLs.	98% test case passed
12	Proper input sanitization should be checked to prevent code injection or unintended characters. Additionally, handling of special characters, whitespace, and input encoding should be validated.	100% input fields are protected
13	Form submission and data transmission should be tested to ensure that data is correctly processed and sent.	95% test case passed
14	Error messages should be displayed when invalid data is entered, providing clear instructions on how to correct them.	100% accuracy
15	Links to different functionalities should be correct.	100% accuracy
16	API connection to other services (i.e., existing ERP, Google sign in, Google sheet, etc.) should have little error	< 5% API errors
17	The website should display and run as expected on various browsers.	98% of browser tested must passed

## Quality Control Planning

To ensure that our product meets the expected quality, there needs to be a plan of action for each of the quality. Hence, we will be using two methods namely Risk Assessment and Eisenhower Matrix to justify and devise an in-depth plan.

### **1. Ensuring Product meets the Quality defined**

To ensure that each quality stay within the predetermined quality aspect and don't exceed the threshold, they are paired with a matching task

#### Functional Suitability – Functional Correctness

No.	Plan

1	Insert 50 rows of employees with varying degrees of different parameters to observe whether the calculation function is correctly configured.
2	Use a wide range of values for the employee parameters on individual modules and observe whether the final calculation is correct.
3	Verify the displayed information on the website against the database
4	Make 20 employees updates on the website and observe their data in the database
5	Use a wide range of value on the employees' parameters and observe whether the chart function is correctly configured
6	Use the search function with a variety of keywords and random words to note the displayed information.
7	Create 20 samples of fingerprint and create 20 employee accounts according to the prints, then observe it in the database
8	Create 25 fingerprints on the reader and observe whether the website made a mistake when transferring fingerprint data to the database
9	Check whether all input field are programmed with the "required" tag in the HTML code
10	Create 20 employees accounts with different emails to log into the system
11	Create 20 test cases that simulate XSS, or URLs exploiting and test the website against them
12	Test all input field on the website against 20 different test cases of varying error
13	Test form submission input on the website against 20 different test cases and note any difference between the website and the database
14	Wrongly execute any input and observe whether error messages appear
15	Click on all the tabs presented on each page of the website and verify them against the sitemap
16	Intentionally disrupt network connectivity during an ongoing API request and verify that the system handles the timeout or connection failure gracefully
17	Run the HRM website on Opera, Chrome, Coc Coc, Firefox, Brave, and Edge, and note any differences or issues.

*Figure 1.1: Plan for each quality*

## 2. Risk Assessment

We can rank the most significant attributes while considering the associated risks by using the quality checklist to ascertain whether each quality meets any risk criteria. These risks will be ranked from least to highest influence on business functions, with a distinction made between risks with less impact, like browser compatibility, and those with greater impact, like SQL injection.

As part of this categorization process, assigning each attribute to the relevant risk category lays the groundwork for action planning. There are three different priority levels: Normal, Important, and Urgent. Urgent tasks are prioritized, followed by Important tasks, and finally Normal ones.

We can make the following risk assessment chart by following the item numbers on the checklist as a reference:

		Severity				
		Negligible	Minor	Moderate	Significant	Severe
Likelihood	Very Likely		9, 14	3		
	Likely		1, 15	6	16	7, 12
	Possible	17	5	13	2, 10	11
	Unlikely		8			4
	Very Unlikely					

*Figure 2.1: Risk assessment chart*

### 3. Planning Quality Review

We can efficiently design and plan tasks to maintain the quality of the HRM website thanks to the risk assessment chart. Then we can use Trello dashboard to help with the management and organizing of all required actions. With the aid of this dashboard, we will be able to organize and classify jobs in a methodical manner according to the Risk Assessment Chart, guaranteeing thorough monitoring and efficient implementation of quality assurance procedures for the HRM website.

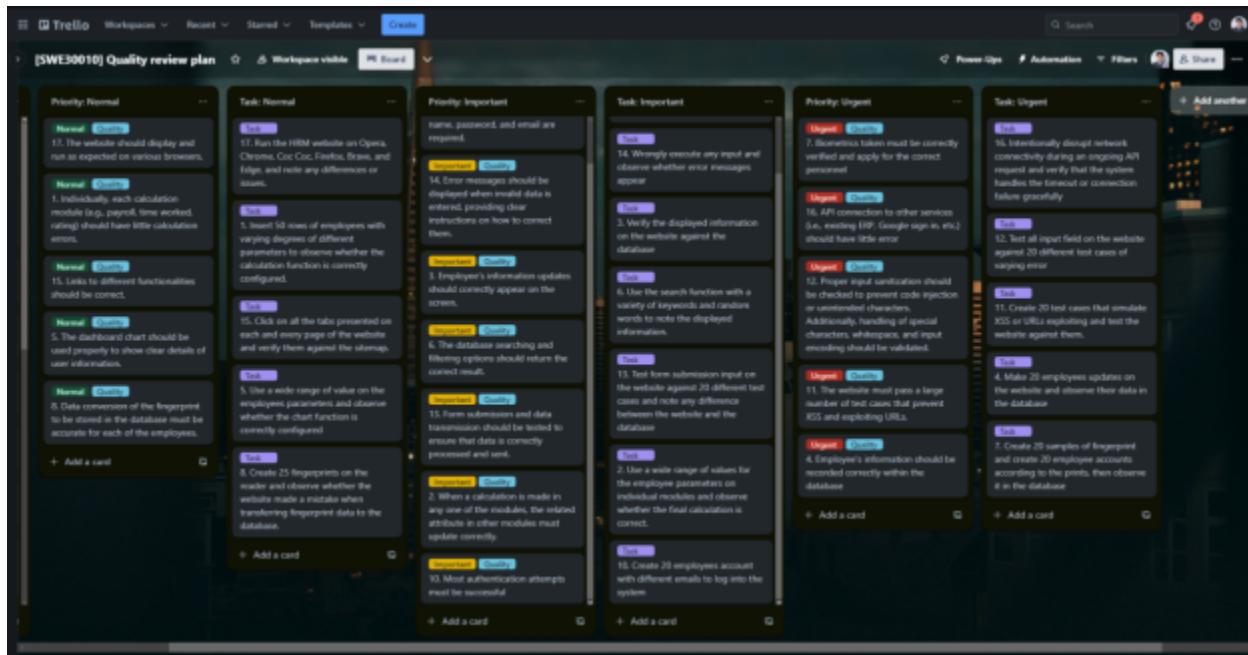


Figure 3.1: Quality Review Plan Trello board

To improve clarity, we will group tasks and qualities according to the color of the risk assessment into three separate groups, as mentioned in the section above.

Green-colored tasks and attributes will be categorized as Normal. Yellow and Orange range represent tasks that are more than minor but still lesser than severe; these tasks are labeled Important. Severe tasks and qualities will be categorized as Urgent and will don the color red.

## TASK 073D: Quality Review

### Conducting Quality Control Measure

#### 1. Implemented Measures

After defining the quality that our product must have for the first sprint, we performed those tasks and observe their progress via Trello, some of the evidence is also collected to determine our quality performance and progress.

We separated the testing into different groups that have similar component for ease of backtracking and review and during this period, we have tested 3 groups of tests that has been done.

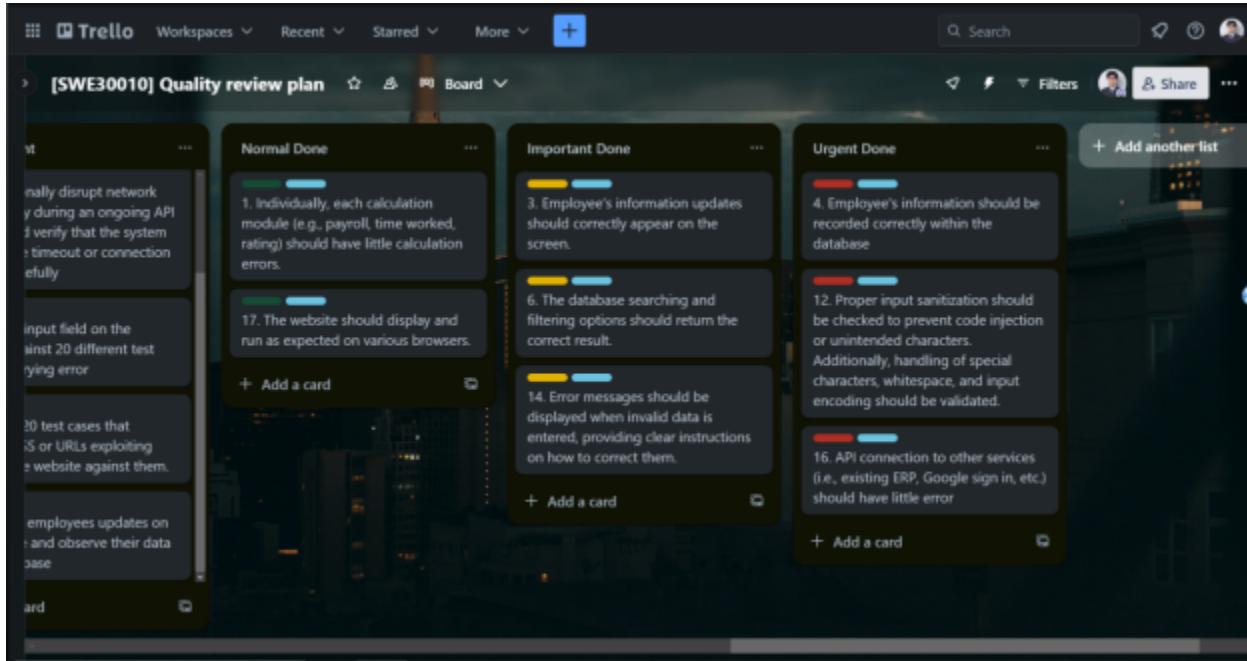


Figure 1.1: Plan for each quality

## 2. Evidence & Report

### Evidence

#### Group 1: Display and record

**Quality 1:** Individually, each calculation module (e.g., payroll, time worked, rating) should have little calculation errors.

**Testing measure:** Insert 50 rows of employees with varying degrees of different parameters to observe whether the calculation function is correctly configured.

## Original Data

A	B	C	D	E	F	G	H	I	J	K	L
1	<b>id</b>	<b>email</b>	<b>base_salary</b>	<b>total_work_hour</b>	<b>bonus</b>	<b>actual_pay</b>					
2	1	mloyna0@iu.qd	\$21,00	7:30:00	\$23,00	\$180,50					
3	2	cstoneham1@timbio.com	\$27,00	1886:00:00	\$79,00	\$51,271,00					
4	3	scannop2@vho.int	\$21,00	480:00:00	\$23,00	\$10,103,00					
5	4	inser3@businesswire.com	\$23,00	7:30:00	\$20,00	\$192,50					
6	5	bsimmers4@glitzmodo.com	\$30,00	12:30:00	\$10,00	\$385,00					
7	6	rhandmar5@amazon.com	\$23,00	28:03:00	\$0,00	\$645,15					
8	7	hdreger6@unc.edu	\$23,00	11:11:00	\$30,00	\$287,22					
9	8	meversen7@reference.com	\$27,00	24:25:00	\$50,00	\$709,25					
10	9	bgnotech8@webx.com	\$23,00	26:23:00	\$50,00	\$696,82					
11	10	babramson9@fcq.com	\$25,00	29:21:00	\$20,00	\$753,75					
12	11	jenylikaa@google.ru	\$24,00	17:46:00		\$428,40					
13	12	kwheewilb@wired.com	\$23,00	27:49:00	\$30,00	\$609,78					
14	13	mfrancyc@elegantthemes.com	\$26,00	20:44:00	\$30,00	\$569,07					
15	14	wcannawayd@wired.com	\$23,00	30:56:00		\$711,47					
16	15	golytee@usatoday.com	\$27,00	10:56:00		\$295,20					
17	16	finehamf@ed.gov	\$26,00	6:33:00		\$170,30					
18	17	lismong@baidu.com	\$26,00	23:42:00	\$20,00	\$636,20					
19	18	slairyh@cnet.com	\$29,00	11:23:00	\$70,00	\$400,12					
20	19	hlahosi@pen.io	\$23,00	8:21:00		\$192,05					
21	20	ngoodallj@tiny.cc	\$27,00	48:08:00		\$1,299,60					
22	21	lrenk@weibo.com	\$22,00	14:53:00		\$327,43					
23	22	rhawkinswoodl@pagekniggle.it	\$27,00	0:00:00		\$0,00					
24	23	wwalickm@spotify.com	\$30,00	3:59:00		\$115,00					
25	24	aplenderleithn@unesco.org	\$21,00	0:00:00		\$0,00					
26	25	spoolo@buzzfeed.com	\$23,00	18:48:00		\$431,63					
27	26	pbuyfieldp@blogger.com	\$26,00	23:30:00		\$611,00					
28	27	alekeyolly@senate.gov	\$24,00	0:00:00	\$7,00	\$7,00					
29	28	hpaitone@neea.com	\$30,00	8:05:00		\$272,50					
30	29	dsternets@csmonitor.com	\$23,00	11:55:00		\$272,17					
31	30	smoglymt@lulu.com	\$26,00	11:42:00		\$304,20					
32	31	kmcarthuru@discuz.net	\$29,00	12:44:00		\$368,27					
33	32	rstandingford@bing.com	\$30,00	13:39:00		\$409,50					
34	33	tkumenekw@umn.edu	\$29,00	9:12:00		\$266,00					
35	34	gianmuzia@webnode.com	\$28,00	35:23:00		\$990,73					
36	35	toonachyy@stanford.edu	\$25,00	6:29:00		\$162,08					
37	36	tsaggenez@hi360.com	\$24,00	0:00:00		\$0,00					
38	37	anorthocote10@del.com	\$23,00	35:24:00		\$814,20					
39	38	ramtorius11@nymag.com	\$22,00	4:56:00		\$108,53					
40	39	kswyersewey12@t.co	\$24,00	17:52:00		\$428,80					
41	40	cgoodlight13@bing.com	\$29,00	9:16:00		\$298,73					
42	41	rlfwdry14@dm.com	\$26,00	2:03:00		\$83,30					
43	42	krays15@vk.com	\$25,00	19:53:00		\$497,08					
44	43	jmorten16@fp.com	\$30,00	34:42:00		\$1,041,00					
45	44	krolling17@hugedomains.com	\$26,00	13:33:00		\$352,30					
46	45	pmerriment18@ucsd.edu	\$25,00	4:09:00		\$103,75					
47	46	acorton19@cnet.com	\$30,00	4:53:00		\$146,50					
48	47	plidgerwood1a@nymag.com	\$28,00	16:33:00		\$463,40					
49	48	ppanyer1b@tuttocitta.it	\$24,00	0:00:00		\$0,00					
50	49	dsimmgenc1@gnu.org	\$25,00	14:27:00		\$361,25					
51	50	smckeeever1d@example.com	\$24,00	19:53:00		\$477,20					
52	51	tgrinvalds1e@github.com	\$21,00	13:23:00		\$281,05					

Figure 2.1, 2.2, 2.3: 50 Original Rows

## After testing

The screenshot shows a Microsoft Excel spreadsheet with the following structure:

	A	B	C	D	E	F	G	H	I	J	K	L
1	id	email	base_salary	total_work_hour	bonus	actual_pay						
2	1	mloym0@ui.edu	\$21,00	7:30:00	\$128,00	\$285,50						
3	2	cstoneham1@zimble.com	\$27,00	1896:00:00	\$41,00	\$51,233,00						
4	3	scannop2@nho.int	\$21,00	480:00:00	\$93,00	\$10,173,00						
5	4	iron13@businesswire.com	\$23,00	7:30:00	\$12,00	\$184,50						
6	5	bsimmers4@jigmodo.com	\$30,00	12:30:00	\$77,00	\$462,00						
7	6	nhandman5@amacon.com	\$23,00	28:03:00	\$135,00	\$780,15						
8	7	hdroger6@unc.edu	\$23,00	11:11:00	\$3,00	\$260,22						
9	8	meverson7@reference.com	\$27,00	24:25:00	\$109,00	\$768,25						
10	9	hgotediffe8@webs.com	\$23,00	26:23:00	\$56,00	\$662,82						
11	10	babramor9@icq.com	\$25,00	29:21:00	\$72,00	\$605,75						
12	11	jwylea@google.ru	\$24,00	17:46:00	\$24,00	\$450,40						
13	12	kehewillb@wired.com	\$23,00	27:49:00	\$147,00	\$788,78						
14	13	mfranc13@elgarrithemes.com	\$26,00	20:44:00	\$8,00	\$547,07						
15	14	wcannawayd@wired.com	\$23,00	30:56:00	\$63,00	\$774,47						
16	15	gdyteef@usatoday.com	\$27,00	10:56:00	\$101,00	\$398,20						
17	16	flinehamf@ed.gov	\$26,00	6:33:00	\$40,00	\$210,30						
18	17	hismong@baidu.com	\$26,00	23:42:00	\$86,00	\$702,20						
19	18	slairyh@cnet.com	\$29,00	11:23:00	\$132,00	\$462,12						
20	19	Mahosi@pen.io	\$23,00	8:21:00	\$19,00	\$211,05						
21	20	ngsodulj@tinycc	\$27,00	48:08:00	\$69,00	\$1,386,60						
22	21	lrenk@weibo.com	\$22,00	14:53:00	\$113,00	\$327,43						
23	22	rhawkwoodl@pageagainst.it	\$27,00	0:00:00	\$9,00	\$9,00						
24	23	wwolickim@sportify.com	\$30,00	3:59:00	\$129,00	\$244,00						
25	24	aplenderleithn@unesco.org	\$21,00	0:00:00	\$33,00	\$33,00						
26	25	spoolo@buzzfeed.com	\$23,00	18:46:00	\$121,00	\$552,63						
27	26	pbyfieldp@blogger.com	\$26,00	23:30:00	\$67,00	\$678,00						
28	27	aleyleyj@senate.gov	\$24,00	0:00:00	\$45,00	\$45,00						
29	28	hpaltorr@xrea.com	\$30,00	9:05:00	\$104,00	\$376,50						
30	29	dinternets@comonitor.com	\$23,00	11:56:00	\$17,00	\$289,17						
31	30	smcglynn@lulu.com	\$26,00	11:42:00	\$81,00	\$385,20						
32	31	kmcarthuru@discuz.net	\$29,00	12:44:00	\$139,00	\$508,27						
33	32	ntstandingford@bing.com	\$30,00	13:39:00	\$23,00	\$432,50						
34	33	tkumekew@umn.edu	\$29,00	9:12:00	\$73,00	\$339,80						
35	34	giannuzzi@webnode.com	\$28,00	35:23:00	\$111,00	\$1,101,73						
36	35	toonachyy@stanford.edu	\$25,00	6:29:00	\$68,00	\$220,08						
37	36	tsaggerz@i360.com	\$24,00	0:00:00	\$62,00	\$62,00						
38	37	anorthocote10@dell.com	\$23,00	35:24:00	\$36,00	\$850,20						
39	38	ramtonies11@nymag.com	\$22,00	4:56:00	\$125,00	\$233,53						
40	39	kwayerusoy12@t.co	\$24,00	17:52:00	\$62,00	\$480,00						
41	40	cgoodlight13@bing.com	\$29,00	9:16:00	\$76,00	\$344,73						
42	41	rlawdry14@bm.com	\$26,00	2:03:00	\$142,00	\$195,30						
43	42	lrays15@vk.com	\$25,00	18:53:00	\$29,00	\$526,08						
44	43	jmorten16@hp.com	\$30,00	34:42:00	\$87,00	\$1,138,00						
45	44	krolling17@hugedomains.com	\$26,00	13:33:00	\$48,00	\$400,30						
46	45	pmerriment18@cscl.edu	\$25,00	4:09:00	\$114,00	\$217,75						
47	46	aconton19@cnr.com	\$30,00	4:53:00	\$61,00	\$207,50						
48	47	pldgertwood1@nymag.com	\$26,00	16:33:00	\$64,00	\$547,40						
49	48	ppanyer1b@tuttocitta.it	\$24,00	0:00:00	\$137,00	\$137,00						
50	49	dairmoran1c@gru.org	\$25,00	14:27:00	\$6,00	\$367,25						
51	50	smckeever1d@example.com	\$24,00	19:53:00	\$103,00	\$680,20						

Figure 2.1, 2.2, 2.3: 50 Post-tested Rows

We employed our database as our calculation module as well, we used the following formula to calculate our employee's pay:

$$=((LEFT(D2 ; FIND(":"; D2) - 1))+(MINUTE(D2)/60))*C2)+E2$$

After testing, all the rows have been calculated correctly and this quality passed our defined threshold.

**Quality 3:** Employee's information updates should correctly appear on the screen.

**Testing measure:** Verify the displayed information on the website against the database.

Employee Information		Payroll Management		Overall Tracking		
ID	Employee Email	Salary (per hour)	WeeklyHours	Bonus	Monthly Salary	Status
1	hiyevet4649@gmail.com	\$21.00	7.30 00	\$108.00	\$289.54	Suspending
2	cotoneham1@jambis.com	\$27.00	1000.00 00	\$41.00	\$51.233.00	Suspending
3	scammp2@vnsr.mt	\$21.00	400.00 00	\$83.00	\$10.172.00	Suspending
4	monk1@businessonline.com	\$21.00	7.30 00	\$12.00	\$184.54	Suspending
5	tormenters4@glanmeds.com	\$30.00	12.00 00	\$77.00	\$452.00	Suspending
6	rhardman5@seacon.com	\$23.00	28.00 00	\$135.00	\$758.15	Suspending
7	tdinger6@unc.edu	\$23.00	11.11 00	\$13.00	\$266.22	Suspending
8	miverson7@reference.com	\$27.00	24.25 00	\$100.00	\$768.25	Suspending
9	tpatriciu8@vrtx.com	\$23.00	28.23 00	\$96.00	\$892.82	Suspending
10	sabramank9@icq.com	\$25.00	29.21 00	\$72.00	\$805.75	Suspending
11	jnyhead@people.ru	\$24.00	17.48 00	\$24.00	\$458.48	Suspending
12	kukhavitz10@ured.com	\$23.00	27.48 00	\$147.00	\$788.78	Suspending
13	mfayci11@elegantthemes.com	\$26.00	29.44 00	\$8.00	\$547.07	Suspending
14	vorammary12@med.com	\$23.00	39.58 00	\$83.00	\$714.47	Suspending
15	gjytee13@usatoday.com	\$27.00	19.58 00	\$101.00	\$396.28	Suspending
16	fmehami14@ed.gov	\$26.00	6.33 00	\$40.00	\$218.38	Suspending
17	tsimong15@nrau.com	\$26.00	23.42 00	\$86.00	\$702.28	Suspending
18	staylor16@nat.com	\$26.00	11.23 00	\$132.00	\$462.12	Suspending
19	lvlehsu17@enr.io	\$23.00	8.21 00	\$19.00	\$211.05	Suspending
20	ngoodwill18@try.cc	\$27.00	48.00 00	\$89.00	\$1.368.80	Suspending
21	hene19@vete.com	\$22.00	14.53 00	\$113.00	\$327.43	Suspending
22	rhawkins10@pageantplate.it	\$27.00	0.00 00	\$9.00	\$9.00	Suspending
23	uvatocan20@putty.com	\$30.00	3.92 00	\$129.00	\$344.08	Suspending
24	aplaider11@unesco.org	\$21.00	0.00 00	\$33.00	\$33.00	Suspending
25	zpolo22@buzzfeed.com	\$23.00	18.46 00	\$121.00	\$552.63	Suspending
26	pbeyfield23@megger.com	\$25.00	23.30 00	\$87.00	\$570.09	Suspending
27	alexczef12@renault.gov	\$24.00	0.00 00	\$45.00	\$45.00	Suspending
28	npastore24@viva.com	\$30.00	9.05 00	\$104.00	\$376.50	Suspending
29	datamwata25@ccmonitor.com	\$23.00	11.53 00	\$17.00	\$285.17	Suspending
30	smcglynn26@luu.com	\$25.00	11.42 00	\$81.00	\$385.29	Suspending
31	lmcathrina27@dsouz.net	\$29.00	12.44 00	\$130.00	\$586.27	Suspending
32	rolandinefondy28@big.com	\$30.00	13.39 00	\$23.00	\$432.59	Suspending
33	transeteve29@um.edu	\$29.00	9.12 00	\$73.00	\$339.89	Suspending
34	qammuzi30@investrade.com	\$28.00	16.23 00	\$111.00	\$1.181.73	Suspending
35	lcanachiy31@stanford.edu	\$25.00	6.29 00	\$50.00	\$220.00	Suspending
36	traggemz32@hc360.com	\$24.00	0.00 00	\$82.00	\$82.00	Suspending
37	anorthole10@qtel.com	\$23.00	38.24 00	\$35.00	\$889.29	Suspending
38	rantonewitt11@nymag.com	\$22.00	4.56 00	\$125.00	\$333.53	Suspending
39	louisjeanney12@cc	\$24.00	17.52 00	\$52.00	\$880.00	Suspending
40	cpaodriguez13@jpm.com	\$26.00	9.19 00	\$70.00	\$344.73	Suspending
41	rhendry14@ibm.com	\$26.00	2.03 00	\$142.00	\$185.30	Suspending
42	hrer15@itk.com	\$25.00	18.53 00	\$29.00	\$525.00	Suspending
43	jharter16@tpc.com	\$18.00	34.42 00	\$87.00	\$1.138.00	Suspending
44	krating17@nugedonates.com	\$26.00	13.33 00	\$48.00	\$880.30	Suspending
45	pmclementi18@cod.edu	\$25.00	4.09 00	\$114.00	\$217.75	Suspending
46	acerton19@cnet.com	\$30.00	4.53 00	\$81.00	\$387.50	Suspending
47	plidgewood1a@nymag.com	\$28.00	18.33 00	\$84.00	\$547.49	Suspending
48	peanure1b@utscotta.it	\$24.00	0.00 00	\$137.00	\$137.00	Suspending
49	osmoeapar1c@nwu.org	\$25.00	14.27 00	\$6.00	\$187.25	Suspending
50	smckeever1d@example.com	\$24.00	19.53 00	\$103.00	\$580.20	Suspending

Figure 2.4, 2.5, 2.6: Displaying of payroll on website

With the database's pay calculated, we can take a look at the website to see how it displayed the data, and as can be seen, all the 50 rows have been updated correctly.

Employees' information has also been correctly displayed in the employee's management tab:

A1	B	C	D	E	F	G	H	I	J
1	id	first_name	last_name	email	gender	phone	address	department	position
2	1	Dr	Loyns	mloyns0@gs.gd	Male	485-990-3342	260 Fulton Trail	Support	Senior Financial Analyst
3	2	Ms	Stoneham	cstoneham1@zimbio.com	Female	873-386-1381	61188 Cambridge Crossing	Human Resources	Mechanical Systems Engineer
4	3	Ms	Cannop	scannop2@who.int	Female	957-805-7791	167 Emmet Alley	Human Resources	Analog Circuit Design manager
5	4	Dr	Noni	inon3@businesswire.com	Male	842-922-2795	3342 Peterle Avenue	Marketing	Biostatistician III
6	5	Mrs	Simmers	bsimmers4@gizmodo.com	Male	437-233-7823	6 Fremont Point	Training	Accountant I
7	6	Honorable	Handman	rhandman5@amazon.com	Female	225-771-4775	511 American Ash Hill	Services	Nurse Practitioner
8	7	Ms	Dreger	hdreger6@unc.edu	Male	675-254-5559	49830 Pennsylvania Terrace	Product Management	Senior Sales Associate
9	8	Ms	Everson	meverson7@reference.com	Male	635-948-9337	1 Lighthouse Bay Road	Sales	Assistant Professor

Figure 2.7: Employee's information in Database

Employee Information		Payroll Management		Overall Tracking	
<div style="border: 1px solid #ccc; padding: 10px; border-radius: 10px;"> <div style="display: flex; justify-content: space-between;"> <span><input type="text"/> Search employee by Name</span> <span>Add User</span> </div> <div style="display: grid; grid-template-columns: 1fr 1fr 1fr 1fr; gap: 10px;"> <div style="border: 1px solid #ccc; padding: 5px; border-radius: 10px; text-align: center;">  <p><b>Dr Loyns</b> Senior Financial Analyst</p> <p>Address: 260 Fulton Trail Phone: 485-990-3342 Email: mloyns0@gs.gd</p> <p><a href="#">View Profile</a></p> </div> <div style="border: 1px solid #ccc; padding: 5px; border-radius: 10px; text-align: center;">  <p><b>Ms Stoneham</b> Mechanical Systems Engineer</p> <p>Address: 61188 Cambridge Crossing Phone: 873-386-1381 Email: cstoneham1@zimbio.com</p> <p><a href="#">View Profile</a></p> </div> <div style="border: 1px solid #ccc; padding: 5px; border-radius: 10px; text-align: center;">  <p><b>Ms Cannop</b> Analog Circuit Design manager</p> <p>Address: 167 Emmet Alley Phone: 957-805-7791 Email: scannop2@who.int</p> <p><a href="#">View Profile</a></p> </div> <div style="border: 1px solid #ccc; padding: 5px; border-radius: 10px; text-align: center;">  <p><b>Dr Noni</b> Biostatistician III</p> <p>Address: 3342 Peterle Avenue Phone: 842-922-2795 Email: inon3@businesswire.com</p> <p><a href="#">View Profile</a></p> </div> <div style="border: 1px solid #ccc; padding: 5px; border-radius: 10px; text-align: center;">  <p><b>Mrs Simmers</b> Accountant I</p> <p>Address: 6 Fremont Point Phone: 437-233-7823 Email: bsimmers4@gizmodo.com</p> <p><a href="#">View Profile</a></p> </div> <div style="border: 1px solid #ccc; padding: 5px; border-radius: 10px; text-align: center;">  <p><b>Honorable Handman</b> Nurse Practitioner</p> <p>Address: 511 American Ash Hill Phone: 225-771-4775 Email: rhandman5@amazon.com</p> <p><a href="#">View Profile</a></p> </div> <div style="border: 1px solid #ccc; padding: 5px; border-radius: 10px; text-align: center;">  <p><b>Ms Dreger</b> Senior Sales Associate</p> <p>Address: 49830 Pennsylvania Terrace Phone: 675-254-5559 Email: hdreger6@unc.edu</p> <p><a href="#">View Profile</a></p> </div> <div style="border: 1px solid #ccc; padding: 5px; border-radius: 10px; text-align: center;">  <p><b>Ms Everson</b> Assistant Professor</p> <p>Address: 1 Lighthouse Bay Road Phone: 635-948-9337 Email: meverson7@reference.com</p> <p><a href="#">View Profile</a></p> </div> </div> </div>					
Showing 0 to 0 of 305 results					
<span>1</span> <span>2</span> <span>3</span> ... <span>39</span> <span>&gt;</span>					

Figure 2.8: Employee's information on Website

With the website displaying the correct information within the database, this quality passed our benchmark.

**Quality 4:** Employee's information should be recorded correctly within the database.

**Testing measure:** Make 20 employees updates on the website and observe their data in the database.

ID	Employee Email	Salary (per hour)	Weekly Hours	Status	Monthly Salary	Status
1	milymili@n.pt	\$21.00	7.00	<input checked="" type="checkbox"/>	\$147.00	Suspending
2	ctonetham@google.com	\$27.00	18.00	<input checked="" type="checkbox"/>	\$486.00	Suspending
3	zcmmpc2@yuno.int	\$21.00	48.00	<input checked="" type="checkbox"/>	\$912.00	Suspending
4	inom@businesswriters.com	\$21.00	7.00	<input checked="" type="checkbox"/>	\$147.00	Suspending
5	teamware@google.com	\$20.00	12.00	<input checked="" type="checkbox"/>	\$240.00	Suspending
6	manishant@amazon.com	\$21.00	20.00	<input checked="" type="checkbox"/>	\$420.00	Suspending
7	haleemah@unc.edu	\$21.00	15.00	<input checked="" type="checkbox"/>	\$315.00	Suspending
8	inawerlans@reference.com	\$27.00	24.00	<input checked="" type="checkbox"/>	\$648.00	Suspending
9	bigdata@fedex.com	\$23.00	28.00	<input checked="" type="checkbox"/>	\$644.00	Suspending
10	dataman@vix.com	\$26.00	29.00	<input checked="" type="checkbox"/>	\$754.00	Suspending
11	jerome@google.ru	\$24.00	17.00	<input checked="" type="checkbox"/>	\$408.00	Suspending
12	inawerlans@reference.com	\$23.00	27.00	<input checked="" type="checkbox"/>	\$621.00	Suspending
13	mherry@akagithemes.com	\$26.00	29.00	<input checked="" type="checkbox"/>	\$754.00	Suspending
14	worrenivas@wired.com	\$21.00	38.00	<input checked="" type="checkbox"/>	\$798.00	Suspending
15	phflee@crates.com	\$27.00	18.00	<input checked="" type="checkbox"/>	\$474.00	Suspending
16	finsham@ed.gov	\$26.00	8.00	<input checked="" type="checkbox"/>	\$208.00	Suspending
17	timong@edu.com	\$26.00	22.00	<input checked="" type="checkbox"/>	\$572.00	Suspending
18	stonyh@com.com	\$26.00	11.00	<input checked="" type="checkbox"/>	\$286.00	Suspending
19	natone@ppen.ie	\$23.00	8.00	<input checked="" type="checkbox"/>	\$184.00	Suspending
20	ngoodall@tiny.cc	\$27.00	48.00	<input checked="" type="checkbox"/>	\$1272.00	Suspending

Figure 2.9: First 20 employees

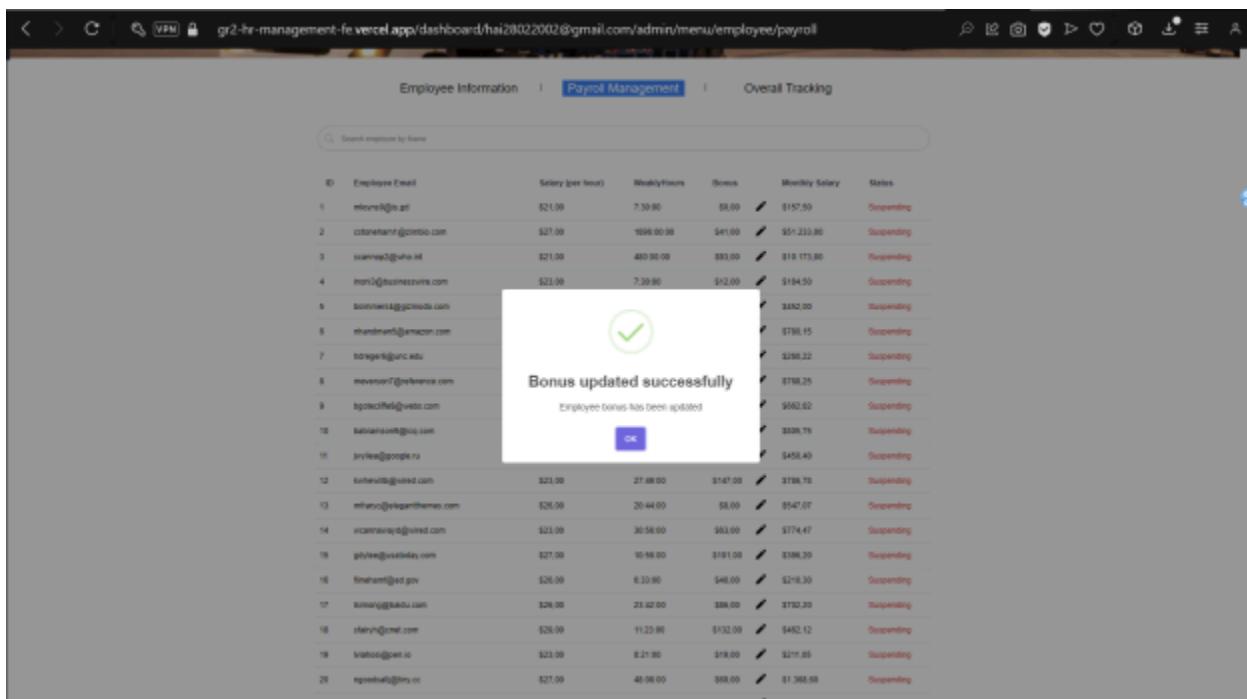
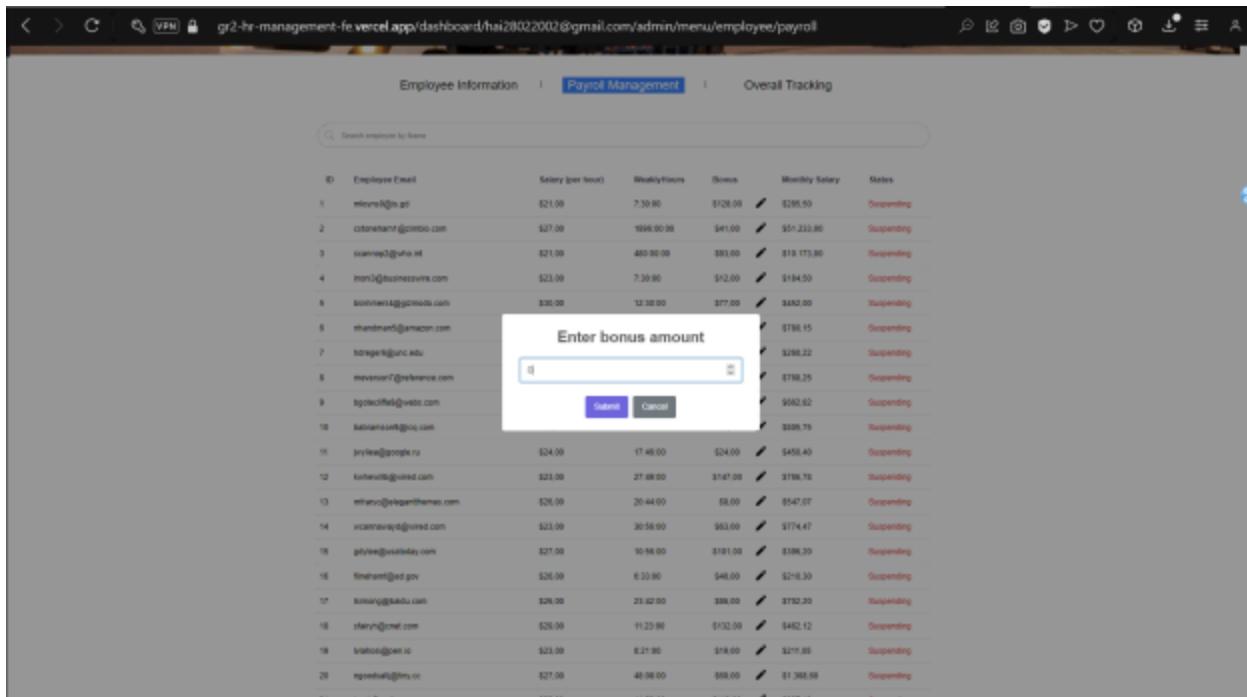


