|  |  |
| --- | --- |
|  | **MINISTRY OF EDUCATION AND TRAINING** |

|  |
| --- |
| **FPT UNIVERSITY** |
| Capstone Project Document |
| Sales Management System for Major Education |

|  |  |
| --- | --- |
| **GSP21SE12** | |
| **Group Members** | Pham Thi Ngoc Ha – SE130460 *(Leader)*  Le Gia Nguyen – SE63042 *(Team member)*  Nguyen Hoang Gia – SE63203 *(Team member)*  ~~Vu Thien Phu – SE63268~~ *~~(Team member)~~* ~~(dropped)~~ |
| **Supervisor** | Mr. Lai Duc Hung *(Instructor)* |
| **Ext Supervisor** | Mr. Phan Duy Tien *(Customer Representative)* |
| **Capstone Project code** | SMSME |

- Ho Chi Minh City, 01/2021 -

**Table of Contents**

[Acknowledgement 4](#_Toc69278987)

[Definition and Acronyms 4](#_Toc69278988)

[I. Project Introduction 5](#_Toc69278989)

[1. Overview 5](#_Toc69278990)

[2. Product Background 5](#_Toc69278991)

[3. Existing Systems 5](#_Toc69278992)

[4. Business Opportunity 8](#_Toc69278993)

[5. Software Product Vision 8](#_Toc69278994)

[6. Project Scope & Limitations 9](#_Toc69278995)

[II. Project Management Plan 10](#_Toc69278996)

[1. Overview 10](#_Toc69278997)

[2. Management Approach 15](#_Toc69278998)

[3. Master Schedule 16](#_Toc69278999)

[4. Project Organization 17](#_Toc69279000)

[5. Project Communication 18](#_Toc69279001)

[6. Configuration Management 19](#_Toc69279002)

[III. Software Requirement Specification 20](#_Toc69279003)

[1. Overall Description 20](#_Toc69279004)

[2. User Requirements 23](#_Toc69279005)

[3. Functional Requirements 77](#_Toc69279006)

[4. Non-Functional Requirements 81](#_Toc69279007)

[5. Other Requirements 82](#_Toc69279008)

[IV. Software Design Description 84](#_Toc69279009)

[1. Overall Description 84](#_Toc69279010)

[2. System Architecture Design 84](#_Toc69279011)

[3. System Detailed Design 88](#_Toc69279012)

[4. Class Specification 98](#_Toc69279013)

[5. Data & Database Design 105](#_Toc69279014)

[V. Software Testing Documentation 110](#_Toc69279015)

[1. Overall Description 110](#_Toc69279016)

[2. Test Plan 112](#_Toc69279017)

[3. Test Cases 114](#_Toc69279018)

[4. Test Reports 114](#_Toc69279019)

[VI. Release Package & User Guides 115](#_Toc69279020)

[1. Deliverable Package 115](#_Toc69279021)

[2. Installation Guides 115](#_Toc69279022)

[3. User Manual 116](#_Toc69279023)

[VII. Appendix 128](#_Toc69279024)

[1. Glossary [Optional] 128](#_Toc69279025)

[2. References [Optional] 128](#_Toc69279026)

[3. Others [Optional] 128](#_Toc69279027)

# Acknowledgement

*[Fill team’s acknowledgement here…]*

# Definition and Acronyms

*[Fill all the definitions, acronyms,… used within the document] in the table format as below]*

|  |  |
| --- | --- |
| **Acronym** | **Definition** |
| AWS | Amazon Web Services |
| BA | Business Analysis |
| BR | Business Rule |
| ERD | Entity Relationship Diagram |
| GUI | Graphical User Interface |
| PM | Project Manager |
| SDD | Software Design Description |
| SPMP | Software Project Management Plan |
| SRS | Software Requirement Specification |
| UAT | User Acceptance Test |
| UC | Use Case |
| API | Application Program Interface |

# I. Project Introduction

## 1. Overview

### 1.1 Project Information

* Project name: Sales Management System for Major Education
* Project code: SMSME
* Group name: GSP21SE12
* Software type: Web Application

### 1.2 Project Team

#### Supervisor

|  |  |  |  |
| --- | --- | --- | --- |
| **Full Name** | **Email** | **Phone Number** | **Title** |
| Lai Duc Hung | HungLD5@fe.edu.vn | 0976.710.580 | Instructor |

#### Team Members

|  |  |  |  |
| --- | --- | --- | --- |
| **Full Name** | **Email** | **Mobile** | **Role** |
| Pham Thi Ngoc Ha | HaPTNSE130460@fpt.edu.vn | 0946.753.053 | Leader |
| Le Gia Nguyen | NguyenLGSE63042@fpt.edu.vn | 0984.272.299 | Member |
| Nguyen Hoang Gia | GiaNHSE63203@fpt.edu.vn | 0799.651.670 | Member |
| ~~Vu Thien Phu (dropped)~~ | ~~PhuVTSE63268@fpt.edu.vn~~ | ~~0946.056.921~~ | ~~Member~~ |

## 2. Product Background

Major Education is an educational organization that collaborates with primary schools, secondary schools or high schools in Ho Chi Minh City, in the fields of English, STEM or Abacus Master. All schools in Ho Chi Minh City are considered as Major Education’s customers/partners. Every school year (from May to September), salesmen from Major Education go to the target schools to introduce and sales their programs.

Currently, the Sales Department of Major Education processes all their works manually, from reports collecting, work-plan planning, salesmen managing and target schools managing, based on free software such as Microsoft Excel, Google Sheet and shared files via Google Drive cloud.

Therefore, they expect a software that can help them, for both salesmen and sales managers, to automate the existing manual workflow. Simultaneously, through this application, sales managers will be able to see the overall picture of the company's business process, to know how many schools are cooperating as well as the school’s state.

## 3. Existing Systems

Nowadays, there are a variety of applications that provide CRM (Customer Relationship Management) solutions for businesses. Major Education suggests our team refer to those softwares to visualize the concept and main features of the future product.

### 3.1 CRMViet

CRMViet is a system that provides comprehensive solutions related to CRM for small and medium-sized enterprises (SME) in Vietnam.

*Link:*

* Web application: <https://crmviet.vn/>.
* Mobile application:
  + Android: <https://play.google.com/store/apps/details?id=vn.altalab.app.crmvietpack>.
  + iOS: <https://apps.apple.com/us/app/crmviet/id1197961185>.

*System analysis:*

1. Roles:

* Administrator
* Manager
* Staff
* … (customized roles depends on each company)

1. Platforms:

* Web app
* Mobile app

1. Main features:

* Manage customers
* Support sales and marketing automation
* Manage workflow, tasks, job reminders,…
* Integrated multi-channel
* …

1. Pros:

* A complete and powerful application
* Fulfill all needs of the Major’s Sales Department

1. Cons:

* Many redundant features
* Price is expensive

### 3.2 GetFly

GetFly is a system that provides comprehensive solutions related to CRM for small and medium-sized enterprises (SME) in Vietnam.

*Link:*

* Web application: <https://getfly.vn/>.
* Mobile application:
  + Android: <https://play.google.com/store/apps/details?id=com.getflyvn.crm>.
  + iOS: <https://apps.apple.com/us/app/getflyvn/id1490060413>.

*System analysis:*

1. Roles:

* Administrator
* Manager
* Staff
* … (customized roles depends on each company)

1. Platforms:

* Web app
* Mobile app

1. Main features:

* Manage customers
* Support marketing automation (via SMS, email, or social network)
* Manage departments, sales team,…
* Manage workflow, tasks, KPIs,…
* Integrate modules extension
* …

1. Pros:

* A complete and powerful application
* User-friendly UI
* Fulfill all needs of the Major’s Sales Department

1. Cons:

* Overpowered, many redundant features
* Require monthly payment

### 3.3 EasyEdu

EasyEdu is a system to manage all activities of many departments in an educational organization, especially for language centers. EasyEdu provides solutions for managing finances, expenses, marketing, human resources, class attendance reports, CRM,…

*Link:*

* Web application: <https://easyedu.vn/>.
* Mobile application:
  + Android:

Teacher: <https://play.google.com/store/apps/details?id=com.easyedu.manager&hl=vi>.

Parent/Student: <https://play.google.com/store/apps/details?id=com.easyedu.customer&hl=vi> .

* + iOS:

Teacher: <https://apps.apple.com/vn/app/easy-manager/id1495280190?l=vi>.

Parent/Student: <https://apps.apple.com/vn/app/easy-edu/id1472824752?l=vi> .

*System analysis:*

1. Roles:

* Administrator
* Staff
* Teacher
* Parent/Student

1. Platforms:

* Web app (for all roles)
* Mobile app (one for Teacher and one for Parent/Student)

1. Main features:

* Manage finances, expenses, marketing campaigns, SMS/chats…
* Manage human resources, departments, agencies,…
* Manage customer relationship
* Manage courses, classes, attendances,…
* Manage tasks, schedules,…
* Provide an online learning platform

1. Pros:

* A complete and powerful application
* Good UI

1. Cons:

* Overpowered and too complicated for what the customer (Sales Department of Major Education) really need
* Is a B2C system (one side is the language center, one side is parent/student)
* Require payment (monthly or annually)

## 4. Business Opportunity

Although the current CRM applications in the market can partially meet the requirements of the business, Major Education still has to pay for many redundant features, which they rarely or would not use. That is why Major Education requests this Sales Management System to specify for their business model. The model of software that we are building is only used for staffs in the Sales Department of Major Education and is not public commercially to the current market. As this Sales Management Software would innovate and improve the effectiveness of Major Education’s manual workflow, so it would save employees’ time and effort to complete their daily tasks as usual.

However, in fact, due to the time and developer team’s experience limitation, we could not covered for the whole Major Education’s sales process yet some features would be upgraded in the next versions of this software.

## 5. Software Product Vision

The target users that SMSME aims to are salesmen and sales managers of the Sales Department of Major Education.

For salesmen, this software would help them manage their work-plan as well as their tasks easier and more visually, submit daily reports more directly and they can also monitor other salesmen’s works.

For sales managers, they would not have to collect and analyze all salesmen’s reports manually. Now they can supervise salesmen or search for reports more easily. Moreover, they can see the overall picture of the company's business process, including what schools are lead customers, who are the most effective salesmen,…

## 6. Project Scope & Limitations

### 6.1 Major Features

FE-01: Sales Manager can manage target schools.

FE-02: Sales Manager can assign salesmen to target schools.

FE-03: Sales Manager can supervise salesmen.

FE-04: Sales Manager and Salesman can manage their daily reports and view all reports.

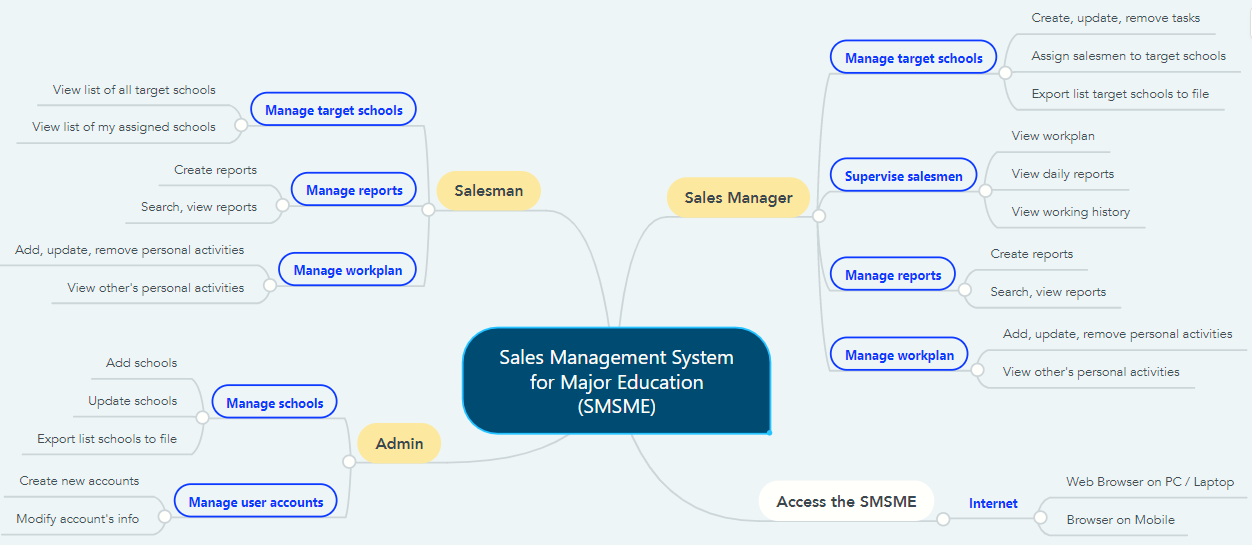
FE-05: Sales Manager and Salesman can manage their work-plan.

FE-06: Sales Manager and Salesman can view target schools, work-plan and daily reports of each other.

FE-07: Sales Manager can comment on other’s

FE-08: Administrator can manage user accounts.

FE-09: Administrator can manage all schools.



*Figure 1. Major Features*

### 6.2 Limitations & Exclusions

LI-1: The SMSME has not supported a Mobile application for role Salesman and Sales Manager. SMSME now is a responsive Web application.

LI-2: The SMSME does not support communication channel with customers.

LI-3: The SMSME has not supported feture “Manage Salesmen” or “Manage team” yet.

# II. Project Management Plan

## 1. Overview

### 1.1 WBS & Estimation

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **WBS Item** | **Complexity** | **Est. Effort**  **(man-days)** |
| ***1*** | ***Initiating*** |  | ***10*** |
| 1.1 | Meet customers at Viet Anh School (Major Education) | Simple | 1 |
| 1.2 | Gain requirements | Medium | 2 |
| 1.3 | Write “Current System Description” | Medium | 4 |
| 1.4 | Confirm “Current System Description” with customers | Simple | 1 |
| 1.5 | Write Report 1 (Introduction) | Medium | 2 |
| 1.6 | Collect stakeholders’ contact information | Simple | 0 |
| ***2*** | ***Planning*** |  | ***4*** |
| 2.1 | Discuss project scope with development team | Simple | 1 |
| 2.2 | Estimate time to finish the project | Simple | 0 |
| 2.3 | Discuss project’s limitations and risks | Simple | 0 |
| 2.4 | Discuss model to use for project management | Simple | 0 |
| 2.5 | Create project plan & work breakdown structure (WBS) | Simple | 1 |
| 2.6 | Assign roles in this project for each team member | Medium | 0 |
| 2.7 | Write Report 2 (Project Management) | Medium | 2 |
| ***3*** | ***Executing*** |  | ***100*** |
| **3.1** | **Analysis** |  | **17** |
| 3.1.1 | Analyse users’ requirements:   * Identify stakeholders * Identify user roles in this system * Identify needs and activities of each role | Complex | 4 |
| 3.1.2 | Analyse materials that Major provided   * Analyse Major current working process (7 steps) * Analyse Excel files (including sample data of target schools, assignment, daily reports,…) | Medium | 2 |
| 3.1.3 | Research business definitions | Simple | 1 |
| 3.1.4 | Use free trial of CRM applications (CRMViet, EasyEdu,…) | Simple | 1 |
| 3.1.5 | Create Use case Diagram | Complex | 3 |
| 3.1.6 | Write Report 3 (Software Requirement Specification – SRS) | Complex | 6 |
| **3.2** | **Designing** |  | **13** |
| 3.2.1 | Create Entity Relationship Diagram (ERD) | Medium | 2 |
| 3.2.2 | Create Database Diagram | Simple | 0 |
| 3.2.3 | Create Architecture Diagram | Medium | 2 |
| 3.2.4 | Standardize provided Excel files | Simple | 1 |
| 3.2.5 | Write Report 4 (Software Design Document – SDD) | Complex | 8 |
| **3.3** | **Prototyping** |  | **9** |
| 3.3.1 | Draw draft major screens on paper | Simple | 2 |
| 3.3.2 | Create prototype for role Admin | Simple | 1 |
| 3.3.3 | Create prototype for role Salesman | Medium | 3 |
| 3.3.4 | Create prototype for role Sales Manager | Medium | 3 |
| **3.4** | **Implementing** |  | **44** |
| 3.4.1 | Install coding environment (Eclipse, Visual Studio Code) | Simple | 0 |
| 3.4.2 | Identify coding conventions (Front-end & Back-end) | Medium | 1 |
| 3.4.3 | Implement Admin module – Iteration 1 |  |  |
| 3.4.3.1 | Update SRS and SDD (Iteration 1) | Medium | 4 |
| 3.4.3.2 | Create test plan and test cases (Iteration 1) | Medium | 2 |
| 3.4.3.3 | Code and perform unit test (Iteration 1) | Complex | 5 |
| 3.4.3.4 | Deploy package 1 | Simple | 0 |
| 3.4.4 | Implement Salesman module – Iteration 2 |  |  |
| 3.4.4.1 | Update SRS and SDD (Iteration 2) | Medium | 4 |
| 3.4.4.2 | Create test plan and test cases (Iteration 2) | Medium | 2 |
| 3.4.4.3 | Code and perform unit test (Iteration 2) | Complex | 7 |
| 3.4.4.4 | Execute integration test and system test (Iteration 2) | Medium | 3 |
| 3.4.4.5 | Deploy package 2 | Simple | 0 |
| 3.4.5 | Implement Sales Manager module – Iteration 3 |  |  |
| 3.4.5.1 | Update SRS and SDD (Iteration 3) | Medium | 4 |
| 3.4.5.2 | Create test plan and test cases (Iteration 3) | Medium | 2 |
| 3.4.5.3 | Code and perform unit test (Iteration 3) | Complex | 7 |
| 3.4.5.4 | Execute integration test and system test (Iteration 3) | Medium | 3 |
| 3.4.5.5 | Deploy package 3 (product version Alpha) | Simple | 0 |
| **3.5** | **Testing** |  | **14** |
| 3.5.1 | Write Report 5 (Testing) | Medium | 5 |
| 3.5.2 | Fix bugs and retest | Medium | 5 |
| 3.5.3 | Get feedback from customers after acceptance test | Medium | 3 |
| 3.5.4 | Deploy full package (product version Beta) | Simple | 1 |
| **3.6** | **Support** |  | **3** |
| 3.6.1 | Write Report 6 (User Guides) | Medium | 2 |
| 3.6.2 | Train customers (end-users) how to use this application | Simple | 1 |
| ***4*** | ***Monitoring and Controlling*** |  | ***7*** |
| 4.1 | Evaluate project’s code quality | Complex | Cannot estimate due to SCRUM model |
| 4.2 | Refractor/Optimize code (optional) | Complex |
| 4.3 | Tracking project’s progress, scope and risks | Medium |
| 4.4 | Write Report 7 (Final Project Report) | Complex | 7 |
| ***5*** | ***Closing*** |  | ***6*** |
| 5.1 | Create final presentation | Simple | 2 |
| 5.2 | Practice Thesis presentation | Medium | 3 |
| 5.3 | Present at Capstone Project Defense | Complex | 1 |
| ***Total Estimated Effort (man-days)*** | | | ***128*** |

### 1.2 Project Objectives

To create an utility, practical and friendly sales management application for Sales Department of Major Education.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Quality Stage** | **No. of Defects** | **% of Defect** | **Notes** |
| 1 | Initiating | 2 | 8.7% | * Gain sufficient and useful information for requirement analysis in the next stage. * All team members must understand clearly customer’s situation and demands. |
| 2 | Planning | 1 | 4.3% | Project plan must be attainable, time-bound and specific. |
| 3 | Analysis | 4 | 17.4% | Confirm SRS (functional requirements, non-functional requirements, business rules) with the customers before moving to “Implementing” stage. |
| 4 | Designing | 4 | 17.4% | Design Use Cases Diagram, ERD and Database based on result of stage 3 (Analysis).  Review carefully with the Instructor before moving to the next stages. |
| 5 | Prototyping | 3 | 13.2% | Design and confirm with customers all screens of all roles before move to “Implementing” stage. |
| 6 | Implementing | 4 | 17.4% | * Strictly follow the general coding convention. * All team members need to update code regularly. |
| 7 | Testing | 2 | 8.7% | * Each developer applys unit test before merge code with each other. * Both Front-end code and Back-end code need to be tested carefully before deploying to the Server. * Have Acceptance test (-test) in the customet-side, in real business environment. |
| 8 | Support | 1 | 4.3% | Project team needs to provide User Guide documents, and also spend time in real business environment to support to the customers at the beginning of the deployment. |
| 9 | Monitoring & Controlling | 1 | 4.3% | * Not only Team Leader needs to track the progress of other team members. * Using version control to keep track of the changes in stages “Designing”, “Prototyping” and “Implementing”. * Applying SCRUM model. |
| 10 | Closing | 1 | 4.3% | All team members need to join slide designing and prepare thoroughly for the final presentation. |
| ***Total*** | | ***23*** | ***100%*** |  |

### 1.3 Project Risks

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Risk Description** | **Impact** | **Possibility** | **Response Plans** |
| **I** | **Requirement risks** | | | |
| 1 | Development team misunderstands or unclears customer requirements | High | High | Meet customers regularly to confirm our thoughts or do not hesitate to ask customers via Zalo group chat. |
| 2 | Customers do not clear about their requirements | High | Medium | Create prototypes or suggest our solutions to customers and confirm with them what features they want to use. |
| 3 | Unexpected scope change | Medium | Low | If new feature is justifiable and time remaining is enough, development team will implement that feature.  If not, team need to deal with customers to implement that feature in the next version. |
| **II** | **Techical risks** | | | |
| 4 | Lack of technical knowledge or poor coding skills | High | Medium | * Enhance self-research. * Exchange knowledge with other team members or other teams. |
| 5 | Poor code quality | Medium | Medium | Strictly follow code convention, carry out acceptance test to ensure quality source code. |
| **III** | **Human risks** | | | |
| 6 | Restricted in direct meeting and communication due to Covid-19 | High | High | * Team can use online meeting platforms (Google Meet, Zoom,…) or send resources via email. * Team can deploy live demo on Ngrok for customers using. |
| 7 | Conflicts between team members | High | High | Team need a meeting, all members share uncomfortable things. |
| 8 | Over deadline | High | High | * Each member has to report their working progress daily. * Team members need to support each other to finish tasks in time. |
| 9 | Lack of real experiences | Low | High | Ask instructor or customer Exchange knowledge with other team members or other teams. |

## 2. Management Approach

The project management model our team chose to manage and implement in this project is Agile methodology, for more specific, is SCRUM framework.

### 2.1 Project Process

We chose Scrum framework because of the following reasons:

* As we are making real product for a real company, we need to satisfy the customers first by early deliverables.
* Scrum requires regular meeting and communication, so development team will closely follow user’s requirements, reduce deviation.
* Scrum welcome changes, even during the developing process.
* Scrum allows us to divide work into small sprints, which will deliver module package each sprint, so we can easily manage bugs, fix bugs faster and reduce risks.

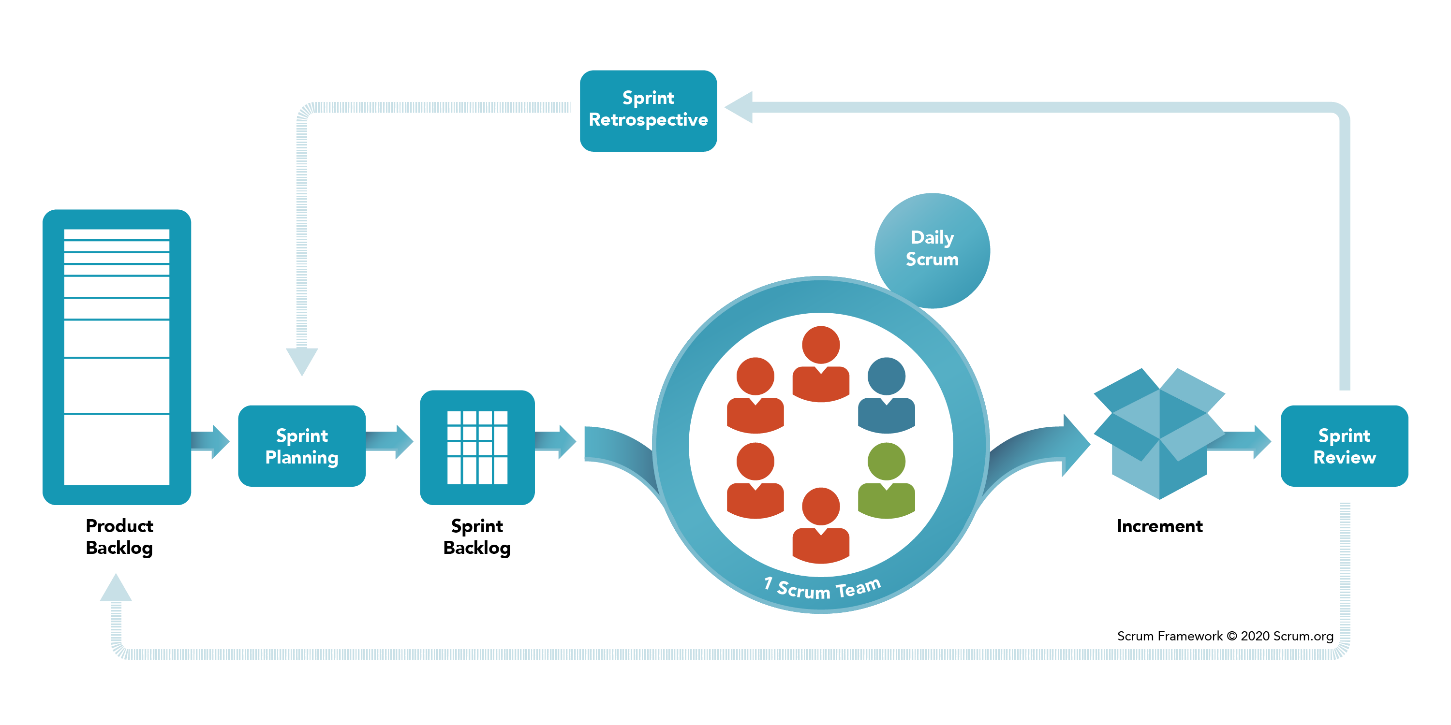


Figure 1. SCRUM Framework

*Reference:* [*https://www.scrum.org/resources/what-is-scrum*](https://www.scrum.org/resources/what-is-scrum)

### 2.2 Quality Management

To enhance the quality, the team have to:

* Follow all code conventions and ensure clean code.
* Refer to and study similar working applications in the market.
* Ensure all team members clearly understand user requirements and prototype.

### 2.3 Training Plan

|  |  |  |  |
| --- | --- | --- | --- |
| Training Area | Participants | When, Duration | Waiver Criteria |
| Java Spring Boot | All team members | Week 3 | Mandatory |
| HTML, CSS, Javascript (ES6) | All team members | Week 1 | Mandatory |
| SCSS | HaPTN, GiaNH | Week 1 | Optional |
| ReactJS | All team members | Week 3 | Mandatory |
| UML 2.0 conventions | All team members | Week 2 | Mandatory |
| Git, Github | All team members | Week 1 | Mandatory |

## 3. Master Schedule

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Deliverable** | **Due Date** | **Deliverable Scope** |
| 1 | Project Introduction (Report 1) | Week 2 |  |
| 2 | Project Plan (Report 2) | Week 2 | Work Breakdown Structure (WBS) |
| 3 | SRS (Report 3) | Week 4 | Use case Diagram and Specification, non functional requirements, screen flow and description |
| 4 | Design (Report 4) | Week 6 | Architecture Diagram, Detailed Design, ERD, Database Diagram, Prototypes |
| 5 | Code Package 1: module “Admin” (Front-end) | Week 8 | Codes & Unit test cases |
| 6 | Code Package 2: module “Salesman” (Front-end) | Week 10 | Codes & Unit test, System test cases |
| 7 | Code Package 3: module “Sales Supervisor”  (Front-end) | Week 12 | Codes & Unit test, System test test cases |
| 8 | Code Package 4: module “Sales Manager”  (Front-end) | Week 12 | Codes & Unit test, System test test cases |
| 9 | Code Package 1  (Back-end) | Week 8 | Codes & Unit test cases |
| 10 | Code Package 2  (Back-end) | Week 10 | Codes & Unit test, System test test cases |
| 11 | UAT Package | Week 14 | Code, System test reports |
| 12 | Final Package | Week 15 | Final codes & documents, User guides |

## 4. Project Organization

### 4.1 Team & Structures

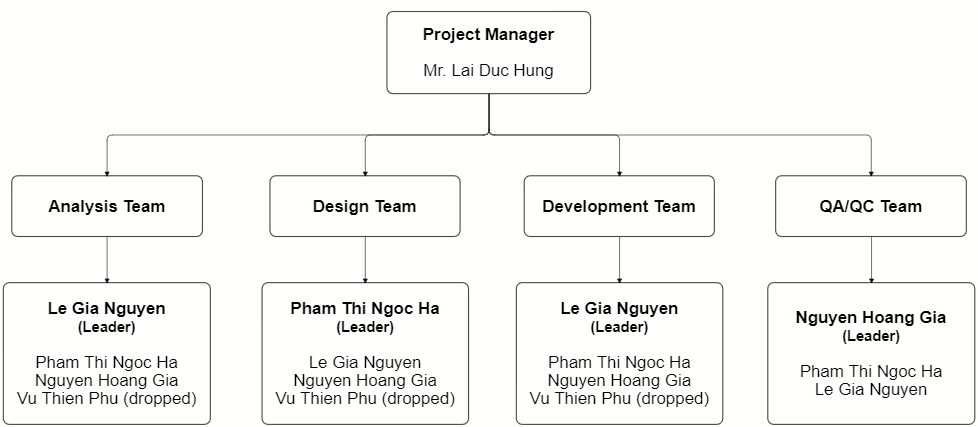


Figure 2. Team & Structures

### 4.2 Roles & Responsibilities

|  |  |
| --- | --- |
| **Role** | **Responsibility** |
| Project Manager | * Be a bridge between Major Education and FPTU development team in the initial stage of the project. * Give guidance, review deliverables during project time. |
| Analysis Leader | * Communicate with the customers to get requirements. * Analyse customer’s requirements and business then explain to team members. * Generate use cases and diagrams in Software Design Description (SDD). * Review and adjust project’s documents made by analysis members. |
| Analysis Member | * Communicate with customers to get requirements. * Analyse customer’s requirements and business. * Draw software diagrams (use cases diagram, ERD,…) and write documents (SRS, SDD). |
| Design Leader | * Design prototypes for website layouts. * Assign tasks for team members. |
| Design Member | * Design prototypes for tasks assigned by the Design Leader. * Review draft prototypes of Design Leader. * Review prototypes of each other. |
| Technical Leader | * Study and decide architecture and technologies using in the project. * Assign tasks for development team members. * Track the work progress of the project. * Review member's code and fix errors. |
| Developer | * Implement code for tasks assigned by the Technical Leader. * Implement unit test. |
| Test Leader | * Write test cases and test plan for the project. * Assign tasks for team members. * Review test results of team members and make test strategy. * Implement the system test. * Collect and analyse customer’s comments at phase UAT. |
| Test Member | Write test cases and test modules as tasks assigned by the Test Leader. |

## 5. Project Communication

### 5.1 Communication Plan

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Communication Item** | **Who/ Target** | **Purpose** | **When, Frequency** | **Type, Tool, Method(s)** |
| First meeting | Project supervisor, Project team, Major team (customers) | Introduce the project | Once | Meeting |
| Weekly meeting | Project supervisor, Project team | Review project status | Weekly | Meeting |
| Meeting minute reports | Project supervisor, Project team | Inform supervisor and team members about weekly meeting contents | Weekly | Document file  Submit via email |

### 5.2 External Interface

#### FU Contacts

|  |  |  |  |
| --- | --- | --- | --- |
| Function | Contact Person  (name, position) | Contact address  (email, telephone) | Responsibility |
| Supervisor | **Lai Duc Hung**  Instructor | [HungLD5@fe.edu.vn](mailto:HungLD5@fe.edu.vn)  0976.710.580 | - Provide document templates. - Give instruction to project team. - Review deliverables. - Supervise project status. |

#### Customer Contacts

|  |  |  |  |
| --- | --- | --- | --- |
| Function | Contact Person  (name, position) | Contact address  (email, telephone) | Responsibility |
| Customer Representor | **Pham Duy Tien** Sales Supervisor | [duytien@truongvietanh.com](mailto:duytien@truongvietanh.com)  0963.281.235 | - Give and describe requirements.  - Review deliverables of the development team. |
| Coordinator | **Tran Thi Xuan Tuyen**  Account Manager | [xuantuyen@truongvietanh.com](mailto:xuantuyen@truongvietanh.com)  0938.137.949 | Introduce project and related person in charge of this project |
| Supporter | **Le Quy Mai Huyen**  Vice president | huyen@truongvietanh.com  0984.388.988 | Support to connect between Major team and development team. |

## 6. Configuration Management

### 6.1 Tools & Infrastructures

|  |  |
| --- | --- |
| **Programming languages** | Java (Java EE 11), Javascript (ES6) |
| **Framework** | Spring Boot |
| **Library** | ReactJS, Syncfusion |
| **API** | RESTful API, Postman, Swagger |
| **DBMS** | MySQL |
| **IDEs/Editors** | Eclipse, Visual Studio Code |
| **UML tools** | StarUML, LucidChart, draw.io, Cacoo |
| **Version Control** | git (GitHub) |
| **Deployment server** | Amazon Web Services (EC2, RDS) |
| **Cloud Services** | Firebase Storage (store images) Firebase Realtime Database (store notifications) |
| **Project management tool** | Jira |

### 6.2 Document Management

We create a shared GitHub account only used only for this project, all team members can access to this account.

In this account, we create a repository for storing all materials related to documents, including 7 reports, requirements analysis, Major’s files, UML diagrams of SDD and weekly meeting minutes.

Link: <https://github.com/salesmgt/documentation>.

### 6.3 Source Code Management

In the shared GitHub account mentioned in part 6.2, we create one repository for prototype, one for front-end codes and one for back-end codes.

Link: <https://github.com/salesmgt/prototype>.

# III. Software Requirement Specification

## 1. Overall Description

### 1.1 Product Overview

The Sales Management System for Major Education is a new software system that replaces the current manual working processes for managing target schools, managing salesmen, collecting daily reports, and planning work-plan in the Sales Department of Major Education.

The system context diagram below demonstrates the human components (actors) and external systems that interacted with SMSME for release 1.0.

The SMSME is expected to automate the existing manual workflow of Major Education and increase the working productivity of Major’s staffs by giving motivation between them.

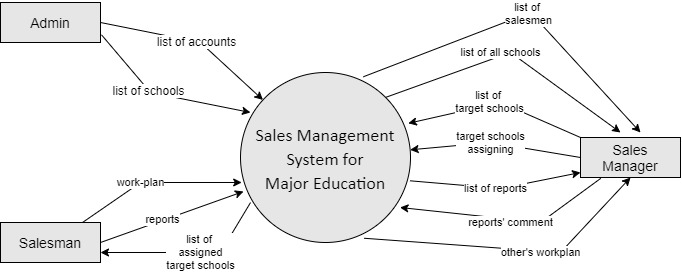


Figure 3. System Context Diagram

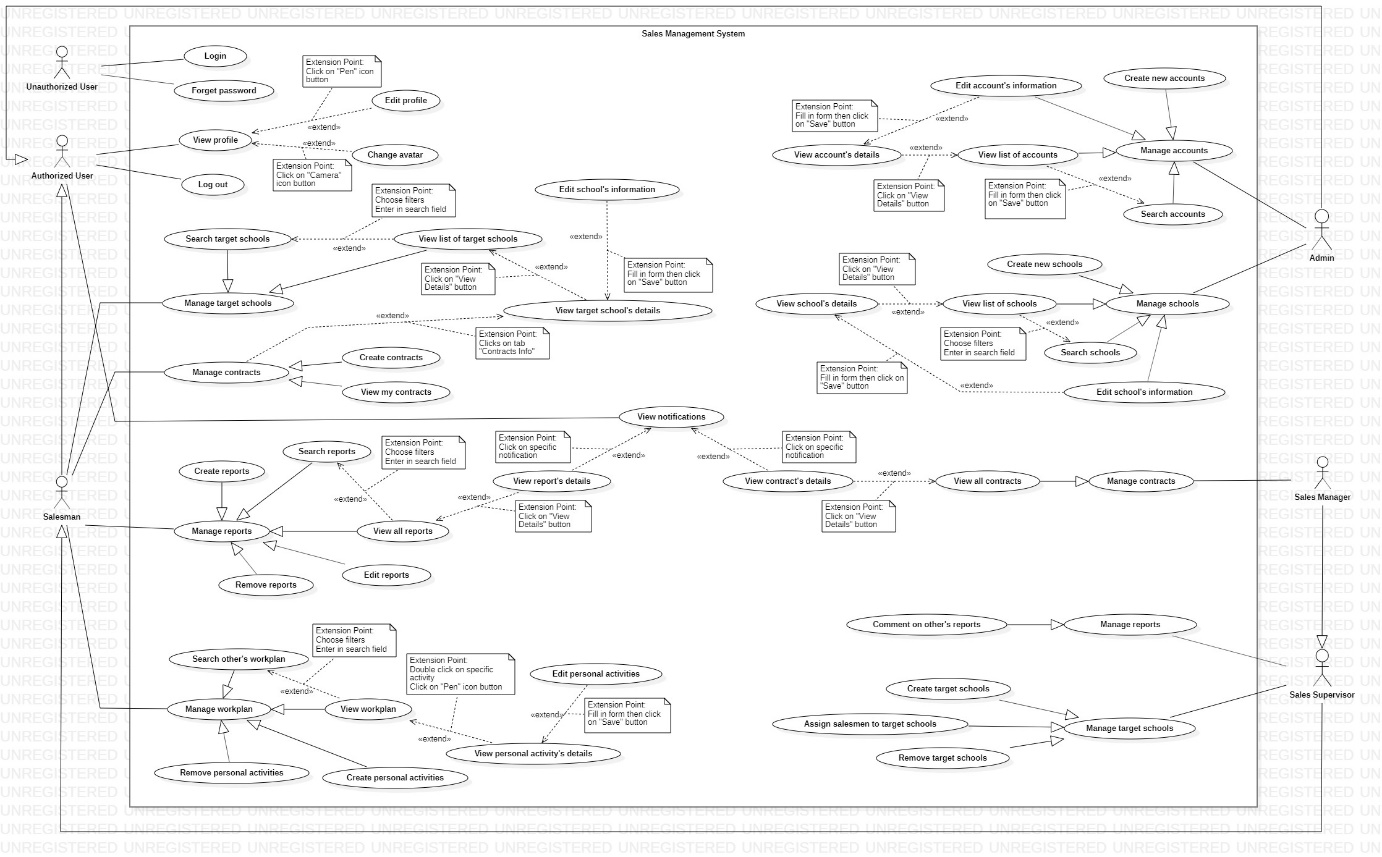
### 1.2 Business Rules

|  |  |
| --- | --- |
| ID | Rule Definition |
| BR-01 | Users shall not register an account. Only Administrators can create new accounts for users. |
| BR-02 | Users shall not have function “Forgot password”. Users must request directly to Administrators and only Administrators can reset users’ forgotten passwords. |
| BR-03 | Users (all roles) can view and edit their profile, including updating basic information (email, phone number, address, birthdate, gender), updating avatar and changing password.  Users shall not edit fields “username” and “working status” (“active”/“inactive”). |
| BR-04 | When an employee quits his/her job or moves to another department, Administrators shall changes his/her “working status” from “active” to “inactive”. |
| BR-05 | Users whose “working status” is “inactive” shall not log into the system. |
| BR-06 | Administrators shall not edit field “username” of the existed accounts. |
| BR-07 | Administrators manage the database of all accounts and schools in the system. |
| BR-08 | Sales Managers shall create lists of target schools before or during phase time/school year. |
| BR-09 | Sales Managers can modify (add, remove, update details or assign) lists of target schools before or during phase time/school year. |
| BR-10 | Sales Managers shall not modify (add, remove, update details or assign) lists of target schools in the past. |
| BR-11 | Sales Managers and Sales Supervisor shall assign Salesmen to target schools. |
| BR-12 | Each target school shall be assigned to only one Salesman at a time. |
| BR-13 | A target school can be assigned many times to different Salesmen. |
| BR-14 | A target school can be unassigned. |
| BR-15 | When Salesmen are assigned to target schools, a notification shall be sent to them. |
| BR-16 | When Sales Managers change the person in charge (PIC) of a target school, notifications shall be sent to both current and new PIC. |
| BR-17 | Only Sales Managers can view list of all target schools.  Salesmen can only view list of their own assigned schools. |
| BR-18 | Target schools are divided into two groups of customer types: “old” and “new”. |
| BR-19 | Business trips to each target school have different purposes, depends on that school’s customer group (old/new). |
| BR-20 | Major’s Sales Department applies the “*Seven-step B2B Sales Process*” as a workflow of the first sales. That process is a guideline and states what to do step-by-step each time a Salesman visits a new target school. |
| BR-21 | Salesmen shall visit their assigned schools. After every business trip to the target schools, Salesmen shall submit a daily report. |
| BR-22 | Salesmen shall successfully meet school’s representatives (principals) of at least 3 new target schools every working day. |
| BR-23 | At the end of phase/school year, if some target schools have not agreed to collaborate with Major yet, the Sales Department will hold a meeting so all Sales Managers and Salesmen can discuss to decide which target schools will be moved into lead nurturing tactics.  Sales Managers can mark a target school as a lead or not.  Salesmen shall not have this function. |
| BR-24 | Salesmen shall not create reports for a date in the past or the future.  The reported date is current date. |
| BR-25 | List of all reports is shared, so any Salesmen or Sales Managers can view it. |
| BR-26 | Sales Managers and Sales Supervisors can comment on other Salesmen’s or Sales Managers’ reports, and shall not comment on their own reports. |
| BR-27 | When Sales Managers comment on reports, notifications shall be sent to the corresponding Salesman or Sales Manager. |
| BR-28 | Sales Managers can only edit or remove comments in the current phase/school year. |
| BR-29 | Salesmen shall not have the function “Comment on reports”. |
| BR-30 | Salesmen and Sales Managers shall not update or remove reports of other Salesmen or Sales Managers. |
| BR-31 | Salesmen can update and remove their own reports only if those reports have not been commented on by Sales Managers yet. |
| BR-32 | Salesmen can only update or remove their own reports in the current phase/school year, not in the past. |
| BR-33 | After a business trip to the target schools, if Salesmen collect some information different from existed school’s details, Salesmen can edit those school’s data. |
| BR-34 |  |
| BR-35 | History of modification in school’s details shall be recorded. |
| BR-36 | Each Salesman or Sales Manager has his/her own work-plan. Anyone can search to view work-plan of the others. |
| BR-37 | An item in work-plan is called “personal activity”.  Both Salesmen and Sales Managers shall not modify (create, edit or remove personal activities) work-plan of the others. |
| BR-38 | Personal activities shall only be created at the current datetime or at a time in the future, shall not be in the past. |
| BR-39 | Salesmen or Sales Managers can only edit or remove personal activities in the current phase/school year. |
| BR-40 | A personal activity has three statuses: “pending” (not started yet or in progress), “completed” and “failed”. |
| BR-41 | When users create a personal activity, the default status shall be “pending”. |
| BR-42 | Any personal activity has endtime.  When the current time reaches endtime, if users have not marked that personal activity as completed, the default status shall be changed into “failed”, otherwise the default status shall be “completed”. |
| BR-43 | Sales Managers supervise Salesmen via daily reports, work-plan and working history (history of assigned schools and their corresponding reports). |
| BR-44 | Sales Managers may play a role as a Salesman, so they may also have their own list of assigned schools, visit target schools, submit daily reports and plan on work-plan. |
| BR-45 | When Admin create a new account, Admin does not create password for users. Password is randomly generated by the system and is sent to the provided user’s email. |
| BR-46 | Only PIC of a target school can view tab “Contract” (to create and view my contracts). |
| BR-47 | Only Sales Manager has right to view all contracts of all target schools.  For other roles, only salesman who is in charge of a target school shall create and view contracts of that target school. |
| BR-48 | The user can only edit fields related to the school’s representative. Any school’s data, the user needs to inform to the Admin and only Admin have right to edit school’s information. |

## 2. User Requirements

### 2.1 Overview

#### Use Case Diagram



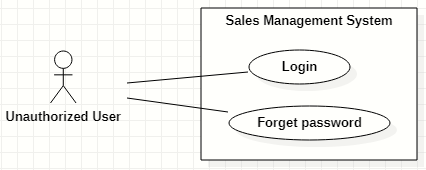
#### System Actors

|  |  |  |
| --- | --- | --- |
| **#** | **Actor** | **Description** |
| 1 | Unauthorized User | The people who have not logged into the system yet. |
| 2 | Authorized User | The people who have logged into the system with the account (Administrator, Salesman, Sales Manager).  All functions of this user are general functions that a normal user can do without considering the role. |
| 3 | Administrator | The Administrators manage data of all accounts and all schools in the system. |
| 4 | Salesman | * The Salesman has business trips to target schools for many purposes, including introducing and selling Major Education’s programs to target schools for cooperation. * Salesman shall create contract and view his/her own assigned target schools’ contracts. |
| 5 | Sales Manager | * Sales Manager manages target schools, Salesmen and supervises salesmen’s reports and work-plans. * Sales Manager also plays a role as Salesman, including business trips. * Sales Manager is not allow to view contracts of others’. |
| 6 | Sales Supervisor | * Sales Supervisor manages target schools, Salesmen and supervises salesmen’s reports and work-plans. * Sales Supervisor also plays a role as Salesman, including business trips. * Sales Supervisor shall view all contracts of all target schools. |