Figure 2.10, 2.11: Update, Confirmation Prompt and Amount

ID	Employee Email	Salary (per hour)	Weekly Hours	Bonus	Monthly Salary	Status
1	mroyns0@in.gd	\$21,00	7.30:00	\$0,00	\$157,50	Suspending
2	ctstoneham1@zimbio.com	\$27,00	1896.00:00	\$0,00	\$51.192,00	Suspending
3	scannop2@who.int	\$21,00	480.00:00	\$0,00	\$10.080,00	Suspending
4	inon3@businessline.com	\$23,00	7.30:00	\$0,00	\$172,50	Suspending
5	bsimmer4@joomla.com	\$30,00	12:30:00	\$0,00	\$375,00	Suspending
6	nhandman5@amazon.com	\$23,00	28.03:00	\$0,00	\$645,15	Suspending
7	hdreger6@unc.edu	\$23,00	11:11:00	\$0,00	\$257,22	Suspending
8	mwerson7@reference.com	\$27,00	24:25:00	\$0,00	\$609,25	Suspending
9	lgottediff8@webs.com	\$23,00	26:23:00	\$0,00	\$606,82	Suspending
10	babramson9@lcaq.com	\$25,00	29:21:00	\$0,00	\$733,75	Suspending
11	jwylea@google.ru	\$24,00	17:46:00	\$0,00	\$426,40	Suspending
12	kschewillb@wired.com	\$23,00	27:49:00	\$0,00	\$639,78	Suspending
13	mfraryc@elegantthemes.com	\$26,00	20:44:00	\$0,00	\$639,07	Suspending
14	wcarinawyd@wired.com	\$23,00	30:56:00	\$0,00	\$711,47	Suspending
15	gbytree@usatoday.com	\$27,00	10:56:00	\$0,00	\$295,20	Suspending
16	flnehamf@ed.gov	\$26,00	6:33:00	\$0,00	\$170,30	Suspending
17	himong@baidu.com	\$26,00	23:42:00	\$0,00	\$616,20	Suspending
18	sliriyh@net.com	\$29,00	11:23:00	\$0,00	\$330,12	Suspending
19	hlahosi@pen.io	\$23,00	8:21:00	\$0,00	\$192,05	Suspending
20	ngoodsall@tiny.cc	\$27,00	48:08:00	\$0,00	\$1.299,60	Suspending

Figure 2.12: After update on Website

A	B	C	D	E	F	G	H	I	J	K	L
1	id	email	base_salary	total_work_hour	bonus	actual_pay					
2	1	mroyns0@in.gd	\$21,00	7.30:00	\$0,00	\$157,50					
3	2	ctstoneham1@zimbio.com	\$27,00	1896.00:00	\$0,00	\$51.192,00					
4	3	scannop2@who.int	\$21,00	480.00:00	\$0,00	\$10.080,00					
5	4	inon3@businessline.com	\$23,00	7.30:00	\$0,00	\$172,50					
6	5	bsimmer4@joomla.com	\$30,00	12:30:00	\$0,00	\$375,00					
7	6	nhandman5@amazon.com	\$23,00	28.03:00	\$0,00	\$645,15					
8	7	hdreger6@unc.edu	\$23,00	11:11:00	\$0,00	\$257,22					
9	8	mwerson7@reference.com	\$27,00	24:25:00	\$0,00	\$609,25					
10	9	lgottediff8@webs.com	\$23,00	26:23:00	\$0,00	\$606,82					
11	10	babramson9@lcaq.com	\$25,00	29:21:00	\$0,00	\$733,75					
12	11	jwylea@google.ru	\$24,00	17:46:00	\$0,00	\$426,40					
13	12	kschewillb@wired.com	\$23,00	27:49:00	\$0,00	\$639,78					
14	13	mfraryc@elegantthemes.com	\$26,00	20:44:00	\$0,00	\$639,07					
15	14	wcarinawyd@wired.com	\$23,00	30:56:00	\$0,00	\$711,47					
16	15	gbytree@usatoday.com	\$27,00	10:56:00	\$0,00	\$295,20					
17	16	flnehamf@ed.gov	\$26,00	6:33:00	\$0,00	\$170,30					
18	17	himong@baidu.com	\$26,00	23:42:00	\$0,00	\$616,20					
19	18	sliriyh@net.com	\$29,00	11:23:00	\$0,00	\$330,12					
20	19	hlahosi@pen.io	\$23,00	8:21:00	\$0,00	\$192,05					
21	20	ngoodsall@tiny.cc	\$27,00	48:08:00	\$0,00	\$1.299,60					

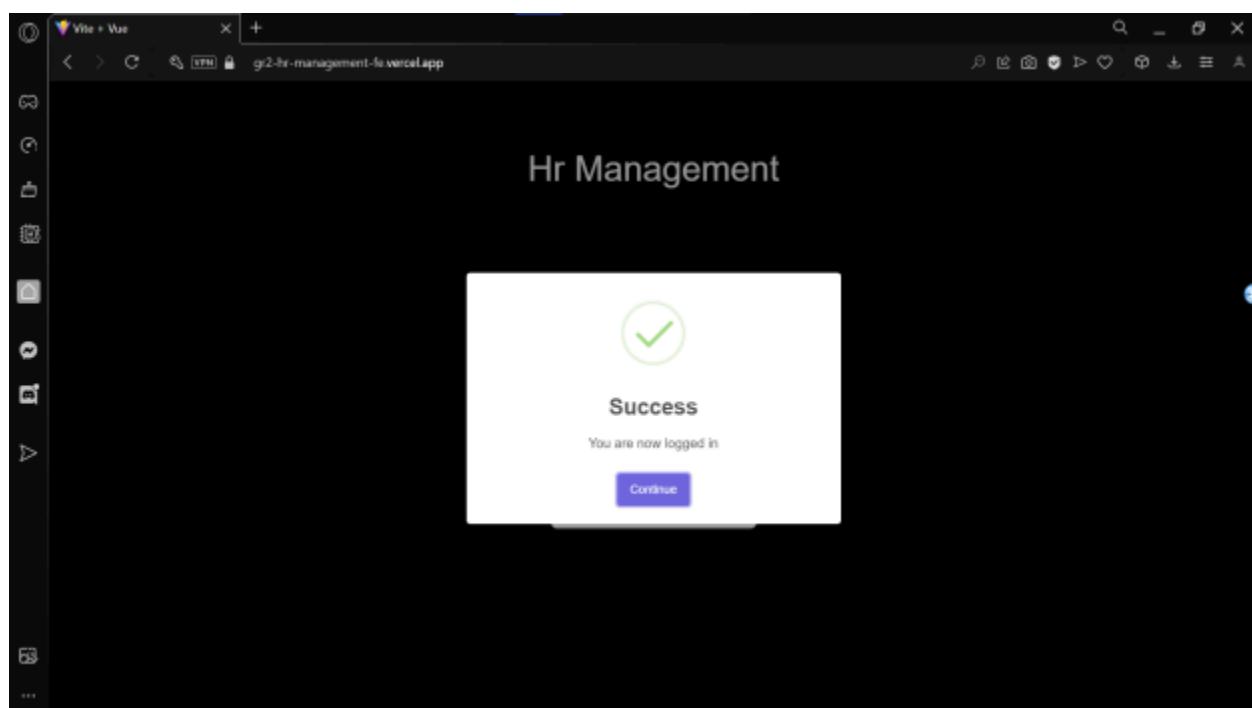
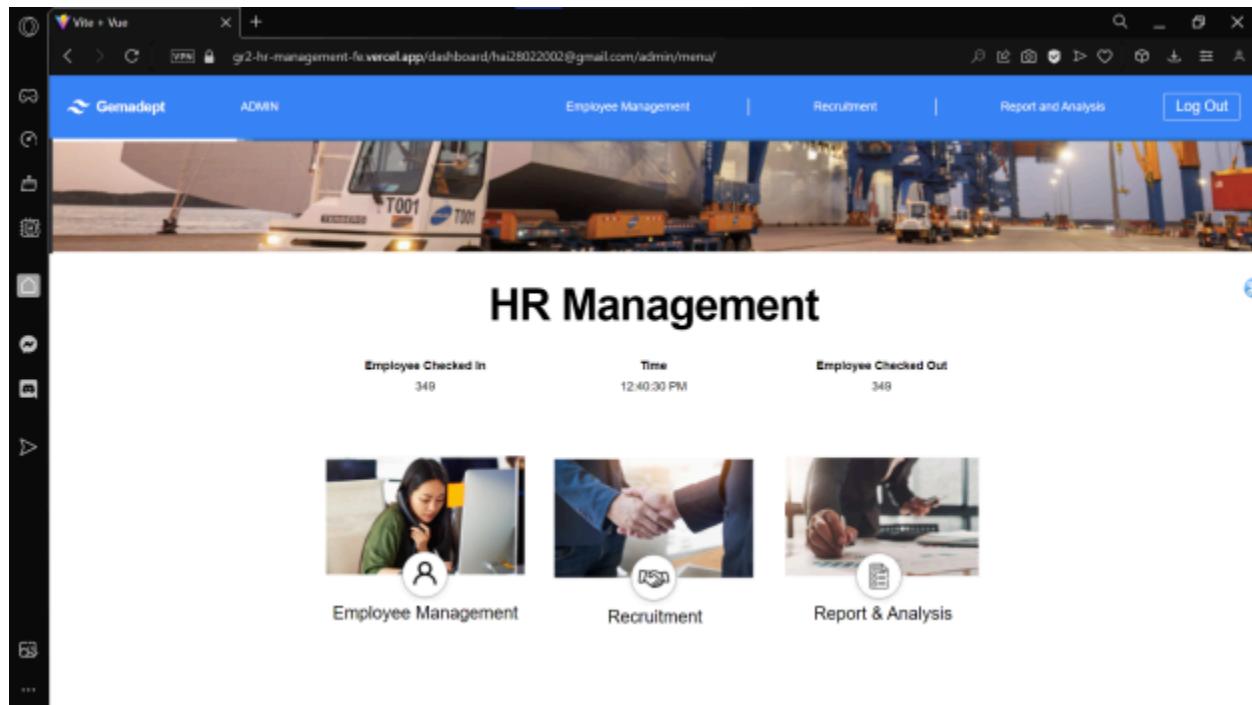
Figure 2.13: Information in Database

The information written on the website matches that of the database, hence, the accuracy is correct, and this quality passed our benchmark.

**Quality 17:** The website should display and run as expected on various browsers.

**Testing measure:** Run the HRM website on Opera, Chrome, Coc Coc, Firefox, Brave, and Edge, and note any differences or issues.

Opera:



The screenshot displays two consecutive pages from a web application, likely a dashboard for managing employees and payroll.

**Employee Management Page:**

- Header:** Shows the title "Gemadept" and "ADMIN".
- Top Navigation:** Includes links for "Employee Management", "Recruitment", "Report and Analysis", and "Log Out".
- Background Image:** A photograph of a port or industrial area with large ships and cranes.
- Section Headers:** "Employee Information", "Payroll Management", and "Overall Tracking".
- Search Bar:** A search input field labeled "Search employee by Name".
- User Profile Cards:** Four cards showing employee profiles:
  - Dr Loyns:** Senior Financial Analyst. Address: 250 Fallon Trail, Phone: 485-990-3342, Email: mloyns@is.gd. Buttons: "View Profile" and "Edit".
  - Ms Stoneham:** Mechanical Systems Engineer. Address: 61988 Cambridge Crossing, Phone: 875-386-1381, Email: cstoneham1@zimbo.com. Buttons: "View Profile" and "Edit".
  - Ms Cannop:** Analog Circuit Design Manager. Address: 157 Briar Alley, Phone: 957-856-7791, Email: scannop2@who.int. Buttons: "View Profile" and "Edit".
  - Dr Noni:** Biostatistician III. Address: 3342 Pecile Avenue, Phone: 842-952-2765, Email: noni3@businesswire.com. Buttons: "View Profile" and "Edit".
- Add User:** A button in the top right corner.

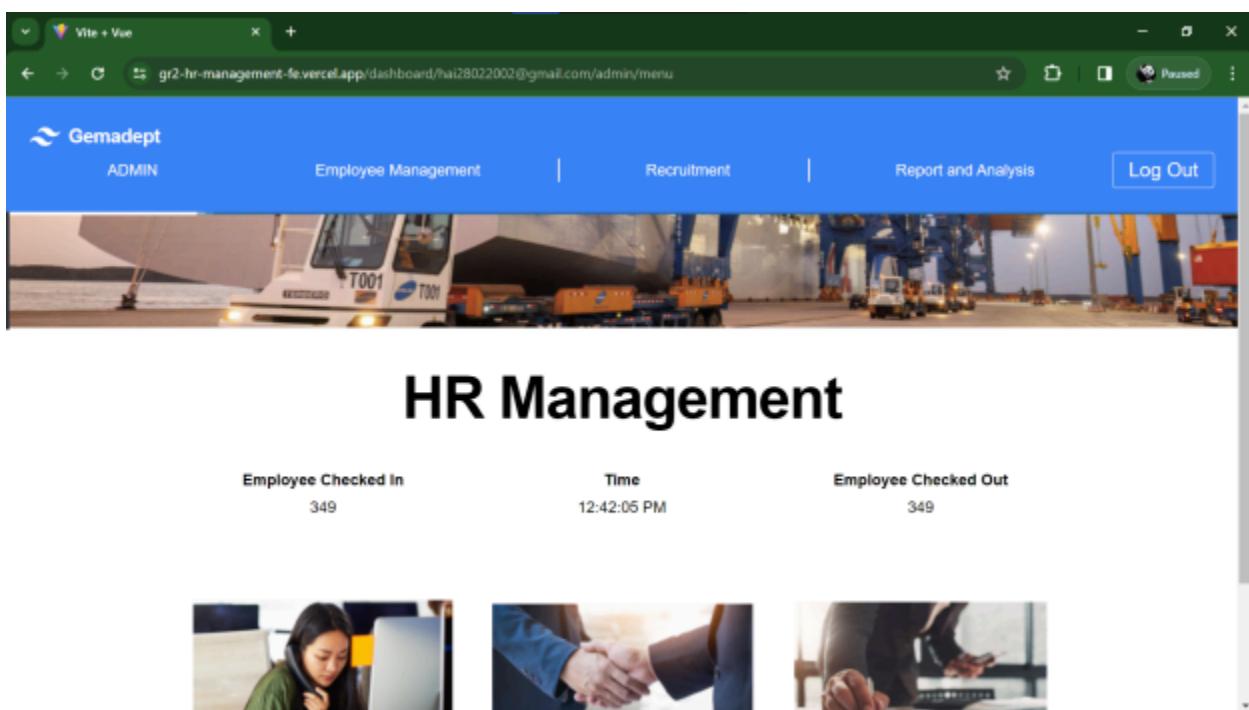
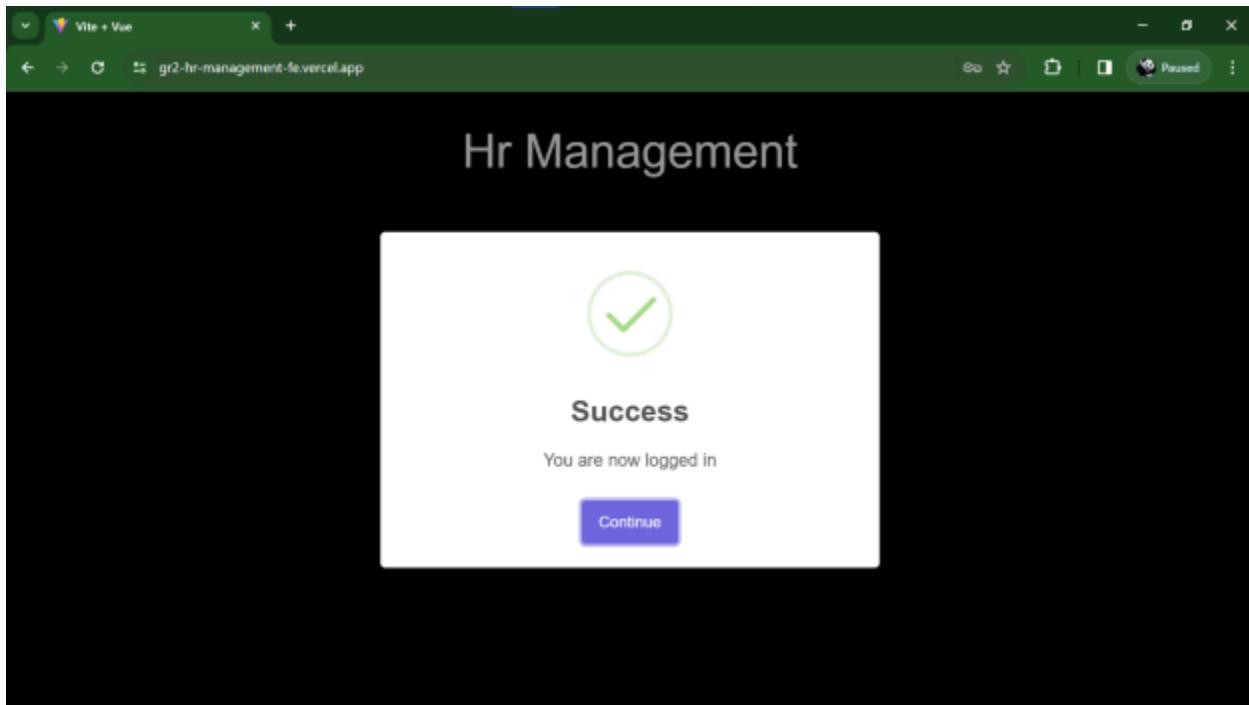
**Payroll Management Page:**

- Header:** Shows the title "Gemadept" and "ADMIN".
- Top Navigation:** Includes links for "Employee Management", "Recruitment", "Report and Analysis", and "Log Out".
- Background Image:** The same photograph of the port or industrial area.
- Section Headers:** "Employee Information", "Payroll Management", and "Overall Tracking".
- Search Bar:** A search input field labeled "Search employee by Name".
- Table:** A grid showing employee details and their status.
 

ID	Employee Email	Salary (per hour)	WeeklyHours	Bonus	Monthly Salary	Status
1	mloyns0@is.gd	\$21.00	7:30:00	\$0.00	\$157.50	Suspending
2	cstoneham1@zimbo.com	\$27.00	1898:00:00	\$0.00	\$51,192.00	Suspending
3	scannop2@who.int	\$21.00	480:00:00	\$0.00	\$10,080.00	Suspending
4	noni3@businesswire.com	\$23.00	7:30:00	\$0.00	\$172.50	Suspending
5	bolmers4@glzmodo.com	\$30.00	12:30:00	\$0.00	\$375.00	Suspending
6	rhandman5@amazon.com	\$23.00	28:00:00	\$0.00	\$645.15	Suspending
7	hdreger6@unc.edu	\$23.00	11:11:00	\$0.00	\$257.22	Suspending

Figure 2.14 – 2.17: Website on Opera Browser

Google Chrome

A screenshot of a dashboard interface titled "HR Management". The top navigation bar is blue and includes links for "Employee Management", "Recruitment", "Report and Analysis", and "Log Out". The main content area features a large banner image of a port or industrial area with several large white trucks and cranes. Below the banner, the title "HR Management" is prominently displayed in a large, bold, black font. Underneath the title, there are three data cards: "Employee Checked In" (349), "Time" (12:42:05 PM), and "Employee Checked Out" (349). At the bottom, there are three small thumbnail images: a woman working at a computer, two people shaking hands, and two people working at a desk.

The screenshot shows the 'Employee Information' section of a web application. It features a search bar at the top left and a 'Paused' status indicator at the top right. Below the header are three tabs: 'Employee Information' (selected), 'Payroll Management', and 'Overall Tracking'. A large search bar is centered above a grid of four employee profiles. Each profile includes a placeholder user icon, the name, title, address, phone number, and email. An 'Add User' button is located in the top right corner of the grid area.

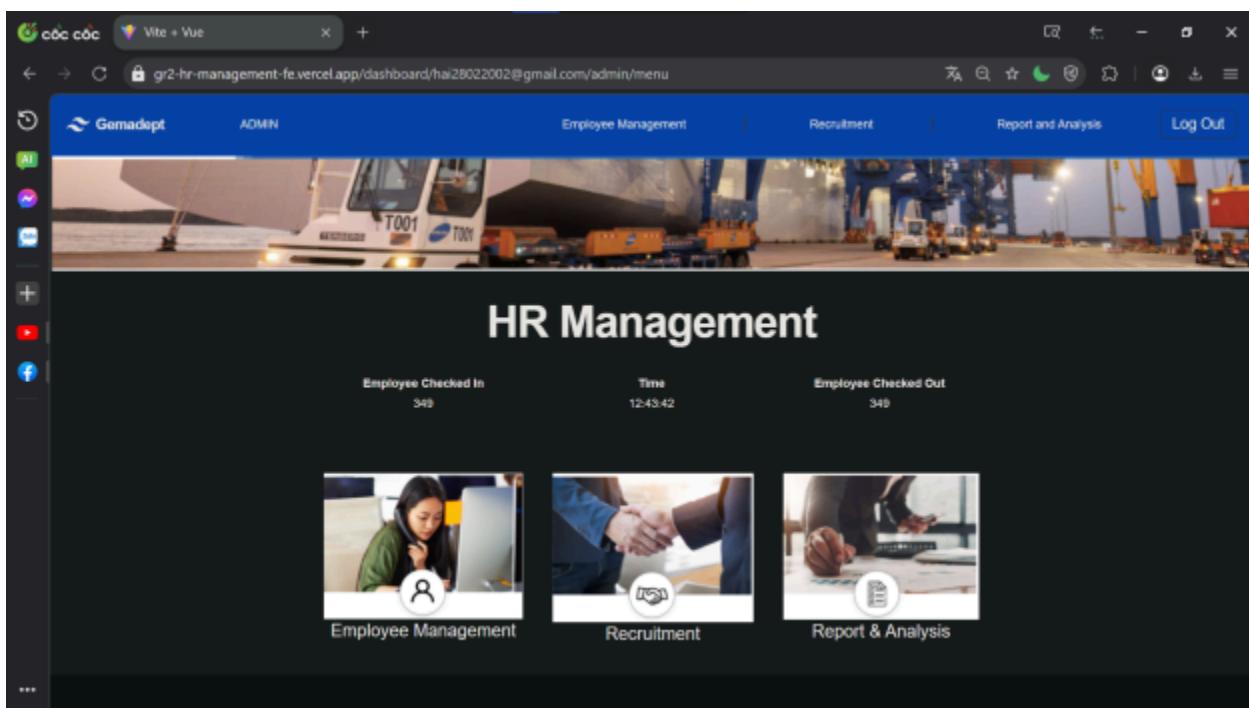
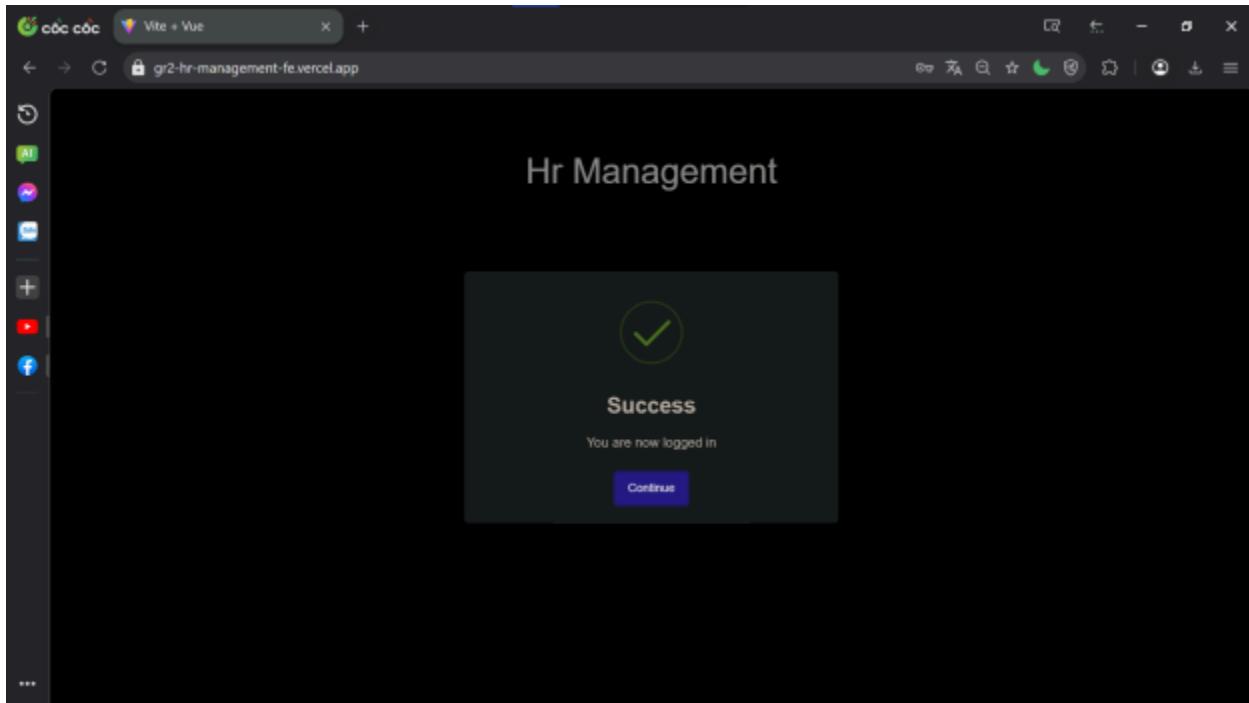
ID	Employee Email	Salary (per hour)	WeeklyHours	Bonus	Monthly Salary	Status
1	mloyns0@is.gd	\$21,00	7:30:00	\$0,00	\$157,50	Suspending
2	cstoneham1@zimbio.com	\$27,00	1896:00:00	\$0,00	\$51.192,00	Suspending
3	scannop2@who.int	\$21,00	480:00:00	\$0,00	\$10.080,00	Suspending
4	inoni3@businesswire.com	\$23,00	7:30:00	\$0,00	\$172,50	Suspending

The screenshot shows the 'Payroll Management' section of the web application. It features a search bar at the top left and a 'Paused' status indicator at the top right. Below the header are four tabs: 'ADMIN' (selected), 'Employee Management', 'Recruitment', and 'Report and Analysis'. A decorative banner image of industrial equipment is displayed below the tabs. The main content area contains a table with employee payroll details, identical to the one shown in the previous screenshot.

Figure 2.18 – 2.21: Website on Google Chrome

Coc Coc:



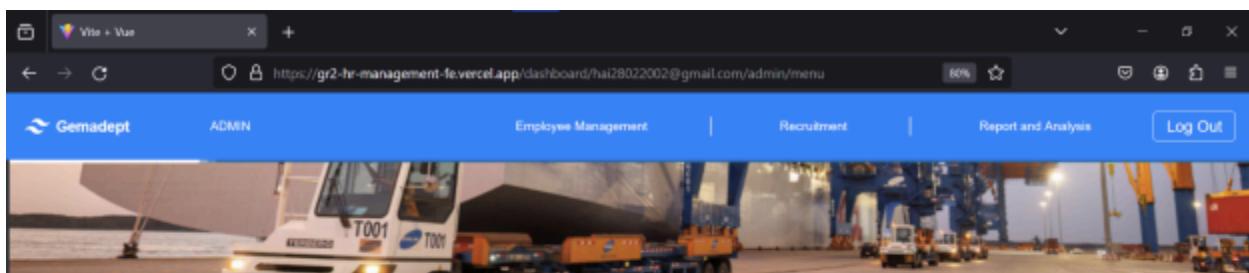
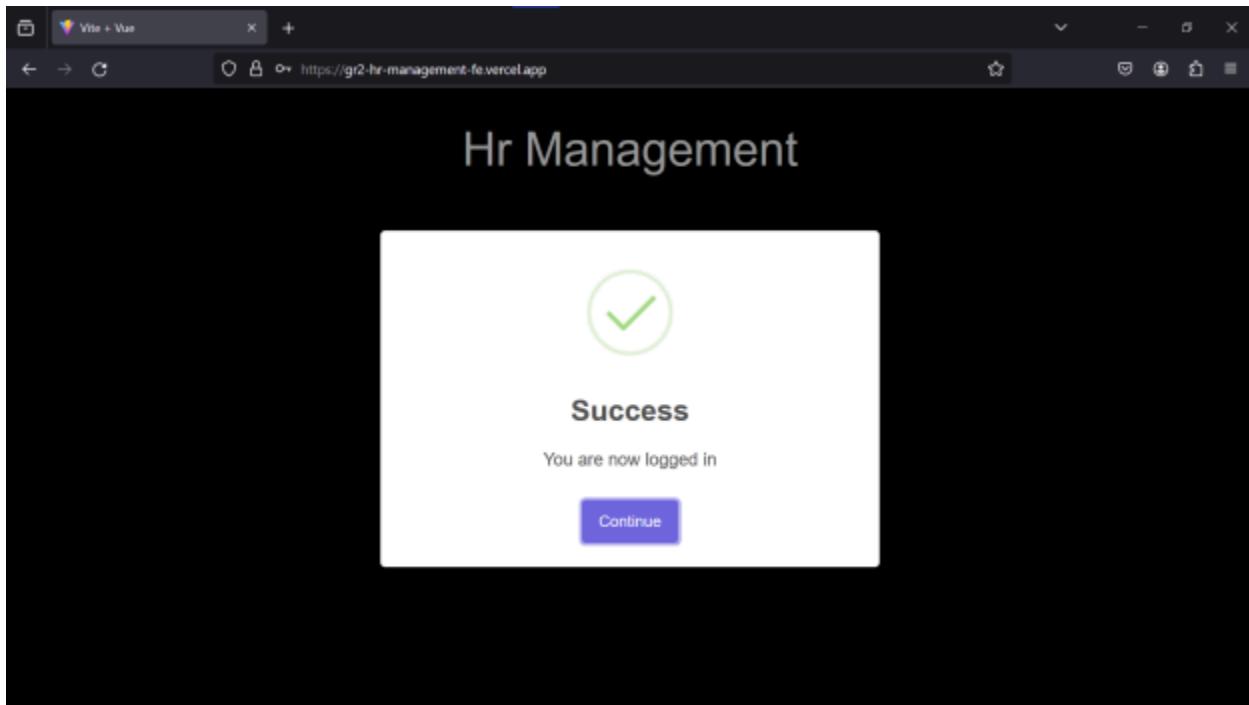
A screenshot of the HR Management dashboard. The top navigation bar includes "Gemadepot" (with a logo), "ADMIN", "Employee Management", "Recruitment", "Report and Analysis", and "Log Out". Below the navigation is a banner image of a port or industrial area with several large white trucks. The main title "HR Management" is centered. Underneath, there are three cards: "Employee Checked In" (349), "Time" (12:43:42), and "Employee Checked Out" (349). At the bottom, there are three more cards: "Employee Management" (with a person icon), "Recruitment" (with a handshake icon), and "Report & Analysis" (with a document icon).

**Employee Management**

ID	Employee Email	Salary (per hour)	WeeklyHours	Bonus	Monthly Salary	Status
1	mloyne0@is.gd	\$21,00	7:30:00	\$0,00	\$167,50	Suspending
2	cstoneham1@zimbio.com	\$27,00	1696:00:00	\$0,00	\$51.192,00	Suspending
3	scannop2@who.int	\$21,00	480:00:00	\$0,00	\$10.080,00	Suspending
4	lnoni3@businesswire.com	\$23,00	7:30:00	\$0,00	\$172,50	Suspending
5	bzimmer4@glamodo.com	\$30,00	12:30:00	\$0,00	\$375,00	Suspending
6	rhandman5@gmail.com	\$23,00	28:03:00	\$0,00	\$645,15	Suspending
7	hdreger6@vnu.edu	\$23,00	11:11:00	\$0,00	\$257,22	Suspending

Figure 2.22 – 2.25: Website on Google Chrome

Firefox:



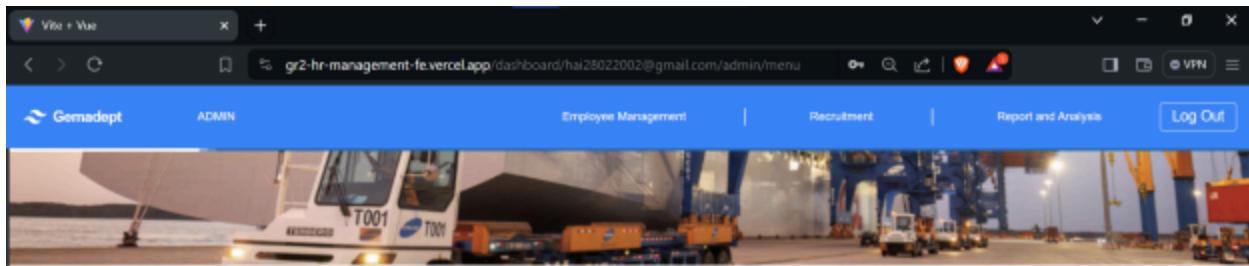
The screenshot shows a web application interface for 'Employee Management'. At the top, there's a navigation bar with 'Gemadept' and 'ADMIN' on the left, and 'Employee Management', 'Recruitment', 'Report and Analysis', and 'Log Out' on the right. Below the navigation is a banner image of a ship being loaded or unloaded at a port. The main content area has three tabs: 'Employee Information' (which is active), 'Payroll Management', and 'Overall Tracking'. A search bar labeled 'Search employee by Name' is present. Below the search bar is a button labeled 'Add User'. There are four user profiles displayed in cards:

User	Title	Address	Phone
Dr Loyns	Senior Financial Analyst	280 Fulton Trail	485-999-3342
Ms Stoneham	Mechanical Systems Engineer	61188 Cambridge Crossing	673-386-1361
Ms Cannop	Analog Circuit Design manager	167 Ennet Alley	987-805-7791
Dr Noni	Microbiologist II	3342 Petree Avenue	842-692-2795

Figure 2.26 – 2.29: Website on Firefox

Brave:

The screenshot shows a web application interface titled 'Hr Management'. The main content area features a large white box with a green checkmark icon. Below the icon, the word 'Success' is centered. Underneath 'Success', a message reads 'You are now logged in'. At the bottom of the box is a blue 'Continue' button. The browser address bar shows the URL 'gr2-hr-management-fe.vercel.app'.



## HR Management

Employee Checked In      Time      Employee Checked Out

349

1:03:02 PM

349



Employee Management



Recruitment



Report & Analysis

A screenshot of the "Employee Management" section of the dashboard. The top navigation bar and port background are identical to the previous screenshot. The main title "Employee Management" is highlighted in blue. Below the title are three tabs: "Employee Information" (highlighted), "Payroll Management", and "Overall Tracking". A search bar labeled "Search employee by Name" is present. Below the tabs, four employee profiles are listed in a grid. Each profile includes a placeholder user icon, the employee's name, their title, and basic contact information.