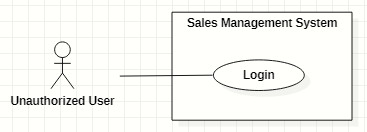
#### Use Cases List

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Use Case** | **Primary Actors** | **Secondary Actors** |
| UC-01 | Login | Unauthorized User |  |
| UC-02 | Forget password | Unauthorized User |  |
| UC-03 | View Profile | Authorized User,  Salesman,  Sales Supervisor,  Sales Manager,  Administrator |  |
| UC-04 | Edit Profile | Authorized User,  Salesman,  Sales Supervisor,  Sales Manager,  Administrator |  |
| UC-05 | Change avatar | Authorized User,  Salesman,  Sales Supervisor,  Sales Manager,  Administrator |  |
| UC-06 | Log out | Authorized User,  Salesman,  Sales Supervisor,  Sales Manager,  Administrator |  |
| UC-07 | View notifications | Authorized User,  Salesman,  Sales Supervisor,  Sales Manager,  Administrator |  |
|  | ***Manage accounts*** | | |
| UC-08 | Create new accounts | Administrator |  |
| UC-09 | Search accounts | Administrator |  |
| UC-10 | View list of accounts | Administrator |  |
| UC-11 | View account’s details | Administrator |  |
| UC-12 | Edit account’s information | Administrator |  |
|  | ***Manage schools*** | | |
| UC-13 | Create new schools | Administrator |  |
| UC-14 | Search schools | Administrator |  |
| UC-15 | View list of schools | Administrator |  |
| UC-16 | View school’s details | Administrator |  |
| UC-17 | Edit school’s information | Administrator |  |
|  | ***Manage target schools*** |  |  |
| UC-18 | Search targetschools | Salesman,  Sales Supervisor,  Sales Manager |  |
| UC-19 | View list of targetschools | Salesman,  Sales Supervisor,  Sales Manager |  |
| UC-20 | View targetschool’s details | Salesman,  Sales Supervisor,  Sales Manager |  |
| UC-21 | Edit school’s information | Salesman,  Sales Supervisor,  Sales Manager |  |
|  | ***Manage contracts*** | | |
| UC-22 | Create contracts | Salesman,  Sales Supervisor,  Sales Manager |  |
| UC-23 | View my contracts | Salesman,  Sales Supervisor,  Sales Manager |  |
|  | ***Manage reports*** | | |
| UC-24 | Create reports | Salesman,  Sales Supervisor,  Sales Manager |  |
| UC-25 | Search reports | Salesman,  Sales Supervisor,  Sales Manager |  |
| UC-26 | View all reports | Salesman,  Sales Supervisor,  Sales Manager |  |
| UC-27 | View report’s details | Salesman,  Sales Supervisor,  Sales Manager |  |
| UC-28 | Edit reports | Salesman,  Sales Supervisor,  Sales Manager |  |
| UC-29 | Remove reports | Salesman,  Sales Supervisor,  Sales Manager |  |
|  | ***Manage workplan*** | | |
| UC-30 | Create personal activities | Salesman,  Sales Supervisor,  Sales Manager |  |
| UC-31 | Search other’s workplan | Salesman,  Sales Supervisor,  Sales Manager |  |
| UC-32 | View workplan | Salesman,  Sales Supervisor,  Sales Manager |  |
| UC-33 | View personal activity's details | Salesman,  Sales Supervisor,  Sales Manager |  |
| UC-34 | Edit personal activities | Salesman,  Sales Supervisor,  Sales Manager |  |
| UC-35 | Remove personal actitivities | Salesman,  Sales Supervisor,  Sales Manager |  |
|  | ***Manage target schools*** | | |
| UC-36 | Create target schools | Sales Supervisor,  Sales Manager |  |
| UC-37 | Assign salesmen to target schools | Sales Supervisor,  Sales Manager |  |
| UC-38 | Remove target schools | Sales Supervisor,  Sales Manager |  |
|  | ***Manage reports*** | | |
| UC-39 | Comment on salesmen's reports | Sales Supervisor,  Sales Manager |  |
|  | ***Manage Contracts*** | | |
| UC-40 | View all contracts | Sales Manager |  |
| UC-41 | View contract’s details | Sales Manager |  |

### 2.2 <<Unauthorized User>> Overview Use Case

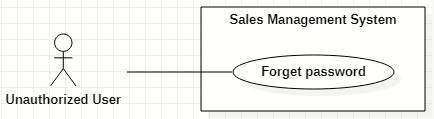


#### Login



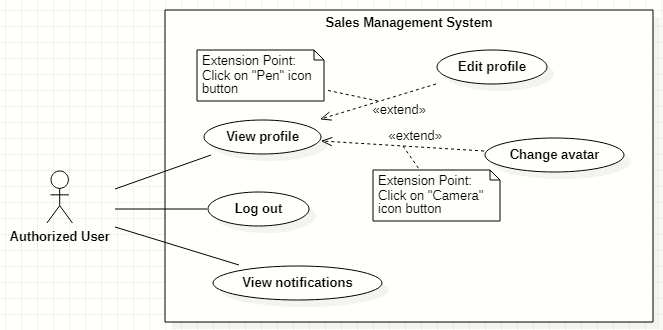
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-01 Login** | | | | |
| Created By: | GiaNH | | Date Created: | 05/02/2021 |
| Primary Actor: | Unauthorized User | | Secondary Actors: | N/A |
| Description: | This use case allows the user to log into the system to use the services of the system. | | | | |
| Trigger: | The user clicks on the “Login” button on the Login screen. | | | | |
| Preconditions: | PRE-1. The user has to own an account with a username and password.  PRE-2. The user has not logged into the system yet. | | | | |
| Post-conditions: | POST-1. The user successfully logs into the system, and be redirected to the Home screen. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user opens the login view in web browsers. | | The system requires identifying information from the user:   * “Username”: text input, required. * “Password”: text input, required. | |
| 2 | The user inputs into the “username” and “password” fields on Login form. | | [Exception 1] | |
| 3 | The user clicks on “Login” button. | | The system allows the user to access into and redirects to the corresponding Home screen (based on the role of user).  [Exception 2]  [Exception 3] | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | The user inputs invalid format content into the “username” and “password” fields. | | The system shows the error message: “Invalid username or password”. | |
| 2 | The user inputs the wrong username or password. | | The system shows the error message: “Wrong username or password”. | |
| 3 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| Priority: | Normal | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | * The required information are username, password. * The username must be 8 – 30 characters length and must not include special characters. * The password must be 8 – 30 characters length and must be hashed encrypted. * Time out for login screen is less than 180 seconds. | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### Forget password

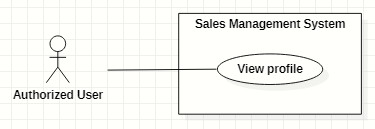


|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-02 Forget password** | | | | |
| Created By: | HaPTN | | Date Created: | 12/04/2021 |
| Primary Actor: | Unauthorized User | | Secondary Actors: | N/A |
| Description: | This use case allows the user to reset their forgotten password. | | | | |
| Trigger: | The user clicks on “Forgot password?” link on the Login screen. | | | | |
| Preconditions: | PRE-1. The user has to own an account with a username and password.  PRE-2. The user has not logged into the system yet. | | | | |
| Post-conditions: | POST-1. The user’s password has been updated in the database.  POST-2. The user is redirected to Login screen. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user clicks on “Forgotten password?” link on the Login screen in Web browsers. | | The system redirects to “Reset your password” screen which requires user to input:   * “Username”: text input, required. * “Email”: text input, required. * “Send me verification code”: button. | |
| 2 | The user inputs into the “username” and “email” fields on “Reset your password” form. | | [Exception 1] | |
| 3 | The user clicks on “Send me verification code” button. | | The system sends a verification code via email that user has just provided.  [Exception 2]  [Exception 3] | |
|  | 4 |  | | The system redirects to “Enter verification code” screen which have:   * “Verification code”: text input. * “Continue”: button. | |
|  | 5 | On “Enter verification code” form, the user inputs into the “verification code” then clicks on “Continue” button. | | The system redirects to “Enter new password” screen which have:   * “New password”: text input, required. * “Re-new password”: text input, required. * “Save”: button.   [Exception 3]  [Exception 4] | |
|  | 6 | The user inputs into the “new password” and “re-new password” fields on “Enter new password”. | | [Exception 5] | |
|  | 7 | The user clicks on “Save” button. | | The system shows the successful message: “Password has been changed successfully”, then redirects to the “Login” screen  [Exception 3] | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | The user inputs invalid format content into the “username” and “email” fields. | | The system shows the error message: “Invalid username or email”. | |
| 2 | The username or email in form do not match to fields “username” and “user\_email” in the database. | | The system shows the error message: “Wrong username or email”. | |
| 3 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
|  | 4 | The user inputs wrong verification code. | | The system shows the error message: “Please check your verification code and try again.”. | |
| Priority: | Normal | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | * The “new password” and “re-new password” must be 8 – 30 characters length and must be hashed encrypted. | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

### <<Authorized User>> Overview Use Case

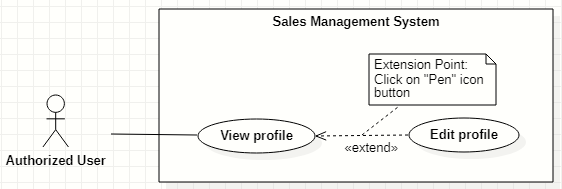


#### View profile



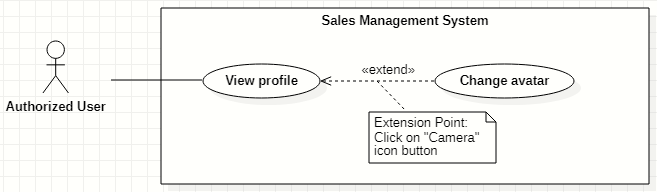
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-03 View profile** | | | | |
| Created By: | GiaNH | | Date Created: | 05/02/2021 |
| Primary Actor: | Authorized User | | Secondary Actors: | N/A |
| Description: | This use case allows the user to view his/her profile. | | | | |
| Trigger: | The user selects “Profile” command in the options menu under the user avatar (at the top right corner of the screen). | | | | |
| Preconditions: | PRE-1. The user has been logged into the system. | | | | |
| Post-conditions: | POST-1. The user’s information is displayed in details as a form. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user clicks on the avatar on the right corner of the topbar. | | The system redirects to the “Profile” screen.  The detail information of the user is displayed on the screen.  [Exception 1] | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| Priority: | Normal | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | N/A | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### Edit profile



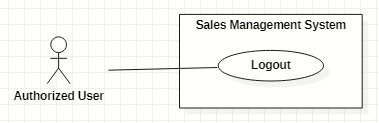
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-04 Edit profile** | | | | |
| Created By: | GiaNH | | Date Created: | | 05/02/2021 |
| Primary Actor: | Authorized User | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to edit his/her profile. | | | | |
| Trigger: | The user clicks on “Pen” icon button. | | | | |
| Preconditions: | PRE-1. The user has been logged into the system.  PRE-2. The user is at the “Profile” screen. | | | | |
| Post-conditions: | POST-1. The user’s information is edited in the system. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | On Profile screen, the user clicks on “Pen” icon button at the right of the information row that he/she wants to edit.  [Alternative 1 ] | | At the selected row, the system opens the collapsed div to show the text field with the following format:   * “Password”: text. * “Address”: text. * “Email”: text. * “Birthdate”: date. * “Gender”: two radio buttons, required, only select one. * “Phone” text. * “Avatar”: image. | |
| 2 | The user inputs the information into the selected text field. | | [Exception 1]  [Exception 2]  [Exception 3] | |
| 3 | The user selects “Save” button.  [Exception 4] | | The system shows the the successful message: “Updated Successfullly”.  [Exception 5] | |
| Alternative Flows: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user selects “Pen” icon button at “Password” line. | | The system shows the accordion that contains following information:   * “Current Password”: text, required. * “New Password”: text, required. * “Confirm New Password”: text, required.   *Continue to step 2 of Normal Flow*. | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | The user inputs invalid format content into fields. | | The system shows the error message: “Incorrect entry”. | |
| 2 | The user inputs the wrong old password. | | The system shows the error message: “Invalid password” and “Update Unsuccessful”. | |
| 3 | The user inputs the confirm-new password that does not match to new password. | | The system shows the error message: “You must enter the same password twice in order to confirm it”. | |
| 4 | The user selects “Cancel” button to cancel updating. | | The system discards all changes and returns the “Profile” screen. | |
| 5 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| Priority: | Normal | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | * All fields related to “password” must be 8 – 30 characters length and must be hashed encrypted. * Fields that could be edited are: password, email, phone number, address and avatar. * Phone must be in format “/(84|0[3|5|7|8|9])+([0-9]{8})\b/g”   *(Resources:* [*https://www.regextester.com/106725*](https://www.regextester.com/106725)*)*   * Fields which are not allowed users to edit are: username, fullname, birthdate and gender. | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### Change avatar



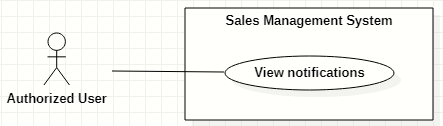
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-05 Change avatar** | | | | |
| Created By: | HaPTN | | Date Created: | | 12/04/2021 |
| Primary Actor: | Authorized User | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to update his/her profile avatar. | | | | |
| Trigger: | The user clicks on “Camera” icon button next to the avatar. | | | | |
| Preconditions: | PRE-1. The user has been logged into the system.  PRE-2. The user is at the “Profile” screen. | | | | |
| Post-conditions: | POST-1. The user’s new avatar is edited in the system. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | On Profile screen, the user clicks on “Camera” icon button next to the avatar image. | | The system opens the Windows Explorer pop-up for user to upload an image. | |
| 2 | The user chooses an image in the Windows Explorer pop-up and clicks on “Open” button.  [Exception 1]  [Exception 2] | | The system updates avatar with the chosen image and shows the the successful message: “Updated Successfullly”.  [Exception 3] | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | The user chooses invalid image format. | | The system shows the error message: “Update Unsuccessful”. | |
| 2 | The user closes the Windows Explorer (by clicking on “x” button or “Cancel” button). | | The system shows the error message: “Update Unsuccessful”. | |
| 3 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| Priority: | Normal | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | * In the Windows Explorer pop-up, the user can only choose one image at a time. * The system only accepts image files in type “.png”, “.jpg”, “.jpeg”. | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### Logout



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-06 Logout** | | | | | |
| Created By: | GiaNH | | Date Created: | 05/02/2021 | |
| Primary Actor: | Authorized User | | Secondary Actors: | N/A | |
| Description: | This use case allows the users to log out of the system. | | | | | |
| Trigger: | The user clicks on “Logout” button. | | | | | |
| Preconditions: | PRE-1. The user has been log into the system. | | | | | |
| Post-conditions: | POST-1. The user logs out the system. | | | | | |
| Normal Flow: | **Step** | **Actor Action** | | | **System Response** | |
| 1 | The user clicks on the “Avatar” icon button to open “Profile” screen. | | | The system redirects to the “Profile” screen. | |
| 2 | The user clicks on “Log out” button at the top right corner of the “Profile” screen. | | | The system clears cookies and access token then returns the “Login” screen. | |
| Alternative Flows: | N/A | | | | | |
| Exceptions: | N/A | | | | | |
| Priority: | Normal | | | | | |
| Frequency of Use: | Usually | | | | | |
| Business Rules: | User’s token is removed in client and the session is destroyed in server. | | | | | |
| Other Information: | N/A | | | | | |
| Assumptions: | N/A | | | | | |

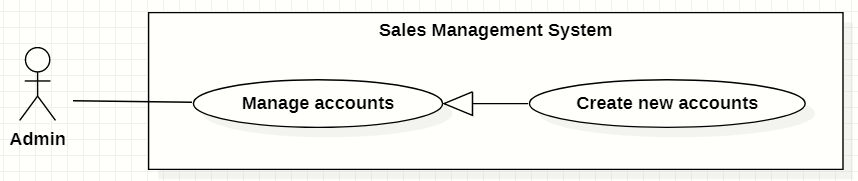
#### View notifications

******

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-07 View notifications** | | | | | |
| Created By: | HaPTN | | Date Created: | 12/04/2021 | |
| Primary Actor: | Authorized User | | Secondary Actors: | N/A | |
| Description: | This use case allows the users. | | | | | |
| Trigger: | The user clicks on “Bell” icon button (at the top right corner of the screen). | | | | | |
| Preconditions: | PRE-1. The user has been log into the system. | | | | | |
| Post-conditions: | POST-1. The user is redirect to the corresponding screen depends on what notification he/she selects. | | | | | |
| Normal Flow: | **Step** | **Actor Action** | | | **System Response** | |
| 1 | The user clicks on “Bell” icon button (at the top right corner of the screen) to view list of notifications. | | | The system shows list of notifications in a popover, near the “Bell” icon button. | |
|  | 2 | The user select specific notification.  [Alternative 1]  [Alternative 2] | | | [Exception 1] | |
| Alternative Flows: | **Step** | **Actor Action** | | | **System Response** | |
| 1 | The user clicks on notification type “report”. | | | The system redirects to “Report details” screen for user to view comment of supervisor/manager. | |
| 2 | The user clicks on notification type “contract”. | | | The system redirects to “Contract details” screen for user to view contract’s details. | |
| Exceptions: | **No** | **Cause** | | | **System Response** | |
| 1 | The user’s device does not connect to the Internet or the server has internal error. | | | The system shows the error message: “Connection Failed or Internal Error”. | |
| Priority: | Normal | | | | | |
| Frequency of Use: | Usually | | | | | |
| Business Rules: |  | | | | | |
| Other Information: | N/A | | | | | |
| Assumptions: | N/A | | | | | |

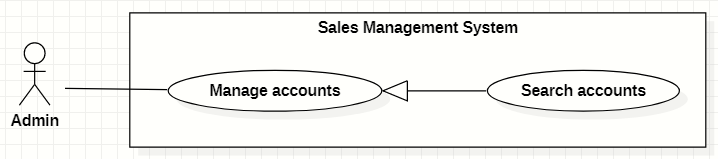
### <<Administrator>> Overview Use Case

#### Create new accounts



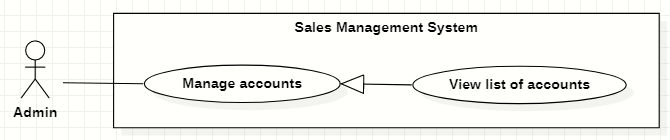
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-08 Create new accounts** | | | | |
| Created By: | GiaNH | | Date Created: | | 15/02/2021 |
| Primary Actor: | Administrator | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to create new account. | | | | |
| Trigger: | The user clicks on the “Create” button in “Accounts” screen. | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2. The user is on the “Accounts” screen. | | | | |
| Post-conditions: | POST-1. The new account is added in the system. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user selects “Accounts” on the navigation bar (sidebar). | | The system redirects the Accounts screen. | |
| 2 | The user selects on “Create” button in screen. | | The system displays the “Create Account” form with the following required information:   * “Username”: input text filed, required. * “Role”: selections (Salesman, Sales Supervisor, Sales Manager, Admin). * “Full name”: input text field, required. * “Email”: input text field. * “Phone”: input text field, required. * “Address”: input text field. * “Gender”: radio buttons, required. * “Birthdate”: date picker. * “Save”: button. * “Cancel”: button. | |
| 3 | The user inputs into required information fields. | | [Exception 1]  [Exception 2] | |
| 4 | The user selects on the “Save” button.  [Exception 4] | | The system shows the successful message: “New account has been created successfully. Please check your email to get the password.”.  [Exception 3]  [Exception 5] | |
| 5 |  | | The system sends the new user an email which contains the his/her password.  [Exception 3] | |
| 6 |  | | The system updates the table of list of schools in the “Schools” screen.  [Exception 3] | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | The user leaves the blank fields or inputs the invalid format content. | | The system shows the error message: “Please input the valid format content”. | |
| 2 | The content that the user inputs into “Confirm password” is not matched the “Password” field. | | The system show the error message: “Confirm password is not matched the password”. | |
| 3 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| 4 | The user selects “Cancel” button. | | The system closes the “Create Account” view and returns the “Accounts” screen. | |
| 5 | The content that the user inputs into “username” field is duplicate in the system. | | The system shows the error message: “The username has already existed”. | |
| Priority: | High | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | * The username must be 8 – 30 characters, not include special characters. * Admin does not create password for users. * Password is randomly generated by the system and is sent to the provided user’s email. * The password which is stored in the Database must be hasheded encrypted by Bcript. * The email must be validate with Regular Expression. * Phone must be in format “/(84|0[3|5|7|8|9])+([0-9]{8})\b/g”   *(Resources:* [*https://www.regextester.com/106725*](https://www.regextester.com/106725)*)* | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### Search accounts



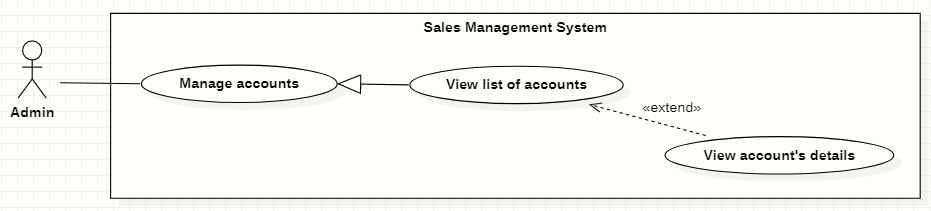
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-09 Search accounts** | | | | | | |
| Created By: | GiaNH | | | Date Created: | | | 16/02/2021 |
| Primary Actor: | Administrator | | | Secondary Actors: | | | N/A |
| Description: | This use case allows the users to find the accounts based on the role, status, username, phone, address,… | | | | | | |
| Trigger: | The user selects filters or inputs in search field on the “Accounts” screen. | | | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2. The user is on “Accounts” screen. | | | | | | |
| Post-conditions: | POST-1. The list of accounts is shown. | | | | | | |
| Normal Flow: | **Step** | | **Actor Action** | | **System Response** | | |
| 1 | | The user inputs in the search field or selects filters to looking for accounts. | | The system displays the list of accounts as the table.  Available filters on this screen:   * “isActive”: selections (true, false) * “Roles”: selections (Admin, Sales Manager, Sales Supervisor, Salesman). * Search field: text field. | | |
| 2 | |  | | The list of Accounts is shown as a table on the Accounts screen.  Table will includes these fields:   * “Username”: text. * “Fullname”: image (avatar) and text. * “Phone”: text. * “Email”: text. * “Role”: Label. | | |
| Alternative Flows: | N/A | | | | | | |
| Exceptions: | **No** | **Cause** | | | | **System Response** | |
| 1 | The content of search field or selecting filter that the user inputted is not matched any data in database. | | | | The system displays “No records found” line. | |
| Priority: | Normal | | | | | | |
| Frequency of Use: | Usually | | | | | | |
| Business Rules: | * The default values of filter “isActive” is “true” * The default values of filter “Role” is “All”. | | | | | | |
| Other Information: | N/A | | | | | | |
| Assumptions: | N/A | | | | | | |

#### View list of accounts



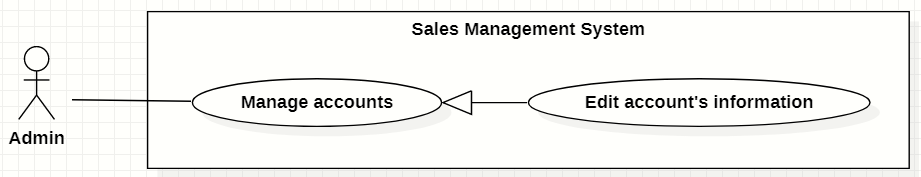
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-10 View list of accounts** | | | | |
| Created By: | GiaNH | | Date Created: | | 15/02/2021 |
| Primary Actor: | Administrator | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to view list of accounts. | | | | |
| Trigger: | The user selects “Accounts” on the navigation bar (sidebar). | | | | |
| Preconditions: | PRE-1. The user has been log into system. | | | | |
| Post-conditions: | POST-1. The list of Accounts is shown as a table on the “Accounts” screen. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user selects “Accounts” on the navigation bar (sidebar). | | The system redirects the Accounts screen. | |
| 2 |  | | The list of Accounts is shown as a table on the Accounts screen.  Table will includes these fields:   * “Username”: text. * “Fullname”: image (avatar) and text. * “Phone”: text. * “Email”: text. * “Role”: Label. | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | The data is null or empty in database. | | The system shows “No records found” line on the table. | |
| Priority: | Medium | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | * The default values of filter “isActive” is “true” * The default values of filter “Role” is “All”. | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### View account’s details



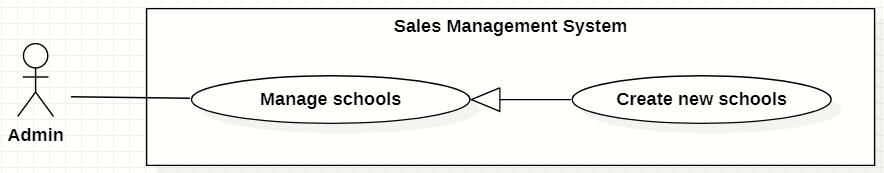
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-11 View account’s details** | | | | |
| Created By: | GiaNH | | Date Created: | | 16/02/2021 |
| Primary Actor: | Administrator | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to view the detail information of an account. | | | | |
| Trigger: | The user clicks on the “View details” command on the menu options at the end of row in table list of accounts. | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2. The user is on the “Accounts” screen. | | | | |
| Post-conditions: | POST-1. The detail information of account is shown as a form. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | On the “Accounts” screen, at the end of each row in the table, the user clicks on action icon button, then selects “View detail” command on the menu options. | | The system displays “Account’s details” view, the account’s information is displayed as a form.   * “Username”: input text filed, required. * “Password”: input password text field, required. * “Confirm password: input password text field, required. * “Role”: selections (Salesman, Sales Supervisor, Sales Manager, Admin). * “Full name”: input text field, required. * “Email”: input text field. * “Phone”: input text field, required. * “Gender”: radio buttons, required. * “Birthdate”: date picker. * “Address”: input text field. * “Save”: button. * “Cancel”: button. | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | N/A | | | | |
| Priority: | Medium | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | N/A | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### Edit account’s information



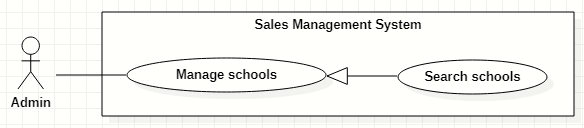
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-12 Edit account’s information** | | | | | | |
| Created By: | GiaNH | | | Date Created: | | | 16/02/2021 |
| Primary Actor: | Administrator | | | Secondary Actors: | | | N/A |
| Description: | This use case allows the users to edit the account’s information. | | | | | | |
| Trigger: | The user edits on fields in “Account’s details” screen then clicks “Save” button. | | | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2. The user is on the “Account’s details” screen. | | | | | | |
| Post-conditions: | POST-1. The Account information is edited in database. | | | | | | |
| Normal Flow: | **Step** | | **Actor Action** | | | **System Response** | |
| 1 | |  | | | The system displays the “Account’s detail” screen as a form with the following required information:   * “Username”: text field, disabled. * “Full name”: input text field. * “Email”: input text field. * “Phone”: input text field, required. * “Gender”: radio buttons. * “Birthdate”: date picker. * “Address”: input text field. * Role: selections (Admin, Sales Manager, Sales Supervisor, Salesman). | |
| 2 | | The user edits on fields in this form. | | | [Exception 1] | |
| 3 | | The user clicks “Save” button when finish. | | | The system has successful edit and displays the successful message: “Updated Successfully”. [Exception 2] | |
| Alternative Flows: | N/A | | | | | | |
| Exceptions: | **No** | **Cause** | | | **System Response** | | |
| 1 | The user inputs invalid format content into fields. | | | The system informs the users that the content of fields is invalid format. | | |
| 2 | The user’s device does not connect to the Internet or the server has internal error. | | | The system shows the error message: “Connection Failed or Internal Error”. | | |
| Priority: | Medium | | | | | | |
| Frequency of Use: | Usually | | | | | | |
| Business Rules: | * Phone must be in format “/(84|0[3|5|7|8|9])+([0-9]{8})\b/g”   *(Resources:* [*https://www.regextester.com/106725*](https://www.regextester.com/106725)*)* | | | | | | |
| Other Information: | N/A | | | | | | |

#### Create new schools

****

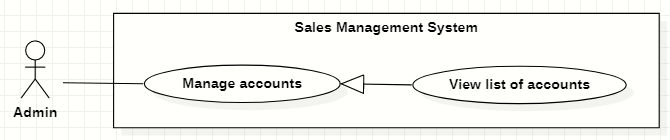
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-13 Create new schools** | | | | |
| Created By: | GiaNH | | Date Created: | | 15/02/2021 |
| Primary Actor: | Administrator | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to create new school. | | | | |
| Trigger: | The user clicks on the “Create” button in “Schools” screen. | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2. The user is on the “Schools” screen. | | | | |
| Post-conditions: | POST-1. The new account is added in the system. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user selects “Schools” on the navigation bar (sidebar). | | The system redirects the Schools screen. | |
| 2 | The user selects on “Create” button in screen. | | The system displays the “Create new schools” form with the following required information:  *School’s information:*   * “School name”: input text filed, required. * “School phone”: input text field. * “District”: selections, required. * “Address”: input text field. * “isActive”: boolean, default value is true. * “Description”: input text field. * “School type”: selections (Công lập, Ngoài công lập, Bán công), required. * “Educational level”: selections (Tiểu học, THCS, THPT), required. * “School scale”: selections (Lớn, Vừa, Nhỏ). * “School status”: selections (Leads, Customer, Ngưng hợp tác).   ----------  *Representative’s information:*   * “Full name”: input text field, required. * “Email”: input text field. * “Phone”: input text field, required. * “Gender”: radio buttons, required. | |
| 3 | The user inputs into information fields. | | [Exception 1] | |
| 4 | The user selects on the “Save” button.  [Exception 3] | | The system shows the successful message: “Create Successfully”.  [Exception 2] | |
| 5 |  | | The system updates the table of list of schools in the “Schools” screen.  [Exception 2] | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | The user leaves the blank fields or inputs the invalid format content. | | The system shows the error message: “Please input the valid format content”. | |
| 2 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| 3 | The user selects “Cancel” button. | | The system closes the “Create School” view and returns the “Schools” screen. | |
| Priority: | High | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | * Phone must be in format “/(84|0[3|5|7|8|9])+([0-9]{8})\b/g”   *(Resources:* [*https://www.regextester.com/106725*](https://www.regextester.com/106725)*)* | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### Search schools



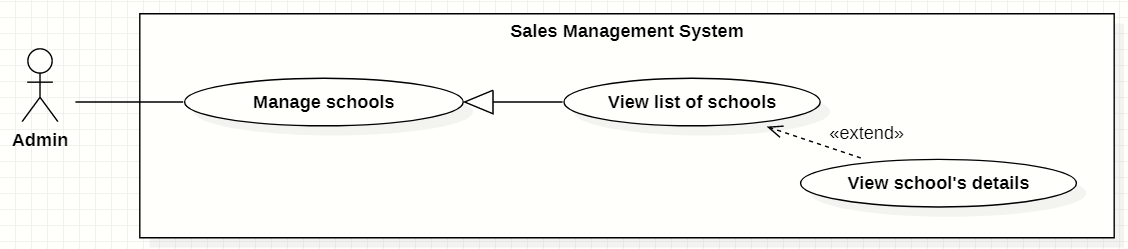
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-14 Search schools** | | | | | | |
| Created By: | GiaNH | | | Date Created: | | | 16/02/2021 |
| Primary Actor: | Administrator | | | Secondary Actors: | | | N/A |
| Description: | This use case allows the users to find the schools based on school’s name, district, school status, school educational level, school type, address,… | | | | | | |
| Trigger: | The user selects filters or inputs in search field on the “Schools” screen. | | | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2. The user is on “Schools” screen. | | | | | | |
| Post-conditions: | POST-1. The list of schools is shown. | | | | | | |
| Normal Flow: | **Step** | | **Actor Action** | | **System Response** | | |
| 1 | | The user inputs in the search field or selects filters to looking for schools. | | The system displays the list of accounts as the table.  Available filters on this screen:   * “Districts”: selections (24 districts in Ho Chi Minh City) * “School Statuses”: selections (Leads, Customer, Ngưng hợp tác). * “School Types”: selections (Công lập, Ngoài công lập, Bán công). * “School Levels”: selections (Tiểu học, THCS, THPT). * “School Scales”: selections (Lớn, Vừa, Nhỏ). * Search field: text field. | | |
| 2 | |  | | The list of Schools is shown as a table on the “Schools” screen.  Table will includes these fields:   * “School Name”: text. * “Address”: text. * “Principal”: text. * “Status”: Label. | | |
| Alternative Flows: | N/A | | | | | | |
| Exceptions: | **No** | **Cause** | | | | **System Response** | |
| 1 | The content of search field or selecting filter that the user inputted is not matched any data in database. | | | | The system displays “No records found” line. | |
| Priority: | Normal | | | | | | |
| Frequency of Use: | Usually | | | | | | |
| Business Rules: | The default values of all filters are “All”. | | | | | | |
| Other Information: | N/A | | | | | | |
| Assumptions: | N/A | | | | | | |

#### View list of schools



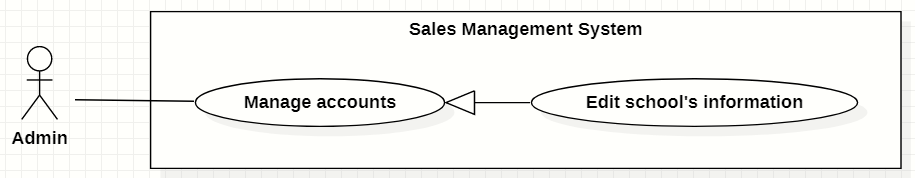
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-15 View list of schools** | | | | |
| Created By: | GiaNH | | Date Created: | | 15/02/2021 |
| Primary Actor: | Administrator | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to view list of schools. | | | | |
| Trigger: | The user selects “Schools” on the navigation bar (sidebar). | | | | |
| Preconditions: | PRE-1. The user has been log into system. | | | | |
| Post-conditions: | POST-1. The list of Schools is shown as a table on the “Schools” screen. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user selects “Schools” on the navigation bar (sidebar). | | The system redirects the “Schools” screen. | |
| 2 |  | | The list of Schools is shown as a table on the “Schools” screen.  Table will includes these fields:   * “School Name”: text. * “Address”: text. * “Principal”: text. * “Status”: Label. | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | The data is null or empty in database. | | The system shows “No records found” line on the table. | |
| Priority: | Medium | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | The default values of all filters are “All”. | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### View school’s details



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-16 View school’s details** | | | | |
| Created By: | GiaNH | | Date Created: | | 16/02/2021 |
| Primary Actor: | Administrator | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to view the detail information of a school. | | | | |
| Trigger: | The user clicks on the “View details” command on the menu options at the end of row in table list of schools. | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2. The user is on the “Schools” screen. | | | | |
| Post-conditions: | POST-1. The detail information of school is shown as a form. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | On the “Schools” screen, at the end of each row in the table, the user clicks on action icon button, then selects “View detail” command on the menu options. | | The system displays “School’s details” view, the school’s information is displayed as a form.  *School’s information:*   * “School name”: input text filed, required. * “School phone”: input text field. * “District”: selections (24 districts in Ho Chi Minh City). * “Address”: input text field. * “isActive”: boolean, default value is true. * “Description”: input text field. * “School type”: selections (Công lập, Ngoài công lập, Bán công), required. * “Educational level”: selections (Tiểu học, THCS, THPT), required. * “School scale”: selections (Lớn, Vừa, Nhỏ). * “School status”: selections (Leads, Customer, Ngưng hợp tác).   ----------  *Representative’s information:*   * “Full name”: input text field. * “Email”: input text field. * “Phone”: input text field. * “Gender”: radio buttons (Male, Female). | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | N/A | | | | |
| Priority: | Medium | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | N/A | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### Edit school’s information



|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-17 Edit school’s information** | | | | | | |
| Created By: | GiaNH | | | Date Created: | | | 16/02/2021 |
| Primary Actor: | Administrator | | | Secondary Actors: | | | N/A |
| Description: | This use case allows the users to edit the school’s information. | | | | | | |
| Trigger: | The user edits on fields in “School’s details” screen then clicks “Save” button. | | | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2. The user is on the “School’s details” screen. | | | | | | |
| Post-conditions: | POST-1. The School information is edited in database. | | | | | | |
| Normal Flow: | **Step** | | **Actor Action** | | | **System Response** | |
| 1 | |  | | | The system displays the “School’s detail” screen as a form with the following required information:  *School’s information:*   * “School name”: input text filed, required. * “School phone”: input text field. * “District”: selections (24 districts in Ho Chi Minh City). * “Address”: input text field. * “isActive”: boolean, default value is true. * “Description”: input text field. * “School type”: selections (Công lập, Ngoài công lập, Bán công), required. * “Educational level”: selections (Tiểu học, THCS, THPT), required. * “School scale”: selections (Lớn, Vừa, Nhỏ). * “School status”: selections (Leads, Customer, Ngưng hợp tác).   ----------  *Representative’s information:*   * “Full name”: input text field. * “Email”: input text field. * “Phone”: input text field. * “Gender”: radio buttons (Male, Female). | |
| 2 | | The user edits on fields in this form.  [Exception 1] | | |  | |
| 3 | | The user clicks “Save” button when finish. | | | The system has successful edit and displays the successful message: “Updated Successfully”.  [Exception 2] | |
| Alternative Flows: | N/A | | | | | | |
| Exceptions: | **No** | **Cause** | | | **System Response** | | |
| 1 | The user inputs invalid format content into fields. | | | The system informs the users that the content of fields is invalid format. | | |
| 2 | The user’s device does not connect to the Internet or the server has internal error. | | | The system shows the error message: “Connection Failed or Internal Error”. | | |
| Priority: | Medium | | | | | | |
| Frequency of Use: | Usually | | | | | | |
| Business Rules: | * Phone must be in format “/(84|0[3|5|7|8|9])+([0-9]{8})\b/g”   *(Resources:* [*https://www.regextester.com/106725*](https://www.regextester.com/106725)*)* | | | | | | |
| Other Information: | N/A | | | | | | |

### <<Salesman>> Overview Use Case

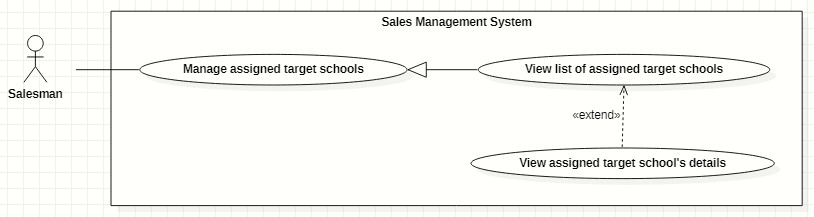
#### Search target schools

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-18 Search target schools** | | | | |
| Created By: | GiaNH | | Date Created: | | 07/02/2021 |
| Primary Actor: | Salesman, Sales Supervisor, Sales Manager | | Secondary Actors: | | N/A |
| Description: | This use case allows the users to find the target schools based on school’s name, district, school status, school educational level, school type, address,… | | | | |
| Trigger: | The user selects filters or inputs in search field on the “Target Schools” screen. | | | | |
| Preconditions: | PRE-1. The user has been log into the system.  PRE-2. The user is on the “Target Schools” screen. | | | | |
| Post-conditions: | POST-1. The list of target schools is shown as a table. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user inputs in the search field or selects filters to looking for schools. | | The system displays the list of accounts as the table.  Available filters on this screen:   * “PICs”: selections (all salesmen in the Sales Department) * “Purposes”: selections (   - Leads: Sales mới, Theo dõi;  - Customer: Chăm sóc, Tái ký hợp đồng, Ký mới hợp đồng;  - Ngưng hợp tác  )   * “Districts”: selections (24 districts in Ho Chi Minh City) * “School Years”: selections (2020-2021, 2019-2020, 2018-2019,…). * “School Types”: selections (Công lập, Ngoài công lập, Bán công). * “School Levels”: selections (Tiểu học, THCS, THPT). * “School Scales”: selections (Lớn, Vừa, Nhỏ). * Search field: text field. | |
| 2 |  | | The system displays the list of target schools as a table.   * “School Name”: text. * “Principal”: text. * “PIC”: image (avatar) and text. * “School Year”: text. * “Purpose”: Label.   [Exception 1]  [Exception 2] | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | The content of search field or selecting filter that The user inputted do not match any stored data in the system. | | The system displays “No records found” line on the table. | |
| 2 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| Priority: | High | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | The default values of all filters are “All”. | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### View list of target schools

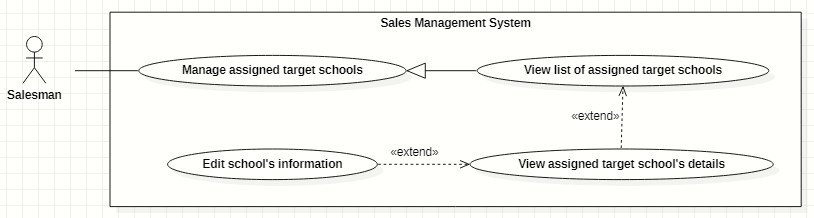
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-19 View list of target schools** | | | | |
| Created By: | GiaNH | | Date Created: | | 07/02/2021 |
| Primary Actor: | Salesman, Sales Supervisor, Sales Manager | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to view list of his assigned target schools. | | | | |
| Trigger: | The user selects “Target Schools” on the navigation bar (sidebar). | | | | |
| Preconditions: | PRE-1. The user has been log into system. | | | | |
| Post-conditions: | POST-1. The list of target schools is shown as a table. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user selects “Target Schools” on the navigation bar (sidebar). | | The system redirects to the “Target schools” screen and shows the target schools table with following columns:   * “School Name”: text. * “Principal”: text. * “PIC”: image (avatar) and text. * “School Year”: text. * “Purpose”: label.   Available filters on this screen:   * “PICs”: selections (all salesmen in the Sales Department) * “Purposes”: selections (   - Leads: Sales mới, Theo dõi;  - Customer: Chăm sóc, Tái ký hợp đồng, Ký mới hợp đồng;  - Ngưng hợp tác  )   * “Districts”: selections (24 districts in Ho Chi Minh City) * “School Years”: selections (2020-2021, 2019-2020, 2018-2019,…). * “School Types”: selections (Công lập, Ngoài công lập, Bán công). * “School Levels”: selections (Tiểu học, THCS, THPT). * “School Scales”: selections (Lớn, Vừa, Nhỏ). * Search field: text field.   [Exception 1]  [Exception 2] | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | No stored target schools data in the system. | | The system displays “No records found” line on the table. | |
| 2 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| Priority: | High | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | The default values of all filters are “All”. | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### View target school’s details



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-20 View target school’s details** | | | | |
| Created By: | GiaNH | | Date Created: | | 07/02/2021 |
| Primary Actor: | Salesman, Sales Supervisor, Sales Manager | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to view the detail information of the target school. | | | | |
| Trigger: | The user clicks on the “View details” command on the menu options at the end of row in table list of target schools. | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2. The user is at the “Target Schools” screen. | | | | |
| Post-conditions: | POST-1. The detail information of target school is shown as a form. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | On the “Target Schools” screen, at the end of each row in the table, the user clicks on action icon button, then selects “View detail” command on the menu options. | | The system shows “Target School’s Details” view, the target school’s information is displayed as a form with following information:  *School’s information:*   * “School name”: input text filed, required. * “School phone”: input text field. * “District”: selections (24 districts in Ho Chi Minh City). * “Address”: input text field. * “isActive”: boolean, default value is true. * “Description”: input text field. * “School type”: selections (Công lập, Ngoài công lập, Bán công), required. * “Educational level”: selections (Tiểu học, THCS, THPT), required. * “School scale”: selections (Lớn, Vừa, Nhỏ). * “School status”: selections (Leads, Customer, Ngưng hợp tác).   ----------  *Representative’s information:*   * “Full name”: input text field. * “Email”: input text field. * “Phone”: input text field. * “Gender”: radio buttons (Male, Female).   ----------  *Assign information:*   * “PIC”: text. * “PIC’s phone”: text. * “PIC’s email”: text. * “Purpose”: text.   [Exception 1] | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| Priority: | Medium | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | N/A | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### Edit school’s information

******

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-21 Edit school’s information** | | | | |
| Created By: | HaPTN | | Date Created: | | 12/04/2021 |
| Primary Actor: | Salesman, Sales Supervisor, Sales Manager | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to view the detail information of the target school. | | | | |
| Trigger: | The user edits on fields in “Target School’s details” screen then clicks “Save” button. | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2. The user is on the tab “General Info” in “Target School’s details” screen. | | | | |
| Post-conditions: | POST-1. The detail information of target school is edited in database. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
|  | 1 | The user edits on fields in “Target School’s details” screen then clicks “Save” button. | | The system displays the “Target School’s detail” screen, on the tab “General Info”, there is a form with the following information:  *School’s information:*   * “School name”: input text filed, read only. * “School phone”: input text field, read only. * “District”: selections (24 districts in Ho Chi Minh City), read only. * “Address”: input text field, read only. * “isActive”: boolean, read only. * “Description”: input text field, read only. * “School type”: selections (Công lập, Ngoài công lập, Bán công), read only. * “Educational level”: selections (Tiểu học, THCS, THPT), read only. * “School scale”: selections (Lớn, Vừa, Nhỏ), read only. * “School status”: selections (Leads, Customer, Ngưng hợp tác), read only.   ----------  *Representative’s information:*   * “Full name”: input text field. * “Email”: input text field. * “Phone”: input text field. * “Gender”: radio buttons (Male, Female). | |
|  | 2 | The user edits on fields in this form.  [Exception 1] | |  | |
|  | 3 | The user clicks “Save” button when finish. | | The system has successful edit and displays the successful message: “Updated Successfully”.  [Exception 2] | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | The user inputs invalid format content into fields. | | The system informs the users that the content of fields is invalid format. | |
| 2 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| Priority: | Medium | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | * The user can only edit fields related to the school’s representative: full name, gender, email, phone. * Phone must be in format “/(84|0[3|5|7|8|9])+([0-9]{8})\b/g”   *(Resources:* [*https://www.regextester.com/106725*](https://www.regextester.com/106725)*)*   * Any school’s data, the user needs to inform to the Admin and only Admin have right to edit school’s information. | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### Create contracts