Dr Loyns

Senior Financial Analyst

Address: 260 Fulton Trail  
Phone: 455-999-0342  
Email: drloyns@gti.com

Ms Stoneham

Mechanical Systems Engineer

Address: 4116 Cambridge Crossing  
Phone: 873-565-1384  
Email: cstoneham1@zimbo.com

Ms Cannop

Analog Circuit Design manager

Address: 157 Emmet Alley  
Phone: 957-805-7791  
Email: scannop2@viro.it

Dr Noni

Electrokinetics R&D

Address: 3542 Peltier Avenue  
Phone: 842-922-2756  
Email: mnoni3@businessware.com

ID	Employee Email	Salary (per hour)	WeeklyHours	Bonus	Monthly Salary	Status
1	mloymoD@is.gd	\$21,00	7:30:00	\$0,00	\$157,90	Suspending
2	calonwham1@zimbio.com	\$27,00	1896:00:00	\$0,00	\$51.192,00	Suspending
3	scannop2@who.int	\$21,00	480:00:00	\$0,00	\$10.080,00	Suspending
4	inoni3@businesswire.com	\$23,00	7:30:00	\$0,00	\$172,50	Suspending
5	bsimmers4@glzmodo.com	\$30,00	12:30:00	\$0,00	\$375,00	Suspending
6	nhandman6@amazon.com	\$23,00	28:03:00	\$0,00	\$645,15	Suspending
7	htreger6@unc.edu	\$23,00	11:11:00	\$0,00	\$257,22	Suspending

Figure 2.30 – 2.33: Website on Brave

Edge:

Hr Management

Success

You are now logged in

Continue

The screenshot shows the HR Management dashboard. At the top, there are three main navigation tabs: "Employee Management", "Recruitment", and "Report and Analysis". On the far right, there is a "Log Out" button. Below the tabs, there is a large banner image of a port or industrial area with several large ships and cranes. The main title "HR Management" is centered below the banner. Underneath the title, there is a section for "Employee Checked In" which shows a count of 349 and a timestamp of 12:38:37 PM. To the right, there is a section for "Employee Checked Out" which also shows a count of 349. Below these sections, there are three small thumbnail images: one of a person working at a computer, one of two people shaking hands, and one of a person using a smartphone.

The screenshot shows the "Employee Management" page. At the top, there are four main navigation tabs: "Employee Management", "Recruitment", "Report and Analysis", and "Log Out". Below the tabs, there is a large banner image of a port or industrial area. The main content area has three tabs: "Employee Information" (which is selected and highlighted in blue), "Payroll Management", and "Overall Tracking". Below the tabs, there is a search bar with the placeholder "Search employee by Name" and a "Add User" button. The main content area displays four employee profiles in cards:

Employee Name	Title	Address	Phone	Email
Dr Loyns	Senior Financial Analyst	280 Fulton Trail	(485) 986-3342	nloyns0@n.gf
Ms Stoneham	Mechanical Systems Engineer	81180 Cambridge Crossing	(873) 386-1381	cstoneham1@zimbo.com
Ms Cannop	Analog Circuit Design manager	1627 Everett Alley	(957) 885-7791	scannop2@jeho.inf
Dr Noni	Electronician III	3342 Petterle Avenue	(640) 823-2795	inoni3@wixieswite.com

ID	Employee Email	Salary (per hour)	WeeklyHours	Bonus	Monthly Salary	Status
1	mibyme0@is.gd	\$21,00	7:30:00	\$0,00	\$157,50	Suspending
2	ctstoneham1@zimbio.com	\$27,00	1896:00:00	\$0,00	\$51,192,00	Suspending
3	scannop2@who.int	\$21,00	480:00:00	\$0,00	\$10,080,00	Suspending
4	inoni3@businesswire.com	\$23,00	7:30:00	\$0,00	\$172,50	Suspending
5	bsimmers4@gizmodo.com	\$30,00	12:30:00	\$0,00	\$375,00	Suspending
6	rhandman5@amazon.com	\$23,00	28:03:00	\$0,00	\$645,15	Suspending
7	hdreger6@unc.edu	\$23,00	11:11:00	\$0,00	\$257,22	Suspending

Figure 2.34 – 2.37: Website on Edge

We tested login in and other tests on various browser and our website held itself well. Hence, this quality passed our benchmark.

### Group 2: Search and Input

**Quality 6:** The database searching and filtering options should return the correct result.

**Testing measure:** Use the search function with a variety of keywords and random words to note the displayed information.

Address: 88000 Continental Road	Address: 80432 Badeau Avenue
Phone: 327-422-3817	Phone: 161-921-1559
Email: aupjohn32@geocities.jp	Email: frabjohn18@wulfoo.com

[Employee Information](#) | [Payroll Management](#) | [Overall Tracking](#)

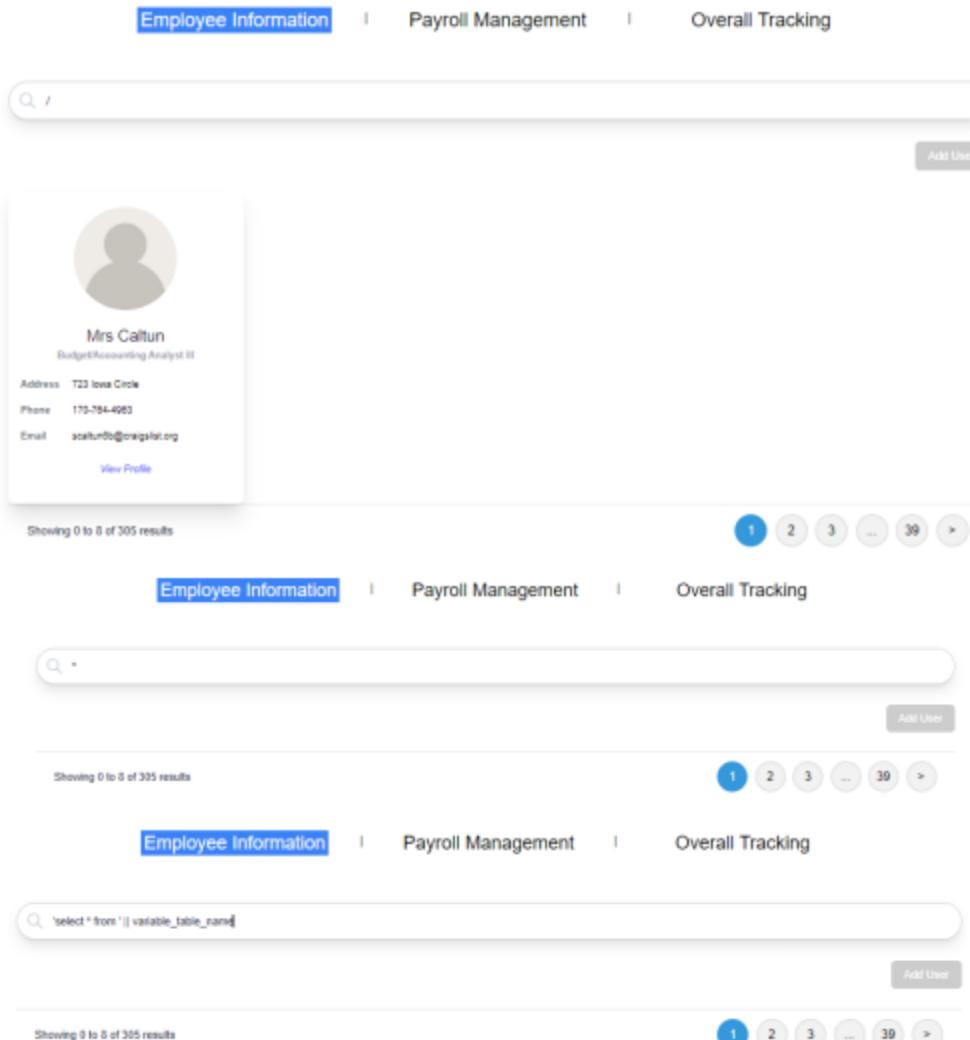
[Add User](#)

 <p><b>Dr Loyns</b> Senior Financial Analyst</p> <p>Address: 280 Fulton Trail Phone: 485-890-5542 Email: nloyns0@ra.gr</p> <p><a href="#">View Profile</a></p>	 <p><b>Dr Noni</b> Biostatistician III</p> <p>Address: 3342 Potters Avenue Phone: 842-422-2195 Email: inoni3@businesswir.com</p> <p><a href="#">View Profile</a></p>	 <p><b>Ms Dreger</b> Senior Sales Associate</p> <p>Address: 49830 Pennsylvania Terrace Phone: 815-254-8569 Email: ndreger@uh.edu</p> <p><a href="#">View Profile</a></p>	 <p><b>Dr Wylie</b> Biostatistician I</p> <p>Address: 83337 Meissnermidt Way Phone: 850-571-1958 Email: jwylie@goole.ru</p> <p><a href="#">View Profile</a></p>
 <p><b>Dr Ren</b> Programmer Analyst II</p> <p>Address: 6197 Cody Street Phone: 736-731-0086 Email: trenk@busines.com</p>	 <p><b>Dr Plenderleith</b> Computer Systems Analyst III</p> <p>Address: 6287 Stephen Center Phone: 368-680-5517 Email: adelenda@busines.com</p>	 <p><b>Dr McGlynn</b> Senior Quality Engineer</p> <p>Address: 89058 Carberry Junction Phone: 138-675-7503 Email: amcglynn@bu.com</p>	 <p><b>Rev Goodright</b> Junior Executive</p> <p>Address: 8 Rutledge Terrace Phone: 480-783-4310 Email: execgood12@busines.com</p>

[Employee Information](#) | [Payroll Management](#) | [Overall Tracking](#)

[Add User](#)

 <p><b>Ms Stoneham</b> Mechanical Systems Engineer</p> <p>Address: 61188 Cambridge Crossing Phone: 873-286-1381 Email: cstoneham@simio.com</p> <p><a href="#">View Profile</a></p>	 <p><b>Rev Gotecliffe</b> Software Engineer II</p> <p>Address: 827 Canary Plaza Phone: 248-614-3281 Email: tgotecliffe@wets.com</p> <p><a href="#">View Profile</a></p>	 <p><b>Mr Lineham</b> Software Test Engineer I</p> <p>Address: 2 Lany Parkway Phone: 381-377-0938 Email: klineham@ad.gov</p> <p><a href="#">View Profile</a></p>	 <p><b>Mr Hawkswood</b> Product Engineer</p> <p>Address: 60813 Lakewood Parkway Phone: 807-824-4814 Email: mhawkswood@sageagile.it</p> <p><a href="#">View Profile</a></p>
 <p><b>Mrs Paiton</b> Mechanical Systems Engineer</p> <p>Address: 278 Stone Corner Pass Phone: 168-811-7889 Email: bpaiton@softwra.com</p>	 <p><b>Dr McGlynn</b> Senior Quality Engineer</p> <p>Address: 89058 Carberry Junction Phone: 138-675-7683 Email: amcglynn@bu.com</p>	 <p><b>Honorable Corton</b> Geological Engineer</p> <p>Address: 9 Washington Parkway Phone: 911-855-2170 Email: scorton@fbi.org</p>	 <p><b>Mrs Rennebeck</b> Senior Quality Engineer</p> <p>Address: 7 Antalt Court Phone: 794-820-5494 Email: drennebeck1@busines.com</p>

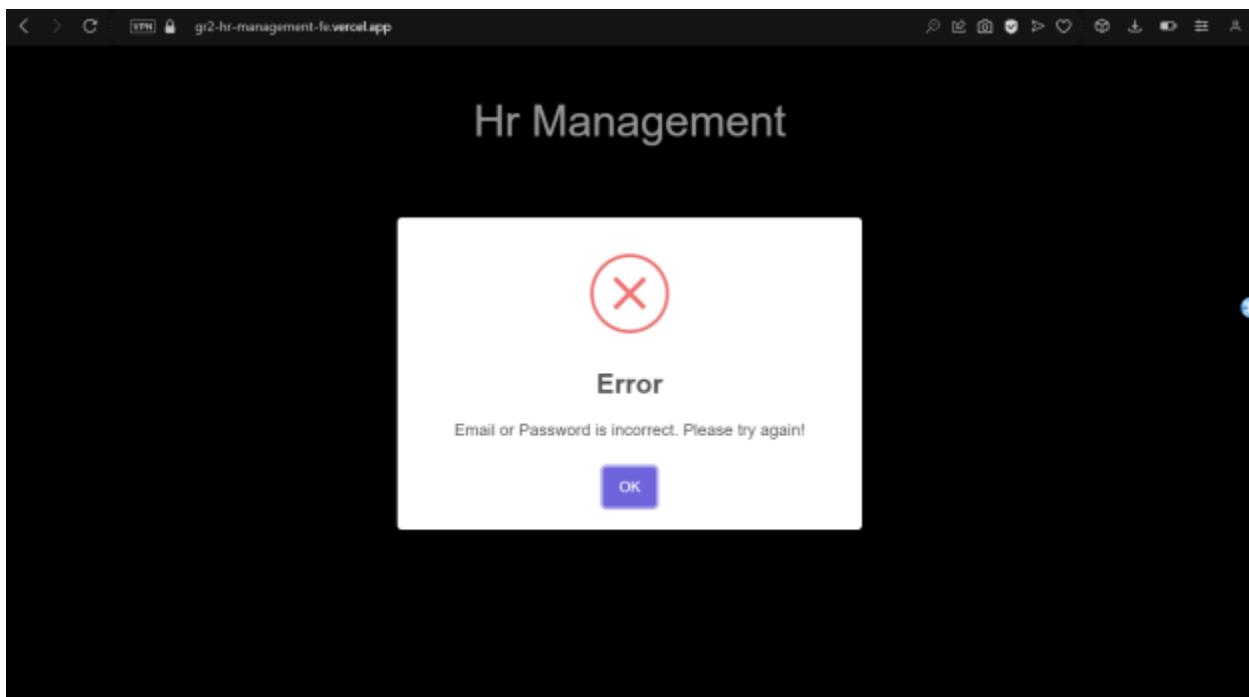
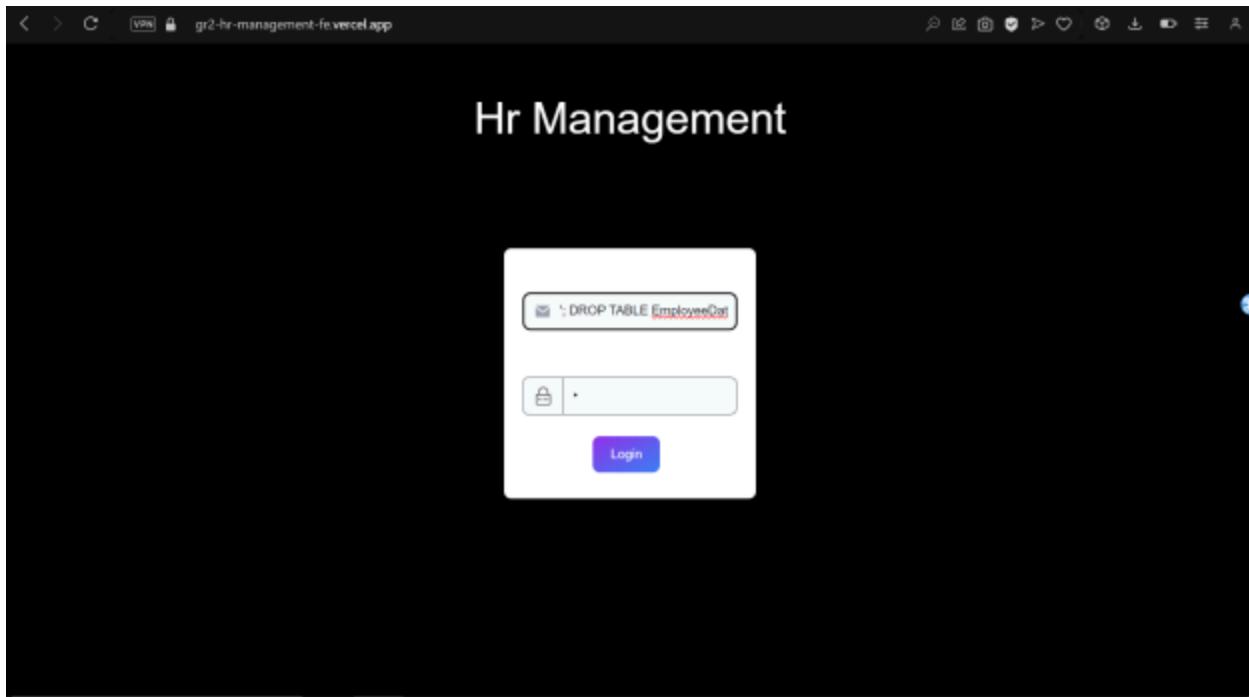


*Figure 2.38 – 2.43: Testing search query*

We tested the website's query with a variety of input from words, names, to special characters, and as can be seen, the search returns accurate results and does not allow special characters, hence this quality passed our benchmark.

**Quality 12:** Proper input sanitization should be checked to prevent code injection or unintended characters. Additionally, handling of special characters, whitespace, and input encoding should be validated.

**Testing measure:** Test all input fields on the website against 20 different test cases of varying error.



Showing 0 to 8 of 305 results

1 2 3 ... 39 >

Showing 0 to 8 of 305 results

**Ms Lillywhite**  
Environmental Tech

Address: 357 Portage Road  
Phone: 815-603-3522  
Email: lillywhite4@businessinsider.com

Gemadepot ADMIN Employee Management | Recruitment | Report and Analysis Log Out

Employee Information | Payroll Management | Overall Tracking

Enter bonus amount

ID	Employee Email	Monthly Salary	Status
1	mbyrne0@is.gd	\$157,50	Suspending
2	cstoneham1@zimbio.com	\$51,192,00	Suspending
3	scannop2@who.int	\$10,080,00	Suspending
4	lioni3@businesswire.com	\$172,50	Suspending
5	bsimmers4@gizmodo.com	\$375,00	Suspending
6	nhandman5@amazon.com	\$645,15	Suspending
7	hdreger6@unc.edu	\$257,22	Suspending

ID	Employee Email	Monthly Salary	Status
1	mloyiso@is.gd	\$157,50	Suspending
2	cstoneham1@zimbio.com	\$61,192,00	Suspending
3	scampop2@who.int	\$10,080,00	Suspending
4	lironi3@businesswire.com	\$172,50	Suspending
5	bsimmers4@gizmodo.com	\$375,00	Suspending
6	rhandmard@amazon.com	\$645,15	Suspending
7	hdreger6@unc.edu	\$257,22	Suspending

Employee Information | Payroll Management | Overall Tracking

SELECT \* FROM EmployeeData LIKE 'John%' OR '1'='1'

Add User

Showing 0 to 8 of 305 results

1 2 3 ... 39 >

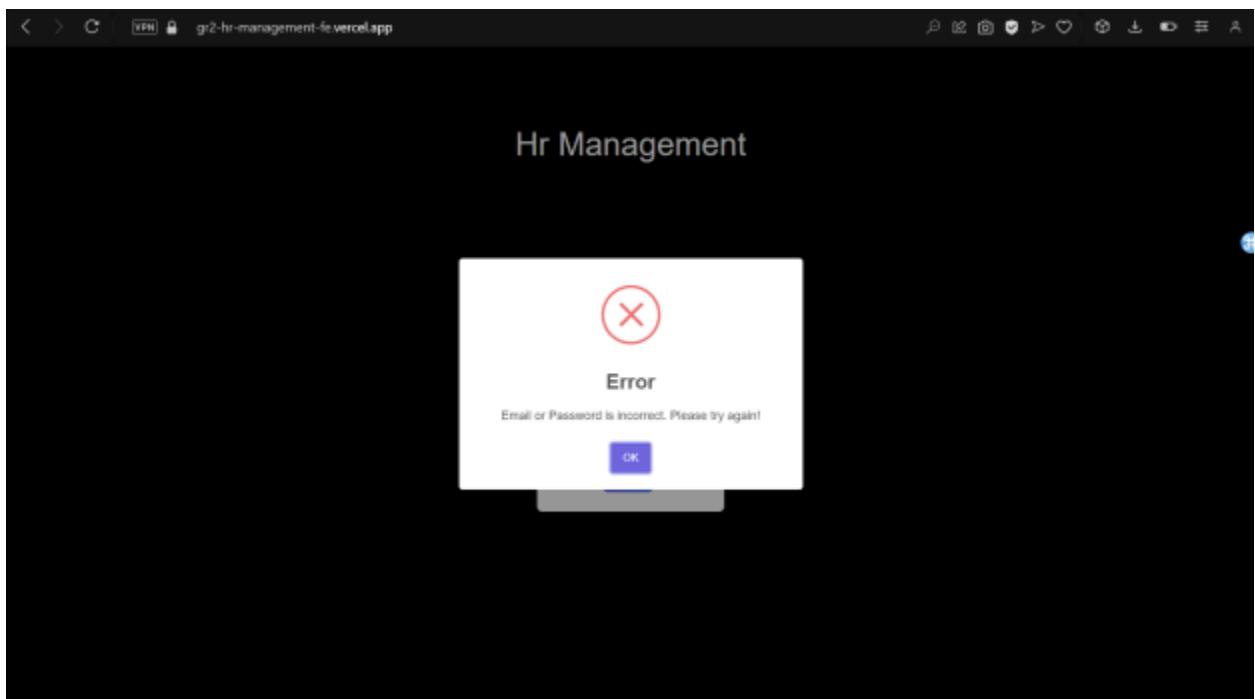
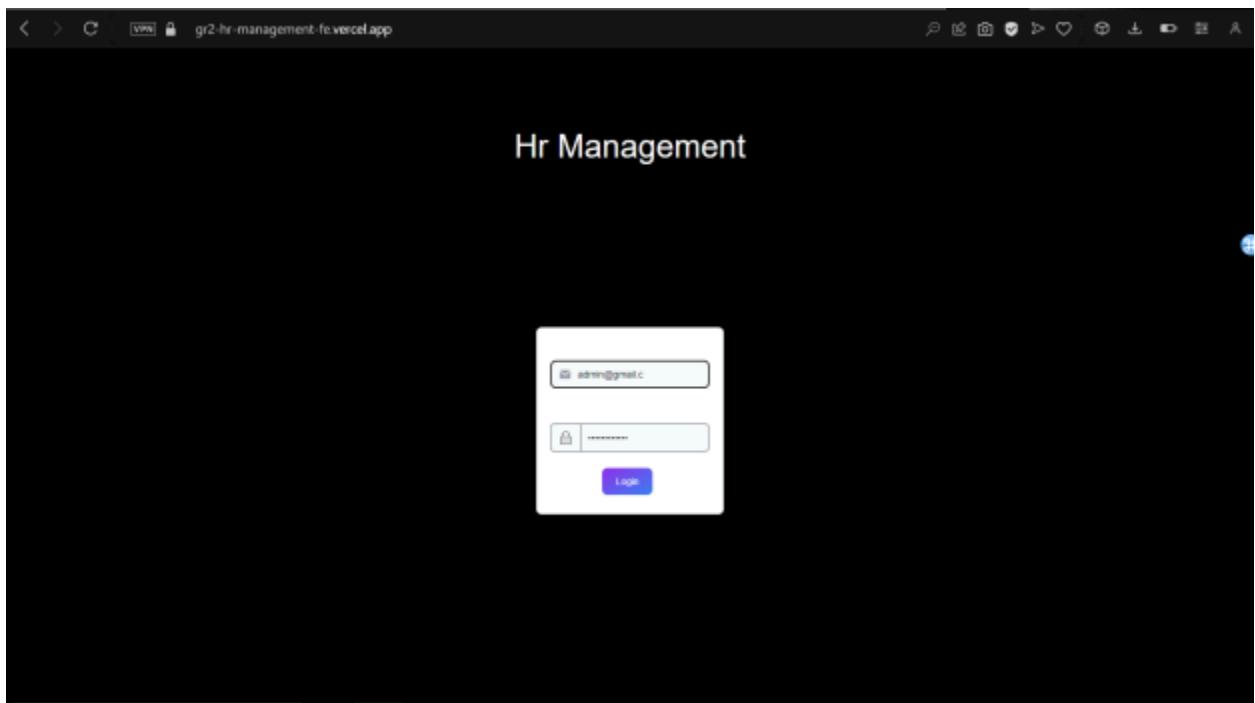
*Figure 2.44 – 2.51: Input Testing*

After testing, we can see that our input has passed a good number of test cases, and those with severe priority are addresses. Hence, this quality passes our benchmark.

### Group 3: Other Tests

**Quality 14:** Error messages should be displayed when invalid data is entered, providing clear instructions on how to correct them.

**Testing measure:** Wrongly execute any input and observe whether error messages appear.



The figure consists of two vertically stacked screenshots of a web-based payroll management system. Both screenshots have a header with the logo 'Gemadept', user role 'ADMIN', and menu items 'Employee Management', 'Recruitment', 'Report and Analysis', and 'Log Out'. The background of both screenshots shows a blurred image of a port or industrial setting with shipping containers and cranes.

**Screenshot 1 (Top):**

- Modal Dialog:** 'Enter bonus amount' with an input field containing 'e'.
- Table:** Employee Information with 7 rows of data.

ID	Employee Email	Monthly Salary	Status
1	mloyna0@is.gd	\$157,50	Suspending
2	cstoneham1@zimbio.com	\$51,192,00	Suspending
3	scannop2@who.int	\$10,080,00	Suspending
4	inon3@businesswire.com	\$172,50	Suspending
5	bsimmers4@gizmodo.com	\$375,00	Suspending
6	nhandman5@amazon.com	\$645,15	Suspending
7	hdreger6@unc.edu	\$257,22	Suspending

**Screenshot 2 (Bottom):**

- Modal Dialog:** 'Enter bonus amount' with an input field containing 'e' and an error message: 'Please enter a number.'
- Table:** Employee Information with 7 rows of data.

ID	Employee Email	Monthly Salary	Status
1	mloyna0@is.gd	\$157,50	Suspending
2	cstoneham1@zimbio.com	\$51,192,00	Suspending
3	scannop2@who.int	\$10,080,00	Suspending
4	inon3@businesswire.com	\$172,50	Suspending
5	bsimmers4@gizmodo.com	\$375,00	Suspending
6	nhandman5@amazon.com	\$645,15	Suspending
7	hdreger6@unc.edu	\$257,22	Suspending

Figure 2.52 – 2.55: Instruction Testing

The invalid text prompt is correctly showing on our website. Hence this quality passed out benchmark.

**Quality 16:** API connection to other services (i.e., existing ERP, Google sign in, etc.) should have little error.

**Testing measure:** Intentionally disrupt network connectivity during an ongoing API request and verify that the system handles the timeout or connection failure gracefully.



Gemadept ADMIN Employee Management | Recruitment | Report and Analysis Log Out

Employee Information | Payroll Management | Overall Tracking

Search employee by Name:

ID	Employee Email	Salary (per hour)	WeeklyHours	Bonus	Monthly Salary	Status
1	mboym0@ix.gd	\$21.00	7.00 00	\$0.00	\$157.50	Suspending
2	cstoneham1@zimbio.com	\$27.00	1666.00 00	\$0.00	\$51.192.00	Suspending
3	scannop2@who.int	\$21.00	480.00 00	\$0.00	\$10.080.00	Suspending
4	inon3@businesswire.com	\$23.00	7.00 00	\$0.00	\$172.50	Suspending
5	tsimmers4@gizmodo.com	\$30.00	12.00 00	\$0.00	\$375.00	Suspending
6	nhandman5@amazon.com	\$23.00	28.03 00	\$0.00	\$645.15	Suspending
7	hdreger6@unc.edu	\$23.00	11.11 00	\$0.00	\$257.22	Suspending
8	meverson7@reference.com	\$27.00	24.25 00	\$0.00	\$659.25	Suspending
9	bgotediff8@webs.com	\$23.00	26.23 00	\$0.00	\$606.82	Suspending



Gemadept ADMIN Employee Management | Recruitment | Report and Analysis Log Out

Employee Information | Payroll Management | Overall Tracking

Search employee by Name:

Enter bonus amount

1000

Request failed. TypeError: Failed to fetch

Submit Cancel

ID	Employee Email	Salary (per hour)	WeeklyHours	Bonus	Monthly Salary	Status
1	mboym0@ix.gd	\$21.00	7.00 00	\$0.00	\$157.50	Suspending
2	cstoneham1@zimbio.com	\$27.00	1666.00 00	\$0.00	\$51.192.00	Suspending
3	scannop2@who.int	\$21.00	480.00 00	\$0.00	\$10.080.00	Suspending
4	inon3@businesswire.com	\$23.00	7.00 00	\$0.00	\$172.50	Suspending
5	tsimmers4@gizmodo.com	\$30.00	12.00 00	\$0.00	\$375.00	Suspending
6	nhandman5@amazon.com	\$23.00	28.03 00	\$0.00	\$645.15	Suspending
7	hdreger6@unc.edu	\$23.00	11.11 00	\$0.00	\$257.22	Suspending
8	meverson7@reference.com	\$27.00	24.25 00	\$0.00	\$659.25	Suspending
9	bgotediff8@webs.com	\$23.00	26.23 00	\$0.00	\$606.82	Suspending

L7	A	B	C	D	E	F	G	H	I	J	K	L
1	Id	email	base_salary	total_work_hour	bonus	actual_pay						
2	1	mloyno0@is.gd	\$21,00	7:30:00	\$0,00	\$157,50						
3	2	cstoneham1@zimbio.com	\$27,00	1896:00:00	\$0,00	\$51,192,00						
4	3	scannop2@who.int	\$21,00	480:00:00	\$0,00	\$10,080,00						
5	4	inon3@businesswire.com	\$23,00	7:30:00	\$0,00	\$172,50						
6	5	bimmers4@joomla.com	\$30,00	12:30:00	\$0,00	\$375,00						
7	6	nhandman5@amazon.com	\$23,00	28:03:00	\$0,00	\$645,15						
8	7	hdreger6@unc.edu	\$23,00	11:11:00	\$0,00	\$257,22						
9	8	meverson7@reference.com	\$27,00	24:26:00	\$0,00	\$659,25						
10	9	lgotecliffe8@wells.com	\$23,00	26:23:00	\$0,00	\$608,82						
11	10	babramon9@eq.com	\$29,00	29:21:00	\$0,00	\$733,75						
12	11	jwylea@google.ru	\$24,00	17:46:00	\$0,00	\$426,40						
13	12	kwhewillb@mired.com	\$23,00	27:49:00	\$0,00	\$639,78						
14	13	mfrancyc@elegantthemes.com	\$26,00	20:44:00	\$0,00	\$539,07						
15	14	wcannawayd@wired.com	\$23,00	30:56:00	\$0,00	\$711,47						
16	15	gdyter@matadory.com	\$27,00	10:56:00	\$0,00	\$295,20						
17	16	finehamf@ed.gov	\$26,00	6:33:00	\$0,00	\$170,30						
18	17	hismong@baidu.com	\$26,00	23:42:00	\$0,00	\$616,20						
19	18	sfairyh@cnet.com	\$29,00	11:23:00	\$0,00	\$330,12						
20	19	Mahosij@parcio	\$23,00	8:21:00	\$0,00	\$192,05						
21	20	ngoodallj@tiny.cc	\$27,00	48:08:00	\$0,00	\$1,299,00						
22	21	lrenk@weibo.com	\$22,00	14:53:00	\$113,00	\$327,43						

Trang tính1 EmployeeRole EmployeeData Employee Attendance 05/2023 Salary of May

Figure 2.56 – 58: API connectivity check

We tested the API connectivity by disconnecting the device from the internet and attempted to add information on the website. As can be seen, the API failed to fetch, and data was not added to the database. This quality hence passed our benchmark.

## Report

So, we were able to conduct quality testing for 6 out of 17 of our quality tests that were defined in the previous document. This means that we have yet to achieve the substantial amount of quality that was defined, and our product is lacking a significant amount. Moreover, some important and urgent quality has not been done.

The reason that we have not completed all the quality defined is due to the fact that our sprint was originally planned to be as long as 21 days, but we have only done 10 days of work as of this period of time. Hence, we can state that our backlog item (HRM website) does not satisfy the quality requirements defined.

## 3. Rationale and Analysis

### What Worked:

- **Clear Task Definition:** The sprint backlog items, in my opinion, had well-defined goals and acceptance criteria. This helps teamwork become better while also making tasks in the sprint easier to follow through.
- **Achievable Thresholds:** We created achievable thresholds for our sprint goals and adhered to the SMART framework, so we had reasonable goals to strive for. Which adds to our ability to achieve a lot of quality in a limited amount of time.

- **Efficient Communication:** We had multiple channels of communication, which became very useful for exchanging ideas, solutions and urge one another to do tasks.

#### **What Did Not Work:**

- **Rushed Development:** As we had to do our sprint in a timeframe lower than what was planned, development activities were rushed, which leads to some compromise in code quality and could potentially introduce defects.
- **Inefficient Time Allocation:** With a limited timeframe, time was not always allocated efficiently, which results in delays or unfinished tasks towards the end of the sprint.
- **Too Many Qualities:** We tried to achieve too many quality metrics at the same time and may have stretched the team's focus too thin.

#### **Future Recommendations:**

- **Improved Time Management:** We believe that we should put time tracking and task prioritizing strategies into practice. Which will hopefully help avoid last-minute rushes, and divide work into smaller, more manageable chunks and allot time wisely.
- **Quality Over Quantity:** We need to focus on tasks that have a higher priority first (e.g., urgent, and important task). We believe that would help us to concentrate efforts on key quality aspects, and the team can deliver more impactful results within the sprint timeframe, while putting the less important task for later.
- **Continuous Improvement:** As a team, we should strive for greater growth by valuing input, introspection, and using lessons from the past. To assess sprint performance, pinpoint areas for improvement, and execute iterative modifications to improve procedures and practices, more meetings are held.

## **TASK 081HD: Software Project Planning, Design and Quality Management**

### **Project Proposal: *Cloud-based work planner and organizer system***

#### **1. Background / Problem Description**

Task management is critical in today's world, from studying to working it is crucial that we keep track of our tasks, progress and development in order to manage our time and resources in the most efficient way possible. However, traditional methods of task tracking like paper based or standalone software introduces drawbacks that limits scalability and collaboration with other people; which is why our project proposed a cloud-based task planner and tracking system that would allow stream-lined management processes and enhanced collaboration within teams and organizations.

## **2. Scope**

### **2.1 Objectives:**

The cloud-based planner app will assist in the streamlining management process as it allows users to organize tasks, track progress, and manage deadlines effectively. This can be achieved as users can create, group and categorize projects based on their relativity or priority. The app will allow for collaboration between users which would facilitate increased collaboration and we aimed to have it working on both mobile and desktop which will aid in the overall productivity. Another notable objective we aimed for is deadline notification, which will notify users about tasks that are about to be due shortly.

### **2.2 Solutions:**

To achieve the objectives listed, this software will include but not limited to these key features:

- Task Organization: Allow users to categorize and prioritize tasks via task groups, priority, or category, etc.
- Collaboration: Enable users to share tasks and projects with one another, assign tasks to team members, and track collaborative progress.
- Progress Tracking: Provide tools (checklists, progress bar, etc.) for users to track the progress of their tasks and projects, completion percentages, and milestone tracking.
- Integration: Integrate with other tools, plug-ins and platforms commonly used for productivity, like email, calendar apps, or communication tools like Slack.
- Customization: Offer customization options for users, which could include customizable task fields, color-coding options, or the ability to create custom task templates.
- Mobile Accessibility: The planner should be accessible on mobile devices via website or mobile app.
- Notifications: Provide customizable notification via email or in-app notifications settings to remind users of upcoming deadlines, task assignments, or changes to tasks.

### 3. Stakeholders

<b>Stakeholders</b>	<b>Interest</b>
Users	The primary stakeholders of the work organizer website are the users who will utilize the task management application.
Management	Higher-level management within the organization, who may provide oversight, guidance, support for the project or have strategic goals, objectives tied to the project's outcomes.
Project Team	The project team members, including developers, project managers, and quality assurance engineers, who are directly involved in the planning, development, and delivery of the website and have clear interest in creating the website.

### 4. Deliverable and schedule

#### Schedule:

This project is expected to take around 2 weeks (10 working days) to complete, while the maintenance support will be an on-going process. The timeline for each stage (from designing to implementation) of the project is outlined as follow:

System Design and Prototyping: 1 day.

Development and Testing: 7 days.

Fine tuning and Implementation: 3 days.

Post-implementation support and maintenance: On-going from go live date

#### Initial Release Schedule of the Product backlog items

No.	Item	Dependences	Business Value (1 least – 10 most)	Release Schedule (Sprint 1   2   3   ...)
F1	Website design and wireframe.	None	9	Sprint 1
F2	Basic website with task card creation function.	None	8	Sprint 1
F3	Feature to hold tasks card defined.	F2	7	Sprint 1

F4	Urgency assigned for tasks.	F3	7	Sprint 1
F5	Signup and Login feature	F2	8	Sprint 1
F6	Feature to set deadlines to tasks.	F2, F3	7	Sprint 2
F7	Email users on impending deadline feature.	F3, F4	9	Sprint 2
F8	Further customization options (background, color, etc.)	None	5	Sprint 2
F9	Feature to assign users to different tasks.	F2, F3, F5	8	Sprint 2
F10	Integration with Google calendar.	F5, F9	7	Sprint 3
F11	UAT Testing and suggestion collection.	After all prior product	7	Sprint 3
F12	Back and front-end thorough testing.	F11	8	Sprint 3
F13	Final bug fix and documentation.	F12	9	Sprint 4
F14	Go live and ongoing maintenance	F13	10	Sprint 4

Business value rationale:

Sprint	Item	Business Value	Reasons
Sprint 1	Website design and wireframe.	9	Essential for setting the overall tone and impression of the website.
Sprint 1	Basic website with task card creation function.	8	Forms the backbone of the website's core functionality.