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-22 Create contracts** | | | | |
| Created By: | HaPTN | | Date Created: | | 12/04/2021 |
| Primary Actor: | Salesman, Sales Supervisor, Sales Manager | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to create contracts of his/her assigned target schools. | | | | |
| Trigger: | The user selects the “Create” button on “Contracts” screen. | | | | |
| Preconditions: | PRE-1. The user has been log into the system.  PRE-2. The user is on the tab “Contracts” in “Target School’s details” screen. | | | | |
| Post-conditions: | POST-1. A new contract is created in the system. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user selects the “Create” button on “Contracts” screen. | | The system shows “Create Contract” view that contains following information fields:   * “Target Schools”: text. * “Duration”: input text field, required. * “Service”: checkbox, required. * “Revenue Criteria”: checkbox, required. * “Note”: input text field, required. | |
| 2 | The user inputs the information into the required fields. | |  | |
| 3 | The user selects “Save” button to create a new contract.  [Exception 3] | | The system shown the successful message: “Create Successfully”.  [Exception 1]  [Exception 2] | |
| 4 |  | | The system updates the table of list of contracts in the “Contracts” screen.  [Exception 2] | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | The user leaves the blank fields or inputs invalid format content. | | The system shows the error message: “Please input the valid format content”. | |
| 2 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| 3 | The user selects “Cancel” button. | | The system closes the “Create Contract” view and returns the Contracts screen. | |
| Priority: | High | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: |  | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### View my contracts

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-23 View my contracts** | | | | |
| Created By: | HaPTN | | Date Created: | | 12/04/2021 |
| Primary Actor: | Salesman, Sales Supervisor, Sales Manager | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to view the list of contract’s of his/her assigned target schools. | | | | |
| Trigger: | The user selects tab “Contracts” in “Target School’s details” screen. | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2. The user is on the tab “Contracts” in “Target School’s details” screen. | | | | |
| Post-conditions: | POST-1. Contract’s list is displayed as the table in “Contracts” screen. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user selects tab “Contracts” in “Target School’s details” screen. | | The system displays “Contracts” screen, contract’s list is displayed as the table.   * “Target Schools”: text. * “Duration”: text. * “Service”: text. * “Revenue Criteria”: text. * “Note”: text. | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | No stored contracts data in the system. | | The system displays “No records found” line on the table. | |
|  | 2 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| Priority: | High | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | N/A | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### Create reports



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-24 Create reports** | | | | |
| Created By: | GiaNH | | Date Created: | | 08/02/2021 |
| Primary Actor: | Salesman, Sales Supervisor, Sales Manager | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to create daily reports. | | | | |
| Trigger: | The user selects the “Create” command on “Reports” screen. | | | | |
| Preconditions: | PRE-1. The user has been log into the system.  PRE-2. The user is on the “Reports” screen. | | | | |
| Post-conditions: | POST-1. A new report is created in the system. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user selects the “Create” button on “Reports” screen.  [Alternative] | | The system shows “Create Report” view that contains following information fields:   * “Target Schools”: input text field, required. * “Date”: date, read only, default is current date. * “Result”: input text field, required. * “Description”: input text field, required. * “Positivity”: input text field. * “Difficulty”: input text field. * “Future plan”: input text field. | |
| 2 | The user inputs the information into the required fields.  [Exception 1] | |  | |
| 3 | The user clicks on “Save” button to create a new daily report.  [Alternative 1] | | The system shows the successful message: “Created successfully”.  [Exception 2]  [Exception 3] | |
| 4 |  | | The system updates the table of list of reports in the “Reports” screen.  [Exception 2] | |
| Alternative Flows: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user clicks on “+” button to continue create daily report for another school. | | The system resets the “Create Report” form and shows the previous inputted reports in “Preview” table.  Table contains these columns:   * “School Name”: text. * “Result”: text. * “Description”: text.   *Continue to step 2 of Normal Flow*. | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | The user leaves the blank fields or inputs invalid format content. | | The system shows the error message: “Please input the valid format content”. | |
| 2 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| 3 | The user selects “Cancel” button. | | The system closes the “Create Report” view and returns the Reports screen. | |
| Priority: | High | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: |  | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### Search reports



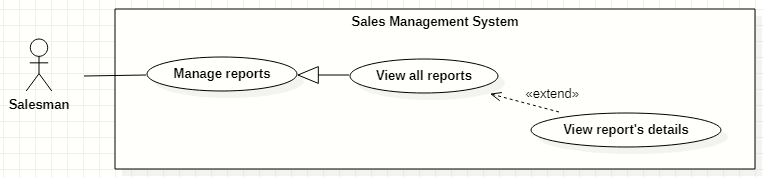
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-25 Search reports** | | | | |
| Created By: | GiaNH | | Date Created: | | 09/02/2021 |
| Primary Actor: | Salesman, Sales Supervisor, Sales Manager | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to find the reports based on target school’s name, district, PIC’s name, from date to date,…. | | | | |
| Trigger: | The user selects on the searching command on “Reports” screen. | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2. The user is on the “Reports” screen. | | | | |
| Post-conditions: | POST-1. The list of reports is displayed as the table. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user inputs the search field or select searching filters on toolbar to find reports. | | The system displays the list of reports as the table.  Available filters on this screen:  “PICs”: selections (all salesmen in the Sales Department)   * “Districts”: selections (24 districts in Ho Chi Minh City) * “School Years”: selections (2020-2021, 2019-2020, 2018-2019,…). * “From date” – “To date”: date range picker, disable future days. * “School Statuses”: selections (Leads, Customer, Ngưng hợp tác). * “Purposes”: selections (   - Leads: Sales mới, Theo dõi;  - Customer: Chăm sóc, Tái ký hợp đồng, Ký mới hợp đồng;  - Ngưng hợp tác  )   * Search field: text field. | |
|  | 2 |  | | The system displays the list of target schools as a table.   * “Date”: date. * “School Name”: text. * “PIC”: image (avatar) and text. * “Purpose”: Label. * “Result”: text. * “Description”: text.   [Exception 1]  [Exception 2] | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | The content of search field or selecting filter that The user inputted is not matched any stored data in the system. | | The system displays “No records found” line on the table. | |
| 2 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| Priority: | High | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | The default values of all filters are “All”. | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### View all reports



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-26 View all reports** | | | | |
| Created By: | GiaNH | | Date Created: | | 08/02/2021 |
| Primary Actor: | Salesman, Sales Supervisor, Sales Manager | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to view list of all daily report’s of all members in Major Sales Department. | | | | |
| Trigger: | The user selects “Reports” on the navigation bar (sidebar). | | | | |
| Preconditions: | PRE-1. The user has been log into system. | | | | |
| Post-conditions: | POST-1. Report’s list is displayed as the table in “Reports” screen. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user selects “Reports” on the navigation bar (sidebar). | | The system displays “Reports” screen, report’s list is displayed as the table.   * “Date”: date. * “School Name”: text. * “PIC”: image (avatar) and text. * “Purpose”: label. * “Result”: text. * “Description”: text.   Available filters on this screen:  “PICs”: selections (all salesmen in the Sales Department)   * “Districts”: selections (24 districts in Ho Chi Minh City) * “School Years”: selections (2020-2021, 2019-2020, 2018-2019,…). * “From date” – “To date”: date range picker, disable future days. * “School Statuses”: selections (Leads, Customer, Ngưng hợp tác). * “Purposes”: selections (   - Leads: Sales mới, Theo dõi;  - Customer: Chăm sóc, Tái ký hợp đồng, Ký mới hợp đồng;  - Ngưng hợp tác  )   * Search field: text field.   [Exception 1]  [Exception 2] | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | No stored reports data in the system. | | The system displays “No records found” line on the table. | |
|  | 2 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| Priority: | High | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | N/A | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### View report’s details

****

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-27 View report’s details** | | | | |
| Created By: | HaPTN | | Date Created: | | 12/04/2021 |
| Primary Actor: | Administrator | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to view the detail information of a report. | | | | |
| Trigger: | The user clicks on the “View details” command on the menu options at the end of row in table list of reports. | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2. The user is on the “Reports” screen. | | | | |
| Post-conditions: | POST-1. The detail information of report is shown as a form. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | On the “Reports” screen, at the end of each row in the table, the user clicks on action icon button, then selects “View detail” command on the menu options. | | The system displays “Report’s details” view, the report’s information is displayed as a form.   * “School Year”: text. * “Date”: text. * “School Name”: text. * “PIC”: image (avatar) and text. * “Purpose”: text. * “Result”: input text field, required. * “Description”: input text field, required. * “Positivity”: input text field. * “Difficulty”: input text field. * “Future plan”: input text field. * “Supervisor comment”: input text field, read only.   [Exception 1] | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| Priority: | Medium | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | N/A | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### Edit reports

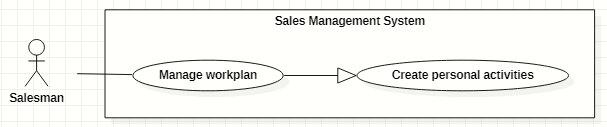
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-28 Edit reports** | | | | |
| Created By: | HaPTN | | Date Created: | | 12/04/2021 |
| Primary Actor: | Salesman, Sales Supervisor, Sales Manager | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to edit a daily report. | | | | |
| Trigger: | The user edits on fields in “Report’s details” screen then clicks “Save” button. | | | | |
| Preconditions: | PRE-1. The user has been log into the system.  PRE-2. The user is on the “Report’s details” screen. | | | | |
| Post-conditions: | POST-1. A new report is updated in the database. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user edits on fields in “Report’s details” screen then clicks “Save” button. | | The system displays “Report’s details” view, the report’s information is displayed as a form.   * “School Year”: text. * “Date”: text. * “School Name”: text. * “PIC”: image (avatar) and text. * “Purpose”: text. * “Result”: input text field, required. * “Description”: input text field, required. * “Positivity”: input text field. * “Difficulty”: input text field. * “Future plan”: input text field. * “Supervisor comment”: input text field, read only.   [Exception 1] | |
| 2 | The user edits on fields in this form.  [Exception 1] | |  | |
| 3 | The user clicks “Save” button when finish. | | The system has successful edit and displays the successful message: “Updated Successfully”.  [Exception 2] | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | The user inputs invalid format content into fields. | | The system informs the users that the content of fields is invalid format. | |
|  | 2 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| Priority: | High | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | The user can only edit reports which have not been commented by the supervisor or the manager. | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### Remove reports

****

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-29 Remove reports** | | | | |
| Created By: | HaPTN | | Date Created: | | 12/04/2021 |
| Primary Actor: | Salesman, Sales Supervisor, Sales Manager | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to remove a daily report. | | | | |
| Trigger: | The user clicks on the “Remove” command on the menu options at the end of row in table list of reports. | | | | |
| Preconditions: | PRE-1. The user has been log into the system.  PRE-2. The user is on the “Reports” screen. | | | | |
| Post-conditions: | POST-1. A daily report is removed in the database. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
|  | 1 | The user clicks on the “Remove” command on the menu options at the end of row in table list of reports. | | The system shows a “Confirm Remove” pop-up to alert.   * “Cancel”: button. * “Remove”: button. | |
|  | 2 | The user clicks on “Remove” button.  [Exception 1] | | The system shows a successful message: “Removed successfully”, and closes the confirm pop-up. | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | The user selects “Cancel” button. | | The system closes the “Confirm Remove” pop-up and returns the Reports screen. | |
| 2 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| Priority: | High | | | | |
| Frequency of Use: |  | | | | |
| Business Rules: | The user can only remove reports which have not been commented by the supervisor or the manager. | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### Create personal activities

******

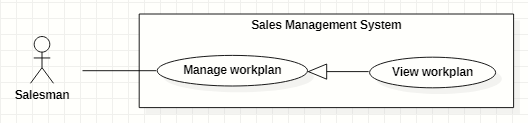
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-30 Create personal activities** | | | | |
| Created By: | GiaNH, HaPTN | | Date Created: | | 12/04/2021 |
| Primary Actor: | Salesman, Sales Supervisor, Sales Manager | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to create one or multiple personal activities in his/her workplan. | | | | |
| Trigger: | The user wants to create one or multiple personal activities. | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2. The user is on the “Work-plan” screen. | | | | |
| Post-conditions: | POST-1. New personal activity/personal activities is/are added in work-plan schedule. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user clicks on “Create” button on the “Work-plan” screen. | | The system displays the “Create Activity” form dialog that contains the following format fields:   * “Title”: text, required. * “Description”: text. * “Remark”: text. * “Start time”: date time, required. * “End time”: date time, required. * “Status”: text, read only, default is “Not yet”. | |
|  | 2 | The user inputs into the information fields. | |  | |
|  | 3 | The user selects the “Save” button. | | The system shows the successful message: “Done”.  [Exception 1]  [Exception 2]  [Exception 3] | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Actor Action** | | **System Response** | |
| 1 | The user leaves the blank fields or inputs invalid format content into fields. | | The system shows the error message: “Please input the valid format content”. | |
|  | 2 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
|  | 3 | The user inputs the “Start time” or the “End time” within the time of another activity item. | | The system shows the error message: “This activity has the same time with another” | |
| Priority: | High. | | | | |
| Frequency of Use: | Usually. | | | | |
| Business Rules: | * “Start time” has the time later than “End time”. * At the time, only activity is existed. | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### Search other’s workplan



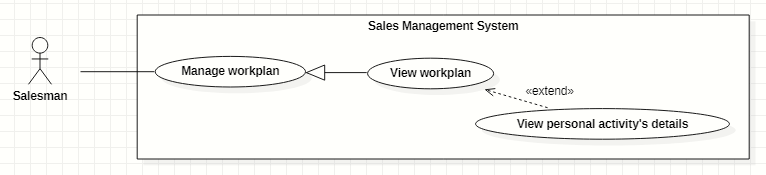
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-31 Search other’s workplan** | | | | |
| Created By: | GiaNH, HaPTN | | Date Created: | | 12/04/2021 |
| Primary Actor: | Salesman, Sales Supervisor, Sales Manager | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to search the work-plan of other users. | | | | |
| Trigger: | The user input in the search field on “Work-plan” screen. | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2. The user is on the “Work plan” screen. | | | | |
| Post-conditions: | POST-1. The work plan schedule is displayed on the screen. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user inputs into the search field. | | The system provides suggestions that presents Salesman to users as they enter their search query into the search box. | |
|  | 2 | The user selects a Salesman item in suggestions. | | The system displays “work-plan” screen. The Work-plan has the personal activity items that present as a schedule | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Actor Action** | | **System Response** | |
| 1 | The content of search field or selecting filter that the user inputted is not matched any stored data in the system. | | The system displays “No records found” line on the table. | |
|  | 2 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| Priority: | Normal. | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | * The Sales Manager can view the Salesman’s work-plan and other Sales Manager’s work-plan. * The Salesman can only view the work-plan of other Salesman. | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### View workplan



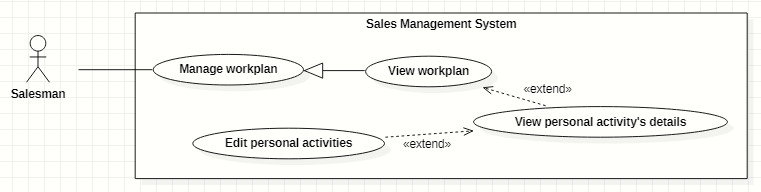
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-32 View workplan** | | | | |
| Created By: | GiaNH, HaPTN | | Date Created: | | 12/04/2021 |
| Primary Actor: | Salesman, Sales Supervisor, Sales Manager | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to view a whole workplan of himself/herself or other’s. | | | | |
| Trigger: | The user selects “Workplans” on the navigation bar (sidebar). | | | | |
| Preconditions: | PRE-1. The user has been log into system. | | | | |
| Post-conditions: | POST-1. The personal activities are displayed in workplan schedule. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user selects “Work-plan” command on the sidebar. | | The system displays “work-plan” screen. The Work-plan has the personal activity items that present as a schedule  [Exception 1] | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| Priority: | Normal | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | * The system displays the “Work-plan” as a week’s schedule with the activity item is a cell in the schedule. * The activity items have different colors to distinguish based on status. | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### View personal activity’s details



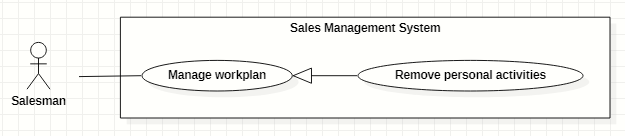
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-33 View personal activity’s details** | | | | | |
| Created By: | GiaNH, HaPTN | | | Date Created: | | 12/04/2021 |
| Primary Actor: | Salesman, Sales Supervisor, Sales Manager | | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to view the detail information of personal activity. | | | | | |
| Trigger: | The user clicks on the personal activity item on work-plan screen. | | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2. The user is on the “Work-plan” screen. | | | | | |
| Post-conditions: | POST-1. The detail information of personal activity is shown as a form. | | | | | |
| Normal Flow: | **Step** | **Actor Action** | | | **System Response** | |
| 1 | The user selects any item on work-plan schedule. | | | The system displays “Personal Activity’s Details” view, the personal activity’s information is displayed as a form.  [Exception 1] | |
| Alternative Flows: | N/A | | | | | |
| Exceptions: | **No** | | **Actor Action** | | **System Response** | |
| 1 | | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| Priority: | Normal | | | | | |
| Frequency of Use: | Usually | | | | | |
| Business Rules: | N/A | | | | | |
| Other Information: | N/A | | | | | |
| Assumptions: | N/A | | | | | |

#### Edit personal activities



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-34 Edit personal activities** | | | | | |
| Created By: | GiaNH, HaPTN | | Date Created: | | | 12/04/2021 |
| Primary Actor: | Salesman, Sales Supervisor, Sales Manager | | Secondary Actors: | | | N/A |
| Description: | This use case allows the user to edit personal activities in his/her workplan. | | | | | |
| Trigger: | The user wants to edit one or multiple personal activities. | | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2.1. The user is on the “Personal Activity’s Details” view.  PRE-2.2. The user is on the “Workplan” screen. | | | | | |
| Post-conditions: | POST-1. The personal activity/personal activities is/are updated in the database. | | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | | |
| 1 | The user selects the “Edit” command. | | The system changes the text information to the inputting text fields with the following format content:   * “Title”: text, required. * “Description”: text. * “Remark”: text. * “Start time”: date time, required. * “End time”: date time, required. * “Status”: contains selections. | | |
|  | 2 | The user inputs the new content in the fields. | |  | | |
|  | 3 | The user clicks “Save” button.  [Exception 1] | | The system responses the successful editing message: “Done”.  [Exception 2]  [Exception 3] | | |
| Alternative Flows: | N/A | | | | | |
| Exceptions: | **No** | **Actor Action** | | | **System Response** | |
| 1 | The user selects “No” button to cancel the editing. | | | The system returns the “Personal Activity’s Details” screen. | |
| 2 | The user’s device does not connect to the Internet or the server has internal error. | | | The system shows the error message: “Connection Failed or Internal Error”. | |
| 3 | The user inputs the “Start time” or the “End time” within the time of another activity item. | | | The system shows the error message: “This activity has the same time with another” | |
| Priority: | High. | | | | | |
| Frequency of Use: | Usually | | | | | |
| Business Rules: | * “Start time” has the time later than “End time”. * At the time, only activity is existed. | | | | | |
| Other Information: | N/A | | | | | |
| Assumptions: | N/A | | | | | |

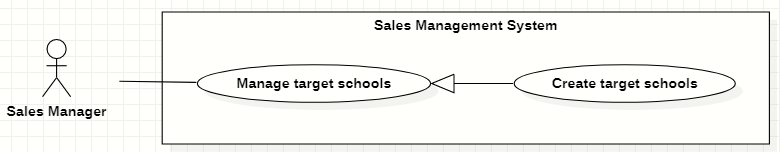
#### Remove personal activities



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-35 Remove personal activities** | | | | |
| Created By: | GiaNH, HaPTN | | Date Created: | | 12/04/2021 |
| Primary Actor: | Salesman, Sales Supervisor, Sales Manager | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to remove personal activities in his/her workplan. | | | | |
| Trigger: | The user wants to remove one or multiple personal activities. | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2.1. The user is on the “Personal Activity’s Details” view.  PRE-2.2. The user is on the “Workplan” screen. | | | | |
| Post-conditions: | POST-1. The personal activity/personal activities is/are removed in the database. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user selects “Remove” command on the “Personal Activity’s Details” view. | | The system shows the removing confirmation message. | |
|  | 2 | The user selects “Yes” button to confirm the removing.  [Exception 1] | | The system shows the successful message: “Done”.  [Exception 2] | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Actor Action** | | **System Response** | |
| 1 | The user selects “No” button to cancel the removing. | | The system returns the “Personal Activity’s Details” screen. | |
|  | 2 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| Priority: | Normal. | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | N/A | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

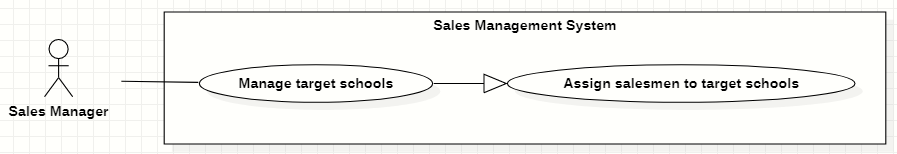
### <<Sales Supervisor>> Overview Use Case

#### Create target schools



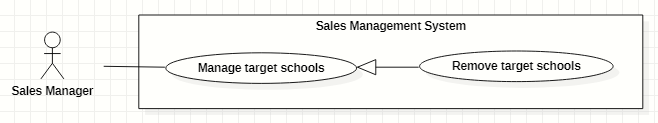
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-36 Create target schools** | | | | |
| Created By: | GiaNH | | Date Created: | | 15/02/2021 |
| Primary Actor: | Sales Manager | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to create new target schools from the list of all schools. | | | | |
| Trigger: | The user clicks on the “Create” button on “Target Schools” screen. | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2. The user is on the “Target Schools” screen. | | | | |
| Post-conditions: | POST-1. The new target schools are created in the system. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user clicks on the “Create” button on “Target Schools” screen. | | The system displays the “Create Target School” form with following required information:   * “School year”: text, default is the current school year, read only. * A table of all schools which has the following columns: * “”: checkbox. * “School Name”: text. * “Principal”: text. * “School Status”: text. * “Purpose”: selections (   - Leads: Sales mới, Theo dõi;  - Customer: Chăm sóc, Tái ký hợp đồng, Ký mới hợp đồng;  - Ngưng hợp tác  ), value of purpose depends on value of “School Status”.   * “Note”: text. * Preview panel: div (to display school’s details). * “Cancel”: button. * “Save”: button. | |
| 2 | The user ticks the check boxes.  [Alternative 1] | |  | |
| 3 | The user selects “Save” button.  [Exception 2] | | The system shows the successful message: “Created successfully”.  [Exception 1] | |
| 4 |  | | The system updates the table of list of target schools in the “Target Schools” screen.  [Exception 1] | |
| Alternative Flows: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The users clicks on a table row. | | The systems display that school’s details in the preview panel.   * “School Name”: text. * “School level”: text. * “School type”: text. * “School scale”: text. * “School status”: text. * “Address”: text. * “School phone”: text. * “Representative name”: text. * “Representative phone”: text. * “Representative email”: text. * “Representative gender”: text. * “This school also be targeted in”: text, list of school year.   *Continue to step 2 of Normal Flow*. | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| 2 | The user selects “Cancel” button. | | The system closes the “Create Target Schools” form and returns to the “Target Schools” screen. | |
| Priority: | High | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | * Sales Supervisor/Sales Manager can only create the target schools at the current school year. | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### Assign salesmen to target schools

******

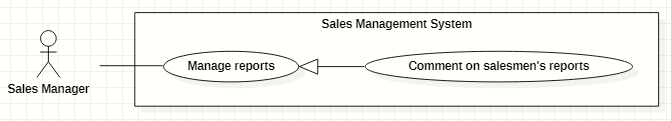
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-37 Assign salesmen to target schools** | | | | |
| Created By: | GiaNH | | Date Created: | | 11/02/2021 |
| Primary Actor: | Sales Manager | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to assign salesmen to one or multiple target schools. | | | | |
| Trigger: | The user clicks on the “Assign” button in “Target schools” screen. | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2. The user is on the “Target Schools” screen. | | | | |
| Post-conditions: | POST-1. The target schools are assigned to salesmen. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user ticks on the target school rows that he/she wants to assign, then clicks on “Assign” button on the “Target Schools” screen.  [Alternative 1] | | The system displays the “Assign Salesmen” form with the following information:  - “PICs”: input text field, required.  - Table of target schools, which contains columns:   * “School Name”: text. * “PIC”: text (get value from “PICs” input text field above. * “Purpose”: selections (   - Leads: Sales mới, Theo dõi;  - Customer: Chăm sóc, Tái ký hợp đồng, Ký mới hợp đồng;  - Ngưng hợp tác  ).   * “Note”: input text field / editable table cell.   - “Cancel”: button.  - “Save”: button. | |
| 2 | The user input/chooses a PIC in the input text field. | |  | |
| 3 | The user selects “Save” button.  [Exception 3] | | The system shows the successful message: “Done”.  [Exception 1]  [Exception 2] | |
| Alternative Flows: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user selects “Assign” commands on the selected target school row. | | The system displays the “Assign” form with the following information:  -“Salesman”: input text field, required.  -“Start date”: date picker, required, default is current date.  -“End date”: date picker.  - “Note”: input text field.  *Continue step 2 of Normal Flow* | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | The target school has been already assigned in the same school year. | | The system shows the error message: “The target school has been already assigned in the same school year”. | |
| 2 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| 3 | The user selects “Cancel” button. | | The system closes the “Assign” form and returns the “Target School” screen. | |
| Priority: | Usually | | | | |
| Frequency of Use: | N/A | | | | |
| Business Rules: | N/A | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### Remove target schools



|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-38 Remove target schools** | | | | |
| Created By: | GiaNH | | Date Created: | | 15/02/2021 |
| Primary Actor: | Sales Manager | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to remove the target schools | | | | |
| Trigger: | The user selects the “Remove” command on the target school row. | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2. The user is on the “All Target Schools” screen. | | | | |
| Post-conditions: | POST-1. The target schools are removed out of table. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
| 1 | The user selects “remove” command on the selected target school row. | | The system shows the confirmation dialog. | |
|  | 2 | The user selects “Yes” to confirm the removing.  [Exception 1] | | The system responses the successful removing message: “Done”.  [Exception 2 ] | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
| 1 | The user selects “Cancel” button. | | The system closes the confirmation dialog and returns the “All Target School” screen. | |
| 2 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| Priority: | Normal | | | | |
| Frequency of Use: | Usually | | | | |
| Business Rules: | N/A | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

#### Comment on Salesmen’s reports



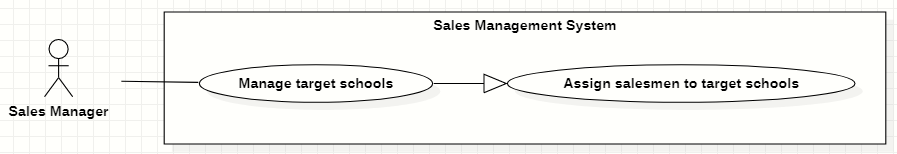
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-39 Comment on Salesmen’s reports** | | | | | | |
| Created By: | GiaNH | | | | Date Created: | | 12/02/2021 |
| Primary Actor: | Sales Manager | | | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to comment on Salesmen’s Reports. | | | | | | |
| Trigger: | The user selects the “Comment” command. | | | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2. The user is on the “Reports” screen. | | | | | | |
| Post-conditions: | POST-1. The comment is added on Salesmen’s reports. | | | | | | |
| Normal Flow: | **Step** | | **Actor Action** | | | **System Response** | |
| 1 | | The user selects “comment” command on the selected report row.  [Alternative 1]  [Alternative 2] | | | The system shows the “Add comment” form. With the following information:  -“Sales Manager’s comment”: input text filed. | |
| 2 | | The user inputs information into the text fields. | | |  | |
| 3 | | The user selects “Save” button. | | | The system shows successful message: “Done”.  [Exception 1]  [Exception 2]  [Exception 3] | |
| Alternative Flows: | **No** | **Step** | | **Actor Action** | | **System Response** | |
| 1 | 1 | | The user selects “comment” command on the commented report. | | The system notifies that this report has already commented and confirm to edit. | |
| 2 | | The user selects “Yes” button. | | The system shows the “Add comment” form. With the following information:  -“Sales Manager’s comment”: text.  *Continue to step 2 Normal Flow.* | |
| 2 | 1 | | On “Target School” screen. The user selects the action button on target school row. | | The system shows the options menu. | |
| 2 | | The user selects the “Report” command on the options menu. | | The system redirects the “Reports” screen.  *Continue to step 1 Normal Flow* | |
| Exceptions: | **No** | | **Cause** | | | **System Response** | |
| 1 | | The user selects the “Cancel” button without typing content. | | | The system closes the “Add comment” form and returns the “Reports” screen. | |
| 2 | | The user selects the “Cancel” button after typing content. | | | The system shows the confirmation message: “Discard unsaved changes?” | |
| 3 | | The user’s device does not connect to the Internet or the server has internal error. | | | The system shows the error message: “Connection Failed or Internal Error”. | |
| Priority: | High. | | | | | | |
| Frequency of Use: | Usually | | | | | | |
| Business Rules: | * Sales Manager is not allowed to comment on his report. * Sales Manager is allowed to comment on reports of other Sales Managers | | | | | | |
| Other Information: | N/A | | | | | | |
| Assumptions: | N/A | | | | | | |

### <<Sales Manager>> Overview Use Case

#### View all contracts

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-43 View all contracts** | | | | | |
| Created By: | HaPTN | | | Date Created: | | 12/04/2021 |
| Primary Actor: | Sales Manager | | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to view all contracts of all target schools. | | | | | |
| Trigger: | The user selects tab “Contracts” in “Target School’s details” screen. | | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2. The user is on the tab “Contracts” in “Target School’s details” screen. | | | | | |
| Post-conditions: | POST-1. Contract’s list is displayed as the table in “Contracts” screen. | | | | | |
| Normal Flow: | **Step** | | **Actor Action** | | **System Response** | |
| 1 | | The user selects tab “Contracts” in “Target School’s details” screen. | | The system displays “Contracts” screen, contract’s list is displayed as the table.   * “Target Schools”: text. * “Duration”: text. * “Service”: text. * “Revenue Criteria”: text. * “Note”: text.   [Exception 1] | |
| Alternative Flows: | N/A | | | | | |
| Exceptions: | **No** | **Cause** | | | **System Response** | |
| 1 | No stored contracts data in the system. | | | The system displays “No records found” line on the table. | |
| 2 | The user’s device does not connect to the Internet or the server has internal error. | | | The system shows the error message: “Connection Failed or Internal Error”. | |
| Priority: | High. | | | | | |
| Frequency of Use: | Usually | | | | | |
| Business Rules: |  | | | | | |
| Other Information: | N/A | | | | | |
| Assumptions: | N/A | | | | | |
| Assumptions: | N/A | | | | | |

#### View contract’s details

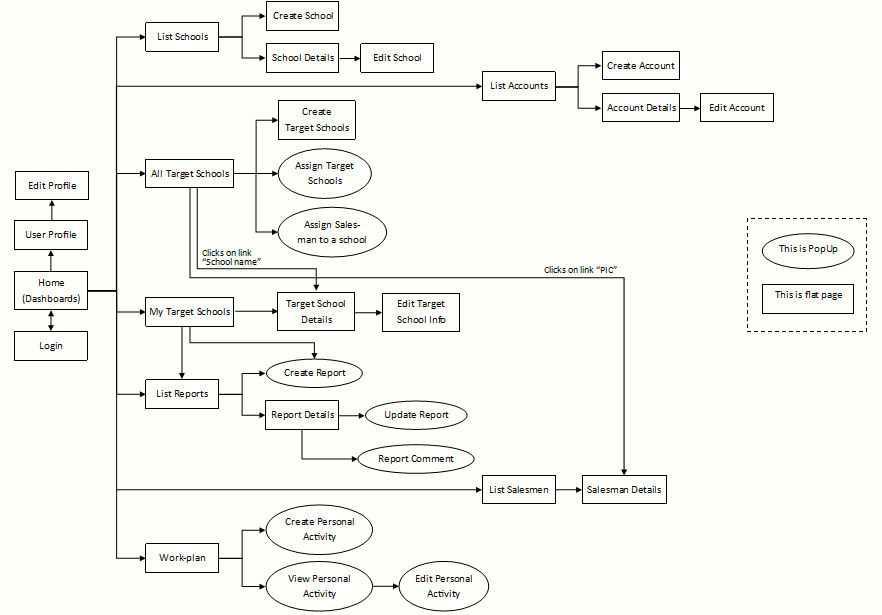
******

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and Name: | **UC-44 View contract’s details** | | | | |
| Created By: | HaPTN | | Date Created: | | 12/04/2021 |
| Primary Actor: | Sales Manager | | Secondary Actors: | | N/A |
| Description: | This use case allows the user to view a contract’s details. | | | | |
| Trigger: | The user selects a specific contract in the list of contracts in tab “Contracts” on “Target School’s details” screen. | | | | |
| Preconditions: | PRE-1. The user has been log into system.  PRE-2. The user is on the tab “Contracts” in “Target School’s details” screen. | | | | |
| Post-conditions: | POST-1. Contract’s details is displayed as the dialog in “Contracts” screen. | | | | |
| Normal Flow: | **Step** | **Actor Action** | | **System Response** | |
|  | 1 | The user clicks on a specific contract in the list of contracts in tab “Contracts” on “Target School’s details” screen. | | The system displays the “Contract’s details” dialog with the following information:   * “Target Schools”: text. * “Duration”: text. * “Service”: text. * “Revenue Criteria”: text. * “Note”: text.   [Exception 1] | |
| Alternative Flows: | N/A | | | | |
| Exceptions: | **No** | **Cause** | | **System Response** | |
|  | 1 | The user’s device does not connect to the Internet or the server has internal error. | | The system shows the error message: “Connection Failed or Internal Error”. | |
| Priority: | Usually | | | | |
| Frequency of Use: | N/A | | | | |
| Business Rules: | N/A | | | | |
| Other Information: | N/A | | | | |
| Assumptions: | N/A | | | | |

## 3. Functional Requirements

### 3.1 System Functional Overview

#### Screen Flow



#### Screen Details

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Feature** | **Screen** | **Description** |
| 1 | Login | Login |  |
| 2 | View Dashboards | Home (Dashboards) |  |
| 3 | View Profile | User Profile |  |
| 4 | Edit Profile | Edit Profile |  |
| 5 | * View list of schools * Search schools | List Schools |  |
| 6 | Create new schools | Create School |  |
| 7 | View school’s details | School Details |  |
| 8 | Edit school’s information | Edit School |  |
| 9 | * View list of accounts * Search accounts | List Accounts |  |
| 10 | Create new accounts | Create Account |  |
| 11 | View account’s details | Account Details |  |
| 12 | Edit account’s information | Edit Account |  |
| 13 | View list of all target schools | All Target Schools |  |
| 14 | Create target schools | Create Target Schools |  |
| 15 | Assign Salesmen to target schools | Assign Target Schools |  |
| 16 | Assign Salesmen to target schools | Assign Salesman to a school |  |
| 17 | * View list of assigned target schools * Search assigned target schools | My Target Schools |  |
| 18 | View target school’s details | Target Schools Details |  |
| 19 | Edit school’s information | Edit Target School Info |  |
| 20 | * View all reports * Search reports | List Reports |  |
| 21 | Create reports | Create Report |  |
| 22 | View report’s details | Report Details |  |
| 23 | Update reports | Update Report |  |
| 24 | Comment on Salesmen’s reports | Report Comment |  |
| 25 | * View list of Salesmen * Search Salesmen | List Salesmen |  |
| 26 | View Salesman’s details | Salesman Details |  |
| 27 | View Workplan | Workplan |  |
| 28 | Create Personal Activities | Create Personal Activities |  |
| 29 | View Personal Activity’s details | View Personal Activity |  |
| 30 | Edit Personal Activities | Edit Personal Activity |  |

#### Screen Authorization

*[Provide the system roles authorization to the system features (down to screens, and event to the screen activities if applicable) in the table form as below – replace Role1, Role2,… with the specific system user role names]*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Screen** | **Salesman** | **Sales Supervisor** | **Sales Manager** | **Admin** |
| <<Screen Name1>> | X |  |  | X |
| <<Screen Activity>> |  |  |  | X |
| <<Screen Name2>> | X |  |  | X |
| Query All Data | X |  |  |  |
| Query Own Data |  |  |  | X |
| Query Managed Data |  |  |  | X |
| Add New Data |  |  |  | X |
| Update All Data |  |  |  |  |
| Update Own Data |  |  |  |  |
| Update Managed Data |  |  |  |  |
| Delete Data |  |  |  |  |
| … |  |  |  |  |

In which:

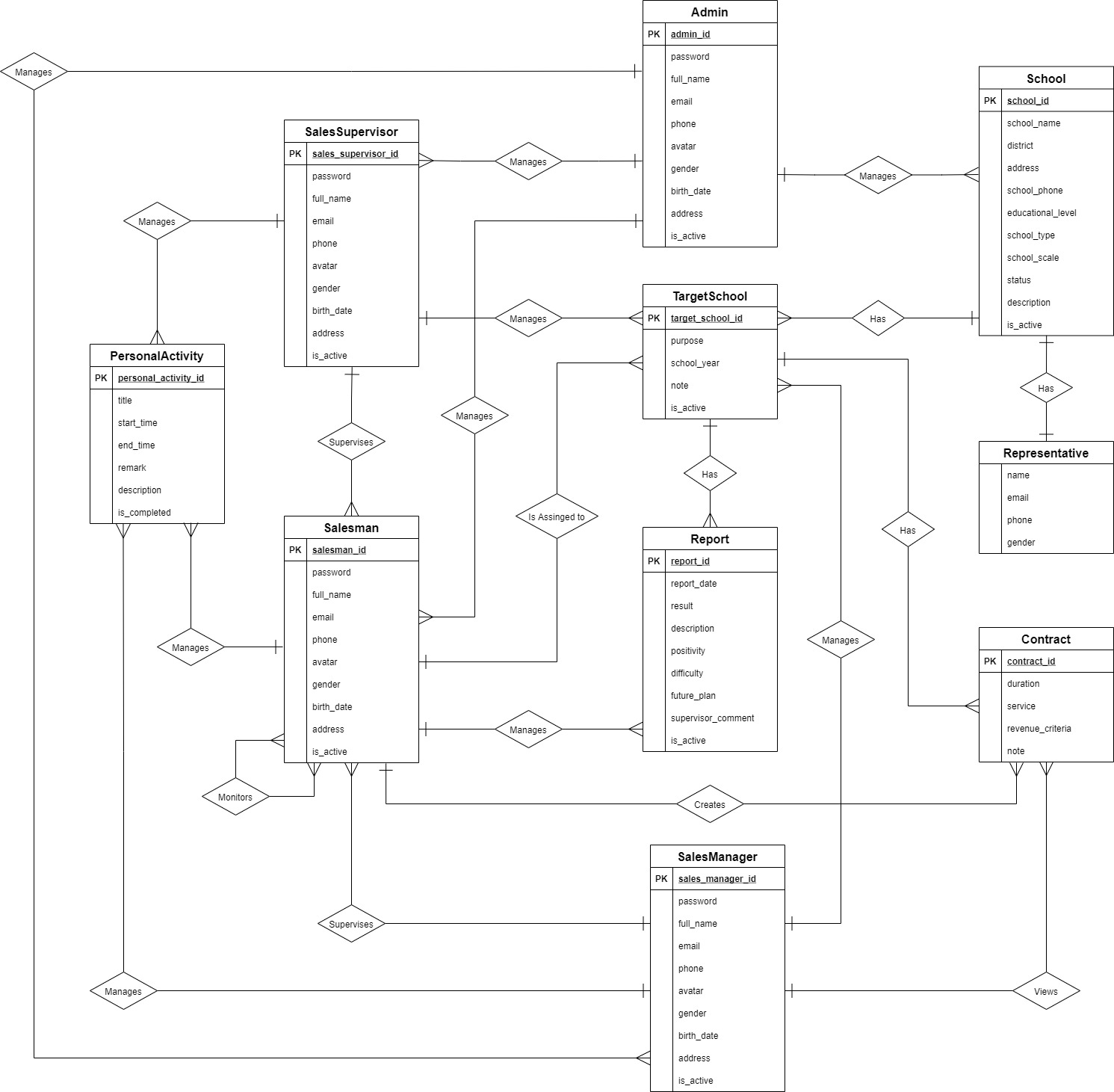
* Salesman:
* Sales Supervisor:
* Sales Manager:
* Administrator:

#### Non-Screen Functions

*[Provide the descriptions for the non-screen system functions, i.e batch/cron job, service, API, etc.]*

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Feature** | **System Function** | **Description** |
| 1 | <<Feature Name>> | <<Function Name1>> | <<Function Name1 Description>> |
| 2 | … |  |  |

#### Entity Relationship Diagram



**Entities List**

|  |  |  |
| --- | --- | --- |
| **#** | **Entity** | **Description** |
| 1 | Admin | The Admin is an entity presenting a person who manages accounts and whole schools data of sale department.  Containing information of admin (id, password, full name, phone, email, avatar, address, birthday, gender, isActive). |
| 2 | Salesman | The Salesman is an entity presenting a person who goes to work with the assigned target schools.  Containing information of Salesman (id, password, full name, phone, email, avatar, address, birthday, gender, isActive). |
| 3 | Sales Manager | The Sales Manager is an entity presenting a person who manages and assigns the Target Schools to the Salesmen.  Containing information of Sales Manager (id, password, full name, phone, email, avatar, address, birthday, gender, isActive). |
| 4 | Sales Supervisor | The Sales Supervisor is an entity presenting a person who manages, assigns the Target Schools to the Salesmen and also have right to view all contracts of all target schools.  Containing information of Sales Manager (id, password, full name, phone, email, avatar, address, birthday, gender, isActive). |
| 5 | School | The School is an entity presenting the school that was collected information and was select to become the target.  Contain information of School (id, name, district, address, phone, scale, type, description, educational level, status, isActive). |
| 6 | Target School | The Target School is an entity presenting the target that Sales Manager creates base on Schools list and assigns to Salesmen.  Containing information of School (id, purpose, school year, note, isActive). |
| 7 | Report | The Report is an entity presenting the report that Salesmen must submits to Sales Manager.  Containing the report information (id, date, result, description, positivity, difficulty, futurePlan, supervisorComment). |
| 8 | Personal Activity | The Personal Activity is an entity presenting an activity that Salesmen or Sales Managers create in work-plan and execute based on that.  Containing the Personal Activity information (id, title, startTime, endTime, remark, description, isCompleted). |
| 9 | Representative | The Representative is an entity presenting a person who is the representative for the school to contact.  Containing information of Representative (full name, gender, email, phone). |
| 10 | Contract | The Contract is an entity presenting a contract between a target school and Major Education.  Containing information of Contract (id, duration, service, revenueCriteria, note). |

### 3.2 <<Feature Name 1>>

#### <<Function Name 1>>

*[A function can be a screen or a non-screen function (listed in the part 5.1 above). In this part, you need to provide the details on the related function, focus on mentioning below information*

* *Function trigger: how this function is triggered (navigation path, a timing frequency, etc.*
* *Function description: actors/roles, purpose, interface, data processing, etc.*

*Screen layout: mockup prototype of the screen*

## 4. Non-Functional Requirements

### 4.1 External Interfaces

UI-1: The user interfaces in general should be consistent, simple, clear, intuitive, and reminiscent.

* Some design principles will be taken into consideration: <https://material.io/design/introduction>

UI-2: The user interfaces for the software shall be compatible with: Chrome (version 52 or above), Firefox (version 51 or above), Safari (version 10 or above) browser.

UI-3: The user interfaces should be responsive on multiple screen devices: Laptop, Tablet, and Mobile.

UI-4: The user interfaces should be clear and intuitive message for the users when there is a warning or an error.

UI-5: The user interfaces should ask the users to confirm the action when having major operating actions.

#### Software Interfaces

SI-1: This web application works with: Chrome (version 52 or above), Firefox (version 51 or above), Edge (version 14 or above), Safari (version 10 or above), or Opera (version 38 or above) browser.

#### Hardware Interfaces

HT-1: Since this system is a web application, which it shall work on Laptop, Tablet and Mobile.

HT-2: The system must run over the Internet; all the hardware shall require connect to the Internet.

#### Communications Interfaces

CI-1: The system shall communicate through HTTP protocol over the Internet.

### 4.2 Quality Attributes

#### Usability

- The system is designed simple, and intuitive which the users can easily manage their work.

- The default system language is English.

- The system shall take two hours of training for the users to familiar with usage.

#### Availability

- The system is available 24/7.

#### Security

- The system demands unauthorized users to log in for using the system.

- Each authentic user has a different role and scope for accessing a set of system functions.

- The password field must be obscured with special characters representing typed characters.

- The active session time for this system is two hours, after that the user shall be demanded to log in again.

#### Maintainability

- The source code should be based on a unified convention coding so that developers can easily manage.

- The system is divided into separate components and modules which giving the ability to update or scale up the system.

## 5. Other Requirements

### 5.1 Appendix1 - Messages List

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Message code** | **Message Type** | **Content** | **Content** |
| 1 | MSG01 | In red, above the text box | Invalid username or password | *Wrong format or invalid entry when user logs in.* |
| 2 | MSG02 | In red, above the text box | Wrong username or password | *Wrong username or password when user logs in.* |
| 3 | MSG03 | In red, under the text box | Invalid username or email | *Wrong format or invalid entry when user resets forgotten password.* |
| 4 | MSG04 | In red, under the text box | Wrong username or email | *Wrong username or password when user resets forgotten password.* |
| 5 | MSG05 | In red, above the text box | Invalid password | *Wrong current password when user changes password.* |
| 6 | MSG06 | In red, under the text box | Incorrect entry | *Wrong format or invalid entry when user updates something.* |
| 7 | MSG07 | Toast message, green | Password has been reset successfully | *Reset forgotten password successfully.* |
| 8 | MSG08 | Toast message, green | Updated Successfully | *Update something successfully.* |
| 9 | MSG09 | Toast message, red | Update Unsuccessfully | *Update something failed.* |
| 10 | MSG10 | Toast message, green | Create Successfully | *Create something successfully.* |
| 11 | MSG11 | In line, italic, gray | No records found | *Table does not have data to show.* |
| 12 | MSG12 | In line, red | Connection Failed or Internal Error | *Trouble with the Internet connection.* |

# IV. Software Design Description

## 1. Overall Description

### 1.1 Assumptions

This system is designed basing on these following assumptions:

* Web application
* ReactJS
* Spring boot
* MySQL
* 3-tier architecture

### 1.2 Design Constraints

This system should be complied with following items:

* The system should work on Laptop, Tablet, Mobile.
* The system must run over the Internet; all the hardware shall require connect to the Internet.
* The system shall communicate through HTTP protocol over the Internet.
* The system shall available 24/7.