Sprint 1	Feature to hold tasks card defined.	7	Fundamental to the website's purpose and usability. But is less significant as it is a feature, not backbone.
Sprint 1	Urgency assigned for tasks.	7	Enhances efficiency in task management and prioritization. But is less significant compared to other items in the sprint.
Sprint 1	Signup and Login feature	8	Essential for user authentication and engagement.
Sprint 2	Feature to set deadlines to tasks.	7	Promotes time management and productivity. But is less significant as it is a feature, not backbone.
Sprint 2	Email users on impending deadline feature.	9	Significantly enhances user engagement and task completion.
Sprint 2	Further customization options (background, color, etc.)	5	While adds value in user satisfaction, not critical to core functionality.
Sprint 2	Feature to assign users to different tasks.	8	Facilitates collaboration and delegation, essential for team-based projects. Foster collaboration, one of the key (but not the most important) feature
Sprint 3	Integration with Google calendar.	7	Synchronizes tasks and deadlines with a widely used calendar platform. But is less significant compared to other items in the sprint
Sprint 3	UAT Testing and suggestion collection.	7	Ensures functionality and usability meet user expectations. Significant, but need completion from other features in the sprint.

Sprint 3	Back and front-end thorough testing.	8	Crucial for reliability, performance, and user experience.
Sprint 4	Final bug fix and documentation	9	Ensures stability, maintainability, and smooth transition to a live environment.
Sprint 4	Go live and ongoing maintenance	10	Critical for launching a website and ensuring continued operation and improvement.

#### Work Breakdown Structure:

We will use the Work Breakdown structure to estimate the effort it will take to develop our first sprint of the project. As we have 2 weeks to complete the project, and each person will work at most 8 hours per week which leaves us with 64 hours to plan our project.

As the first sprint has the most number of products with a great degree of complexity, half of the effort will be put into developing this sprint. This is because the subsequent sprint relies on this sprint as the backbone, hence, its completeness is the utmost priority. So 34 hours will be put into the first sprint, while the remaining 30 hours will be spread between the other 3.

## **TASK 082D: Software Development Methodology Comparison**

### **Title of the article**

**Scrum is better than CPM – a case study of a software project “HRM system with Biometric Attendance”.**

**Nguyen Dinh Nhat Minh**

**Student id 103802490**

**103802490@student.swin.edu.au**

## **Abstract**

This essay attempts to compare the effectiveness of Scrum, an agile software development approach versus a traditional approach to software project management, the Critical Path Method. The comparison will be achieved by applying scrum or CPM to address challenges or enhance development management within a case study of a project named HRM system with biometric attendance. The effectiveness of the scrum methodology when it comes to the modern software development process outweighs that of the traditional Critical Path Method because the scrum agile methodology shines in terms of increasing adaptability, changes responsiveness, and collaboration between different parties.

### **1. Introduction**

Building a software involves a lot of elements, from planning the scope to developing a product, these elements can contribute to making the cycle of software development complex as the duration of the project goes on and tasks start increasing. Hence, it is of the utmost importance that the project management bodies select and use the right Software Development Life Cycle models to devise a plan that would accommodate their project effectively [1, 2].

Within this essay, we will attempt to discuss how the scrum and CPM approach aid in addressing the challenge brought about by a software project, which is the case study that will be used as our analysis medium title: “Human Resource Management system with Biometric Attendance” or HRMwBA in lieu of a better acronym.

### **2. Background**

#### **2.1 Software Project "HRMwBA":**

The project HRMwBA is a web-based application development project created to address the need for stream-lining human resources management within the Gemadept corporation. The application will consist of all the human resource management needs from payroll, employees information tracking, to timely visualised report on the employee's progress.

This application will be further improved by adding a fingerprint scanner-based biometric attendance system. With the use of this technology, the HR department can precisely track when employees clock in, which makes it possible to calculate their actual work hours for a variety of purposes, including payroll and performance reviews. As a result, this is an improvement over conventional HRM systems, where regulation is lax, and attendance is frequently not formally documented.

The objectives set the stage for the team coming up with the features that would satisfy the requirement for this project:

- Employee Information Management.
- Recruitment and Applicant Tracking.
- Onboarding and Offboarding.
- Payroll Management.
- Time and Attendance Tracking.
- Reporting and Analytics.
- Performance Management.
- Learning and Development.
- Integration with Existing Systems.

These features are expected to be achieved in a period of 4 sprints each lasting around 4 weeks (21 days) and have around 5 to 6 products to be done. As of the time of this research, our team had successfully developed our first sprint and created a usable first version of the product.

## **2.2 Overview of Scrum:**

A method that is based on the agile framework, scrum according to Schwaber and Beedle [3] is an empirical approach that reintroduces flexibility, adaptability and increased productivity into system development. The Scrum process cuts through complexity and focuses exclusively on building software that meets business needs.

Scrum breaks a project down into iterations of fixed duration (one to four weeks) called Sprints. And personnel can assume one of the three main roles.

- Product Owner: responsible for the product vision represented by a Product Backlog.
- The Scrum / Delivery team: a group of people that is in charge of developing the software.
- The Scrum Master: guides the team through the process and helps the team to resolve issues that may occur.

Scrum gained its effectiveness [4] since it promotes regular inspection and adaptation through a variety of activities like frequent meetings and check ups to make sure everyone stays on progress, and work from everyone can be synchronized better. Moreover at the end of each sprint, retrospective review or sprint demo that would be used as reports on progress of the project.

These elements combined well with one another and create a very realistic, effective and efficient model that is loved by every project manager in the world.

### **2.3 Overview of Critical Path Method (CPM):**

The Critical Path Method, a traditional software development methodology used to determine the critical path in a project's network, project duration, and efficient use of resources. CPM is used to plan and manage complex projects with interdependent activities as it follows a sequential approach, meaning that in order to achieve one product, the prior product must be done first.

In CPM, the project is represented as a network of activities [5], each with its duration and dependencies on other activities. The critical path in a CPM is meant to figure out the longest sequence of activities that determines the minimum duration of the project, which means that the duration of the project is fixed and changes within any activities will directly translate to the project's overall duration. Management bodies can leverage CPM's critical path analysis to identify critical activities and allocate resources accordingly to ensure the smooth delivery of the project.

Just like any traditional project management framework, CPM would perform especially well in projects that have well defined scope and detailed plan from the get-go, or within an environment of robust communication and strong commitment [6]. Which is one of the reasons why software developed with this method does not have any extra features but does carry a higher chance of delivering software to clients.

## **3. Discussion**

### **3.1 Main Challenges of the Software Project:**

Throughout the development of the HRMwBA project, several challenges emerged like (1) *Tight deadlines* as we originally devised each sprint in our project to be 4 weeks long, however, due to time constraint, we must shorten the time to 2 weeks, which made our time get more tight. (2) *Expanded scope* because we are determined to incorporate one of our later sprint's into the first sprint midway through the project, which made our scope for sprint 1 expanded a little. (3) *Rushed development* due to time constraints, hence, we must rush our website's development. (4) *Knowledge deficit* among team members, as our team members studied IT, hence, we have little knowledge when it comes to human resource management, which is why it took us quite some time to research and develop a plan that would suit HRM business needs.

### **3.2 Scrum's Approach to Addressing Challenges:**

It is not surprising to learn that scrum helped us address almost all of our problems with our HRMwBA project. Because we designed our product based on scrum's product backlog, we can adjust the priority of the products at any time during the project. Which allows us to develop the product sooner than planned.

*Figure 3.2.1: Original Product Backlog*

*Figure 3.2.2: Adapted Product Backlog*

We also apply scrum's frequent meeting practice and conduct meetings twice to thrice a week to discuss what each member of the team is doing, and what news there to be had. This led to the team being able to adjust the time for each product as well as speed up the human resource research for all of us as we can exchange our findings to one another within those meetings.

*Figure 3.2.3: Meeting minute summary example*

### **3.3 CPM's Approach to Addressing Challenges:**

The CPM analysis did help us combat the rushed development drawback of our project. By creating a work breakdown structure (WBS) we were able to determine the most critical product that needed to be done, and it helped us know what task we need to be doing, how long it should take, these elements and make our development move more smoothly.

*Figure 3.3.1: WBS for one of the products*

Even though this style of project management can pull through when detailed objectives are made, should we pursue this approach from the get-go, we would not be able to develop our website how we envisioned it and will slow down development as we need to progress each product step-by-step. Despite this benefit, it can be easy to overestimate or underestimate the time it could take to carry out any product, which makes time planning of this approach unrealistic at times [7].

### **3.4 Comparison and Conclusion:**

Within the HRMwBA project, Scrum's characteristics like adaptability, flexibility, and its practice of regular discussion did create more work for us to do and expanded our original scope a little. However, it made up for this by addressing most of our main challenges with those same characteristics. In addition, this method allows us to develop our sprint's product however we like and does not need to rely on complex dependencies of other products.

Even though CPM's approach was not useless as it helped us pull through our rushed schedule with its scheduling, it does not prove as significant as Scrum. It was only taken into account when doing initial planning for each product (which changes as the project

goes on) and diving head first into creating an intricate network of tasks and dependencies will be too complex and cost too much precious time and resources for our team.

#### 4. Conclusion

Through the case study of the Human Resource Management system with Biometric Attendance, it's evident that the Scrum methodology outperforms the CPM approach due to its ease of use and other characteristics. Scrum's adaptability and simplicity make it well-suited for the rapid development of software [8], especially in our project's context. In contrast, CPM's reliance on extensive planning and complex task networks does not align well with the dynamic nature of today's software development landscape and our project's specific needs.

#### References

- [1] R. S. Pressman, *Software Engineering: A Practitioner's Approach*. McGraw-Hill Science, Engineering & Mathematics, 2010.
- [2] R. K. Wysocki, *Effective Project Management : Traditional, Agile, Extreme*. Somerset: Wiley, 2011.
- [3] K. Schwaber and M. Beedle, *Agile Software Development with Scrum*. Pearson, 2002.
- [4] C. Scharff and R. Verma, "Scrum to support mobile application development projects in a just-in-time learning context," *Proceedings of the 2010 ICSE Workshop on Cooperative and Human Aspects of Software Engineering - CHASE '10*, 2010, doi: <https://doi.org/10.1145/1833310.1833315>.
- [5] J. E. Kelley and M. R. Walker, "Critical-path planning and scheduling," *Papers presented at the December 1-3, 1959, eastern joint IRE-AIEE-ACM computer conference on - IRE-AIEE-ACM '59 (Eastern)*, 1959, doi: <https://doi.org/10.1145/1460299.1460318>.
- [6] P. A. Hofmann, "Critical Path Method: An Important Tool for Coordinating Clinical Care," *The Joint Commission Journal on Quality Improvement*, vol. 19, no. 7, pp. 235–246, Jul. 1993, doi: [https://doi.org/10.1016/s1070-3241\(16\)30004-9](https://doi.org/10.1016/s1070-3241(16)30004-9).
- [7] M. Lu and H. Li, "Resource-Activity Critical-Path Method for Construction Planning," *Journal of Construction Engineering and Management*, vol. 129, no. 4, pp. 412–420, Aug. 2003, doi: [https://doi.org/10.1061/\(asce\)0733-9364\(2003\)129:4\(412\)](https://doi.org/10.1061/(asce)0733-9364(2003)129:4(412)).

[8] P. Adi, "Scrum Method Implementation in a Software Development Project Management," *International Journal of Advanced Computer Science and Applications*, vol. 6, no. 9, 2015.

## TASK 15P: Peer Review

### SWE30010 Managing IT Projects

#### Self and Peer Review Assessment Form [Sprint #1]

Date: 26/3/2024

Your Team: Group 2

Your Name: Nguyen Dinh Nhat Minh

Use the instructions (see below) to fill in scores for each category A to J.

Team Members (Name)	A	B	C	D	E	F	G	H	I	J	Total
Self	3	3	2	2	2	3	4	3	3	2	2.7
Nguyen Nhat Huy	4	4	3	4	4	5	5	4	4	4	4.1
Le Hoang Hai	5	3	4	5	4	4	5	4	5	4	4.3
Nguyen Ngoc Minh Thy	3	3	3	3	5	4	4	4	3	4	3.6

**Your Reasoning / Justification** (You must write a paragraph about each team member below. Incomplete reviews will not be accepted.)

Name, student number	Comments (complete sentences required)

Self	A basic leader, does enough for the team to strive. Can use improvement in time management and quality of work (writing, creating plan, etc.) can definitely use improvement. Overall, still achieve the necessary means that allow the team do their best.
Nguyen Nhat Huy	Huy is an excellent designer and writer, his contribution of fine tuning the structure of our assignments, creating quality mockups and suggest improvement of our writing had made our team's quality increased. It is safe to say that Huy can benefit from managing his time better, and voicing his idea more to the group. All in all, he is a solid team player and emits a sense of professionalism in his work.
Le Hoang Hai	Hai's is a fast thinker and contribute a myriad of ideas when facing any problems/tasks. He definitely fits the "do"-ers type of people as he had put a great emphasis on creating the codes for our project. Hai is a person that needs a positive environment in order to thrive, as the opposite could leads to his ineffectiveness. Overall, this guy had help push the process of doing assignment a ton faster and with his coding effort, made our task much faster and easier.
Nguyen Ngoc Minh Thy	Thy can be said to be the glue that holds our group together, should there be discourse among our self, she usually comes out and suggest finding the middle ground. She is also a fast and varied worker, able to deliver the finished any task thrown at her rather quickly. Thy can use some improvement in her quality of work, and discuss troublesome tasks to the group. However, she proves to be agile, being able to work on various task and has a good sense of responsibility in her work.

# **Group Tasks:**

## **TASK 02P: Scope and Product Backlog**

### **Synopsis**

Gemadep Corporation is a leading logistics and shipping company headquartered in Ho Chi Minh City, Vietnam. With a strong commitment to excellence, we are seeking a comprehensive Human Resource Management Software System to enhance our HR operations.

We would like to have a complete HR Management Software System that meets our specific requirements and integrates seamlessly with our existing systems. This system will be expected to streamline HR processes, increase efficiency, and improve employee management.

### **1. Background**

Gemadep, a leading logistics and shipping company headquartered in Ho Chi Minh City, Vietnam, is requesting a HRM software to streamline HR processes, check attendance of employees, employee management, and improve overall organizational efficiency. The primary objective of this proposal is to introduce a robust Human Resource Management (HRM) system tailored specifically for Gemadep. This proposal aims to establish the creation of a comprehensive HRM infrastructure, spanning the stages of design, development, and maintenance.

### **2. Scope**

#### **a. Objectives:**

The main goal of this solution is to build a complete HRM system that is further improved by adding a fingerprint scanner-based biometric attendance system. With the use of this technology, the HR department can precisely track when employees clock in, which makes it possible to calculate their actual work hours for a variety of purposes, including payroll and performance reviews. As a result, this is an improvement over conventional HRM systems, where regulation is lax, and attendance is frequently not formally documented.

#### **b. Solutions:**

To achieve the objectives listed, this software will include but not limited to these key features:

- Employee Information Management: Centralized database for storing and managing employee data, such as personal details, employment history, and contact information.
- Recruitment and Applicant Tracking: Tracking of applied applications and managing the recruitment pipeline.

- Onboarding and Offboarding: Structured procedures for integrating new hires into the organization and managing the exit process for departing employees.
- Payroll Management: Adjustable automated calculation of employee salaries, deductions, and taxes.
- Time and Attendance Tracking: Uses a fingerprint scanner for monitoring employee attendance, tracking work hours, etc.
- Reporting and Analytics: Generation of customizable reports and dashboards to provide insights into employee's information and status.
- Performance Management: System for setting performance goals, conducting evaluations, and providing feedback to employees.
- Learning and Development: Scheduling training sessions and tracking employee progress in skill development.
- Integration with Existing Systems: Integration with other business systems such as ERP and accounting software.

### **3. Deliverables and schedule**

#### **a. Deliverables Overview:**

Our goal is to develop a comprehensive prototype that includes both the HRM application and the fingerprint scanner. The application will be accessible via a website and will have the capability to store, update, and remove employee information. To ensure scalability, we will store data from both the website and the scanner in a scalable database.

#### **b. Schedule:**

This project is expected to take around 12 weeks (3 months) to completed and the timeline for each stage of the project is outlined as follow:

- System Design and Prototyping: 2 weeks.
- Development and Testing: 6 weeks.
- Implementation and Training: 4 weeks.
- Post-implementation support and maintenance: On-going from go live date.

## Initial Release Schedule of the Product backlog items

No.	Item	Dependencies	Business Value (1 least – 10 most)	Release Schedule (Sprint 1   2   3   ...)
F1	Product UI/UX Design	None	7	Sprint 1
F2	Website for HRM	None	9	Sprint 1
F3	HR database schema design	F2	8	Sprint 1
F4	Manager Portal	F1, F3	7	Sprint 1
F5	Employee Feedback and Survey Module	F1, F3	6	Sprint 2
F6	Employee information form implementation	F3	8	Sprint 2
F7	Fingerprint scanner implementation/installation	F3	8	Sprint 2
F8	Recruitment module development	F3	7	Sprint 2
F9	Payroll calculation module development	F3	7	Sprint 2
F10	On and Offboarding template implementation	F3, F6	6	Sprint 3
F11	Leave request feature implementation	F3, F6	6	Sprint 3
F12	Performance evaluation module development	F3, F6	6	Sprint 4
F13	Implement reporting dashboard	F9	8	Sprint 4

# **TASK 04P: Solution Direction and Design**

## **1. SOLUTION DIRECTION**

### **a. Problem Domain**

An attendance management system is a cloud-based HR tech tool that enables organizations to automate their attendance management operations and keep track of employee working hours. It allows employees to mark their attendance through its website integrated with the biometric attendance system to sync attendance entries automatically. These attendance entries are centralized in a single system and can be accessed by HR teams from anywhere, at any time. The key aspects of the problem domain include:

- **Attendance Tracking and Accuracy:** Ensuring accurate recording of employee attendance, clock-ins, and clock-outs.
- **Integration with Payroll:** Seamlessly integrating attendance data with payroll systems.
- **Compliance and Legal Requirements:** Adhering to labor laws, regulations, and company policies related to attendance.
- **User-Friendly Interfaces:** Designing intuitive interfaces for employees to mark attendance and for HR teams to manage records.
- **Shift Scheduling and Management:** Efficiently managing employee shifts, rotations, and work schedules.
- **Biometric Integration and Security:** Integrating biometric systems (such as fingerprint or facial recognition) for attendance tracking.
- **Reporting and Analytics:** Generating insightful reports on attendance trends, absenteeism, and productivity.
- **Remote Work and Geolocation:** Managing attendance for remote workers and tracking geolocation-based clock-ins.
- **Employee Self-Service:** Empowering employees to view their attendance records, apply for leaves, and manage time off.

In addition to the core functionalities, the problem domain of HRM system also focuses on:

- **Security and Privacy:** Ensuring the protection of biometric data and preventing unauthorized access or misuse.
- **Device Maintenance and Calibration:** Regularly maintaining and calibrating biometric devices for accurate readings.
- **User Training and Adoption:** Educating employees effectively to maximize system utilization.
- **Handling Exceptions and Anomalies:** Addressing unexpected situations, system downtime, or disputes.

- **Data Retention and Archiving:** Properly storing and managing historical attendance records.
- **Scalability for Large Organizations:** Designing the system to accommodate workforce growth.
- **Real-Time Alerts and Notifications:** Promptly notifying HR of unusual attendance patterns.

### b. Solution Domain

#### i. The Discussion of Alternative Solutions

In this section, we will explore different solution directions such as web-based applications, desktop-based applications, mobile applications or hybrid solutions (web + mobile). Considering the unique requirements of HRM and attendance management, we decided to develop a web-based application, other alternatives were discarded due to the following limitations:

Alternative	Discard reasons
Desktop-based application	<ul style="list-style-type: none"> <li>- Tie users to specific workstations, limiting their ability to work remotely or access HRM features from different devices.</li> <li>- Require manual updates on each machine where they are installed.</li> <li>- Attendance systems can be susceptible to time fraud, such as “buddy punching,” where employees clock in for absent colleagues.</li> <li>- Rely on local hardware and infrastructure, making them less flexible for distributed teams or remote work.</li> <li>- May lack the robust security features found in modern web or mobile apps.</li> <li>- Ensuring data security and privacy on individual workstations can be challenging.</li> </ul>
Mobile application	<ul style="list-style-type: none"> <li>- Mobile apps are vulnerable to malware, data breaches, and unauthorized access.</li> <li>- Mobile apps may not work on all devices (e.g., older phones, different operating systems).</li> <li>- Approval process and restrictions can be time-consuming.</li> <li>- Updates and compatibility issues across various devices.</li> </ul>

Hybrid solution	<ul style="list-style-type: none"> <li>- Integrating web and mobile components can be challenging.</li> <li>- Requires managing both web and mobile aspects.</li> <li>- Hybrid solutions may not perform as well as native apps.</li> </ul>
-----------------	---

The Web-Based Application offers the best balance of flexibility, scalability, ease of maintenance, and security. It supports remote work, aligns with modern work practices, and has existing successful competitors. While other solutions have merits, the web-based approach provides the most comprehensive benefits for HRM and attendance management.

Besides, we will deploy the web-based approach on **AWS** instead of other hosting options, such as a dedicated server, or on-premises as it is requested by client, and AWS is also more flexible than other ways in many aspects:

Criteria	AWS	Dedicated server	On-premises
<b>Cost</b>	Variable cost: Pay-as-you-go pricing based on usage. Initial setup costs may be lower.	Fixed cost: Requires purchasing or leasing dedicated hardware. Higher initial investment.	Fixed cost: Requires purchasing and maintaining hardware infrastructure.
<b>Scalability</b>	Highly scalable: Easily adjust resources based on demand.	Limited scalability: Hardware capacity is fixed. Scaling requires additional servers.	Limited scalability: Hardware capacity is fixed. Scaling requires new hardware.
<b>Maintenance</b>	Managed service: AWS handles maintenance, updates, and security.	Self-managed: You're responsible for maintenance, updates, and security.	Self-managed: You're responsible for all aspects of maintenance.
<b>Reliability</b>	Highly reliable: AWS data centers are distributed globally.	Depends on provider: Varies based on service level agreements.	Depends on organization: In-house infrastructure reliability.
<b>Flexibility</b>	Highly reliable: AWS data centers are distributed globally.	Depends on provider: Varies based on service level agreements.	Depends on organization: In-house infrastructure reliability.

<b>Deployment Time</b>	Quick deployment: Provision resources within minutes.	Moderate deployment time: Requires setting up and configuring hardware.	Longer deployment time: Procurement, setup, and configuration.
<b>Location Independence</b>	Global availability: Accessible from anywhere with an internet connection.	Dependent on server location: Limited to physical server location.	Dependent on office premises: Limited to local network.
<b>Security</b>	AWS provides security features: Encryption, firewalls, IAM, etc.	Depends on provider: Security measures vary.	Depends on organization: In-house security policies and practices.

## ii. The KoST Analysis of My Knowledge

### 1. Knowledge (*what you know*)

#### Required knowledge

- **Web-based application:** Proficiency in HTML, CSS, and JavaScript for creating user interfaces, knowledge of server-side programming languages such as Python, Java or Node.js
- **Desktop-based application:** Proficiency in C++, Java, and C#.
- **Hosting server:** Expertise in server management, networking, databases, and understanding of hosting services.
- **Biometric Integration:** Learn about integrating biometric devices (e.g., fingerprint scanners) with your application. Understand how to capture and process biometric data securely.
- **HR Management Basics:** Possess basic knowledge of HR management through engagement in volunteer operations involving personnel allocation and administration.

#### My team's Knowledge

- All of our team have members that enrolled in two web development courses; hence, they can understand the basic understanding of HTML, CSS, JavaScript and PHP.
- Two of our team members had enrolled in IoT programming and can develop IoT related projects.
- Two of us were enrolled in the AWS architecture designing course. Which prompts understanding of networking, DBMS and hosting services.
- One team member is in the HR sector of a club. Which helps with our domain problem.

### 2. Skills (*your experience*)

## Required skills

### Technical Skills:

- Understanding of responsive design principles.
- Ability to develop APIs for communication between front-end and back-end.
- Familiarity with database management (e.g., MySQL, PostgreSQL, or MongoDB).
- Understanding of server inspection and deployment into production environments.
- Knowledge of how to acquire necessary infrastructure (e.g., cloud hosting, virtual servers).
- Experience in creating intuitive and user-friendly interfaces for HRM applications.

### Domain knowledge:

- Understanding of HR functions related to attendance tracking, leave management, and payroll.
- Understanding of the benefits and challenges of using biometrics for attendance management.
- Familiarity with HRM processes specific to attendance, employee records, and time tracking.
- Knowledge of how biometric attendance fits into the overall HR ecosystem.

## My team's skills

- All of us have created and hosted a few websites as our side and university projects.
- One of us had UX/UI designing experiences and is proficient in Adobe XD, Figma.
- Another one had created and managed databases using SQL developer by Oracle.
- The IoT proficient member had built various IoT machines that function well.
- We have had experience observing the HRM pipeline and process of a club.

### **3. Technology (*existing solution*)**

Various contemporary technologies and solutions can be leveraged to develop web-based HRM systems. Key among these are:

- **Proprietary HRM Software:** Esteemed companies such as SAP (SuccessFactors), Oracle (Oracle HCM Cloud), and Workday present comprehensive HRM suites encompassing diverse HR functions. These solutions typically include modules for core HR, talent management, workforce planning, payroll, benefits administration, and analytics.
- **Cloud-based HRM Platforms:** Scalable and adaptable solutions for managing HR processes are offered by cloud-based HRM platforms like BambooHR, Namely, Zenefits, and Gusto. These platforms commonly feature employee self-service portals, tools for onboarding and offboarding, performance management, time and attendance tracking, as well as compliance management.
- **Open-source HRM Solutions:** Flexibility and customization are core benefits of open-source HRM solutions, making them ideal for constructing web-based HRM applications. Well-regarded open-source HRM frameworks and platforms include OrangeHRM, Sentrifugo, IceHrm, and Odoo HR. These solutions provide modules for managing employee information, leave management, attendance tracking, and more.
- **HRM Modules within ERP Systems:** Enterprise Resource Planning (ERP) systems like Microsoft Dynamics 365 and SAP ERP often include HRM modules alongside other business functions such as finance, supply chain, and manufacturing. These integrated solutions provide a unified platform for managing HR processes within the broader context of organizational operations.

### **Rationale**

After a gap analysis using the KsoT framework and a comparison analysis of several solutions, the following factors suggest that creating a web application is the best course of action for this project:

- Web-based apps can be easily integrated and blended in with the existing system because they are easily accessible from a variety of devices and have an intuitive interface.
- The platform can grow to meet the needs of its users thanks to the scalability features offered by Firebase and Google cloud.
- Our team is fairly competent in producing a project this scale.

### **iii. Final Solution**

- The HRM application will be developed as a web-based application with a front-end built with Vue JS and a back-end utilizing a combination of Node Js and Express Js.

- The HRM Application will be developed using Nginx Web Servers, with Firebase hosting the back end. This combination will ensure the necessary scalability and reliability for this project.
- Google Spreadsheet on Google Cloud will be utilized to store user data due to its exceptional capacity to manage vast amounts of data. Additionally, Google's security and compliance measures are of the highest caliber, helping to safeguard critical user information.

## 2. Architecture Design

### a. 3-Tier Architecture

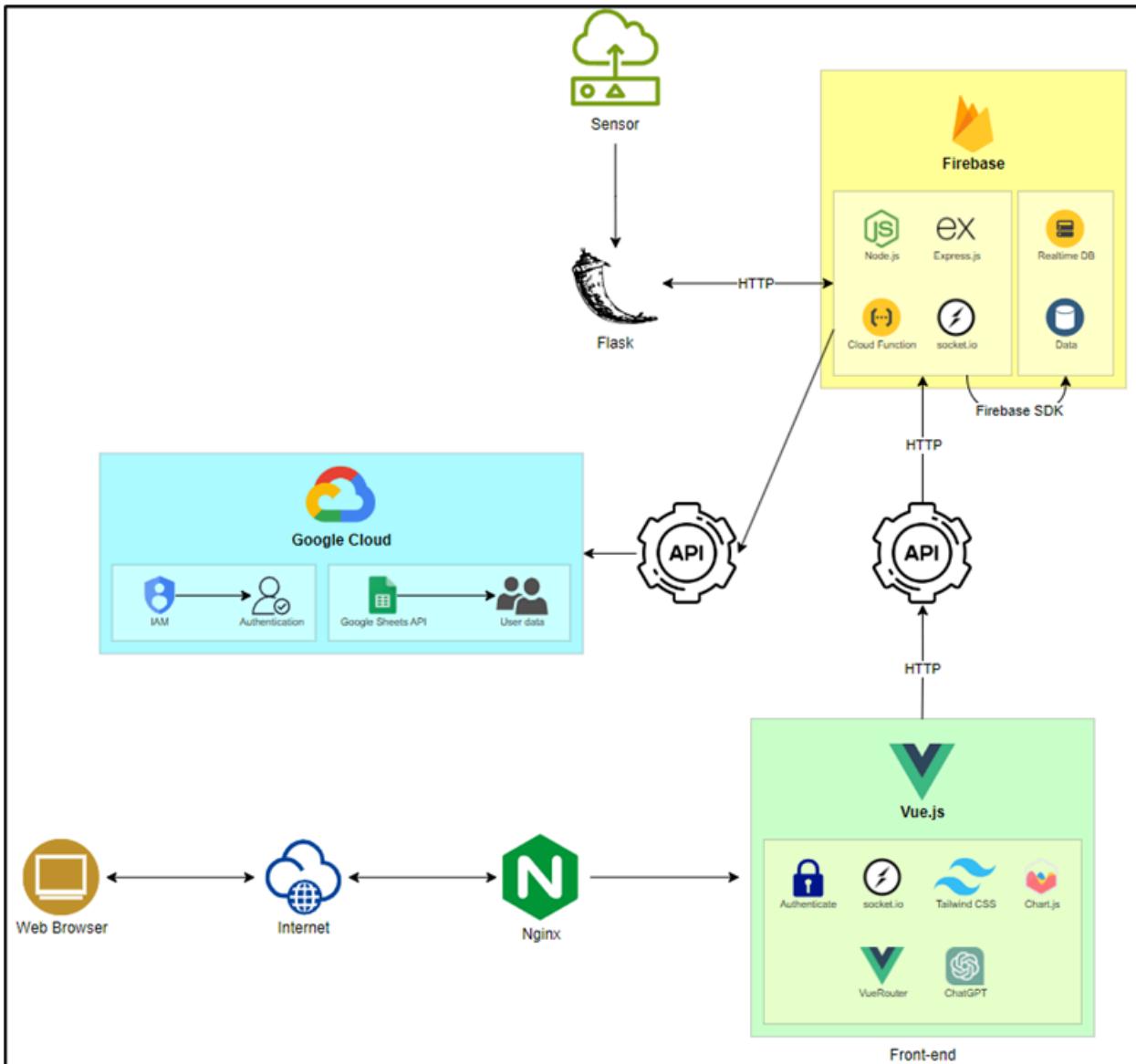


Figure A: 3-tier Website Application Design

#### TIER 1: PRESENTATION

- Cloudflare: Provides secure DNS and website optimization services.
- Vue.js: A progressive JavaScript framework for building user interfaces.
- Vue Router: For client-side routing within the Vue.js application.
- Vuex: State management library for Vue.js applications.
- Vue Auth: Authentication library for handling user authentication.
- Socket.io: Enables real-time client-server communication for notifications.
- Tailwind CSS / Bootstrap: CSS frameworks for styling user interfaces.

- Chart.js: JavaScript library for creating charts and graphs.

## **TIER 2: APPLICATION**

Application 1:

- Flask: A micro web framework written in Python.
- Biometric Attendance: Module for retrieving employee attendance using biometric data.
- Flask API: Creates APIs to facilitate communication between servers.

Application 2:

- Node.js: A JavaScript runtime environment for executing server-side code.
- Express.js: A web application framework for Node.js.
- Socket.io: Enables real-time server-client communication for notifications.
- RESTful API: Provides endpoints for interacting with the application.
- Real-Time Database (RTDB): Firebase's NoSQL database for real-time data storage, suitable for IoT data.
- Google Spreadsheet: Integration with Google Sheets for data storage.
- IAM Service: Registers and manages access permissions for connecting to Google Sheets.

## **TIER 3: DATABASE**

- Google Spreadsheet: Google Sheets used as a database for storing structured data.
- Real-Time Database (RTDB): Firebase's NoSQL database for real-time data storage.

## b. UML DESIGN

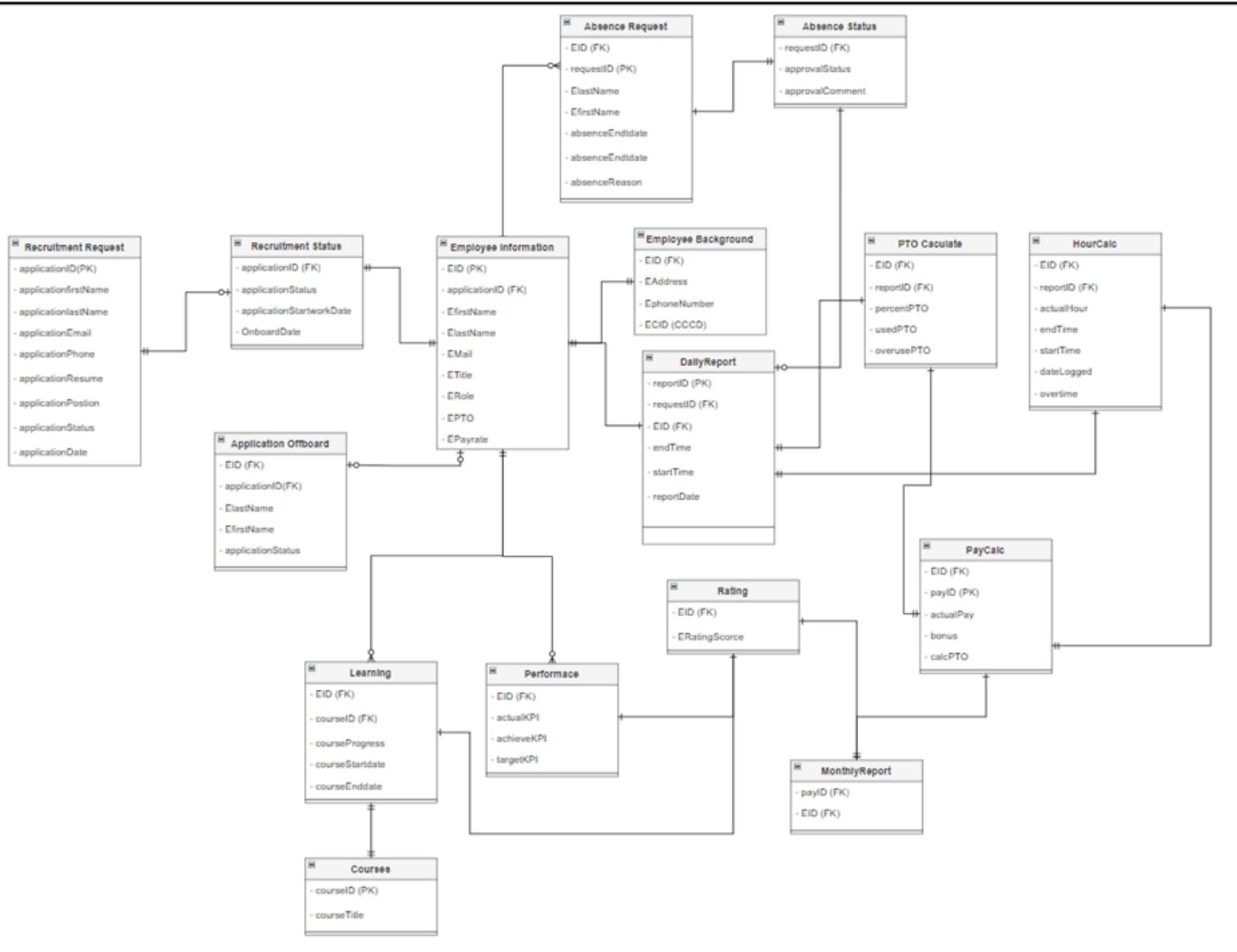


Figure B: UML Design

# **TASK 06P: Software Quality and Definition of Done**

## **1. QUALITY MANAGEMENT**

### **1.1. What is Quality?**

Within the framework of this HR management system with a biometric attendance project, a "quality" platform is defined as the ultimate platform that satisfies all of the client's criteria. The following is a summary of these requirements:

#### **a) Technical requirements**

- The system should seamlessly integrate with biometric devices (such as fingerprint scanners, iris recognition, or facial recognition cameras) to capture employee attendance data, ensure compatibility with various biometric technologies and models.
- Store and manage employee data, attendance records, and biometric templates securely in a database, implement efficient data retrieval and update mechanisms.
- Utilize TCP/IP, wireless, fiber, or copper networks for data transfer between biometric devices and the central system.
- Design the system to handle a growing number of employees and devices.

#### **b) Functional Requirements**

All core functionalities of the Biometric Attendance system can work efficiently and do not have any severe errors that corrupt the system. Core functionalities include:

- Biometric Attendance Tracking
- Employee information Management and Leave Management
- Payroll Integration
- Reporting and Analytics
- Security Enhancement
- Time check and attendance
- Performance and Development tracking.

#### **c) Non-Functional Requirements**

- Ensure data privacy and protection of biometric templates, implement encryption during data transmission and storage, prevent unauthorized access.
- Maintain high system availability (uptime) to avoid disruptions, regularly back up data and enable efficient recovery mechanisms.

- Response time for biometric authentication should be fast, handle concurrent requests efficiently and correct.
- Provide user manuals, technical guides, and change logs, document system architecture and design.

## 1.2. What is Quality?

The Definition of Done (DoD) checklist can be used to assess the quality of Biometric Attendance system development. Before software development is considered complete, this checklist consists of a set of conditions that have to be met. There are several items on the DoD checklist before the release, including:

- Core functionalities development: The platform must be constructed with all of the essential features listed in the project proposal. Both automated test cases and manual tests can be used throughout the testing process.
- Usability and Scalability: Consider factors like reaction time and resource usage while assessing the software's performance under workloads. Verify that the system is capable of handling an unexpected increase in workload.
- Code Quality and Conventions: Verify that the code complies with the requirements for readability, maintainability, naming conventions, and code structure.
- Security: Verify if the platform followed the necessary data protection and authentication guidelines. Performing security testing, reviewing protocols, and reviewing IT policies and documentation are all part of this process.
- Testing and Documentation: Conduct unit testing to verify individual components, integration testing to validate interactions between modules, and user acceptance testing (UAT) involving end-users.
- UX/UI Testing: Make that the UI is responsive, easy to use, and visually consistent. It ought to satisfy every design criterion listed in the proposal.

There are several methods and metrics for assessing quality when it comes to measuring. As a result, the measuring process is standardized in many quality models. IEEE 1061, COSMIC, ISO/IEC 9126, and ISO 25010 are a few of the widely used models. Whichever quality evaluation technique is selected, the DoD Checklist's evaluation criteria must be properly formatted, comprehensible, and appropriate. The DoD can be improved by using S.M.A.R.T. concepts.

## 2. CHECKLIST

When using ISO 25010 as a framework for an HRM (Human Resource Management) project, you would tailor your project's definition and goals to address these quality characteristics in the context of HRM software or systems. For example:

- **Functional Suitability:** Ensuring that the HRM software provides all necessary functions to support HR processes such as recruitment, employee onboarding, performance management, and training.
- **Performance Efficiency:** Ensuring that the HRM software performs tasks such as data processing and report generation within acceptable time frames and with minimal resource consumption.
- **Compatibility:** Ensuring that the HRM software can integrate with existing systems such as payroll software, time-tracking systems, and enterprise resource planning (ERP) systems.
- **Usability:** Ensuring that the HRM software is user-friendly and intuitive, allowing HR professionals and employees to easily navigate and perform tasks within the system.
- **Reliability:** Ensuring that the HRM software operates reliably without frequent crashes or errors, especially during critical HR processes.
- **Security:** Ensuring that the HRM software protects sensitive HR data such as employee personal information, payroll details, and performance evaluations from unauthorized access or modification.
- **Maintainability:** Ensuring that the HRM software can be updated and modified efficiently to accommodate changes in HR policies, regulations, and business requirements without introducing bugs or issues.
- **Portability:** Ensuring that the HRM software can be deployed across different hardware and software environments, such as on-premises servers or cloud platforms, with minimal effort and modifications.

No.	(Sub) Characteristics	Description	Y/N
<b>1. Functionality Suitability</b>			
1.1	Functional Appropriateness	Deploy the platform on the cloud-based hosting server	<input type="checkbox"/>
1.2	Functional Appropriateness	The platform allows user creation for 2 account types: Admin account and employee account	<input type="checkbox"/>
1.3	Functional Appropriateness	The platform allows users to switch between different languages. All content, messages, and alternative text are translated correctly	<input type="checkbox"/>

1.4	Functional Appropriateness	The data analytics tool can generate reports with meaningful insights from the data into visualized format on the website.	<input type="checkbox"/>
1.5	Functional Appropriateness	The platform's core functionalities can integrate with external systems, such as current database, existing ERP system with less than 5% of API conflicts	<input type="checkbox"/>
1.6	Functional Appropriateness	The data analytics tool is integrated with the feedback mechanism to analyze manager feedback to employee	<input type="checkbox"/>
1.7	Functional Correctness	The number of errors of each core functionality of the HRM should be less than 5% of total automated test cases	<input type="checkbox"/>
1.8	Functional Correctness	During the account registration process, important fields such as name, password, and email are required	<input type="checkbox"/>
1.9	Functional Correctness	Employee's information updates correctly appear on the screen	<input type="checkbox"/>
1.10	Functional Correctness	The database searching and filtering options return the correct result	<input type="checkbox"/>
1.11	Functional Correctness	When calculation is made upon any one of the modules, the other with the related attribute must also update correctly.	<input type="checkbox"/>
1.12	Functional Correctness	Different test cases are performed to prevent XSS, exploiting URLs	<input type="checkbox"/>

1.13	Functional Correctness	Dashboard chart should be use properly to show a clearly details of user information	<input type="checkbox"/>
1.14	Functional Completeness	Manager can generate an user report of their attendance each month with a provided sample template with one click	<input type="checkbox"/>
1.15	Functional Completeness	Use the authenticated token to fetch the user details from the system, including user ID, role, and history information.	<input type="checkbox"/>
1.16	Functional Completeness	An process to verify if the authenticated token is expired or not, then alert user to login/ refresh token again	<input type="checkbox"/>
1.17	Functional Completeness	Use the authenticated token to fetch the order details from the system, including order ID, payment information, and shipping methods.	<input type="checkbox"/>
1.18	Functional Completeness	Perform a JWT authentication to obtain a valid token for the user who login the system	<input type="checkbox"/>
<b>2. Performance Efficiency</b>			
2.1	Time Behaviour	<p>The following actions have a response time of less than 5 seconds:</p> <ul style="list-style-type: none"> <li>● Display a list of employees.</li> <li>● Calculate and estimate the payroll.</li> <li>● Display dashboard for admin/ user role.</li> <li>● Show user information.</li> <li>● Generate report.</li> <li>● Login/Logout.</li> </ul>	<input type="checkbox"/>
2.2	Time Behaviour	The loading time of pages is less than 5 seconds.	<input type="checkbox"/>

2.3	Resource Utilization	The CPU utilization when running the platform is below 80% and memory consumption is below 70%	<input type="checkbox"/>
2.4	Capacity	The application can handle a 1,000 employee test case without losing 30% of the performance.	<input type="checkbox"/>
<b>3. Compatibility</b>			
3.1	Co-existence	Run simultaneously with other web applications on the same operating system without conflicts	<input type="checkbox"/>
3.2	Interoperability	Use widely accepted and standardized data formats for communication	<input type="checkbox"/>
3.3	Interoperability	Design clear and well-documented APIs (Application Programming Interfaces) for your web application.	<input type="checkbox"/>
<b>4. Usability</b>			
4.1	Appropriate recognizability	<p>Users complete the following tasks in less than 3 minutes:</p> <ul style="list-style-type: none"> <li>• Register a new account</li> <li>• Watch their personal information</li> <li>• Review their KPI through chart</li> <li>• Generate report</li> </ul>	<input type="checkbox"/>
4.2	Learnability	90% of users (admin staff and employee) can complete basic tasks to serve their needs on the first day of the testing period	<input type="checkbox"/>
4.3	Operability	The average number of errors that test users encountered during a 15- minute testing session is less than 2 errors.	<input type="checkbox"/>
4.4	User interface aesthetics	The platform is responsive for mobile and desktop.	<input type="checkbox"/>

4.5	User interface aesthetics	The minimum average satisfaction score of 20 users about the UX/UI of the platform, including web page structure, text font, color palette, languages, navigation, product display, buttons, etc. is 8 out of 10.	<input type="checkbox"/>
4.6	User error protection	90% of test users can understand the content of the confirmation dialog box displayed when they: <ul style="list-style-type: none"><li>• Delete/edit their review/rating</li><li>• Cancel/Confirm their orders</li><li>• Change the platform settings</li></ul>	<input type="checkbox"/>
4.7	Accessibility	80% of non-text content (video, images) have alternative text descriptions	<input type="checkbox"/>
<b>5. Reliability</b>			
5.1	Maturity	Mean Time Between Failures (MTBF) metrics of the platform is at least 30 days	<input type="checkbox"/>
5.2	Fault Tolerance	Mean Time to Recover (MTTR) takes less than 2 hours to recover platform data from errors or failures	<input type="checkbox"/>
5.3	Recoverability	All functionalities of the platform can be fully restored from the backup within 6s hours after the failure	<input type="checkbox"/>
5.4	Availability	The platform is accessible for 99.9% of operating hours.	<input type="checkbox"/>
<b>6. Security</b>			
6.1	Confidentiality	Block at least 95% of unauthorized access attempts and send alerts to users	<input type="checkbox"/>

6.2	Confidentiality	Data encryption and secure authentication protocols are integrated	<input type="checkbox"/>
6.3	Confidentiality	Payment card industry (PCI) compliance is applied	<input type="checkbox"/>
6.4	Integrity	100% of data is stored and transmitted correctly. Data in the databases is reflected correctly on the user interface and data analytics tools	<input type="checkbox"/>
6.5	Integrity	The platform follows data protection regulations	<input type="checkbox"/>
6.6	Non-repudiation	Implement non-repudiation measures for at least 95% of user actions	<input type="checkbox"/>
6.7	Accountability	100% of user interactions on the platform and the platform events are recorded in the logbook	<input type="checkbox"/>
6.8	Accountability	100% of incident occur on the system is recorded and notify to admin	<input type="checkbox"/>
6.9	Authenticity	95% of user authentication processes are successful	<input type="checkbox"/>
<b>7. Maintainability</b>			
7.1	Analyzability	Reduce time taken to diagnose and resolve reported issues to less than 1 hour 30 minutes	<input type="checkbox"/>
7.2	Modifiability	New change/enhancement is integrated into the platform 1 month after the change request is made	<input type="checkbox"/>

7.3	Testability	Maintain a code coverage of at least 80% through automated test cases	<input type="checkbox"/>
7.4	Modularity	The platform architecture is designed with at least 6 modules	<input type="checkbox"/>
7.5	Reusability	All codes have naming conventions, and comments to modify when needed without affecting other parts (Low coupling – High cohesion)	<input type="checkbox"/>
7.6	Reusability	A minimum of 80% code is reusable	<input type="checkbox"/>
<b>8. Portability</b>			
8.1	Adaptability	The time to deploy the platform on a new hosting server (on-premises or cloud server) is less than 4 days	<input type="checkbox"/>
8.2	Installability	The time to complete the installation and configuration of all platform components in an environment is less than 10 hours	<input type="checkbox"/>
8.3	Replaceability	Could we replace a component of the platform with a migration time of less than 2 weeks without impact on other components?	<input type="checkbox"/>

# **TASK 07P: Project Proposal**

## **1) Background / Problem Description**

Gemadept, a leading logistics and shipping company headquartered in Ho Chi Minh City, Vietnam, is requesting a HRM software to streamline HR processes, check attendance of employees, employee management, and improve overall organizational efficiency. The primary objective of this proposal is to introduce a robust Human Resource Management (HRM) system tailored specifically for Gemadept. This proposal aims to establish the creation of a comprehensive HRM infrastructure, spanning the stages of design, development, and maintenance.

## **2) Scope**

### **a) Objectives**

The main goal of this solution is to build a complete HRM system that is further improved by adding a fingerprint scanner-based biometric attendance system. With the use of this technology, the HR department can precisely track when employees clock in, which makes it possible to calculate their actual work hours for a variety of purposes, including payroll and performance reviews. As a result, this is an improvement over conventional HRM systems, where regulation is lax, and attendance is frequently not formally documented.

### **b) Solutions**

To achieve the objectives listed, this software will include but not limited to these key features:

- **Employee Information Management:** Centralized database for storing and managing employee data, such as personal details, employment history, and contact information.
- **Recruitment and Applicant Tracking:** Tracking of applied applications and managing the recruitment pipeline.
- **Onboarding and Offboarding:** Structured procedures for integrating new hires into the organization and managing the exit process for departing employees.
- **Payroll Management:** Adjustable automated calculation of employee salaries, deductions, and taxes.
- **Time and Attendance Tracking:** Uses a fingerprint scanner for monitoring employee attendance, tracking work hours, etc.
- **Reporting and Analytics:** Generation of customizable reports and dashboards to provide insights into employee's information and status.

- **Performance Management:** System for setting performance goals, conducting evaluations, and providing feedback to employees.
- **Learning and Development:** Scheduling training sessions and tracking employee progress in skill development.
- **Integration with Existing Systems:** Integration with other business systems such as ERP and accounting software.

### 3) Stakeholders

Stakeholder	Interest
Management	Oversee the implementation and usage of the HRM system and expect it to deliver the desired outcomes, which include smooth operation, effectiveness, and fit business functions.
HR Department	Primary users of the HRM software. A more efficient and effective HRM system is expected to streamline their operation and maximize their efficiency in their daily endeavor.
Employees	With the new HRM system, employees can focus more on working on their projects, as this application shall reduce the time it takes to manually input data on their own.
IT Department	Involved in the implementation and maintenance of the HRM software, ensuring the compatibility with existing infrastructures, addressing technical issues, and providing support to users.

Software Vendor	Assists Gemadept IT and business users with maintenance, updates, and issues of the product. They would also ensure to deliver a high-quality product that meets the requirements, standards, and expectations of Gemadept.
-----------------	---

## 4) Deliverables and schedule

### a) Deliverables overview:

Our goal is to develop a comprehensive prototype that includes both the HRM application and the fingerprint scanner. The application will be accessible via a website and will have the capability to store, update, and remove employee information. To ensure scalability, we will store data from both the website and the scanner in a scalable database.

### b) Schedule:

This project is expected to take around 12 weeks (3 months) to completed and the timeline for each stage of the project is outlined as follow:

- System Design and Prototyping: 2 weeks.
- Development and Testing: 6 weeks.
- Implementation and Training: 4 weeks.
- Post-implementation support and maintenance: On-going from go live date.

### Initial Release Schedule

No.	Item	Dependencies	Business Value (1 least – 10 most)	Release Schedule (Sprint 1   2   3   ...)
F1	Product UI/UX Design	None	7	Sprint 1
F2	Website for HRM	None	9	Sprint 1
F3	HR database schema design	F2	8	Sprint 1

F4	Manager Portal	F1, F3	7	Sprint 1
F5	Employee Feedback and Survey Module	F1, F3	6	Sprint 2
F6	Employee information form implementation	F3	8	Sprint 2
F7	Fingerprint scanner implementation/installation	F3	8	Sprint 2
F8	Recruitment module development	F3	7	Sprint 2
F9	Payroll calculation module development	F3	7	Sprint 2
F10	On and Offboarding template implementation	F3, F6	6	Sprint 3
F11	Leave request feature implementation	F3, F6	6	Sprint 3
F12	Performance evaluation module development	F3, F6	6	Sprint 4
F13	Implement reporting dashboard	F9	8	Sprint 4

## 5) Solution Direction

### a) Discussion of alternative solutions

In this section, we will explore different solution directions such as web-based applications, desktop-based applications, mobile applications or hybrid solutions(web + mobile). Considering the unique requirements of HRM and attendance management,

we decided to develop a web-based application, other alternatives were discarded due to the following limitations:

Alternative	Discard reasons
Desktop-based application	<ul style="list-style-type: none"> <li>• Tie users to specific workstations, limiting their ability to work remotely or access HRM features from different devices.</li> <li>• Require manual updates on each machine where they are installed.</li> <li>• Attendance systems can be susceptible to time fraud, such as “buddy punching,” where employees clock in for absent colleagues.</li> <li>• Rely on local hardware and infrastructure, making them less flexible for distributed teams or remote work.</li> <li>• May lack the robust security features found in modern web or mobile apps.</li> <li>• Ensuring data security and privacy on individual workstations can be challenging.</li> </ul>
Mobile application	<ul style="list-style-type: none"> <li>• Mobile apps are vulnerable to malware, data breaches, and unauthorized access.</li> <li>• Mobile apps may not work on all devices (e.g., older phones, different operating systems).</li> <li>• Approval process and restrictions can be time-consuming.</li> <li>• Updates and compatibility issues across various devices.</li> </ul>
Hybrid solution	<ul style="list-style-type: none"> <li>• Integrating web and mobile components can be challenging.</li> <li>• Requires managing both web and mobile aspects.</li> <li>• Hybrid solutions may not perform as well as native apps.</li> </ul>

The Web-Based Application offers the best balance of flexibility, scalability, ease of maintenance, and security. It supports remote work, aligns with modern work practices, and has existing successful competitors. While other solutions have merits, the web-based approach provides the most comprehensive benefits for HRM and attendance management.

## b) The KoST Analysis of My Knowledge

### i) Knowledge (*what you know*)

#### Required knowledge:

- **Web-based application:** Proficiency in HTML, CSS, and JavaScript for creating user interfaces, knowledge of server-side programming languages such as Python, Java or Node.js
- **Desktop-based application:** Proficiency in C++, Java, and C#.
- **Hosting server:** Expertise in server management, networking, databases, and understanding of hosting services.
- **Biometric Integration:** Learn about integrating biometric devices (e.g., fingerprint scanners) with your application. Understand how to capture and process biometric data securely.
- **HR Management Basics:** Possess basic knowledge of HR management through engagement in volunteer operations involving personnel allocation and administration.

#### My team's Knowledge:

- All of our team members had enrolled in two web development courses; hence, they can understand the basic understanding of HTML, CSS, JavaScript and PHP.
- One of our team members has enrolled in IoT programming and can develop IoT related projects.
- Two of us were enrolled in the AWS architecture designing course. Which prompts understanding of networking, DBMS and hosting services.
- One team member is in the HR sector of a club. Which helps with our domain problem.

### ii) Skills (*your experience*)

#### Required skills:

#### **Technical Skills:**

- Understanding of responsive design principles.
- Ability to develop APIs for communication between front-end and back-end.
- Familiarity with database management (e.g., MySQL, PostgreSQL, or MongoDB).
- Understanding of server inspection and deployment into production environments.
- Knowledge of how to acquire necessary infrastructure (e.g., cloud hosting, virtual servers).

- Experience in creating intuitive and user-friendly interfaces for HRM applications.

**Domain knowledge:**

- Understanding of HR functions related to attendance tracking, leave management, and payroll.
- Understanding of the benefits and challenges of using biometrics for attendance management.
- Familiarity with HRM processes specific to attendance, employee records, and time tracking.
- Knowledge of how biometric attendance fits into the overall HR ecosystem.

**My team's skills:**

- All of us have created and hosted a few websites as our side and university projects.
- One of us had UX/UI designing experiences and is proficient in Adobe XD, Figma.
- Another one had created and managed databases using SQL developer by Oracle.
- The IoT proficient members had built various IoT machines that function well.
- We have had experience observing the HRM pipeline and process of a club.

### **iii) Technology (*existing solution*)**

Various contemporary technologies and solutions can be leveraged to develop web-based HRM systems. Key among these are:

- **Proprietary HRM Software:** Esteemed companies such as SAP (SuccessFactors), Oracle (Oracle HCM Cloud), and Workday present comprehensive HRM suites encompassing diverse HR functions. These solutions typically include modules for core HR, talent management, workforce planning, payroll, benefits administration, and analytics.
- **Cloud-based HRM Platforms:** Scalable and adaptable solutions for managing HR processes are offered by cloud-based HRM platforms like BambooHR, Namely, Zenefits, and Gusto. These platforms commonly feature employee self-service portals, tools for onboarding and offboarding, performance management, time and attendance tracking, as well as compliance management.
- **Open-source HRM Solutions:** Flexibility and customization are core benefits of open-source HRM solutions, making them ideal for constructing web-based HRM applications. Well-regarded open-source

HRM frameworks and platforms include OrangeHRM, Sentrifugo, IceHrm, and Odoo HR. These solutions provide modules for managing employee information, leave management, attendance tracking, and more.

- **HRM Modules within ERP Systems:** Enterprise Resource Planning (ERP) systems like Microsoft Dynamics 365 and SAP ERP often include HRM modules alongside other business functions such as finance, supply chain, and manufacturing. These integrated solutions provide a unified platform for managing HR processes within the broader context of organizational operations.

### c) Rationale

After a gap analysis using the KsoT framework and a comparison analysis of several solutions, the following factors suggest that creating a web application is the best course of action for this project:

- Web-based apps can be easily integrated and blended in with the existing system because they are easily accessible from a variety of devices and have an intuitive interface.
- The platform can grow to meet the needs of its users thanks to the scalability features offered by Firebase and Google cloud.
- Our team is fairly competent in producing a project this scale.

### d) Final Solution

- The HRM application will be developed as a web-based application with a front-end built with Vue JS and a back-end utilizing a combination of Node Js and Express Js.
- The HRM Application will be developed using Nginx Web Servers, with Firebase hosting the back end. This combination will ensure the necessary scalability and reliability for this project.
- Google Spreadsheet on Google Cloud will be utilized to store user data due to its exceptional capacity to manage vast amounts of data. Additionally, Google's security and compliance measures are of the highest caliber, helping to safeguard critical user information.

## e) Architecture Design

### i) High Level Design

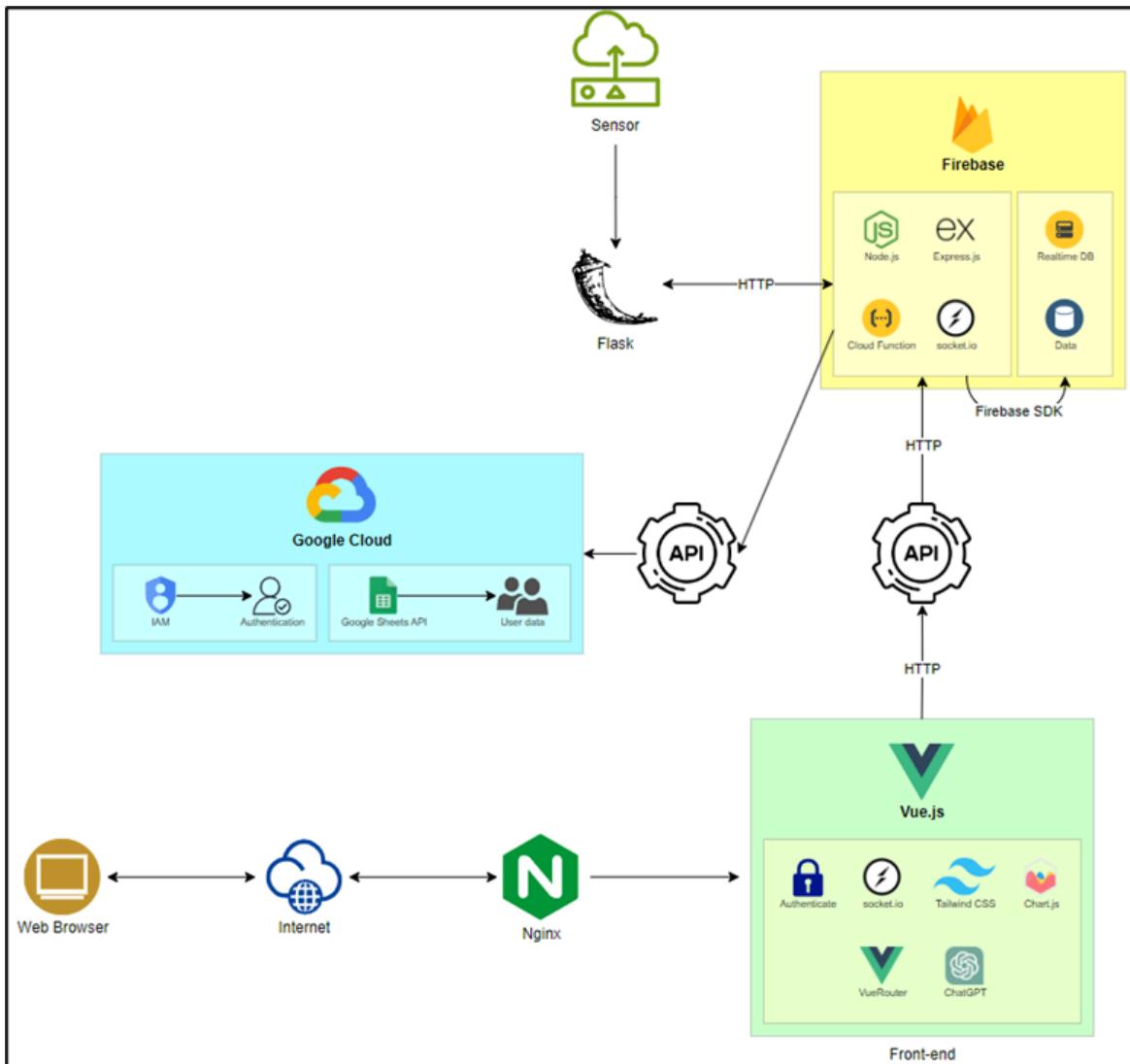


Figure A: 3-tier Website Application Design

### TIER 1: PRESENTATION

- **Cloudflare**: Provides secure DNS and website optimization services.
- **Vue.js**: A progressive JavaScript framework for building user interfaces.
- **Vue Router**: For client-side routing within the Vue.js application.
- **Vuex**: State management library for Vue.js applications.
- **Vue Auth**: Authentication library for handling user authentication.
- **Socket.io**: Enables real-time client-server communication for notifications.
- **Tailwind CSS / Bootstrap**: CSS frameworks for styling user interfaces.
- **Chart.js**: JavaScript library for creating charts and graphs.

## TIER 2: APPLICATION

### Application 1:

- **Flask:** A micro web framework written in Python.
- **Biometric Attendance:** Module for retrieving employee attendance using biometric data.
- **Flask API:** Creates APIs to facilitate communication between servers.

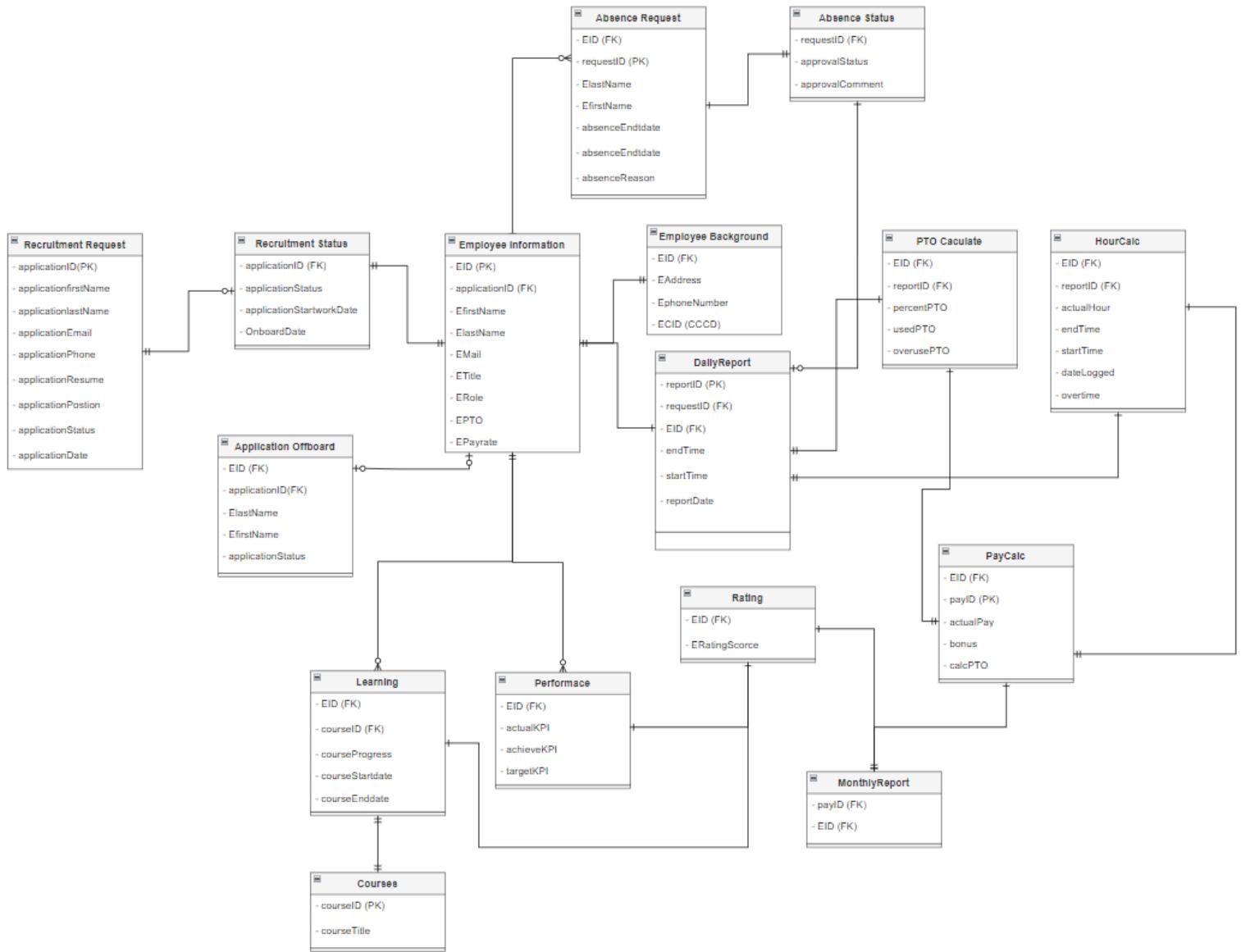
### Application 2:

- **Node.js:** A JavaScript runtime environment for executing server-side code.
- **Express.js:** A web application framework for Node.js.
- **Socket.io:** Enables real-time server-client communication for notifications.
- **RESTful API:** Provides endpoints for interacting with the application.
- **Real-Time Database (RTDB):** Firebase's NoSQL database for real-time data storage, suitable for IoT data.
- **Google Spreadsheet:** Integration with Google Sheets for data storage.
- **IAM Service:** Registers and manages access permissions for connecting to Google Sheets.

## TIER 3: DATABASE

- **Google Spreadsheet:** Google Sheets used as a database for storing structured data.
- **Real-Time Database (RTDB):** Firebase's NoSQL database for real-time data storage.

## ii) Database UML



## 6) Quality Management

Using the S.M.A.R.T guidelines that indicate that goals and achievement should be Specific, Measurable, Achievable, Realistic, and Time-bound. We decided to pursue and align our quality definition on the ISO 25010 standard, which indicates the following objective to ensure the quality of a given software:

- **Functional Suitability:** Ensuring that the HRM software provides all necessary functions to support HR processes such as recruitment, employee onboarding, performance management, and training.

- **Performance Efficiency:** Ensuring that the HRM software performs tasks such as data processing and report generation within acceptable time frames and with minimal resource consumption.
- **Compatibility:** Ensuring that the HRM software can integrate with existing systems such as payroll software, time-tracking systems, and enterprise resource planning (ERP) systems.
- **Usability:** Ensuring that the HRM software is user-friendly and intuitive, allowing HR professionals and employees to easily navigate and perform tasks within the system.
- **Reliability:** Ensuring that the HRM software operates reliably without frequent crashes or errors, especially during critical HR processes.
- **Security:** Ensuring that the HRM software protects sensitive HR data such as employee personal information, payroll details, and performance evaluations from unauthorized access or modification.
- **Maintainability:** Ensuring that the HRM software can be updated and modified efficiently to accommodate changes in HR policies, regulations, and business requirements without introducing bugs or issues.
- **Portability:** Ensuring that the HRM software can be deployed across different hardware and software environments, such as on-premises servers or cloud platforms, with minimal effort and modifications

With ISO 25010 as the reference, we create the detail requirement as the definition of done for our project as follow:

- **Core functionalities development:** The platform must be constructed with all of the essential features listed in the project proposal. Both automated test cases and manual tests can be used throughout the testing process.
- **Usability and Scalability:** Consider factors like reaction time and resource usage while assessing the software's performance under workloads. Verify that the system is capable of handling an unexpected increase in workload.
- **Code Quality and Conventions:** Verify that the code complies with the requirements for readability, maintainability, naming conventions, and code structure.
- **Security:** Verify if the platform followed the necessary data protection and authentication guidelines. Performing security testing, reviewing protocols, and reviewing IT policies and documentation are all part of this process.
- **Testing and Documentation:** Conduct unit testing to verify individual components, integration testing to validate interactions between modules, and user acceptance testing (UAT) involving end-users.
- **UX/UI Testing:** Make sure the UI is responsive, easy to use, and visually consistent. It ought to satisfy every design criterion listed in the proposal.

## 7) Resources

Within our team, we have four members that take on the following roles:

- **Facilitator - Minh:** Takes the lead, understands the assignment and assigns tasks to members based on their task appropriateness.
- **Initiator - Hai:** Comes up ideas, and suggests solutions, the main programmer (IoT, framework, etc.) for the project. He also implemented an API to the application to make it integrate with other software.
- **Compromiser - Thy:** A good negotiator to resolve the group's conflict and mismatch in the collaboration process. She keeps the group running smoothly and efficiently.
- **Critic - Huy:** The content writer, he makes suggestions on our assignment and fine tune small details that can lead to problems. He also is the one to work on the UI design for the application.

## 8) Approval Signatures

### a) Project Team

No.	Name of student	Student Id	Signature
1	Nguyen Dinh Nhat Minh	103802490	
2	Le Hoang Hai	103542974	
3	Nguyen Nhat Huy	103802911	

4	Nguyen Ngoc Minh Thy	103802791	
---	----------------------	-----------	---

### b) Project Sponsor [Your Tutor]

Tutor's name (on behalf of the client)	Signature
Thomas Hang	

## TASK 08P: Sprint Planning Meeting

### INTRODUCTION

In any Agile development project, the process of selecting backlog items for a sprint is crucial for ensuring the successful delivery of features and meeting stakeholder expectations. During the Sprint Planning Meeting, our team engages in discussions to prioritize product backlog items based on various factors. In this project, we will discuss the factors considered by the team and formulate criteria for prioritizing the product backlog items for our hypothetical project.

### BACKLOG

#### Initial Release Schedule of the Product backlog item

For this task, we focused on the first sprint of the full project. The items included in sprint one are as follow:

No.	Item	Dependencies	Business Value (1 least – 10 most)	Release Schedule
F1	Product UI/UX Design	None	7	Sprint 1

F2	Website for HRM	None	9	Sprint 1
F3	HR database schema design	F2	8	Sprint 1
F4	Manager Portal	F1, F3	7	Sprint 1
F5	Employee Feedback and Survey Module	F1, F3	6	Sprint 2
F6	Employee information form implementation	F3	8	Sprint 2
F7	Fingerprint scanner implementation/installation	F3	8	Sprint 2
F8	Recruitment module development	F3	7	Sprint 2
F9	Payroll calculation module development	F3	7	Sprint 2
F10	On and Offboarding template implementation	F3, F6	6	Sprint 3
F11	Leave request feature implementation	F3, F6	6	Sprint 3
F12	Performance evaluation module development	F3, F6	6	Sprint 4
F13	Implement reporting dashboard	F9	8	Sprint 4

## SPRINT DEVELOPMENT

### 1. Important Factors

There are several factors to consider when prioritizing a backlog item to work on for a Sprint. These factors are:

- Business Value: Business value refers to the importance of each backlog item in achieving project goals and delivering value to stakeholders.
- Development Effort: Development effort encompasses the time, resources, and complexity required to implement each backlog item.
- Feature Dependency: Feature dependency refers to the extent to which a backlog item relies on other features or components.
- Date Needed / Timeline: The urgency and timeline constraints associated with each backlog item.

- Risk Involved: Risk assessment involves identifying potential risks and uncertainties associated with each backlog item that will have a negative impact on project timelines.

After careful discussion with our team, we have finalized the following factors to be the most crucial to the success of this project:

- Business Value: Business value is an important factor to this project as it directly correlates to the project's goals and contributes to overall success and stakeholders satisfaction.
- Development Effort: Development effort is another crucial factor to consider for proper resource and time allocation.
- Risk Involved: Understanding the risks involved with a task is vital to delivering a satisfactory product. By prioritizing items with higher risk, you can mitigate potential issues early in the development process.

The factors we discarded are:

- Feature Dependency: Feature dependency is discarded as we have decided that if we depended on it too much as a prioritization factor, we would be unable to pinpoint which task is the most important. An item of higher significance might be dependent on a less important preceding item, resulting in that more important task being locked out.
- Date Needed / Timeline: While timeline considerations are important for overall project planning and scheduling, the team has decided that timeline constraints can be addressed at a higher level during release planning and sprint planning.

## 2. Criteria for Backlog Items prioritization

We will attempt to develop our criteria by answering the following questions:

1. What if not all factors are of equal importance?
2. Why does a particular factor have a higher weight than the other?
3. What should be the proper weighing among these factors?

### **Question 1.**

We need to determine the importance of each factor as it serves as the foundation for effective task management. Without categorizing and weighing factors, it becomes challenging to establish priorities, keep track of tasks, and assign them to team members.

### **Question 2.**

Putting an emphasis on the weight of business value is imperative due to its direct correlation with the project's overall success. By evaluating tasks based on their business value, we align our efforts with strategic objectives, ensuring that resources are directed towards tasks that contribute most significantly to project goals.

Once the business value is determined, we delve into assessing the effort required to develop specific features or tasks. While considering development effort is essential, it takes a backseat to business value in our decision-making process. Tasks with higher business value may warrant greater effort to ensure their successful implementation.