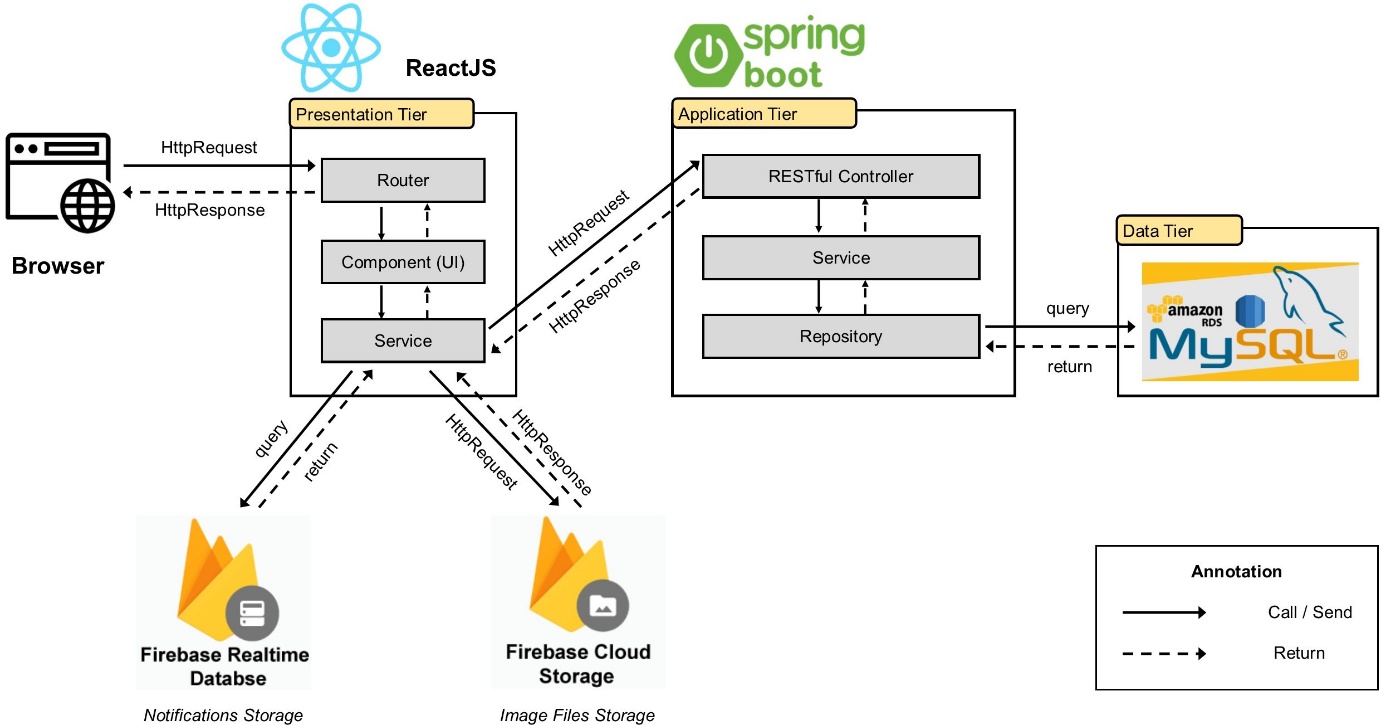
## 2. System Architecture Design

### 2.1 Overall Architecture

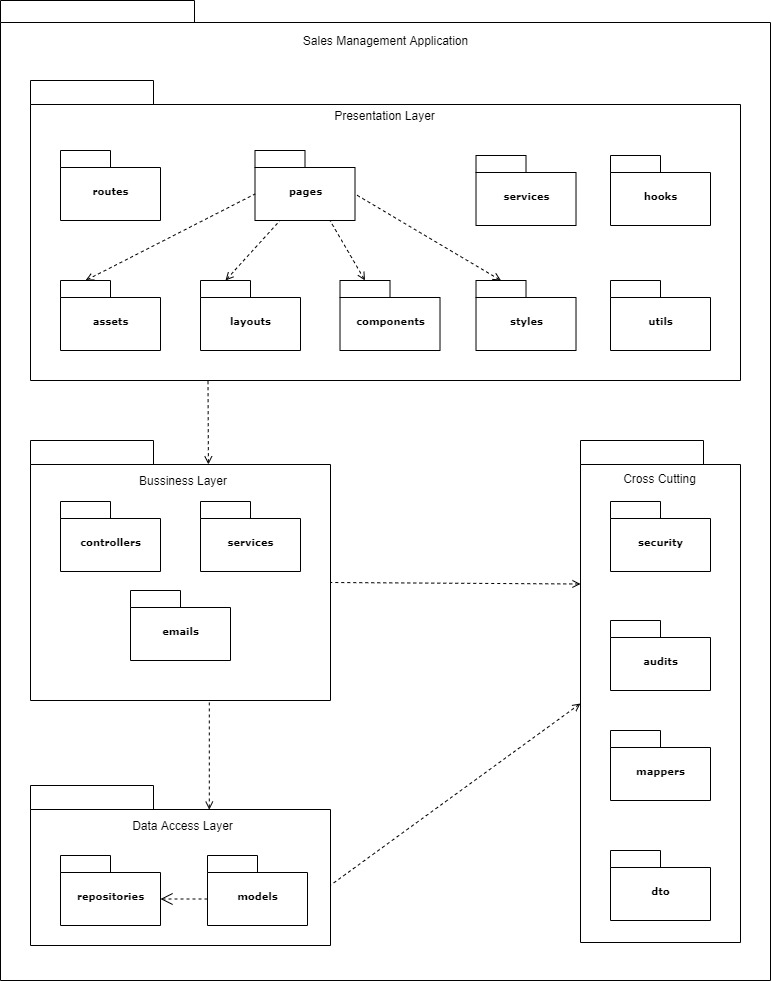
Our system is built on three-tier architecture, which allows us to modularize the user interface, business logic, and data storage layers. As a result, we gain great flexibility, code reusable, and easy maintenance for not only developing but also implementing the system.

|  |  |
| --- | --- |
| **Overall Architecture Data Dictionary** | |
| **Name** | **Description** |
| Presentation Tier | This tier provides a user interface and handles user interactions. It communicates with the application tier which provides the results to the browser. |
| Application Tier | This tier contains a set of rules for processing information and business logic. All communication goes through the application tier. |
| Data Tier | This tier comprises the database where the information processed by the application tier is stored. |

### 2.2 System Architecture

**

### 2.3 Package Diagram

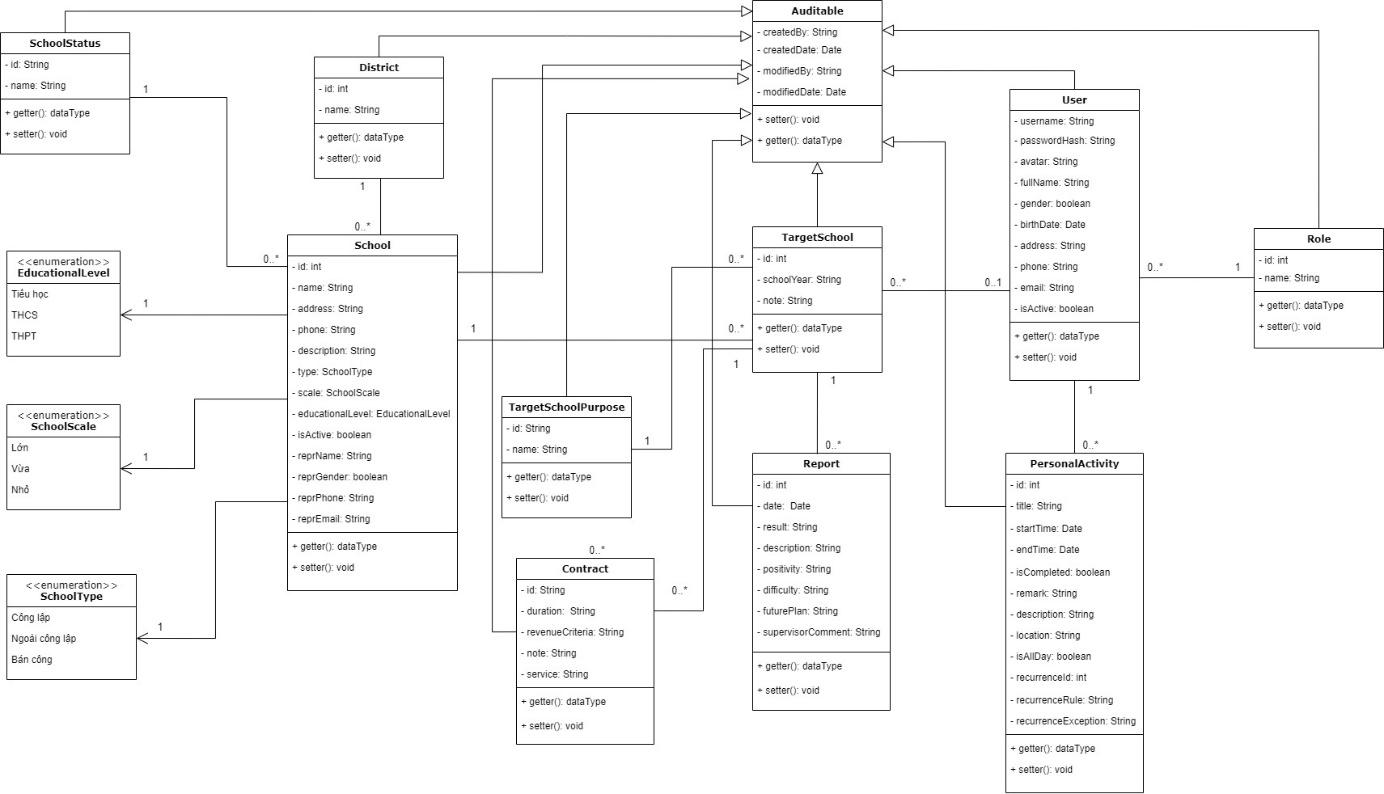


|  |  |  |
| --- | --- | --- |
| **No** | **Package** | **Description** |
| 01 | routes | The package contains functions for app navigation. |
| 02 | pages | The package contains the pages of the routing. |
| 03 | services | The package contains functions to manage API. |
| 04 | hooks | The package contains custom hook functions. |
| 05 | assets | The package contains images and icons. |
| 06 | layouts | The package contains layout components for the app. |
| 07 | components | The package contains UI components that can be shared. |
| 08 | styles | The package contains global CSS, JavaScript for application-level styles. |
| 09 | utils | The package contains all the helper functions. |
| 10 | controllers | The package contains classes provide RESTful web services. |
| 11 | services | The package contains classes for business logic. |
| 12 | emails | The package contains a class provides email service. |
| 13 | repositories | The package contains classes responsible for communicating with data sources, processing queries, and returning data types requested by the Service layer. |
| 14 | models | The package contains classes that represent persisted data to the database. |
| 15 | security | The package contains classes for JWT handling, authentication, and authorization. |
| 16 | audits | The package contains classes for auditing entity objects. |
| 17 | mappers | The package contains a class to convert data between the DTO and entity objects. |
| 18 | dto | The package contains classes to aggregate and encapsulate data for transfer. |

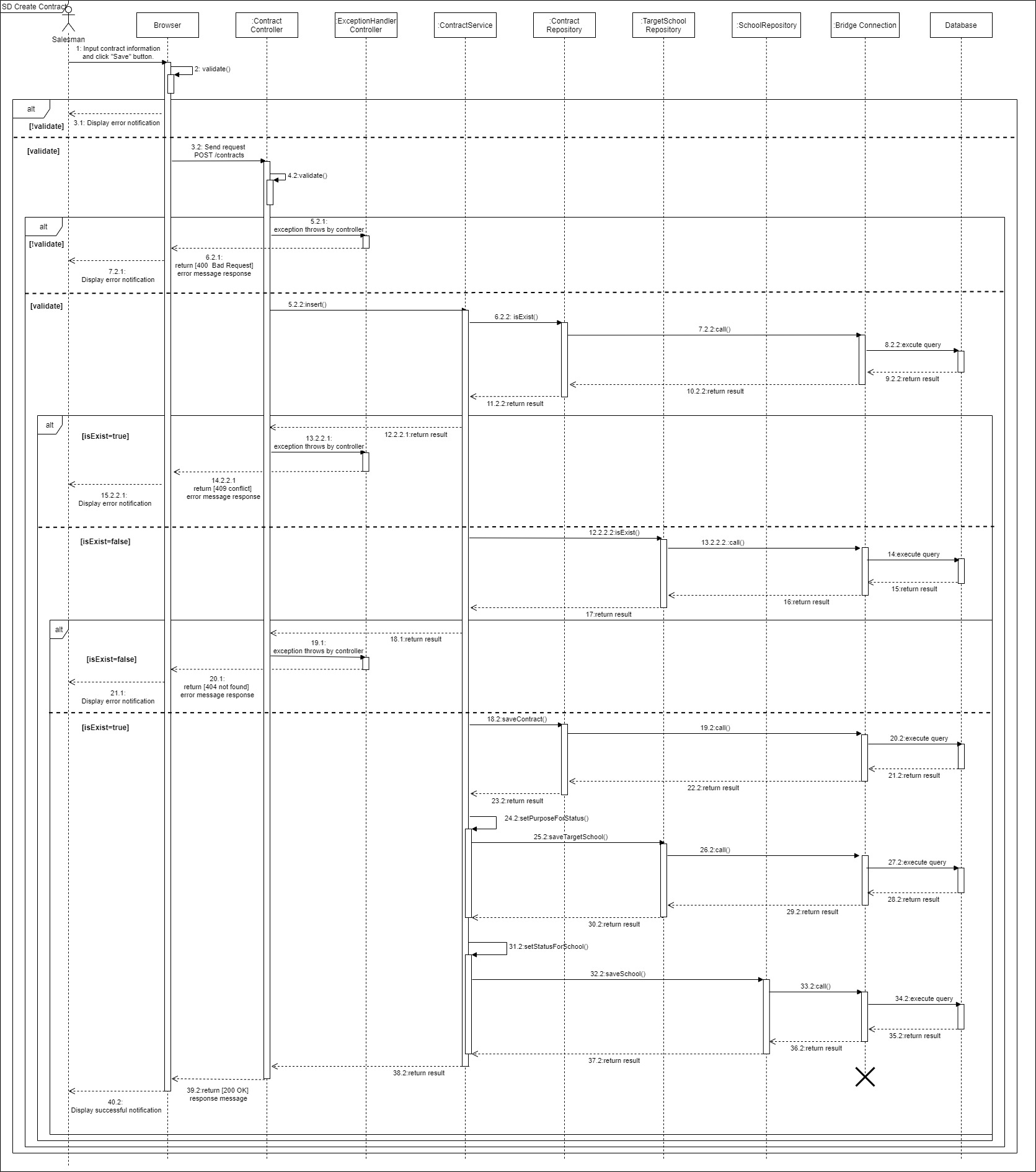
## 3. System Detailed Design

### 3.1 Overall

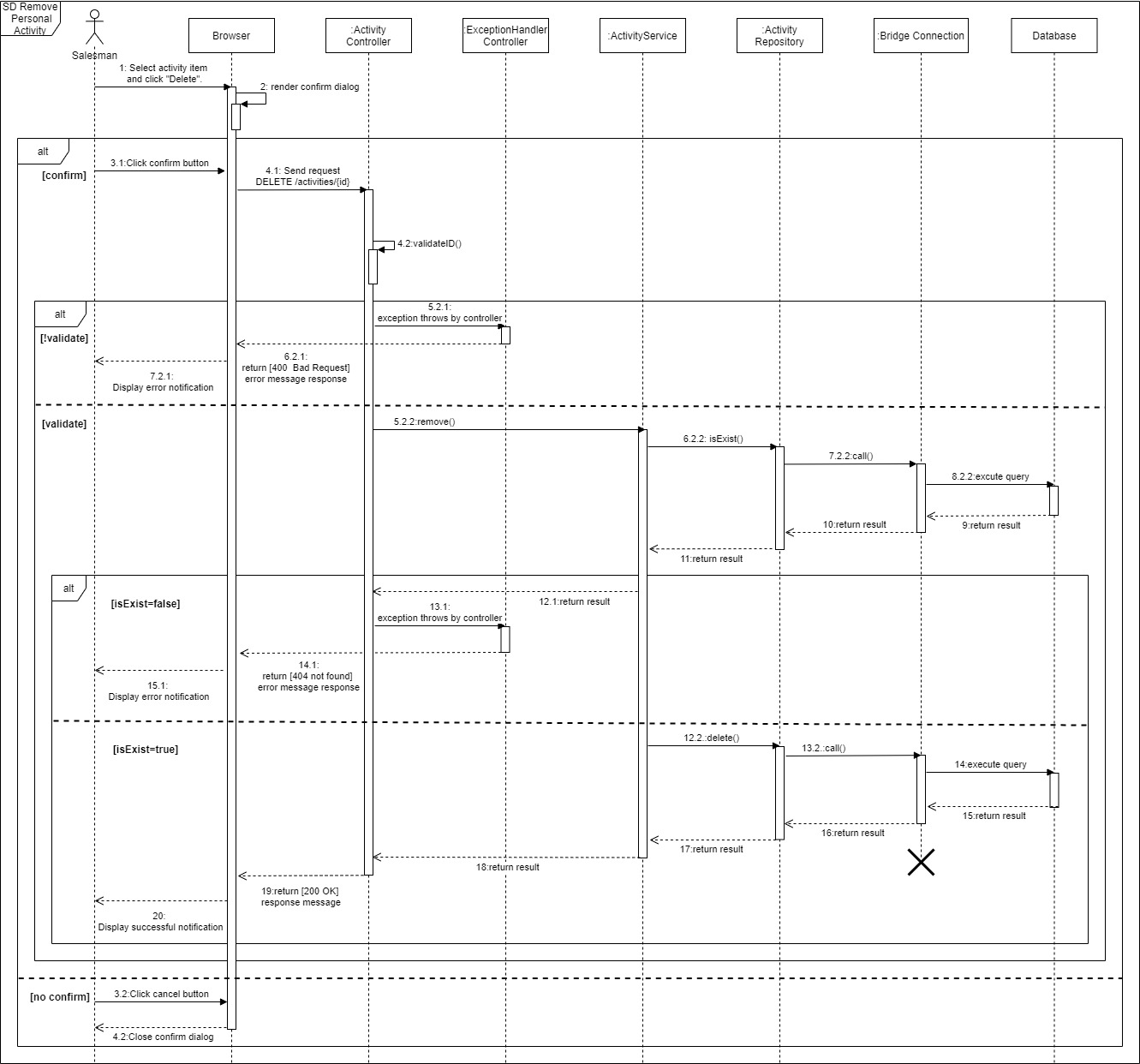
#### Class Diagram



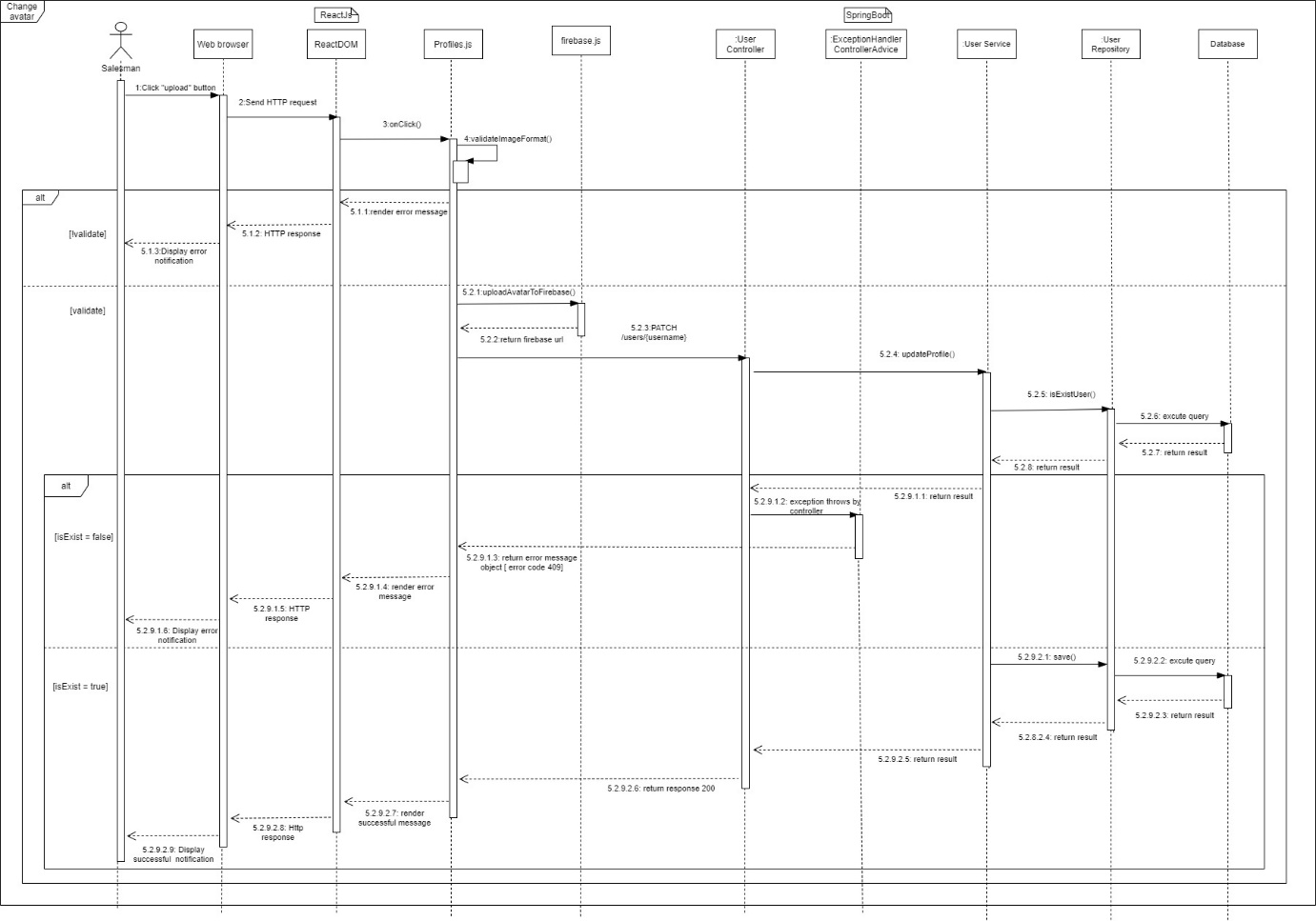
#### Sequence Diagram(s)