Furthermore, we also factor in the risk associated with each task. Assigning weights to risk helps in prioritizing tasks that may pose potential challenges or setbacks to the project. By addressing high-risk tasks proactively, we mitigate potential negative impacts on project timelines and outcomes.

### **Question 3.**

With these considerations, we assign weights to each factor to guide our decision-making process effectively:

- Business Value: 50%
- Development Effort: 30%
- Risk Involved: 20%

### **3. Sprint Backlog Item to develop**

Based on the criteria above, our team has decided that the item *F2*, namely “*Website for HRM*” is the most important task and should be prioritized above all. It has the highest business value in Sprint 1 and is the core of the project, hosting the various functionalities later on, meaning many items are also dependent on it.

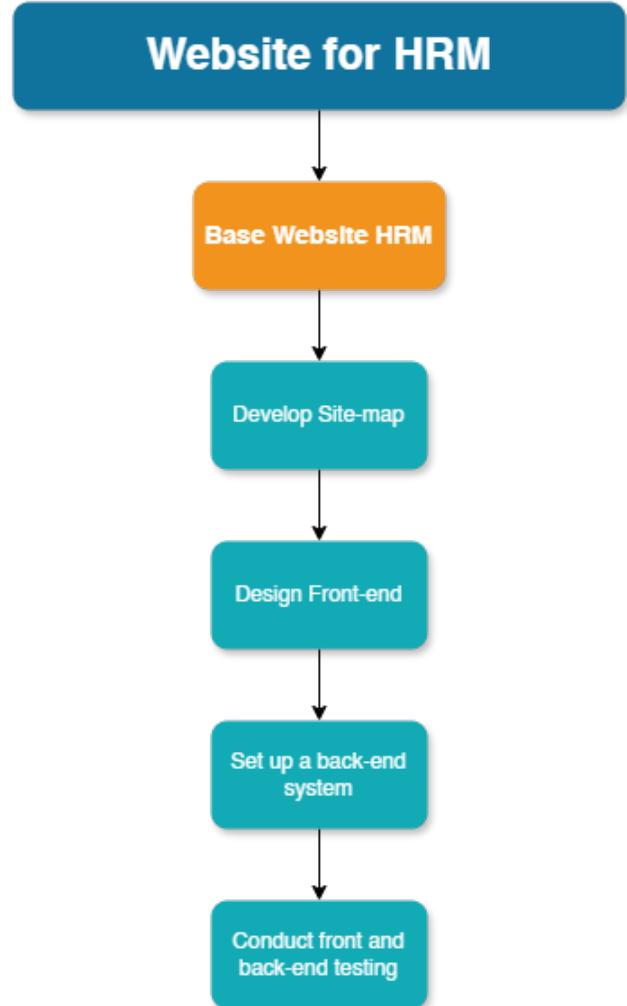
### **4. Work Breakdown Structure**

We first examine the Product Backlog Items list and determine the task needed to be used in the WBS chart, after careful discussion and consideration, there are four (4) items and 20 tasks that need to be done in Sprint 1.

Item	Task ID	Description	Dependency
Product UX/UI Design	T1	Research the overall design of current Gemadept's system.	None
	T2	Sketch initial wireframes on paper or using digital tools to visualize the basic layout and flow.	T1

	T3	Design high-fidelity mockups using design software like Sketch or Figma, incorporating branding elements and visual styles.	T1, T2
	T4	Come up with the UI code and structure.	T3
	T5	Conduct usability testing with a sample group of users to gather feedback on the design and iterate accordingly.	T4
Base website for HRM	T6	Develop a sitemap to organize the structure and navigation of the website.	T5
	T7	Design and implement a responsive front-end using HTML, CSS, and JavaScript frameworks.	T6
	T8	Fine tune the UI code to make it closely align to the UX/UI.	T4, T7
	T9	Set up a back-end system to handle user authentication, database interactions, and business logic using technologies like Node.js or Django.	None
	T10	Conduct front and back-end testing to ensure the application meets business expectations.	T9
HR database schema design	T11	Identify and prioritize the key entities in the HR domain, such as employees, departments, and positions.	None
	T12	Create an entity-relationship diagram (ERD) to visualize the relationships between different entities.	T11
	T13	Define the attributes and data types for each entity based on the information to be stored.	T12

	T14	Normalize the database schema to eliminate redundancy and minimize data anomalies.	T13
	T15	Review and optimize the schema design for performance and scalability, considering factors like indexing and query optimization.	T14, T10
Manager Portal	T16	Gather requirements from managers and stakeholders to understand their needs and expectations.	T10
	T17	Create user stories or use cases to document the desired functionality of the portal.	T16
	T18	Develop features such as employee performance dashboards, task management tools, and reporting capabilities.	T5, T10
	T19	Implement role-based access control to ensure that managers only have access to relevant data and functionality.	T18
	T20	Test the functionality to ensure the feature is up to compliance and security.	T19



*Figure 4: WBS Chart*

## **TASK 09P: Project Set Up**

### **1. TASK ALLOCATION**

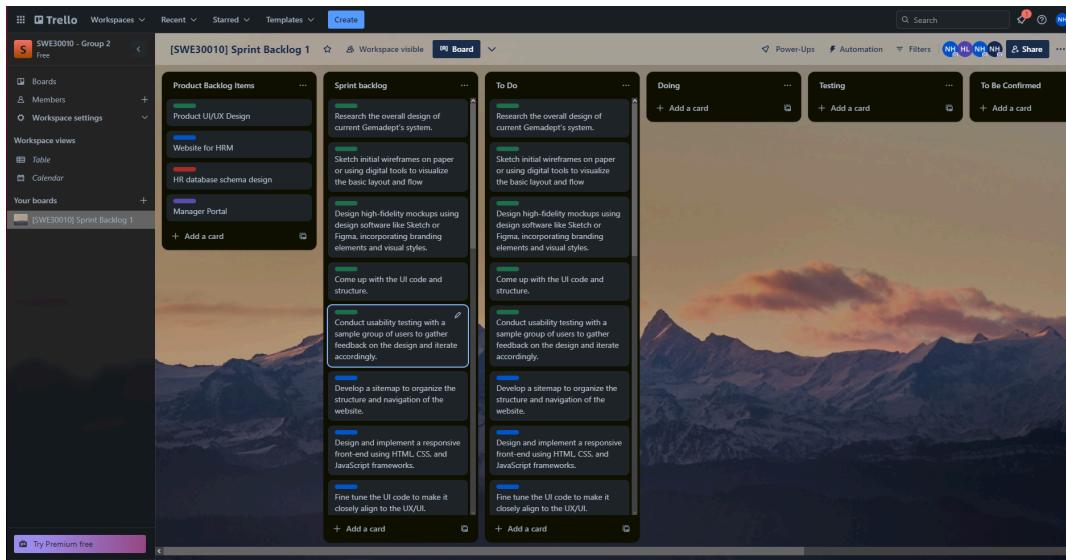
#### **Initializing**

Using Trello, we create a task board completed with the different stages of task completion as follow:

- Product backlog items: Lists all the products in the current sprint.
- Sprint backlog items: Lists all the tasks associated with the backlog items.
- To do: Items not yet started
- Doing: Item currently being developed.
- Testing: Checking of items that are moving from the “doing” to “to be confirmed” state.
- To be confirmed: Item being checked for the last time before being marked done.

- Done: Item successfully developed and test

The items in the sprint and backlog are recorded on Trello as follow:



*Figure 1.1: Screenshot of Trello task board*

## Rationale

As can be seen in the Trello screenshots, we have outlined the items based on the predefined categories above. Since this is day 0, only planning was done and no other work was done.

## 2. BURN-DOWN CHART

### Using burn-down chart

To ensure there is a measurable method of calculating the effort being made or used in the project, we rely on burn-down data, where each task is allocated a specific completion time. We've divided all the products in the product backlog into four sprints, each lasting approximately three weeks, given our maximum project duration of 12 weeks. This enables us to compile a table detailing the total time and duration of each item.

This table serves as the basis for generating the optimal burn-down chart for our project after day 0.

Sprint 1		Burned Down		Balance		Daily Completed
Day		Estimate	Actual	Estimate	Actual	
0		0	0	240	240	0
1		4		236	240	
2		8		228	240	
3		6		222	240	
4		8		214	240	
5		10		204	240	
6		6		198	240	
7		8		190	240	
8		4		186	240	
9		8		178	240	
10		21		157	240	
11		30		127	240	
12		8		119	240	
13		8		111	240	
14		4		107	240	
15		4		103	240	
16		10		93	240	
17		10		83	240	
18		7		76	240	
19		26		50	240	
20		30		20	240	
21		20		0	240	

Figure 2.1: Burn-down table

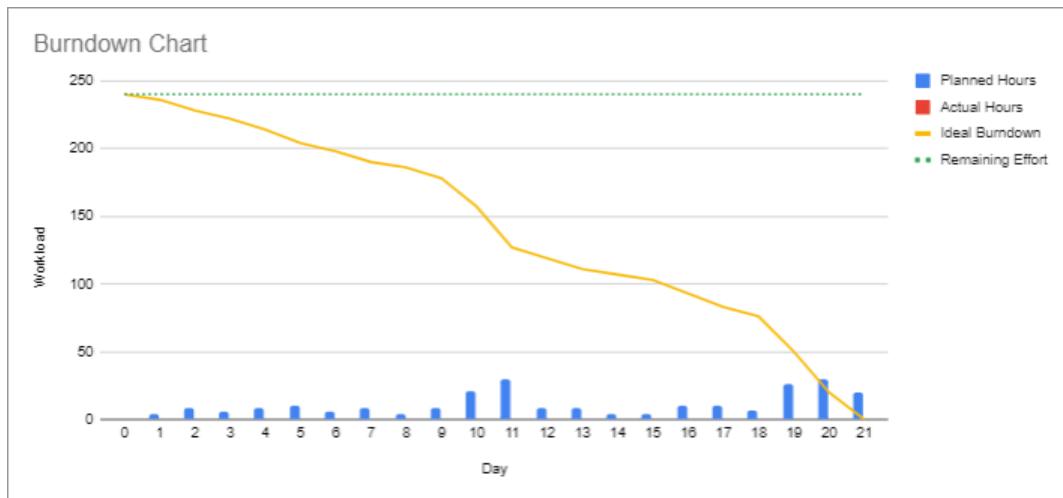


Figure 2.2: Burn-down chart

## Analysis

As can be seen, some of the tasks outlined as done in the task board translate rather well in this ideal burn-down chart, and since this is day 0, we can see only the estimated time of the whole project.

### 3. CODE REPOSITORY

The front and back end code folders of the project were created in the local machine and then pushed onto the Github repository. So far, there has been little to no movement surrounding these folders, but as the project goes on, there will surely be updates being made into these repositories.

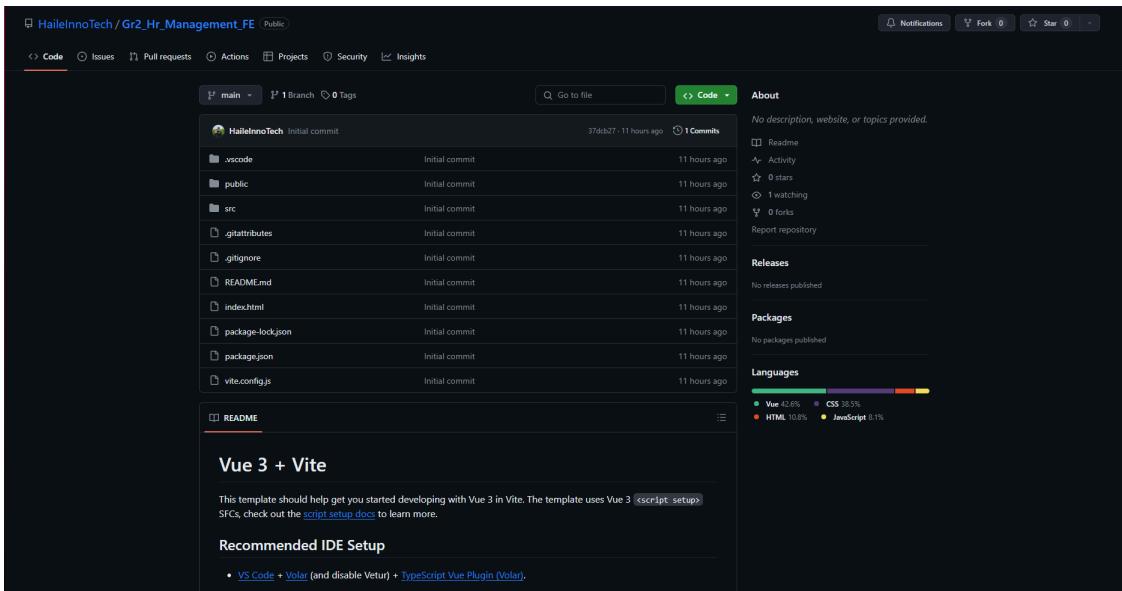


Figure 3.2: Front-end code repo on github

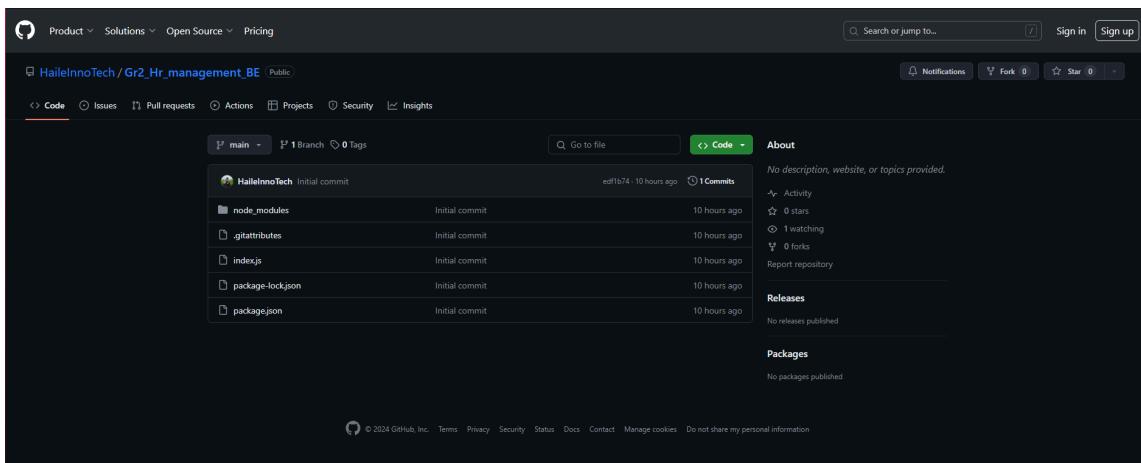


Figure 3.2: Back-end code repo on github

# **TASK 10P: Sprint 1 Mid-Point Progress Reporting**

## **1. Introduction**

This task is concerned with developing the HRM Application project for Gemadept. Each day will consist of a few tasks that have been established in previous work.

Trello is used to keep track of the progress, in our Trello beside the product and sprint backlog, we also include five additional of lists as follow:

- To do: This list contains all the tasks that need to be completed during the current sprint or iteration. These tasks are prioritized based on their importance and dependencies.
- Doing: Tasks that are currently being worked on by team members are moved to this list. It helps in visualizing the ongoing work and ensures that team members are not duplicating efforts.
- Testing: We would define this list as an extension to the Doing list, as should any problem occur during this phase, it should be put back into the doing phase to be further developed.
- To be confirmed: We will check the items the team completed for the day before sending it to the done list (testing for actual work, and to see if the task met requirement). A member cannot check their item as done to avoid bias, hence, checking must be done by another person that did not participate in doing the items.
- Done: Tasks that have been successfully completed and confirmed are moved to this list. It serves as a record of achievements and completed work items throughout the project lifecycle.

This structure helps us efficiently manage our workflow and ensure transparency and accountability within the team.

## **2. Sprint Progress:**

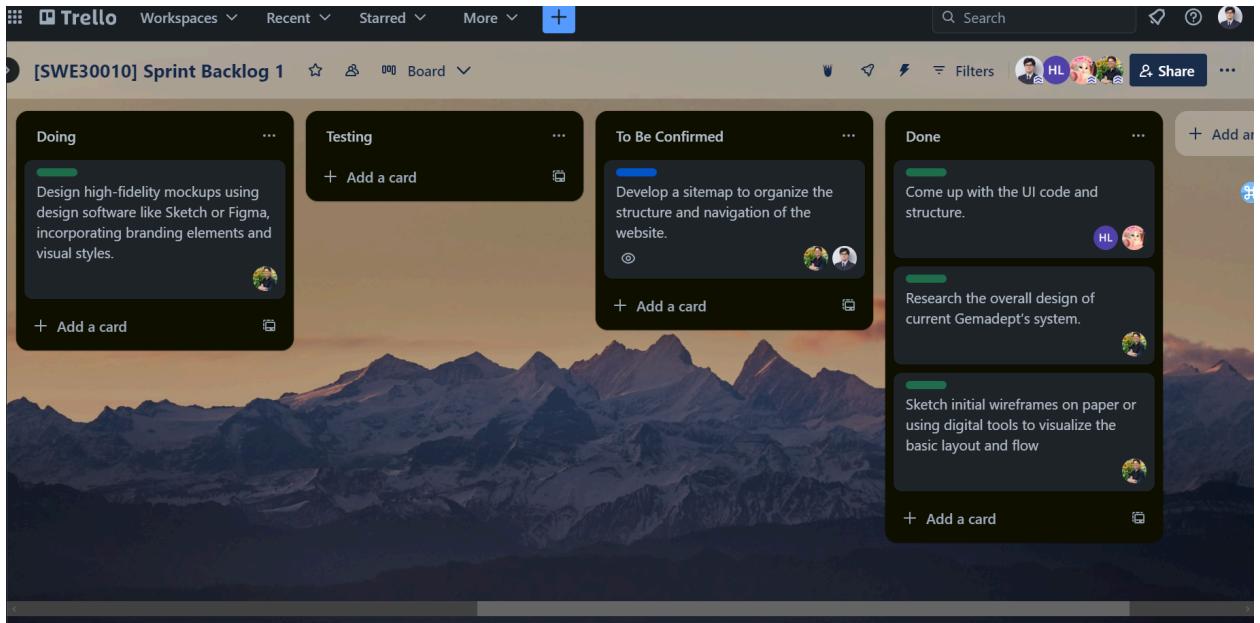
Below are all the items which we have mustered up in the process of doing this sprint.

### **a) Day One**

#### **Task sign-up at the start of the day:**

<b>Item</b>	<b>Description</b>	<b>Member</b>
1. Research Gemadept Design	The designer must conduct research on Gemadept Design to gather insights and inspiration for the project.	Nhat Huy
2. Sketch wireframe	Initial wireframes needed to be sketched out to visualize the layout and structure of the project.	Nhat Huy
3. Design Front-end mockup	Preliminary designs for the front-end interface must be created to outline the visual appearance and user interaction flow.	Nhat Huy
4. Develop sitemap	Designer and Leader develop a sitemap to organize the navigation and structure of the website or application.	Nhat Huy Nhat Minh
5. Develop Front-end code (1st edition, not final)	Initial coding efforts that focused on developing the front-end of the project, with a version aimed at providing basic layout.	Hoang Hai Minh Thy

## Trello task board at the end of the day:



## Code repository on Github:

```
00 0.0 >+1.22 00
1 * console.log("Hello World from index.js!");
2 * require("fs").config();
3 * const jwt = require("jsonwebtoken");
4 *
5 * const bcrypt = require("bcrypt");
6 *
7 * const ( JWT ) = require("nodejs-auth-library");
8 * const GoogleSpreadsheet = require("google-spreadsheet");
9 * const creds = require("./googleAuth.json");
10 * const SHEET_ID = "1gAdldeUdmtEwBspqJm4n8b2SCERFkpU_0jgct";
11 * const SCOPES = [
12 *   "https://www.googleapis.com/auth/spreadsheets",
13 *   "https://www.googleapis.com/auth/drive.file",
14 * ];
15 *
16 * const express = require("express");
17 * const app = express();
18 * const cors = require("cors");
19 * app.use(cors({ origin: true }));
20 * app.use(express.json());
21 *
22 * const port = 3000;
```

*Back end code development*

**Gr2\_Hr\_Management\_FE** (Public)

main · 1 Branch · 0 Tags · Go to file · Add file · Code

**HaileInnoTech** Update Mainpage based on sketch · 07b5ecc · yesterday · 4 Commits

.vscode	Initial commit	last week
public	Initial commit	last week
src	Update Mainpage based on sketch	yesterday
.gitattributes	Initial commit	last week
.gitignore	Initial commit	last week
README.md	Initial commit	last week
index.html	Initial commit	last week
package-lock.json	Add login page template	2 days ago
package.json	Add login page template	2 days ago
postcss.config.js	Add login page template	2 days ago
tailwind.config.js	Add login page template	2 days ago
vite.config.js	Initial commit	last week

**README**

## Vue 3 + Vite

This template should help get you started developing with Vue 3 in Vite. The template uses Vue 3 `<script setup>` SFCs, check out the [script setup docs](#) to learn more.

**About**  
No description, website, or topics provided.

**Readme**  
Activity  
0 stars  
1 watching  
0 forks

**Releases**  
No releases published  
Create a new release

**Packages**  
No packages published  
Publish your first package

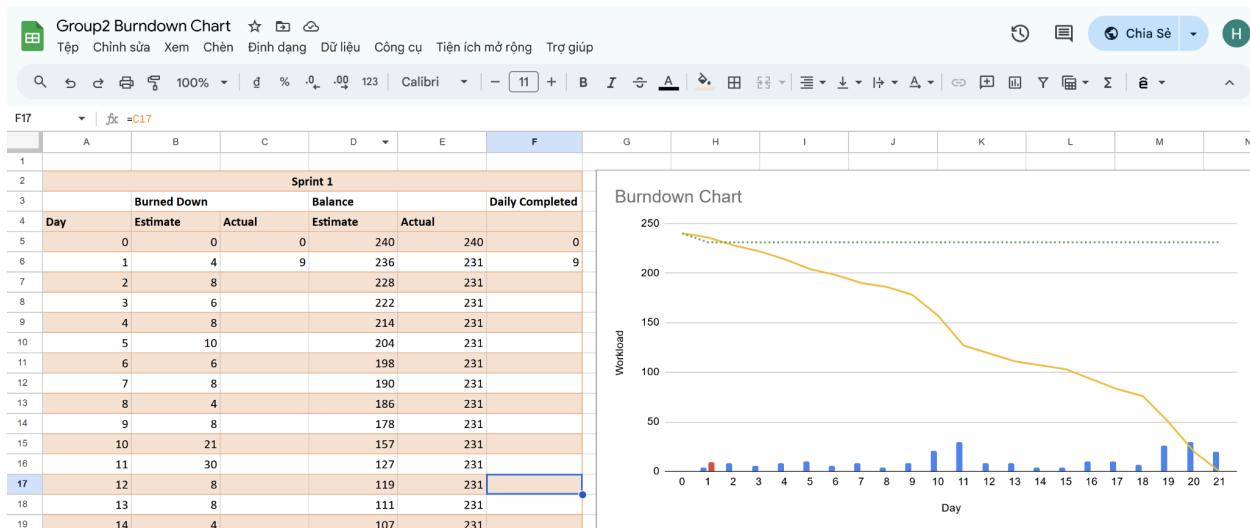
**Languages**  
Vue 89.4% · JavaScript 5.5% · HTML 3.7% · CSS 1.4%

**Suggested workflows**  
Based on your tech stack

**SLSA Generic generator** · Configure · Generate SLSA3 provenance for your existing release workflows

*Back end code development*

## Burndown chart:



## Daily meeting minute:

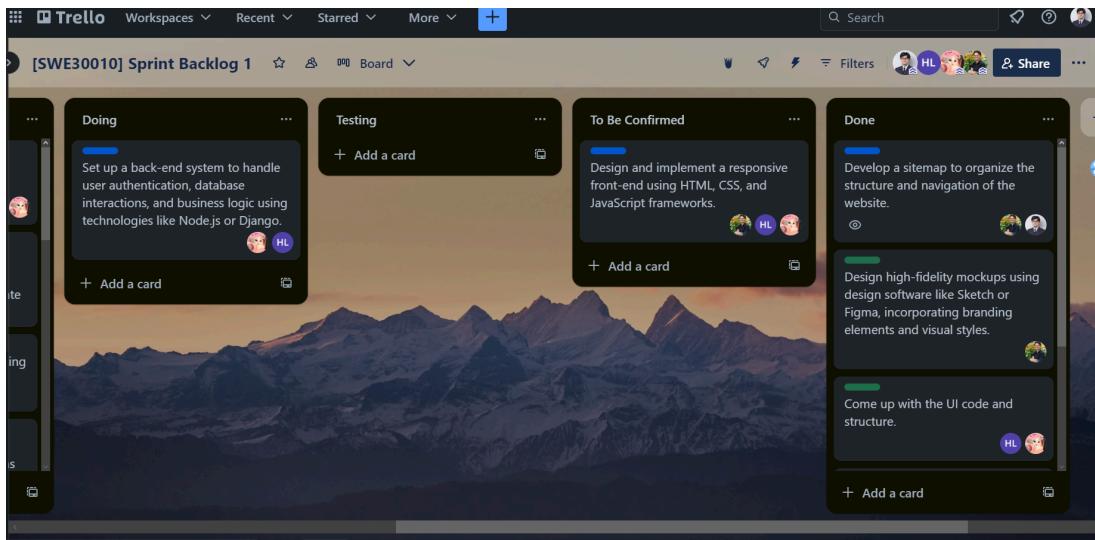
Status	Description	Count
Item Done	Come up with the UI code and structure., Research the overall design of current Gemadept's system., Sketch initial wireframes on paper or use digital tools to visualize the basic layout and flow.	3/5
Item Testing	N/A	0/5
Item Confirming	Develop a sitemap to organize the structure and navigation of the website	1/5
Item Not Done	Design high-fidelity mockups using design software like Sketch or Figma, incorporating branding elements and visual styles.	1/5
Next step	Continue working on the mockup, then move onto doing day 2's tasks	

## b) Day Two

### Task sign-up at the start of the day:

Item	Description	Member
1. Finish website Mockup	Complete the creation of the website mockup, which includes designing the visual layout, user interface elements, and overall aesthetic presentation of the website.	Nhat Huy
2. Setting up back-end functionality	Implement the necessary functionality on the server-side of the website, including database integration, server logic, and API development to support the website's features and interactions.	Hoang Hai Minh Thy
3. Implement responsive and user friendly features to the initial UI code	Add functions of CSS, JS and HTML to the code function to give the website somewhat of a crude usability.	Nhat Huy Hoang Hai Minh Thy
4. Checking of completed items	We will check the items the team completed for the day before sending it to the done list. A member cannot check their item as done to avoid bias, hence, checking must be done by another person that did not participate in doing the items.	Nhat Minh

## Trello task board at the end of the day:



## Code repository on Github:

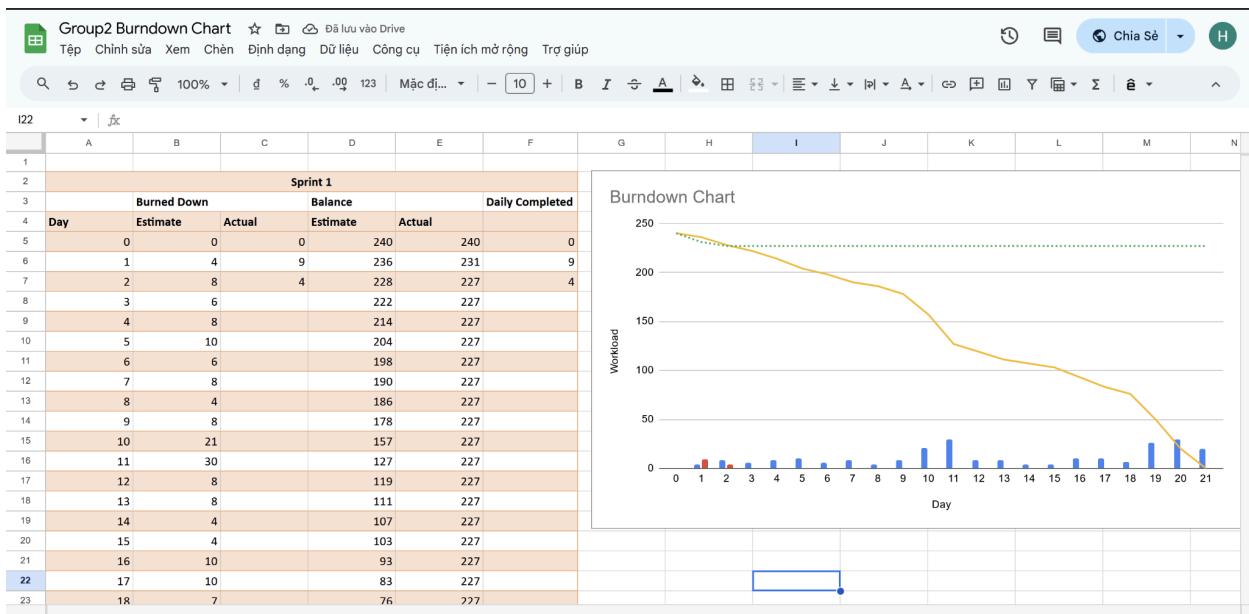
```
Commit
Update api get Employee Data
API for getting employee data from google sheet
Browse file
P main
HalleInnoTech committed yesterday
1 parent 3fbbcc32 commit 8695951b
Showing 1 changed file with 56 additions and 1 deletion
index.js
diff --git a/index.js b/index.js
@@ -19,50 +19,59 @@ const cors = require("cors");
 19   19   app.use(cors({ origin: true }));
 20   20   app.use(express.json());
 21   21
 22 - const port = 3000;
 22 + const port = 4000;
 23 +
 24 + app.listen(port, () => {
 25 +   console.log(`Server is running on port ${port}`);
 26 + });
 27 +
 28 + app.get("/employeedata", async (req, res) => {
 29 +   const data = await getEmployeeData();
 30 +
 31 +   const sheet = doc.sheetsbyIndex[2];
 32 +   const rows = await sheet.getRows();
 33 +   for (let i = 0; i < rows.length; i++) {
 34 +     const id = rows[i].get("id");
 35 +     const firstName = rows[i].get("first_name");
 36 +     const lastName = rows[i].get("last_name");
 37 +     const email = rows[i].get("email");
 38 +     const address = rows[i].get("address");
 39 +     const phone = rows[i].get("phone");
 40 +     const address = rows[i].get("address");
 41 +     const department = rows[i].get("department");
 42 +     const position = rows[i].get("position");
 43 +     data.push({
 44 +       id: id,
 45 +       firstName: firstName,
 46 +       lastName: lastName,
 47 +       email: email,
 48 +       gender: gender,
 49 +       phone: phone,
 50 +       address: address,
 51 +       department: department,
 52 +       position: position,
 53 +     });
 54   }
 55 }

 56
 57
```

Back-end commits on github

*Front-end commits on github*

## Burndown chart:



## Daily meeting minute:

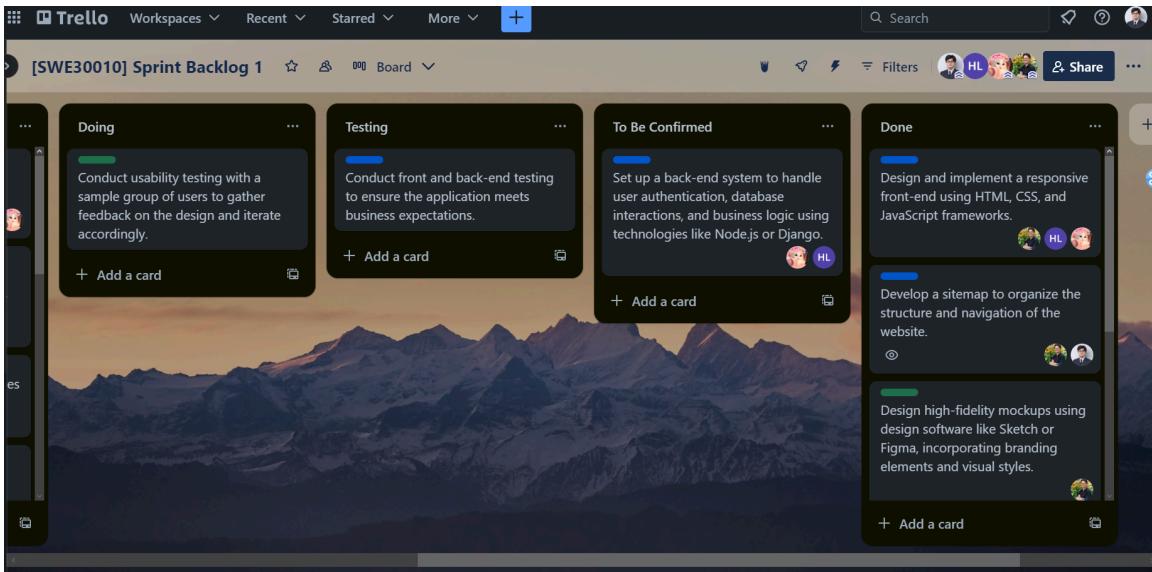
Status	Description	Count
Item Done	Front-end mock up	1/4
Item Testing	N/A	0/4
Item Confirming	Improved front-end design and usability	1/4
Item Not Done	N/A	0/4
Next step	Continue working on the back-end system, check fine tune appropriateness, and move onto next day	

## c) Day Three

### Task sign-up at the start of the day:

Item	Description	Member
1. Continue back-end functionality	Proceed with the development of back-end functionalities, including database integration, server logic implementation, and API development. Ensure that the back-end system is robust, efficient, and capable of supporting the required functionality of the application.	Hoang Hai Minh Thy
2. Testing the UI code	Perform thorough testing of the front-end codebase to identify and address any bugs, errors, or inconsistencies in the user interface. Test for compatibility across different browsers, devices, and screen sizes to ensure a seamless user experience.	Hoang Hai
3. Conduct testing of front end	Conduct comprehensive testing of the front-end components and features to validate their functionality, usability, and performance. Utilize testing techniques such as unit testing, integration testing, and user acceptance testing to identify and resolve any issues or discrepancies.	Nhat Minh Nhat Huy
4. Checking of completed items	We will check the items the team completed for the day before sending it to the done list. A member cannot check their item as done to avoid bias, hence, checking must be done by another person that did not participate in doing the items.	Nhat Minh Hoang Hai

## Trello task board at the end of the day:



## Code repository on Github:

Commit

```
Update api get total attendance
API for getting Employee Attendance by ID from Google Sheet
[{"key": "1", "value": 2}, {"key": "2", "value": 3}, {"key": "3", "value": 2}, {"key": "4", "value": 2}]

main
HaileInnoTech committed yesterday
Showing 3 changed files with 101 additions and 14 deletions.
```

index.js

```
1 // Hello World from index.js!
2 const config = require('dotenv').config();
3 const jwt = require("jsonwebtoken");
4 var bodyParser = require("body-parser");
5
6 const bcrypt = require("bcrypt");
7
8 const port = 4000;
9 app.listen(port, () => {
10   console.log(`Server is running on port ${port}`);
11 });
12
13 + const authenticateWithGoogle = async () => {
14 +   // Authenticate with Google
15 +   const jwt = new JWT({
16 +     email: creds.client_email,
17 +     key: creds.private_key,
18 +     scopes: SCOPES,
19 +   });
20 }
```

*Back-end development*

**Commit**

**Add Horizontal navigation header**

HaileInnoTech committed yesterday

Showing 4 changed files with 124 additions and 2 deletions.

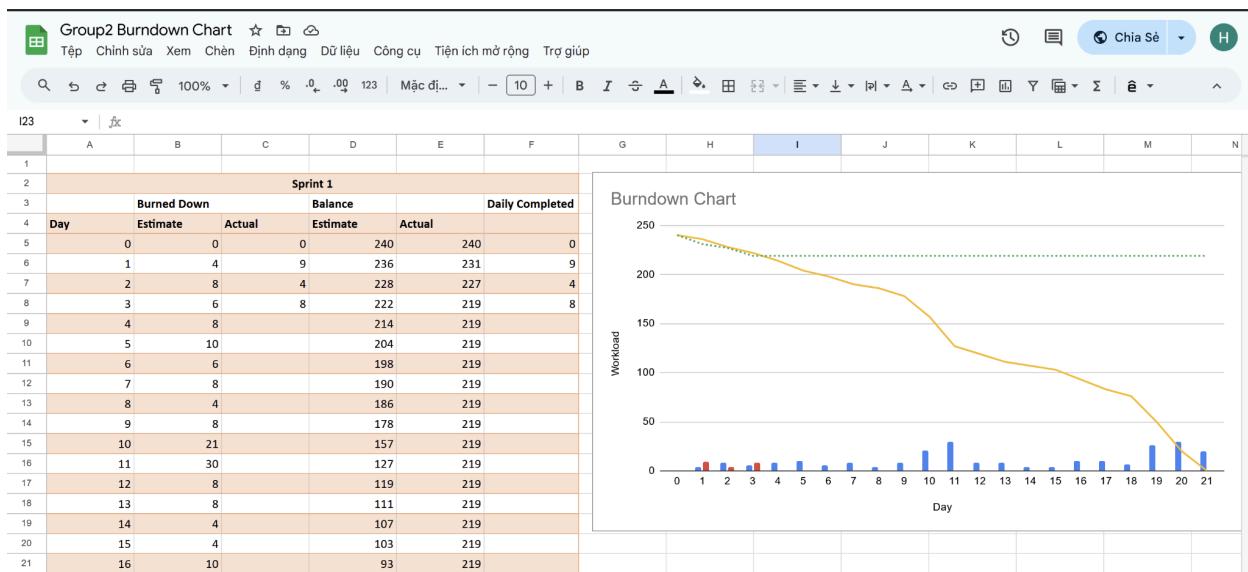
Filter changed files

- src
- App.vue
- components
- Navigate
- Navigation.vue
- image.png
- Pages
- Mainpage.vue

```
✓ 10  src/App.vue
...
1   1    @@ -1,13 +1,19 @@
2   2    <template>
3   3    - <Login />
4   4    + <navigation/>
5   5    + </div>
6   6    + <mainpage/>
7   7    </div>
8   8    </template>
9   9
10  10   <script>
11  11     import Login from "./components/Pages/Login.vue";
12  12     + import Navigation from "./components/Navigate/Navigation.vue";
13  13     + import Mainpage from "./components/Pages/Mainpage.vue";
14  14     +
15  15     export default {
16  16     - components: { Login },
17  17     + components: { Login, Navigation, Mainpage },
18  18   };
19  19   </script>
```

*Front-end development*

## Burndown chart



## Daily meeting minute:

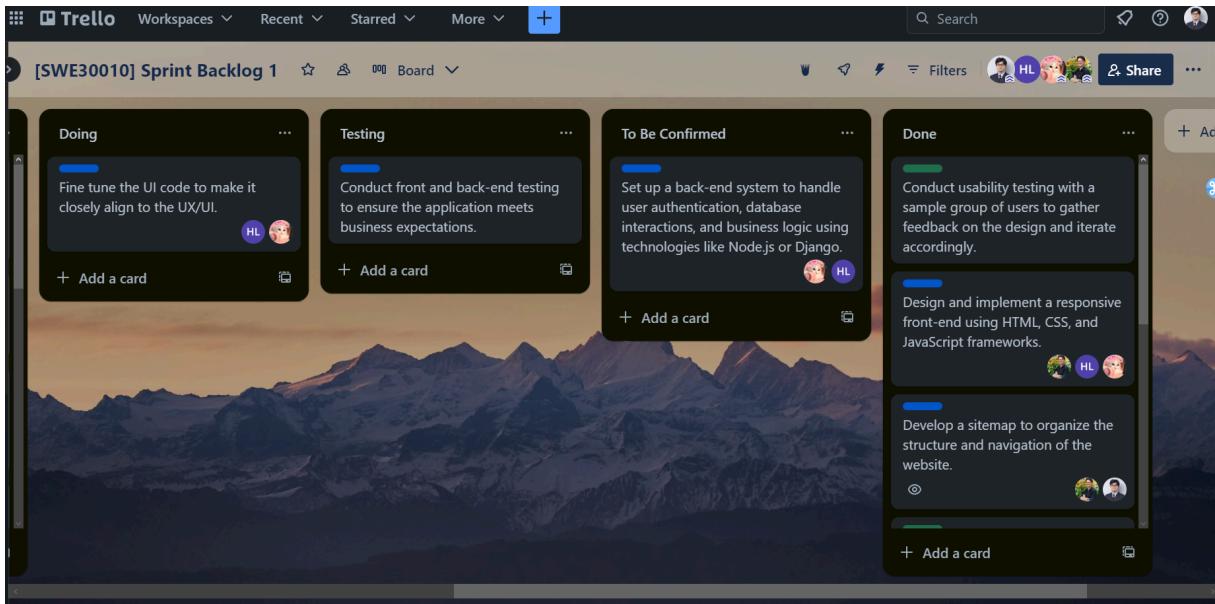
Status	Description	Count
Item Done	Back-end functionality	1/4
Item Testing	Back-end testing (internal testing to ensure correctness)	1/4
Item Confirming	Back-end function	1/4
Item Not Done	Front-end testing (due to time constraint in meeting up with sample users to conduct interview and testing)	1/4
Next step	Reserve more time to meet up with users for testing, and continue with next items.	

## d) Day Four

### Task sign-up at the start of the day:

Item	Description	Member
1. Review of front-end testing	Front-end review to make sure the task is actually completed and all components are met.	Nhat Minh
2. Reviewing of back-end function	Due to the size of the back-end, it can take a lot of time to review each function. Moreover, this task is also tied to the completeness of the testing phase.	Nhat Minh
3. Testing of back-end function	Back-end testing required a lot of time to make sure each function ran correctly. Should any function fail or need improvement, it is also transferred to the doing stage.	Hoang Hai Nhat Minh
4. Fine tune UI code	After meddling with the back-end, to ensure the visual is met, this website must encompass adjustments to CSS styles, HTML structure, and JavaScript functionality to optimize performance, responsiveness, and accessibility across different devices and screens.	Minh Thy

## Trello task board at the end of the day:



## Code repository on Github:

```
Update api get attendance by email
API for getting Employee Attendance by Email include month filter from Google sheet
[{"email": "gmedalle@printfriendly.com", "date": "9/15/2023", "checkin": "9:00 AM", "checkout": "3:15 PM"}, {"email": "gmedalle@printfriendly.com", "date": "9/15/2023", "checkin": "9:00 AM", "checkout": "9:00 AM"}, {"email": "gmedalle@printfriendly.com", "date": "7/16/2023", "checkin": "10:58 AM", "checkout": "5:59 PM"}, {"email": "gmedalle@printfriendly.com", "date": "7/18/2024", "checkin": "11:07 AM", "checkout": "1:01 PM"}, {"email": "gmedalle@printfriendly.com", "date": "7/12/2023", "checkin": "9:00 AM", "checkout": "2:00 AM"}, {"total": 8}
```

Commit

main

HalleInnoTech committed yesterday

Showing 1 changed file with 55 additions and 55 deletions.

1 parent dcb0ae commit b37808

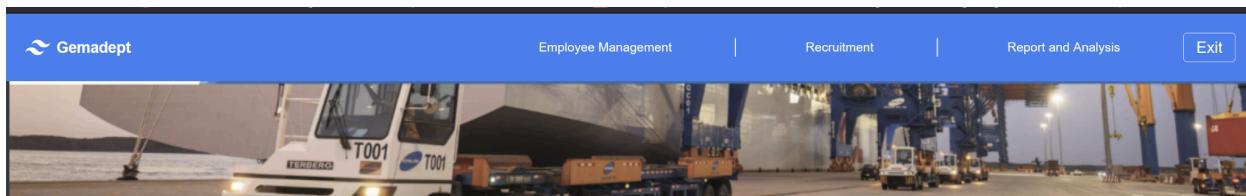
Whitespace Ignore whitespace Split Unified

```

123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352

```

## *Back-end development*

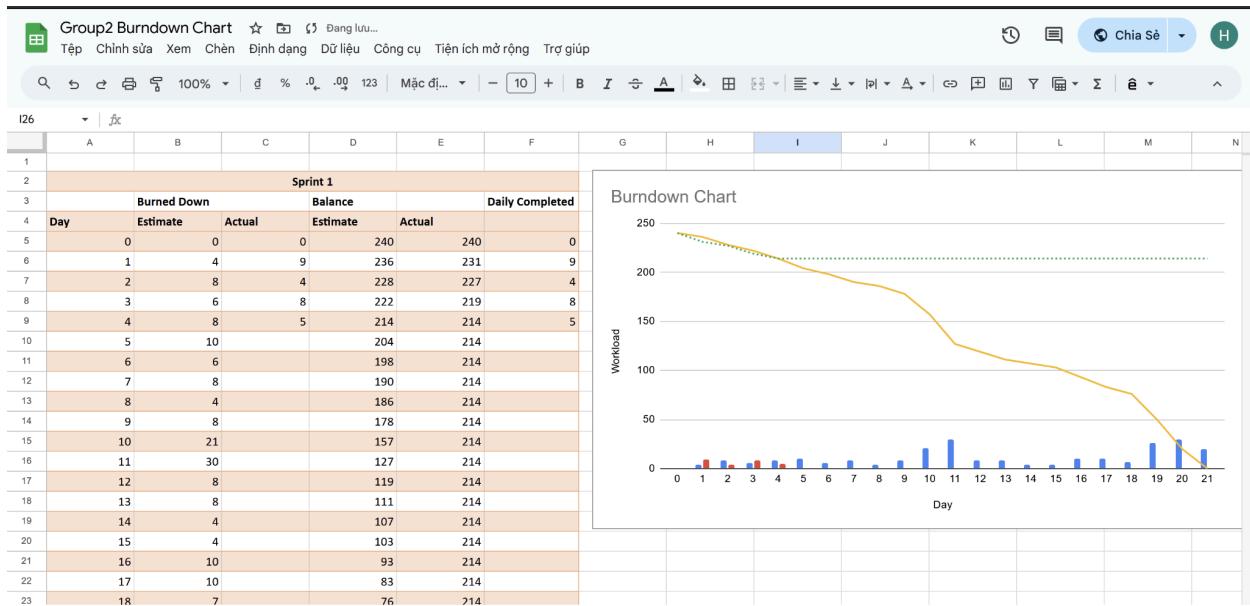


## **HR Management**

Employee Checked In	Time	Employee Checked Out
349	4:31:11 PM	349

## *Front-end code refine*

## Burndown chart



## Daily meeting minute:

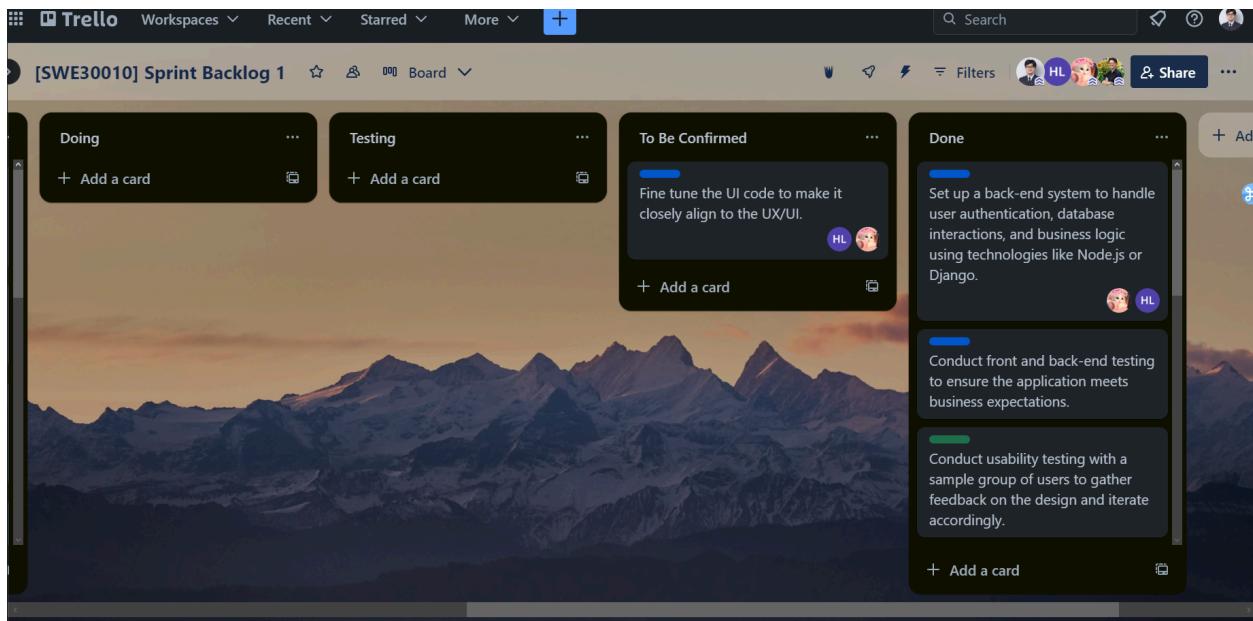
Status	Description	Count
Item Done	N/A	0/4
Item Testing	Back-end testing	1/4
Item Not Done	Fine tune UI code (only partially carried out due to size of back-end)	1/4
Next step	Finish all review and testing	

## e) Day Two

**Task sign-up at the start of the day:**

Item	Description	Member
1. Finish review of previous task	All task not previously done must be done	Nhat Minh
2. Finish UI code	Fine tune UI code to closely align with the mock up	Hoang Hai Minh Thy

**Trello task board at the end of the day:**



## Code repository on Github:

The screenshot shows a GitHub commit page for a file named index.js. A red box highlights a portion of the JSON data being added to the file. The commit message is "Update api get employee data by email". The code changes section shows the addition of a new route to handle employee data by email.

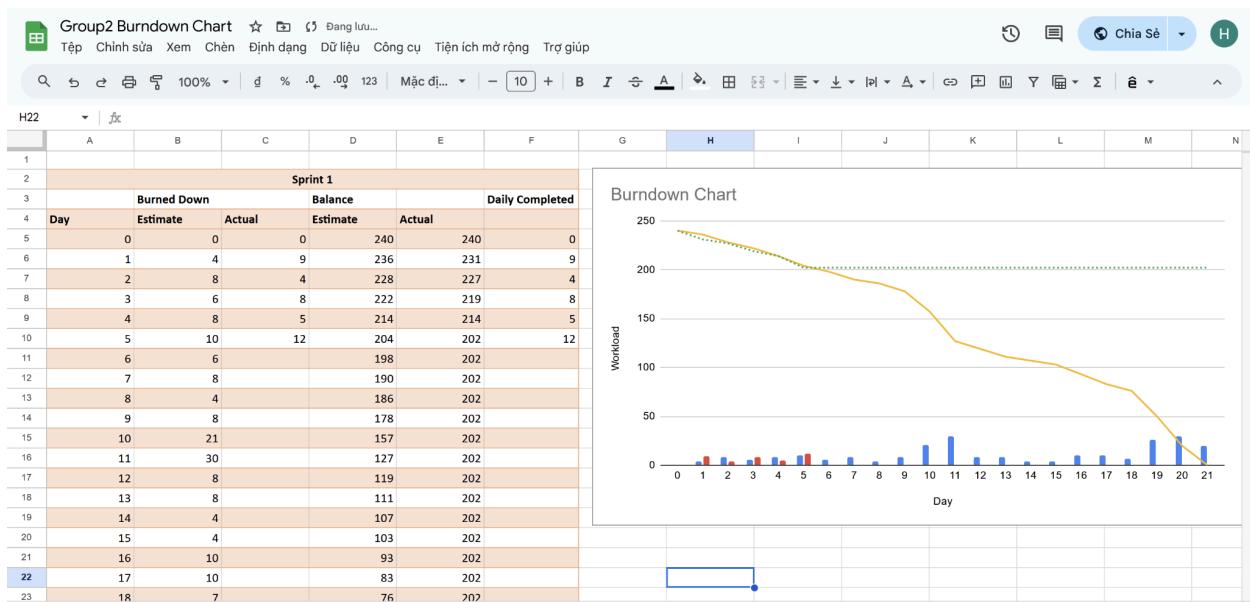
```
Commit
Update api get employee data by email
[{"id": "64",
  "firstname": "Or",
  "lastname": "Danielilot",
  "email": "rdanielilot@ib.com",
  "gender": "Male",
  "phone": "888 888 1234",
  "address": "534 Dunning Parkway",
  "department": "Legal",
  "position": "Electrical Engineer"
}]
12 main
HaleInnoTech committed yesterday
Showing 1 changed file with 41 additions and 1 deletion
1 parent b378d8b commit 9a9f1ed
index.js
@@ -151,7 +151,6 @@ app.get("/countattendancebymail", async (req, res) => {
151   151   )
152   152   }
153   153   }
154   154   data.push({ total: hashMap(req.query.mail) });
155   155   try {
156     156     res.status(400).send("Cannot get data");
157     157   }
158     158   });
159   159   +
160   160   @@ -160,3 +159,4 @@ app.get("/countattendancebymail", async (req, res) => {
161     159     res.status(400).send("Cannot get data");
162     160   }
163     161   );
164   +
165   + app.get('/employeedataemail', async (req, res) => {
166   +   doc = await authenticateWithGoogleId();
167   +   const data = [];
168   +   const emaillookup = req.query.email;
169   +   sheet = doc.sheetsbyIndex[2];
170   +   const rows = await sheet.getRows();
```

```

162
163 + app.get("/employeedatabyemail", async (req, res) => {
164 +   doc = await authenticateWithGoogle();
165 +   const data = [];
166 +   const emaillookup = req.query.email;
167 +   sheet = doc.sheetsByIndex[2];
168 +   const rows = await sheet.getRows();
169 +   for (let i = 0; i < rows.length; i++) {
170 +     const id = rows[i].get("id");
171 +     const firstName = rows[i].get("first_name");
172 +     const lastName = rows[i].get("last_name");
173 +     const email = rows[i].get("email");
174 +     const gender = rows[i].get("gender");
175 +     const phone = rows[i].get("phone");
176 +     const address = rows[i].get("address");
177 +     const derpartment = rows[i].get("department");
178 +     const position = rows[i].get("position");
179 +     if (emaillookup === email && emaillookup !== "") {
180 +       data.push({
181 +         id: id,
182 +         firstName: firstName,
183 +         lastName: lastName,
184 +         email: email,
185 +         gender: gender,
186 +         phone: phone,
187 +         address: address,
188 +         derpartment: derpartment,
189 +         position: position,
190 +       });
191 +     }
192 +   }
193 +   if (!data.length) {
194 +     data.push({ message: "No data found" });
195 +   }
196 +
197 +   try {
198 +     res.status(200).json(data);
199 +   } catch (err) {
200 +     res.status(400).send("Cannot get data");
201 +   }
202 + });

```

## Burndown chart

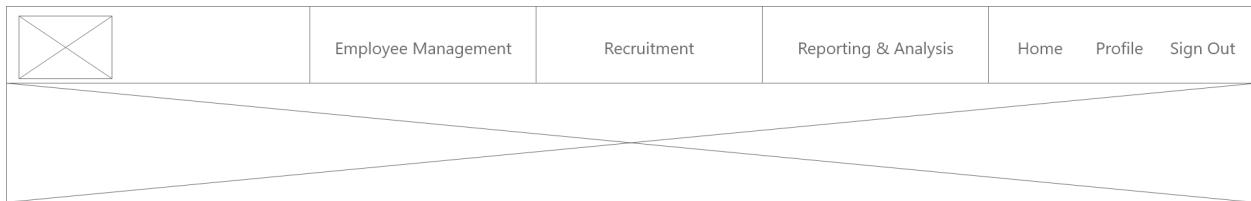


## Daily meeting minute:

Status	Description	Count
Item Done	Review of function	1/2
Item Testing	N/A	0/2
Item Confirming	Flne tune UI code	1/2
Item Not Done	N/A	0/2
Next step	Moving to Database Design and Manager Portal	

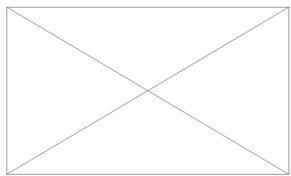
## Appendix of Mockup

### Wireframe:



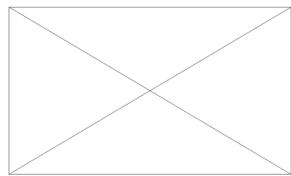
### HRM System

**Employees Checked In**  
320



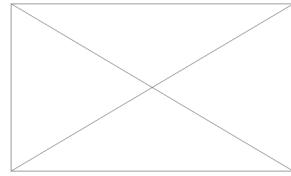
Employee Management

**Time**  
13:32



Recruitment

**Employees Checked Out**  
34



Reporting & Analysis

## Main page mockup:

The main page mockup for the HRM System features a blue header bar with the GEMADEPT logo and tagline "THE WAY FORWARD". The header includes navigation links for "Employee Management", "Recruitment", "Reporting & Analysis", and icons for home, user profile, and logout. Below the header is a large image of a ship being transported on a trailer truck at a port. The main title "HRM System" is centered in a large blue font. Below the title are three sections: "Employees Checked In" (320), "Time" (13:32), and "Employees Checked Out" (34). Each section has a thumbnail image and a circular icon: "Employee Manager" (person icon), "Recruitment" (handshake icon), and "Reporting & Analysis" (document icon). The footer follows the same blue and yellow color scheme as the header.

**Employee Management**

**Recruitment**

**Reporting & Analysis**

**HRM System**

**Employees Checked In**  
320

**Time**  
13:32

**Employees Checked Out**  
34

**Employee Manager**

**Recruitment**

**Reporting & Analysis**

## **Employee Manager mockup:**





# Employee Manager

Employee Information | Payroll Manager | Performance Tracker

X

X

X

Displaying 20 of 412 results

Page 1 of 21



## Recruitment & Applicants mockup:



 Recruitment & Applicants Manager

Applicant's Name...	
Position...	

**Send Onboarding Document to Selected**

**Leave Manager**

Displaying 20 of 112 results

Page 1 of 18

< 1 2 3 4 ... 18 >



# **TASK 11P: Sprint 1 End-of-Sprint Progress Reporting**

## **SPRINT TASK**

### **1. Introduction**

This task is concerned with developing the HRM Application project for Gemadept. This task specifically is going to be a continuation from the previous task of keeping track of day 1 to 5's progress as it is about the progress of day 6 to 10.

For the main content, we will attempt to deliver the manager's portal and then database design and management. These items are not previously done in the last task due to time constraints, however they will be developed here.

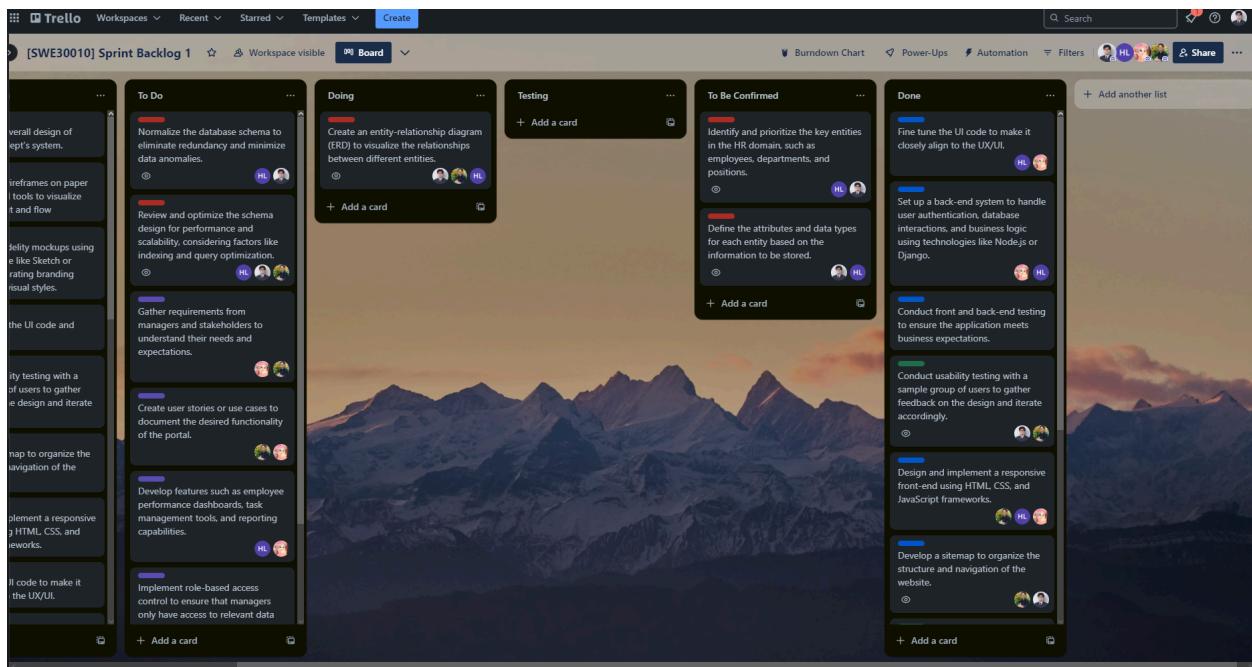
### **2. Task progress**

#### **a. Day 6**

#### **Tasks Sign Up**

<b>Item</b>	<b>Description</b>	<b>Member</b>
Identify and prioritise the key entities in the HR domain.	These entities include employee's name, departments, positions, and additional entities like salaries, and performance reviews. By prioritising these entities, we can ensure that we stay focused on the most critical aspects of HR management that directly impact organisational operations and goals.	Hoang Hai Nhat Minh
Define the attributes and data types for each entity based on the information to be stored.	We will carefully analyse the information to be stored for each entity and determine the appropriate data types, such as string, integer, date, decimal, etc., to accurately capture and represent the data.	Hoang Hai Nhat Minh
Create an ERD to visualise the relationships between different entities.	By creating an ERD, we should have a better understanding of the structure and interactions between HR entities, which will help form effective database design and system development.	Hoang Hai Nhat Minh Nhat Huy

## Trello task board update



## Github repository

Commit

Update API for employee Salary by Email

```
[  
  {  
    "id": "11",  
    "email": "jwyliea@google.ru",  
    "base_salary": "$24,00",  
    "total_work_hour": "17:46:00",  
    "bonus": 0,  
    "actual_payment": "$426,40"  
  }  
]
```

main

HaileInnoTech committed 2 days ago

Showing 1 changed file with 47 additions and 1 deletion.

48 index.js

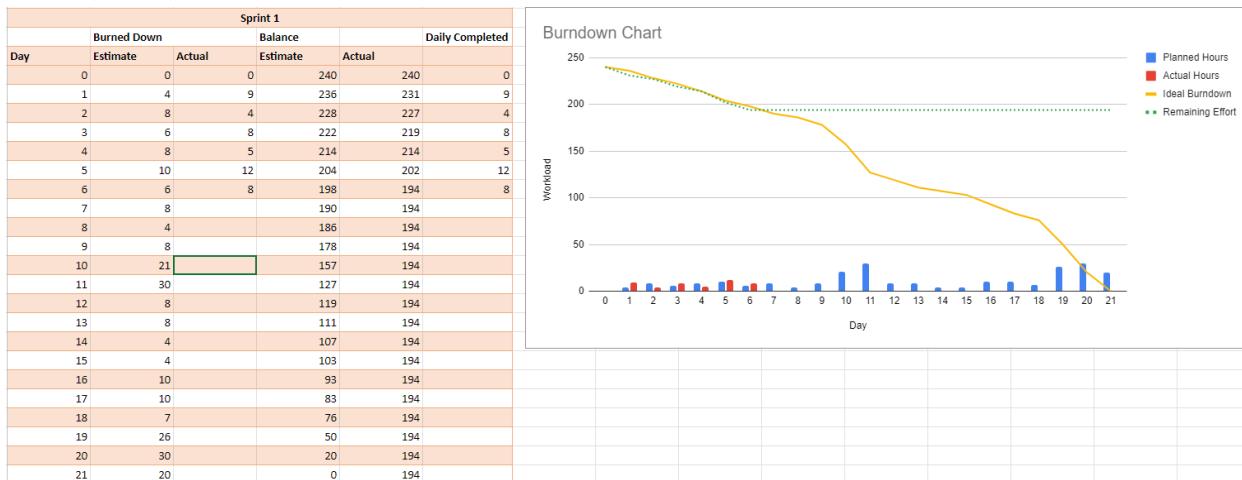
```
@@ -123,6 +123,7 @@ app.get("/counattendancebyemail", async (req, res) => {  
  123 123      date: date,  
  124 124      checkin: checkin,  
  125 125      checkout: checkout,  
  126 +      hourwork: hourwork,  
  127      });  
  128      if (hashMap[email]) {  
  129          hashMap[email]++;  
  ....  
  151 152      }  
  152 153 +    }  
  153 154    }  
  154 -    data.push({ total: hashMap[req.query.email] });  
  155 +    if (!data.length) {  
  156 +      data.push({ message: "No data found" });  
  157 +    }  
}
```

```

210 +
211 + app.get("/employeesalarybyemail", async (req, res) => {
212 +   doc = await authenticateWithGoogle();
213 +   const data = [];
214 +   const emaillookup = req.query.email;
215 +   sheet = doc.sheetsByIndex[4];
216 +   const rows = await sheet.getRows();
217 +   for (let i = 0; i < rows.length; i++) {
218 +     const email = rows[i].get("email");
219 +     if (emaillookup === email){
220 +       const id = rows[i].get("id");
221 +       const base_salary = rows[i].get("base_salary");
222 +       const total_work_hour = rows[i].get("total_work_hour");
223 +       let bonus = rows[i].get("bonus");
224 +       if(bonus === ""){
225 +         bonus = 0;
226 +       }
227 +       const actual_payment = rows[i].get("actual_pay");
228 +       data.push({
229 +         id: id,
230 +         email: email,
231 +         base_salary: base_salary,
232 +         total_work_hour: total_work_hour,
233 +         bonus: bonus,
234 +         actual_payment: actual_payment,
235 +       });
236 +     }
237 +   }
238 +   if (!data.length) {
239 +     data.push({ message: "No data found" });
240 +   }
241 +   try {
242 +     res.status(200).json(data);
243 +   } catch (err) {
244 +     res.status(400).send("Cannot get data");
245 +   }
246 +
247 +
248 + });

```

## Burndown chart



## Daily meeting minute

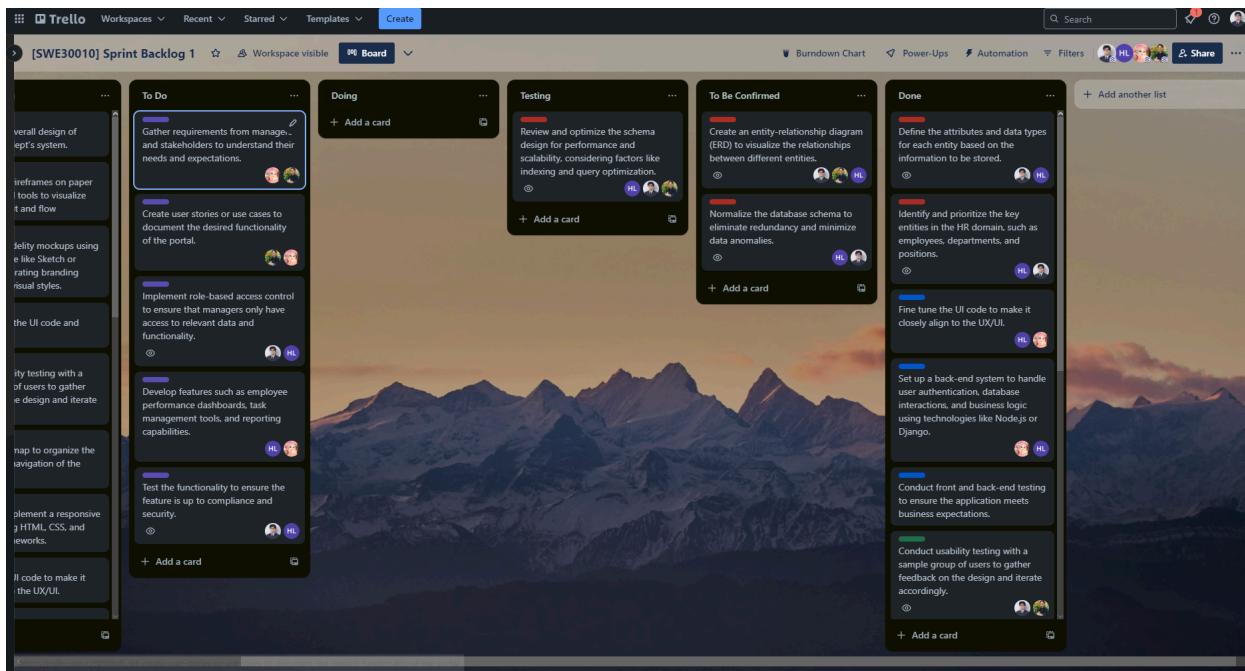
Status	Description	Count
Item Done	N/A	0/3
Item Testing	N/A	0/3
Item Confirming	Identify and define the attributes and data types for each HRM entity based on the information to be stored.	2/3
Item Not Done	ERD Creation, this task takes a long time as ERD designers need to take a lot of consideration into account to create a good model.	1/3
Next step	Review the initial version of the ERD, import some employee data and perform normalisation on it	

## b. Day 7

### Tasks Sign Up

Item	Description	Member
Review of ERD.	Make sure the first database design is adequate, this step can take some time as the reviewer must ensure correct association between entities.	Nhat Minh Nhat Huy Hoang Hai
Normalize the database schema.	We identify entities, define attributes, establish primary keys, and apply normalization rules to ensure data integrity and optimize performance to organise a database schema to eliminate redundancy and minimize data anomalies.	Hoang Hai Nhat Minh
Review and optimize the schema design.	Our team optimises the query statements and or the apis code to make sure the website can function as optimised as possible, i.e., faster and more reliable.	Nhat Minh Nhat Huy Hoang Hai

## Trello task board update



## Github repository

Commit

Update api to set bonus by employee email  
main  
HaileInnoTech committed 2 days ago

Showing 1 changed file with 28 additions and 7 deletions.

```
v ⏺ 35 index.js
  @@ -154,10 +154,8 @@ app.get("/counattendancebyemail", async (req, res) => {
154   154     }
155   155     if (!data.length) {
156   156       data.push({ message: "No data found" });
157 -  }
158 -  else{
157 + } else {
159   158   data.push({ total: hashMap[req.query.email] });
160 - -
161   159   }
162   160
163   161   try {
  @@ -216,12 +214,12 @@ app.get("/employeesalarybyemail", async (req, res) => {
216   214   const rows = await sheet.getRows();
217   215   for (let i = 0; i < rows.length; i++) {
218   216     const email = rows[i].get("email");
219 -   if (emaillookup === email){
217 +   if (emaillookup === email){
220   218     const id = rows[i].get("id");
221 -   const base_salary = rows[i].get("base_salary");
219 +   const base_salary = rows[i].get("base_salary");
222   220     const total_work_hour = rows[i].get("total_work_hour");
223   221     let bonus = rows[i].get("bonus");
224 -   if(bonus === ""){
222 +   if(bonus === ""){

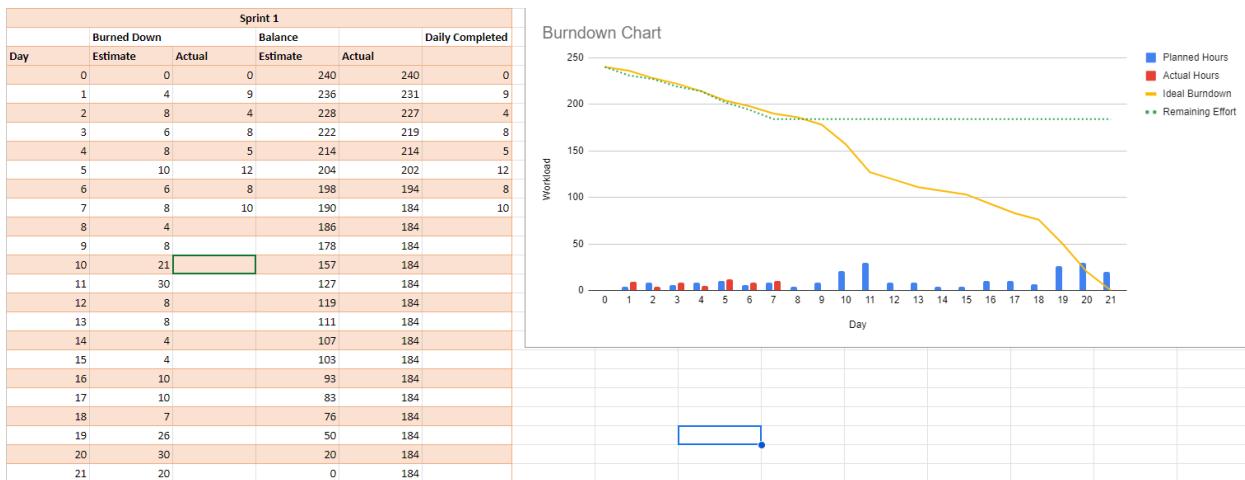
```

```

247
246 + app.post("/updateemployeebonus", async (req, res) => {
247 +   doc = await authenticateWithGoogle();
248 +   const data = [];
249 +   const emaillookup = req.query.email;
250 +   const bonus_update = req.query.bonus;
251 +   sheet = doc.sheetsByIndex[4];
252 +   const rows = await sheet.getRows();
253 +   for (let i = 0; i < rows.length; i++) {
254 +     const email = rows[i].get("email");
255 +     if (emaillookup === email && emaillookup !== "" && bonus_update !== "") {
256 +       await sheet.loadCells("A1:F310");
257 +       const cell = await sheet.getCellByA1(`E${i + 2}`);
258 +       console.log(`E${i + 2}`);
259 +       console.log(cell.value);
260 +       cell.value = Number(bonus_update);
261 +       await sheet.saveUpdatedCells(); // save all updates in one call
262 +     }
263 +   }
264 +   try {
265 +     res.status(200).json("Update bonus successfully");
266 +   } catch (err) {
267 +     res.status(400).send("Cannot get data");
268 +   }
248 269 });

```

## Burndown chart



## Daily meeting minute

Status	Description	Count
Item Done	Identify and define HRM attributes to use in the database	2/5

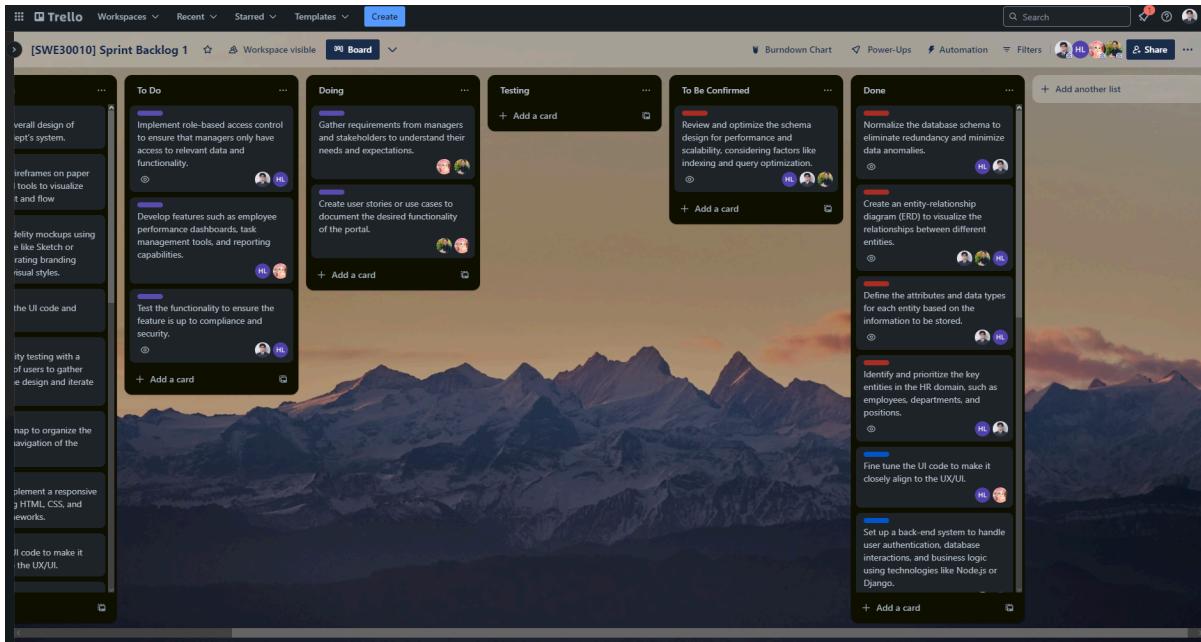
	design	
Item Testing	Review and optimize the schema design.	1/5
Item Confirming	Create an entity-relationship diagram (ERD). Normalize the database schema.	2/5
Item Not Done	N/A	0/5
Next step	Finish normalizing and optimizing the database as well as its relevant code. Start collecting insight to build the Manager Portal.	

### c. Day 8

#### Tasks Sign Up

Item	Description	Member
Finish the ERD schema	Continue working on the ERD to finalise it	Nhat Minh Nhat Huy Hoang Hai
Normalize the data in the database	Continue with cleaning and normalizing the data within the database for better consistency between functions	Nhat Minh Hoang Hai
Review and optimize the schema design.	We review the task at the end to make sure it functions well (those who did not do the coding well review the code, while the one to code will review the schema design).	Nhat Minh Nhat Huy Hoang Hai
Gather requirements from managers and stakeholders.	Our team will engage with managers and stakeholders to gather comprehensive requirements for the portal. As we need to ensure that the portal meets their specific requirements and delivers value to the users.	Nhat Huy Minh Thy
Create user stories or use cases to document the desired functionality of the portal.	We will review the requirements from managers and stakeholders then document the desired functionality of the portal through user stories or use cases. This will help us capture the requirements in a clear and structured manner, allowing for better communication and alignment between the development team and stakeholders.	Nhat Huy Minh Thy

## Trello task board update



## Github repository

Commit

Update Mainpage based on sketch

main

 HaileInnoTech committed 5 days ago

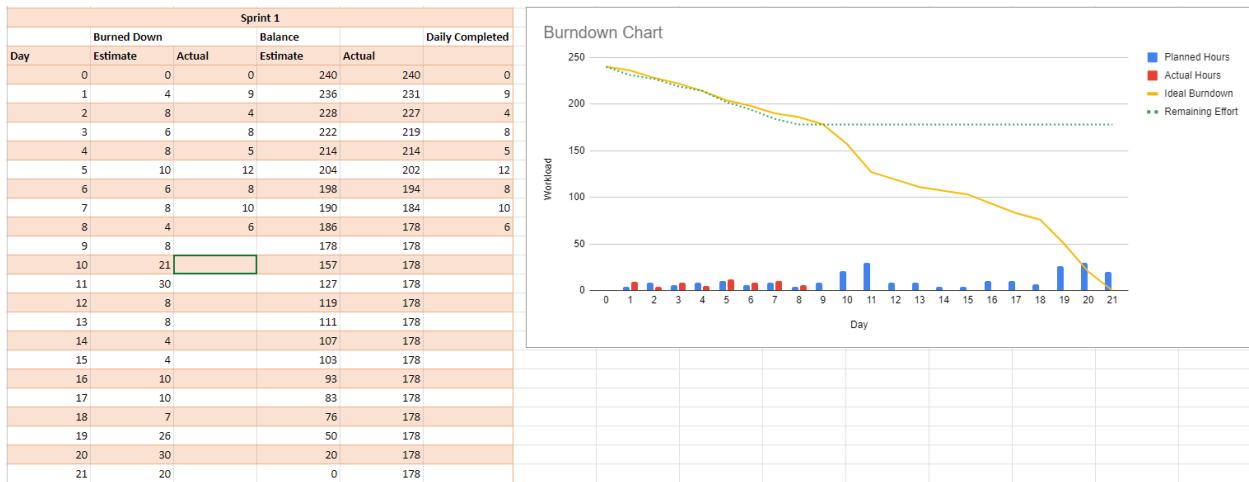
Showing 1 changed file with 2 additions and 2 deletions.

v 4 src/components/Pages/Mainpage.vue

```
... ... @@ -1,4 +1,4 @@  
1 - <!--<template>  
1 + <template>  
2 <div class="relative isolate overflow-hidden bg-white py-10 sm:py-18">  
3 <div class="mx-auto max-w-7xl px-6 lg:px-8">  
4 <h2  
@@ -39,4 +39,4 @@ setInterval(() => {  
39 currentTime.value = new Date().toLocaleTimeString();  
40 40 + currentTime.value = current;  
41 41 }, 1000);  
42 - </script>-->  
42 + </script>
```

0 comments on commit 07b5ecc

## Burndown chart



## Daily meeting minute

Status	Description	Count
Item Done	Create and normalize the schema as well as the database.	2/5
Item Testing	N/A	0/5
Item Confirming	Review and optimize the schema design, basically fine tuning schema for better performance	1/5
Item Not Done	Gather requirements from managers and stakeholders. Create user stories or use cases to document the desired functionality of the portal.	2/5
Next step	As gathering requirements and creating user stories goes together, these tasks need to be done with one another, hence we need to finish them together next time.	

## d. Day 9

### Tasks Sign Up

<b>Item</b>	<b>Description</b>	<b>Member</b>
Finish documenting and getting user's requirement	In order to develop the functions, we need this task to be done first.	Minh Thy Nhat Huy
Develop features such as employee performance dashboards, task management tools, and reporting capabilities.	These features will allow users to track performance metrics, manage their attendance, and generate insightful reports to support decision-making processes. We aim to enhance productivity and enable data-driven decision-making within the organization with the development of these functionalities.	Hoang Hai Minh Thy
Implement role-based access control to ensure that managers only have access to relevant data and functionality.	We perform this task as to ensure that only authorized managers have access to relevant information based on their roles and responsibilities, thereby enhancing data security and confidentiality.	Nhat Minh Hoang Hai
Test the functionality to ensure the feature is up to compliance and security.	Our team will test the functionality of the system to ensure it complies with regulatory requirements and adheres to security standards.	Nhat Minh Hoang Hai

## Trello task board update

To Do

- + Add a card

Doing

- Develop features such as employee performance dashboards, task management tools, and reporting capabilities.
- Implement role-based access control to ensure that managers only have access to relevant data and functionality.

Testing

- Test the functionality to ensure the feature is up to compliance and security.

To Be Confirmed

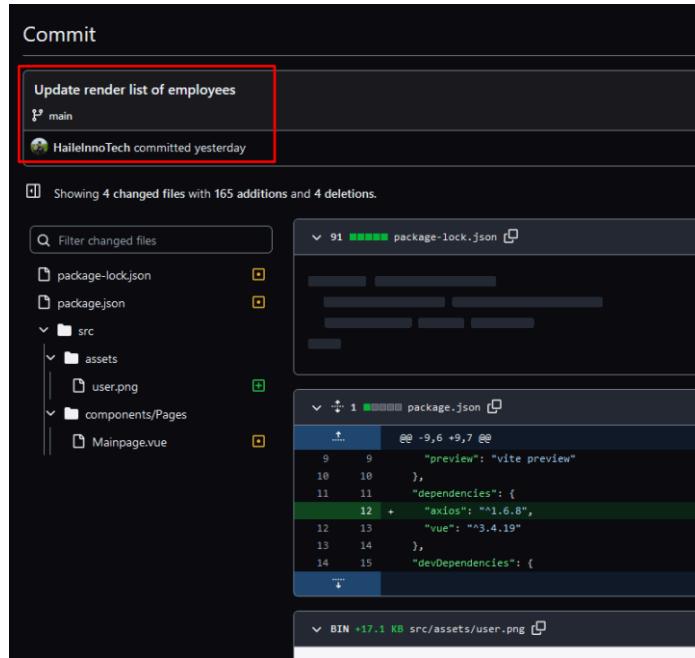
- + Add a card

Done

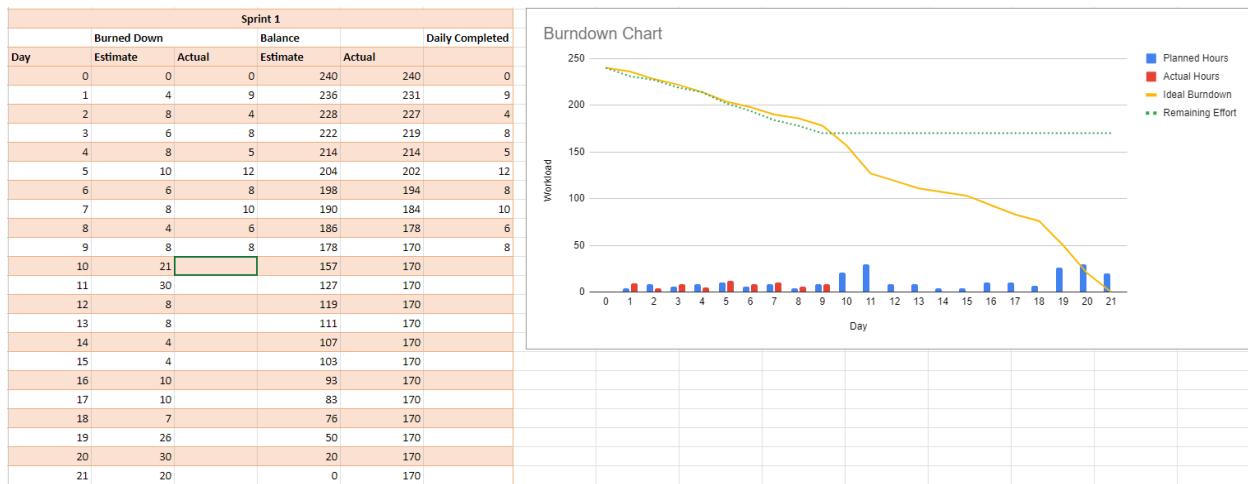
- Create user stories or use cases to document the desired functionality of the portal.
- Gather requirements from managers and stakeholders to understand their needs and expectations.
- Review and optimize the schema design for performance and scalability, considering factors like indexing and query optimization.
- Normalize the database schema to eliminate redundancy and minimize data anomalies.
- Create an entity-relationship diagram (ERD) to visualize the relationships between different entities.
- Define the attributes and data types for each entity based on the information to be stored.

+ Add another list

# Github repository



## Burndown chart



## Daily meeting minute

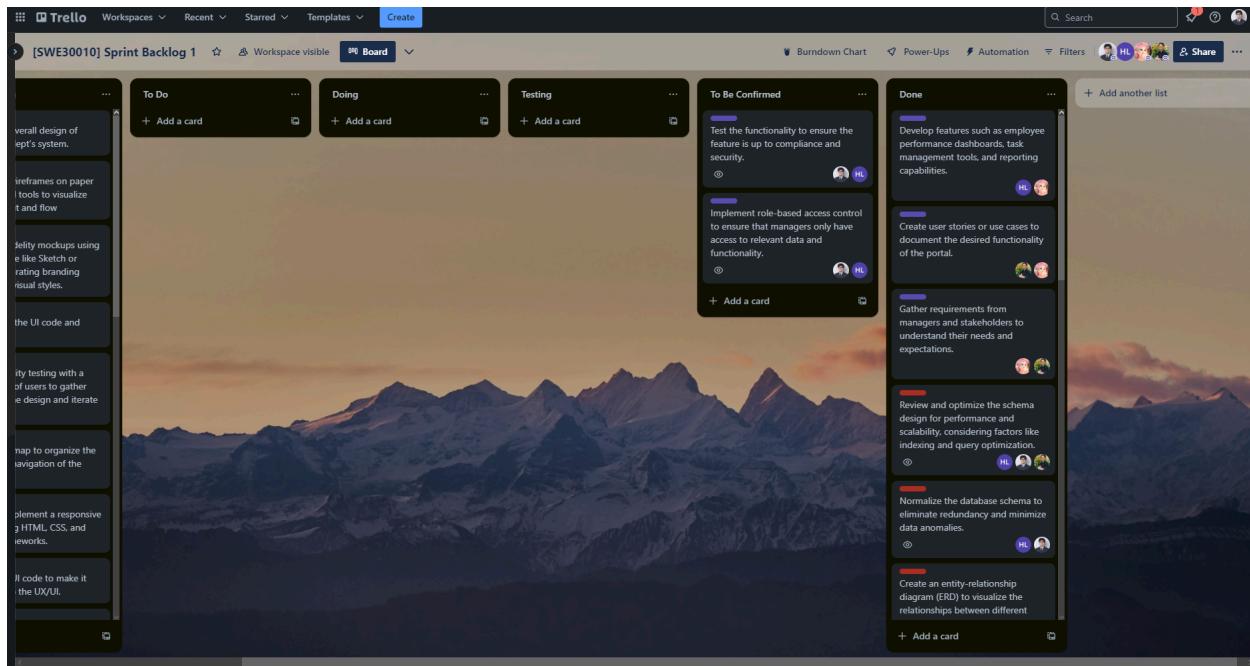
Status	Description	Count
Item Done	Documenting users requirements	1/4
Item Testing	Test the functionality to ensure the feature is up to compliance and security.	1/4
Item Confirming	N/A	0/4
Item Not Done	Implement role-based access control to ensure that managers only have access to relevant data and functionality. Develop features such as employee performance dashboards, task management tools, and reporting capabilities.	2/4
Next step	Finish the website's functions, review testing result and website's security	

## e. Day 10

### Tasks Sign Up

Item	Description	Member
Finish development of manager's functions	We need to ensure the functional aspect of the website to be developed before delivering it to the stakeholders, hence we need to make sure functions are completed	Hoang Hai Minh Thy
Test the functionality to ensure the feature is up to compliance and security.	Testing must be done to ensure functional correctness of the product.	Nhat Minh Hoang Hai
Implement role-based access control to ensure that managers only have access to relevant data and functionality.	Only Managers should have access to the Manager's portal and other unauthorised access should be limited to ensure data security and privacy.	Nhat Minh Hoang Hai

### Trello task board update



## Github repository

The screenshot shows a GitHub commit page for the repository 'HaileInnoTech / Gr2\_Hr\_Management\_FE'. The commit message is 'Update Pagination for employee list' and it was pushed to the 'main' branch by 'HaileInnoTech' 20 hours ago. The commit shows 10 changed files with 406 additions and 39 deletions. A red box highlights the commit message and the file tree on the left. Another red box highlights a specific line of code in the package.json file where the 'vue-awesome-paginate' dependency was updated from '^1.1.46' to '^2.1.0'.

```
diff --git a/package.json b/package.json
index 111111..000000
--- a/package.json
+++ b/package.json
@@ -10,7 +10,10 @@
 },
 "dependencies": {
   "axios": "^1.6.0",
-   "vue": "^3.4.19"
+   "vue": "^3.4.19",
+   "vue-awesome-paginate": "^2.1.0",
+   "vue-router": "^4.3.0",
+   "vuejs-paginate": "^2.1.0"
 },
 "devDependencies": {
   "@itejs/plugin-vue": "^5.0.4",
 }
```

Update employee data view

main

HaileInnoTech committed 20 hours ago

Showing 1 changed file with 2 additions and 2 deletions.

```

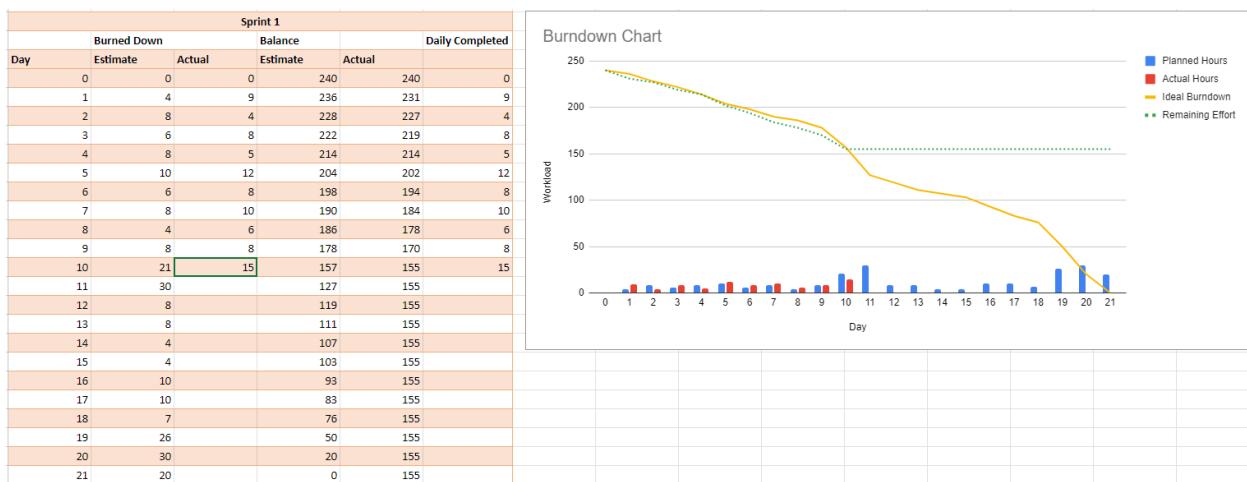
  ↴ 4 src/components/Pages/Individual.vue ↵
  ...
  @@ -1,4 +1,4 @@
  1 - <!-- <template>
  1 + <template>
  2   2   <div class="relative isolate overflow-hidden bg-white py-10 sm:py-18">
  3   3   <div class="mx-auto max-w-7xl px-6 lg:px-8">
  4   4   <button>
  ...
  @@ -130,4 +130,4 @@ export default {
  130  130  };
  131  131  </script>
  132  132
  133 - <style></style> -->
  133 + <style></style>

```

0 comments on commit [616769d](#)

Write Preview

## Burndown chart



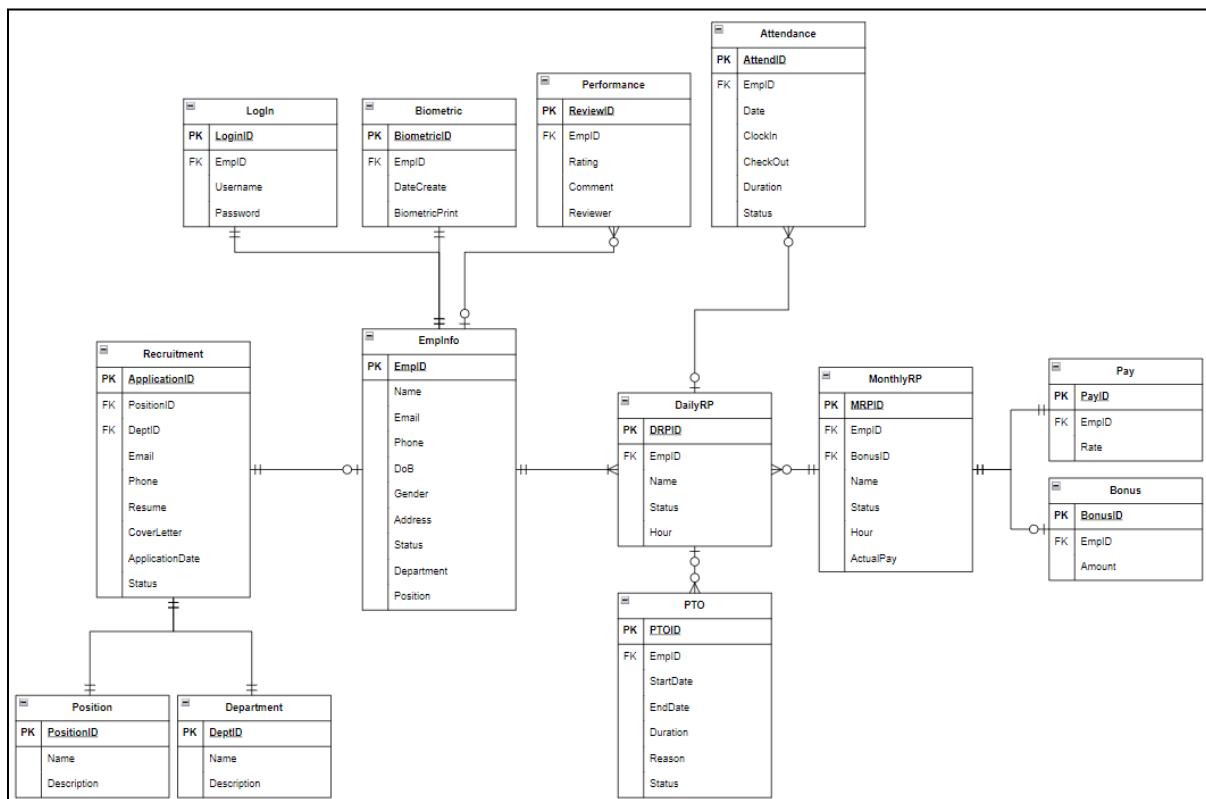
## Daily meeting minute

Status	Description	Count
--------	-------------	-------

Item Done	Develop features such as employee performance dashboards, task management tools, and reporting capabilities	1/3
Item Testing	N/A	0/3
Item Confirming	<p>Test the functionality to ensure the feature is up to compliance and security.</p> <p>Implement role-based access control to ensure that managers only have access to relevant data and functionality.</p>	2/3
Item Not Done	N/A	
Next step		

## APPENDIX

### HRM website's ERD



Desired Database Design for HRM system

## Manager Portal's View

**Employee Manager**

Employee Information | Payroll Manager | Performance Tracker

Employee Name... X  
Department... X Position... X

Displaying 1-8 of 320 results Page 1 of 21

<b>Employee Name</b> Department Position  Address: Phone: Email: <a href="#">View Profile</a>			
<b>Employee Name</b> Department Position  Address: Phone: Email: <a href="#">View Profile</a>			

(1) 2 3 4 ... 40 (2)

Employee Management Window (Manager View)

## *Payroll Management Window (Manager View)*

The screenshot shows the GEMADEPT Employee Management interface. At the top, there's a navigation bar with links for Employee Management, Recruitment, and Reporting & Analysis, along with user icons for home, profile, and logout. A banner image of a port with shipping containers is visible. Below the navigation, a sub-header "Manager" is displayed. The main content area is titled "Employee Management". Underneath, there are three tabs: "Employee Information", "Payroll Manager" (which is selected and highlighted in blue), and "Performance Tracker". Below these tabs are two search input fields: "Employee Name..." and "Status...". A message at the top indicates "Displaying 20 of 403 results" and "Page 1 of 21". The main table has columns for Name, Salary (per hour), Weekly Hours, Monthly Salary, Bonus, and Status. Two rows are visible: Leroy Waslchi (Salary \$29, Bonus \$12, Status Pending) and Ivan Bogen (Salary \$31, Bonus \$0, Status Processed). A modal dialog box is overlaid on the table, titled "Enter Bonus:", containing a text input field with a dollar sign prefix and two buttons: "Confirm" and "Cancel". At the bottom of the page, there are navigation arrows and a footer with the GEMADEPT logo and the tagline "THE WAY FORWARD".

Bonus Editing Window (Manager View)

## **TASK 12P: Sprint Review**

### **SPRINT REVIEW**

When we were defining the tasks for this sprint, we defined 20 general tasks, five for each of our sprint products. Initially, we thought that these tasks were sufficient, however as we developed our product, the scope of it expanded and we inevitably lost track of the tasks at hand and had to resort to creating smaller subtasks in order to keep track of our progress on the website.

These subtasks aren't listed in the WBS. However, our team has noted down the tasks they've gathered from their work independently (and they will be combined in the subsequent section). Therefore, we'll assess the completeness of our product based on both the items outlined in the WBS and those that aren't formally documented.

#### **A. Completed Items**

1. Research the overall design of current Gemadept's system.
2. Create sitemap and navigation paths of the website.
3. Wireframe of the main page of the HRM website.
4. Design mockup of main page, employee manager, payroll manager, manager, reporting dashboard.
5. Design and implement a front-end using HTML, CSS, and JavaScript frameworks.
6. Set up a back-end system to handle user authentication, database interactions, and business logic.
7. Implemented Authentication and Authorization on the website.
8. Added middleware for API security, crypto hashing and salting for authentication.
9. Conduct usability testing.
10. Conduct front and back-end testing.
11. Identify and prioritise the key entities in the HR domain.
12. Define the attributes and data types for each entity based on the information to be stored.
13. Create an entity-relationship diagram (ERD).
14. Gather requirements from managers and stakeholders.
15. Create user stories or use cases to document the desired functionality of the portal.
16. Payroll module is completed ahead of schedule (scheduled for Sprint 2 but finished in Sprint 1)

## **B.Incomplete Items**

Even though we were able to complete most of the website's function:

**a. UI/UX**

1. Responsive design.
2. The reporting dashboard is generally incomplete.
3. Fine tune the UI code to make it closely align to the design mockup.

**b. System functionality**

1. Generate reports.
2. Align the database design to ERD.
3. Deep testing.
4. Mass email sending.
5. Register new accounts.
6. User role page access.
7. Real-time Notification, chatbox support.
8. Biometric system for attendance check.
9. Monthly mockup data for charts.
10. Normalise the database data, review and optimise the schema design.

## **C.Rationale**

There are several reasons why we were unable to complete some of the items on time. One of them was the fact that we underestimated the sheer complexity of some of the tasks. For instance, the objective of Sprint 1 was to implement a base HRM website with some basic HRM operations. However, we severely underestimated the complexity of developing a website as there are many subtasks involved. This meant more time was needed to complete the task, delaying the completion of other tasks significantly. Another problem was that the task descriptions were too vague in the beginning. We learnt the hard way that we should have divided our tasks into smaller, more manageable portions to get better results.

In order to improve this, we would need to research the tasks at hand more thoroughly to be able to understand what needs to be done. With a clearly defined task, we can much more easily determine their complexity, the subtasks involved and have a more accurate time estimation. Furthermore, we should discuss our product and process with the tutor or other experienced personnel to gain insight into how feasible a task is.

## **D.Suggestions for Improvement**

After the sprint review, our stakeholders requested for some adjustments to be made with the current project, including:

- 1. User access management:** Users should only be able to access what is available to them, depending on their role.
- 2. System improvements:** Improvements to design, control management, implementation of Captcha.
- 3. DB improvements:** Transfer current database system (Google Sheets) to a proper one such as MariaDB or MongoDB.

## E. Appendix



*Figure E.1: Our team's sprint review presentation*

# **TASK 13P: Sprint Retrospective**

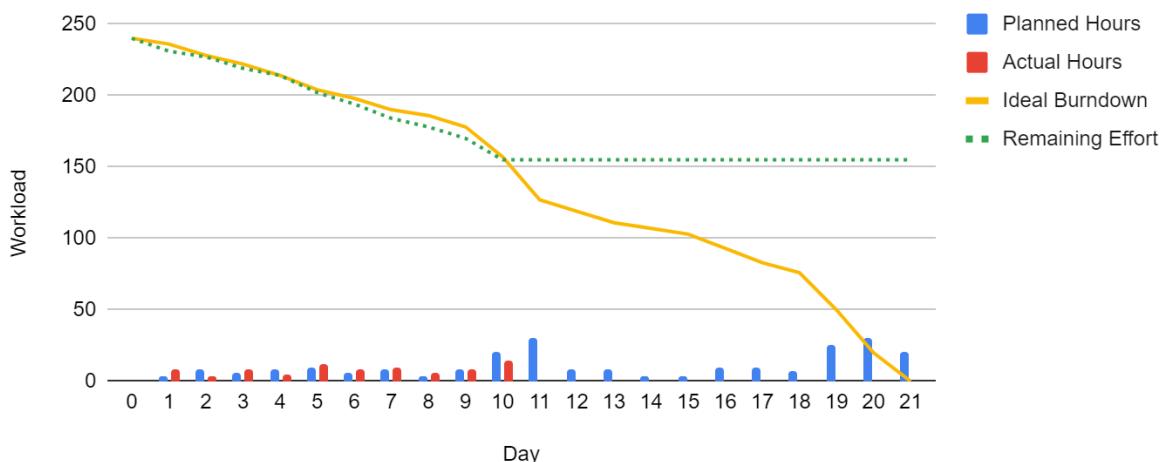
## **Introduction**

After our first sprint is completed, we feel the need to revisit the process of completing our sprint to get a better understanding of what we have done that is capable and what we could have done better. To assist this task, we will attempt to answer some retrospective questions as well as document further ideas that our team has about this sprint.

### **1. Sprint Retrospective**

#### **A. Team's velocity**

Burndown Chart



*Figure 1.1: Burndown Chart*

Sprint 1						
	Burned Down		Balance			Daily Completed
Day	Estimate	Actual	Estimate	Actual		
0	0	0	240	240		0
1	4	9	236	231		9
2	8	4	228	227		4
3	6	8	222	219		8
4	8	5	214	214		5
5	10	12	204	202		12
6	6	8	198	194		8
7	8	10	190	184		10
8	4	6	186	178		6
9	8	8	178	170		8
10	21	15	157	155		15

Figure 1.2: Burndown Table

- Did your team overestimate your ability? Or Did you under-estimate the effort required to complete the tasks?
  - Analysing our team's burndown chart for the initial ten days of the sprint, it appears we may have slightly underestimated the effort required to finish each product within the sprint. However, this discrepancy isn't major, as we're managing to wrap up some work that has been planned later ahead of time. It still highlights the need for us to meticulously reconsider and tweak our approach to subsequent sprint products, which involves breaking down heavier tasks and allocating them to days with fewer hours required, thereby balancing out the days that demand more effort.
- What can you do in order to get a better understanding of the "complexity" of the tasks required? Or What can you do in order to get better time estimates next time?
  - We need to perform regular review with our tutor to refine our plan and adjust our effort accordingly. We would also benefit from observing the plan of other professionals, whose template can be found from the internet.
  - We also need to break down tasks into more detailed descriptions or create additional subtasks for our product's main tasks. This is necessary because we sometimes struggle to keep track of all the tasks we need to do. Moreover, as the project nears completion, the scope of our product expands, and we lack a formal method of recording all the new tasks that arise.

## B. Team's process

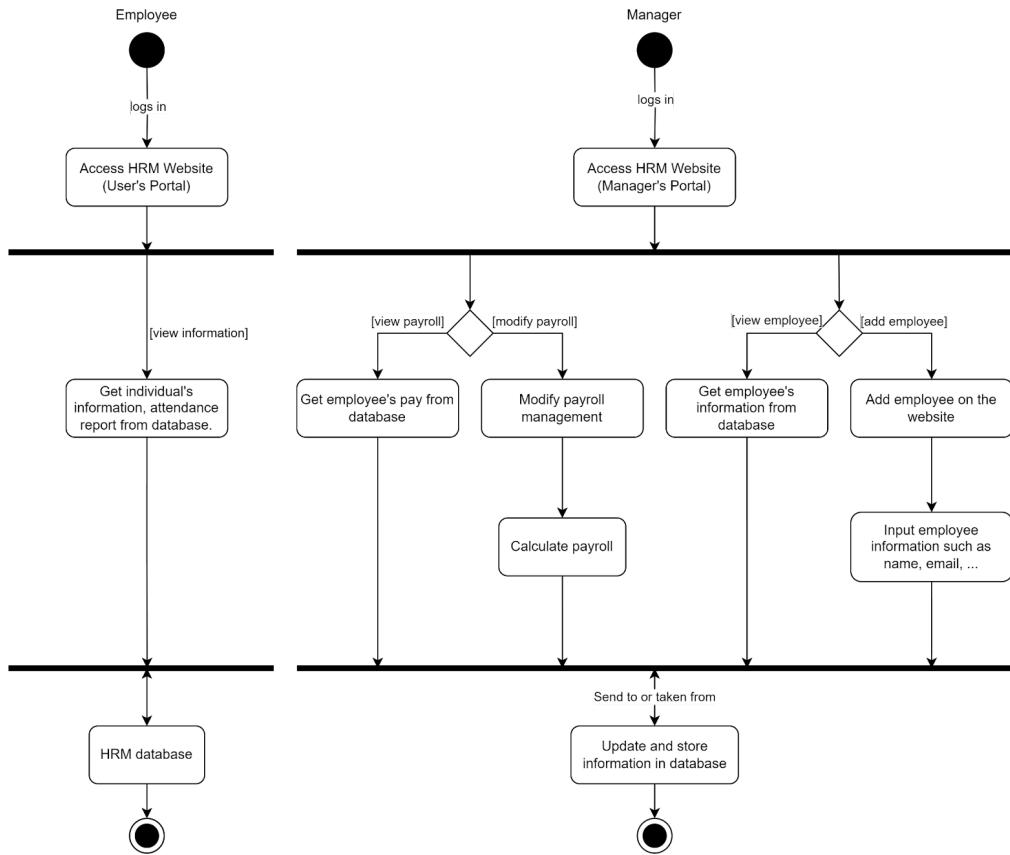
- What is working? Why?
  - Our team's communication is quite effective. We use various channels like Messenger, Discord, and Trello to exchange information about tasks, deadlines, and other notes. This advantage enables us to swiftly pass on new information as soon as it's available.
  - Task allocation is also well done in my opinion. Each team member gets to showcase their strengths. For example, designers handle mockups and writing, developers create and run websites, and the leader handles review and quality control. Overall, I believe our team dynamics are solid.
- What is not working? Why not? Any suggestions to improve the situation if this occurs in the future?
  - Our time management and handling of heavy tasks could be improved. We often make tasks too large and fail to break them down into sufficient detail. As a result, allocating time to these large tasks becomes inaccurate upon closer examination, leading to tight deadlines and lower-quality outcomes.
  - Suggestions for improvement are listed on the former parts, however, our team agrees that in order for our processes and dynamics to be more fluid, we should plan a few weeks further for tasks that have been defined. Moreover, regular meetings to review tasks done by every member would be beneficial.

## TASK 14P: Software Design

### A. Sprint 1 Software Design

With the completion of sprint one, we are able to create an activity UML diagram that captures our website's functionality as well as its design.

Since we have two different kinds of users, employees and managers, their activities differ dramatically. While employees can only view their own attendance, and personal information, managers can do that while also add more employees as well as updating their pay (via the payroll manager) as well.



*Figure A: Software design after Sprint 1*

## B. Justification

### 1. MVC

We followed the Model-View-Controller (MVC) design pattern when creating the post-sprint 1 activity UML design. In this design, the controller (website's backend) coordinates interactions between the model and view, the view (website's frontend) manages the display layer, and the model (website's backend and database) represents the data and business logic. By adhering to the MVC pattern, we can ensure that modifications made to one component don't unduly impact the others, encouraging maintainability and simplifying future development.

### 2. Security

We have implemented various API security measures to protect user data and prevent unauthorized access. Our methods include:

**Salting:** Each password is salted with a random value before hashing.

**Cryptographic Hashing:** Salted passwords are hashed using the bcrypt algorithm, making them resistant to brute-force attacks.

**Limited Database Access:** Access to data in the database is restricted to authorised personnel only (developer and overseer), reducing the risk of data breaches.

**Role-Based Access Control:** Our system incorporates a role-based access control mechanism with two roles: "admin" and "user." This ensures that different users have appropriate levels of access privileges and view features appropriate to their role.

**JSON Web Tokens (JWT):** JWTs are utilised for token generation and management. Upon successful authentication, a unique token is generated and issued to the user. These tokens expire after a set period, mitigating the risk associated with stolen or intercepted tokens.

**Middleware Validation:** The backend API employs middleware to verify the authenticity of incoming requests. Only authenticated users with valid tokens are permitted to make requests, enhancing overall system security.

In this first sprint, our concerns are only about the website's basic functionalities, hence, advanced security has yet to be developed; still, this critical layer of security safeguards our platform against unauthorised access and potential security breaches, reinforcing user trust and confidence in our commitment to protecting their data and privacy.