**

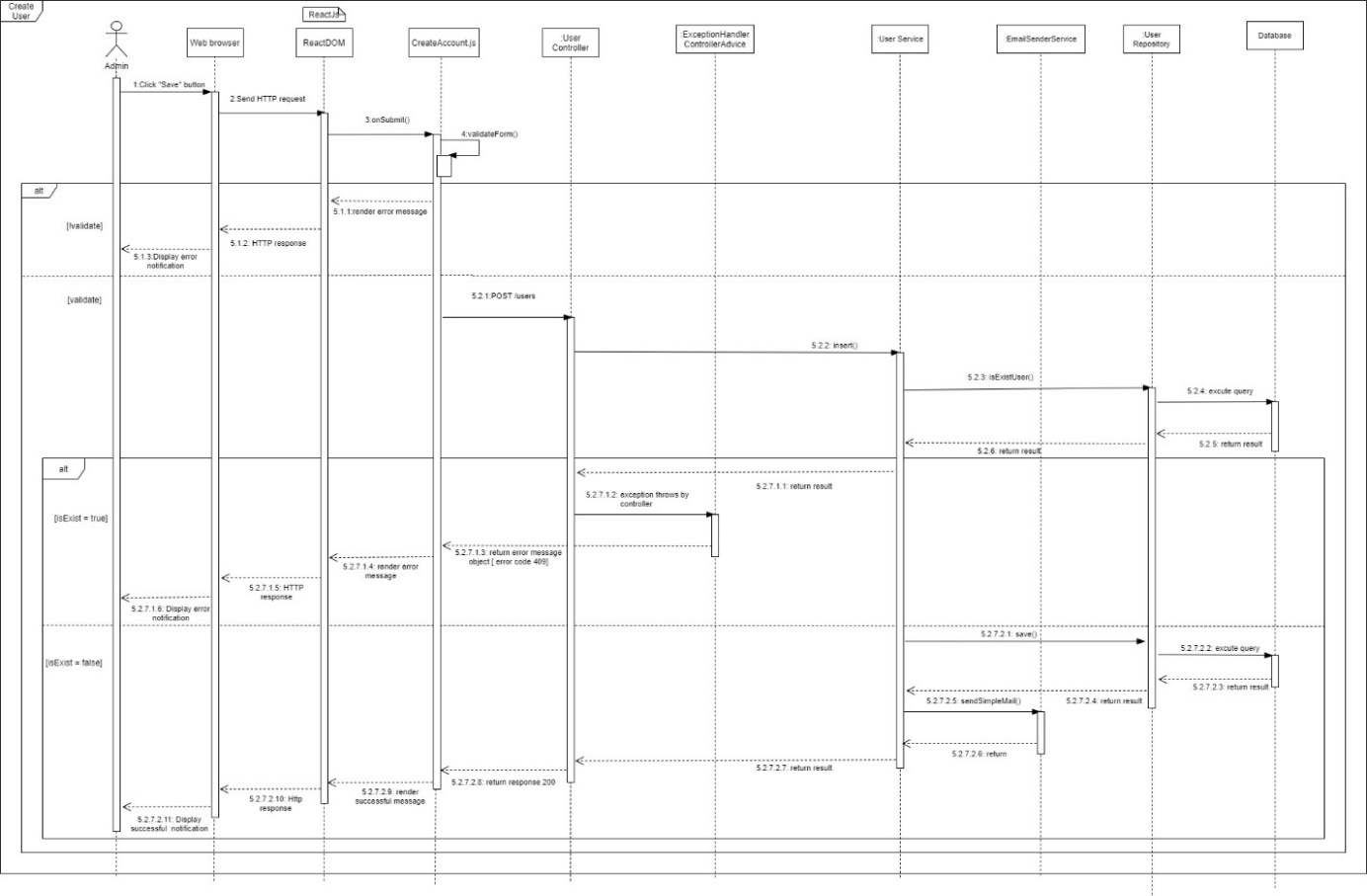
*Create a contract*

**

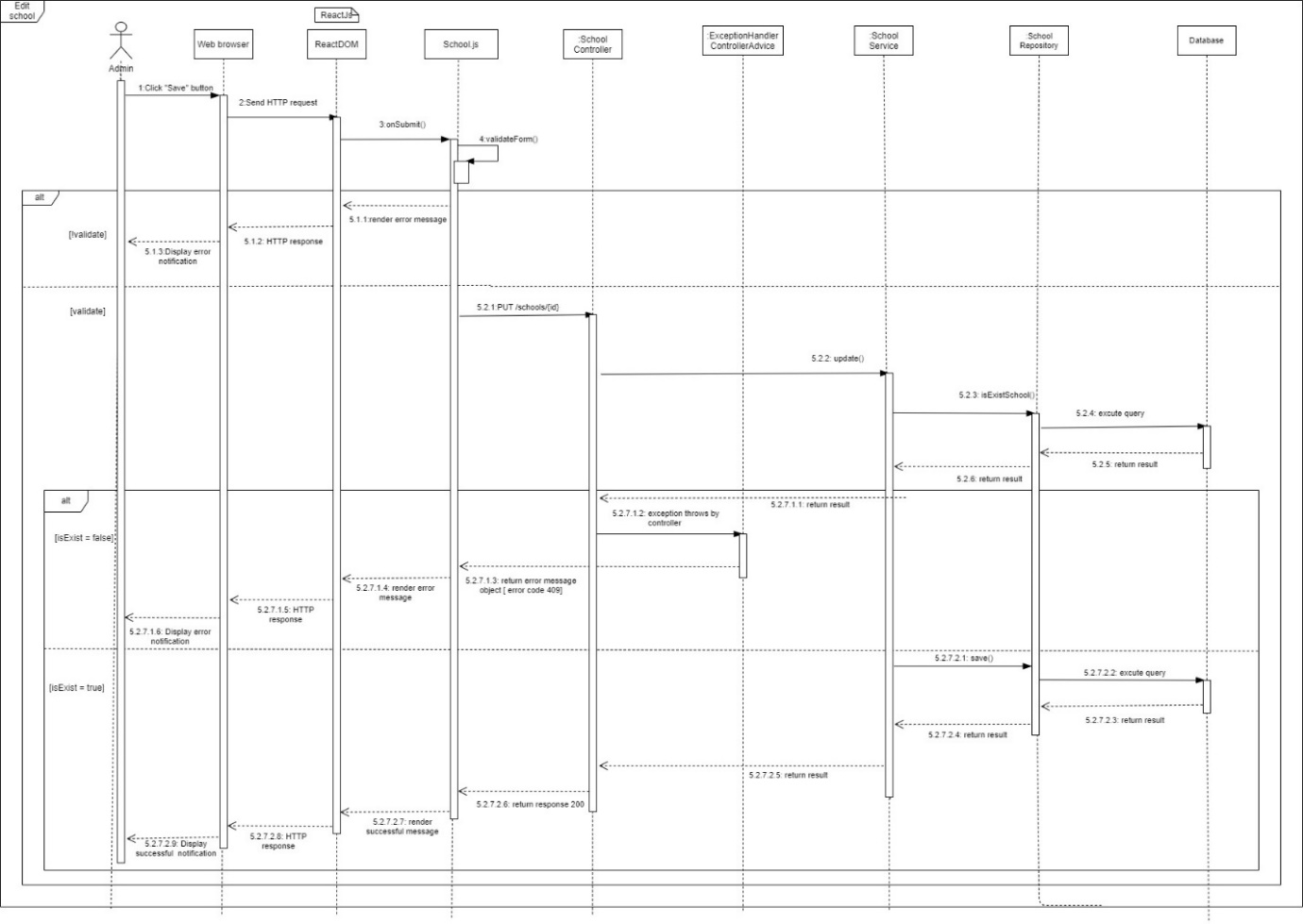
*Remove personal activities*



*Change avatar*



*Create user*



*Edit school*

#### Activity Diagram(s)

1. **<Sales Manager>** **Create target school**

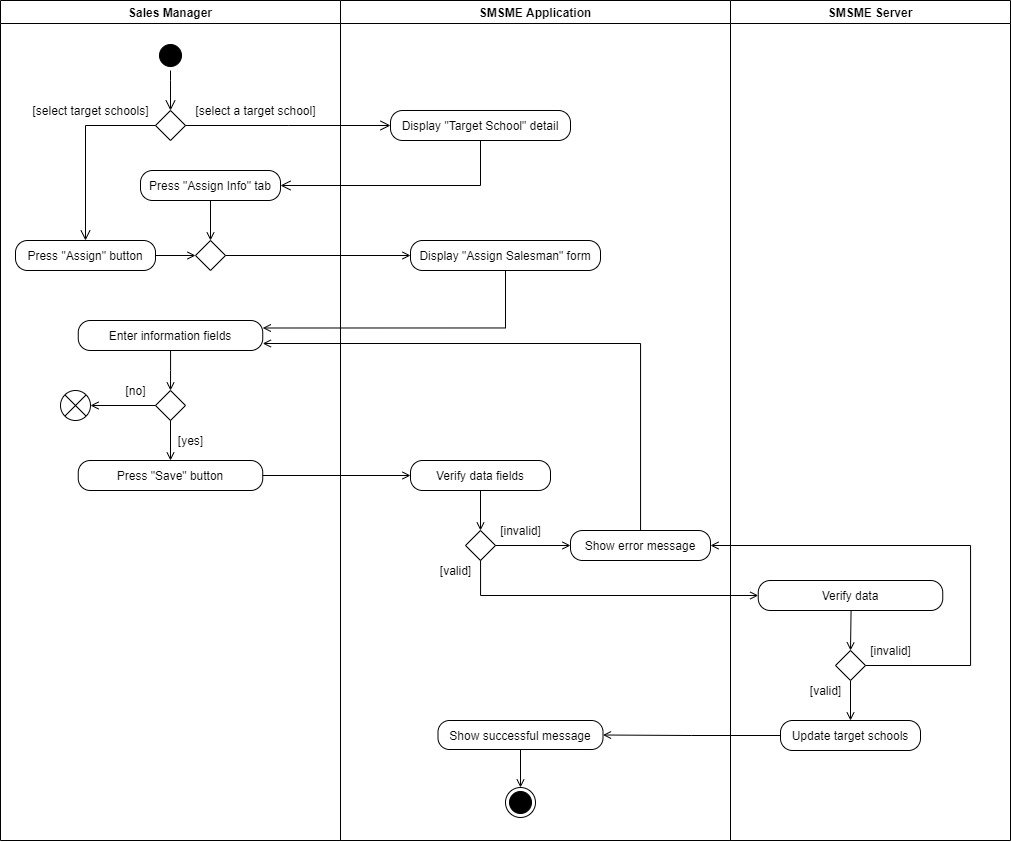
Summary: This diagram shows the process by which the sales manager creates a target school.



Create target school

1. **<Sales Manager> Assign salesman**

Summary: This diagram shows the process by which the sales manager assigns a salesman to a target school.



Assign salesman

1. **<Sales Manager> Update target school**

Summary: This diagram shows the process by which the sales manager updates a target school.

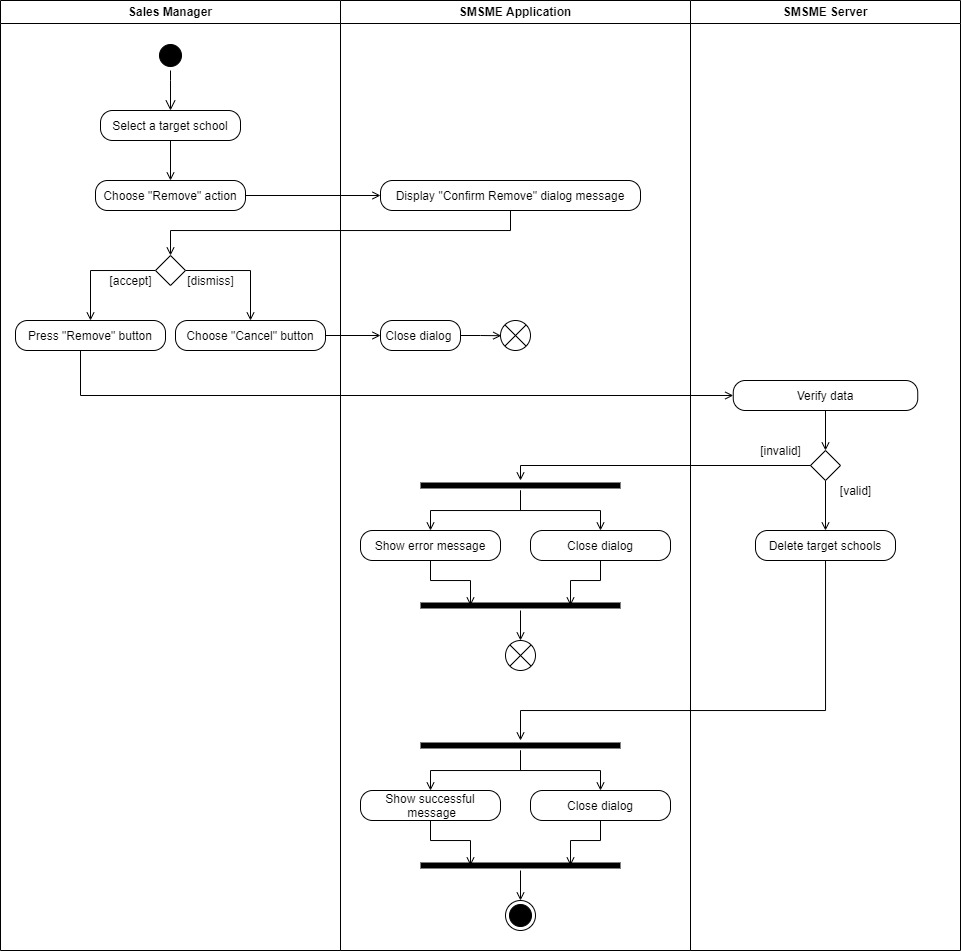
Diagram

Description automatically generated

Update target school

1. **<Sales Manager> Remove target school**

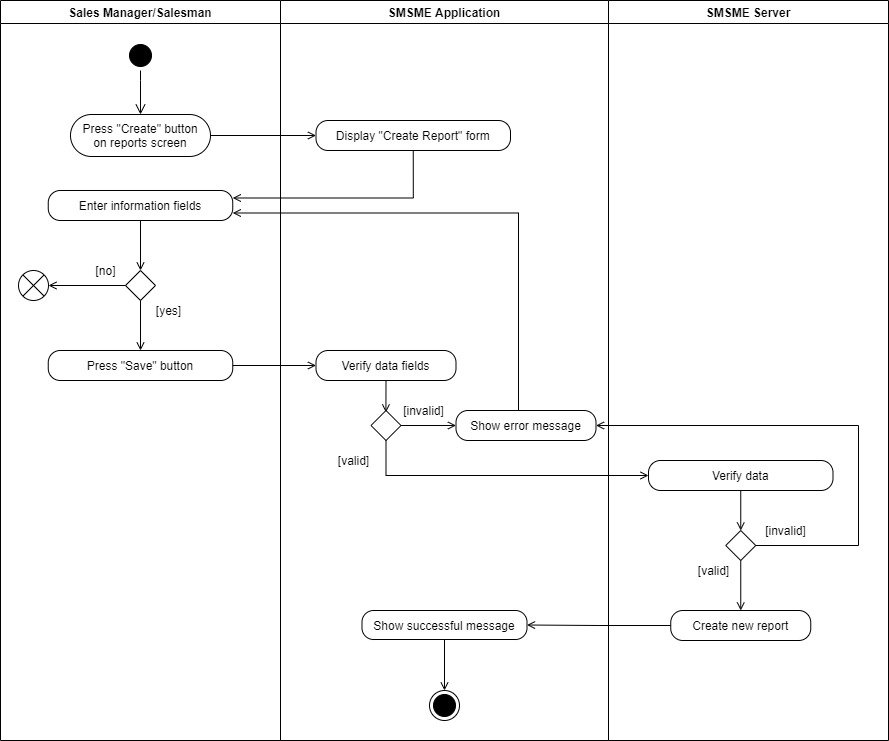
Summary: This diagram shows the process by which the sales manager removes a target school.



Remove a target school

1. **<Sales Manager/ Salesman> Create report**

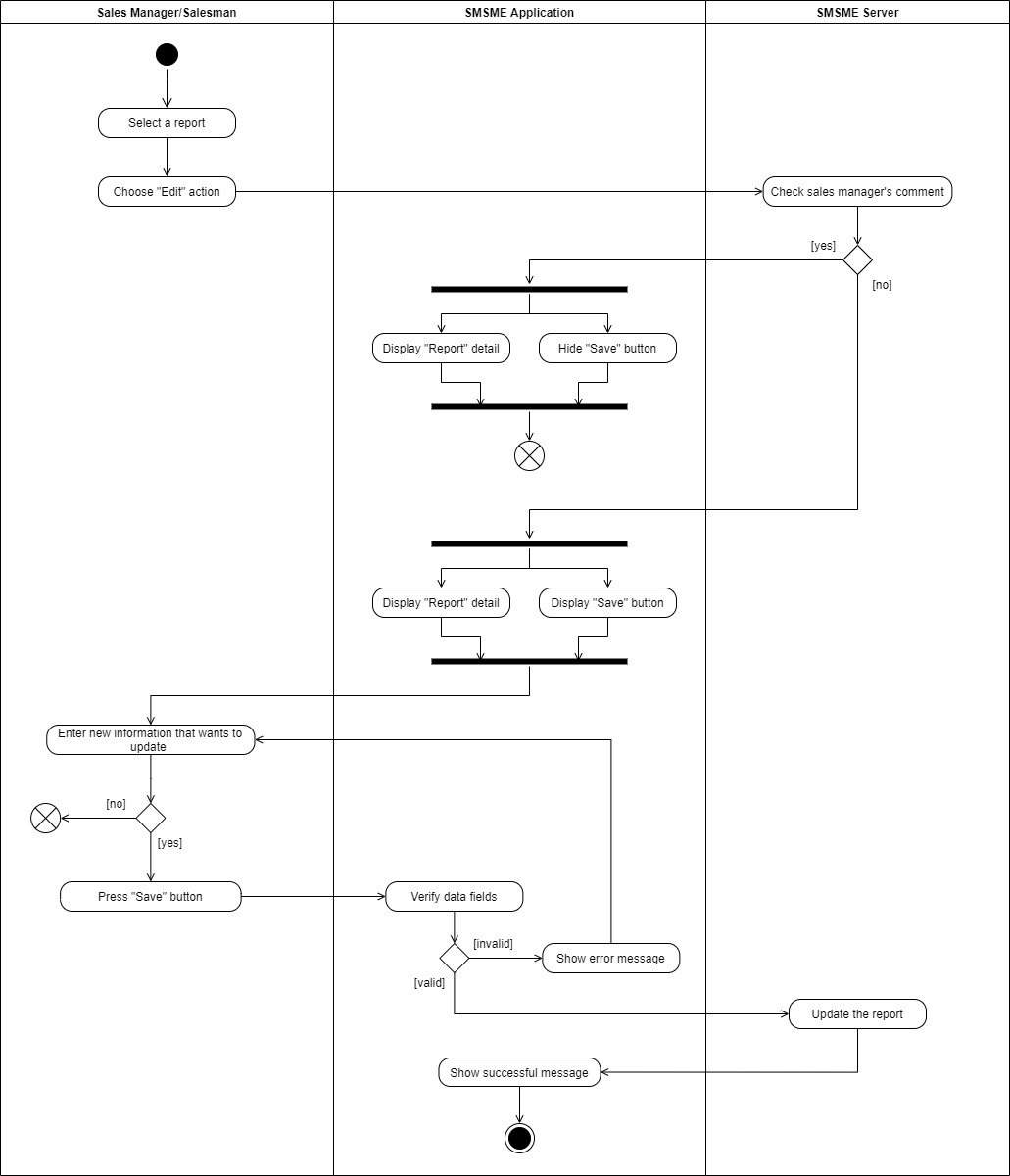
Summary: This diagram shows the process by which the sales manager or salesman create a report.



Create reports

1. **<Sales Manager/ Salesman> Update report**

Summary: This diagram shows the process by which the sales manager or salesman update a report.



Update report

## 4. Class Specification

### 4.1 User

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| username | String | private | The unique identifier of a user |
| passwordHash | String | private | The password hash of the user |
| fullName | String | private | The full name of the user |
| address | String | private | The address of the user |
| email | String | private | The email of the user |
| phone | String | private | The phone number of the user |
| isActive | boolean | private | The active status of the user |
| avatar | String | private | The avatar of the user |
| gender | boolean | private | The gender of the user |
| birthDate | Date | private | The birthdate of the user |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | dataType | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.2 PersonalActivity

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | int | private | The unique identifier of a personal activity |
| title | String | private | The title of the personal activity |
| startTime | Date | private | The start time of the personal activity |
| endTime | Date | private | The end time of the personal activity |
| isCompleted | boolean | private | The completed status of the personal activity |
| remark | String | private | The remark of the personal activity |
| description | String | private | The description of the personal activity |
| location | String | private | The location of the personal activity |
| isAllDay | boolean | private | The all day status of the personal activity |
| recurrenceId | int | private | The unique identifier of a recurrence |
| recurrenceRule | String | private | The rule of the recurrence |
| recurrenceException | String | private | The exception of the recurrence |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | dataType | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.3 Role

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | int | private | The unique identifier of a role |
| name | String | private | The name of the role |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | dataType | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.4 Auditable

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| createdBy | String | private | Auditing who created |
| createdDate | Date | private | The date created for auditing |
| modifiedBy | String | private | Auditing who modified |
| modifiedDate | Date | private | The date modified for auditing |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | dataType | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.5 TargetSchool

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | int | private | The unique identifier of a target school |
| schoolYear | String | private | The school year of the target school |
| note | String | private | The note of the target school |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | dataType | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.6 Report

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | int | private | The unique identifier of a report |
| date | Date | private | The date of the report |
| result | String | private | The result of the report |
| description | String | private | The description of the report |
| positivity | String | private | The positivity of the report |
| difficulty | String | private | The difficulty of the report |
| futurePlan | String | private | The future plan of the report |
| supervisorComment | String | private | The supervisor comment of the report |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | dataType | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.7 School

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | int | private | The unique identifier of a school |
| name | String | private | The name of the school |
| address | String | private | The address of the school |
| phone | String | private | The phone number of the school |
| type | SchoolType | private | The school type of the school |
| scale | SchoolScale | private | The school scale of the school |
| educationalLevel | EducationalLevel | private | The educational level of the school |
| description | String | private | The description of the school |
| isActive | boolean | private | The active status of the school |
| reprName | String | private | The name of the representative |
| reprGender | boolean | private | The gender of the representative |
| reprPhone | String | private | The phone number of the representative |
| reprEmail | String | private | The email of the representative |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | dataType | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.8 District

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | int | private | The unique identifier of a district |
| name | String | private | The name of the district |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | dataType | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.9 SchoolStatus

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | String | private | The unique identifier of a school status |
| name | String | private | The name of the school status |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | dataType | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.10 TargetSchoolPurpose

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| id | String | private | The unique identifier of a target school purpose |
| name | String | private | The name of the target school purpose |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Method** | **Visibility** | **Return Type** | **Description** |
| 1 | getter | public | dataType | get attribute value |
| 2 | setter | public | void | set attribute value |

### 4.11 EducationalLevel

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| Tiểu học |  |  | Enum attribute |
| THCS |  |  | Enum attribute |
| THPT |  |  | Enum attribute |

### 4.12 SchoolScale

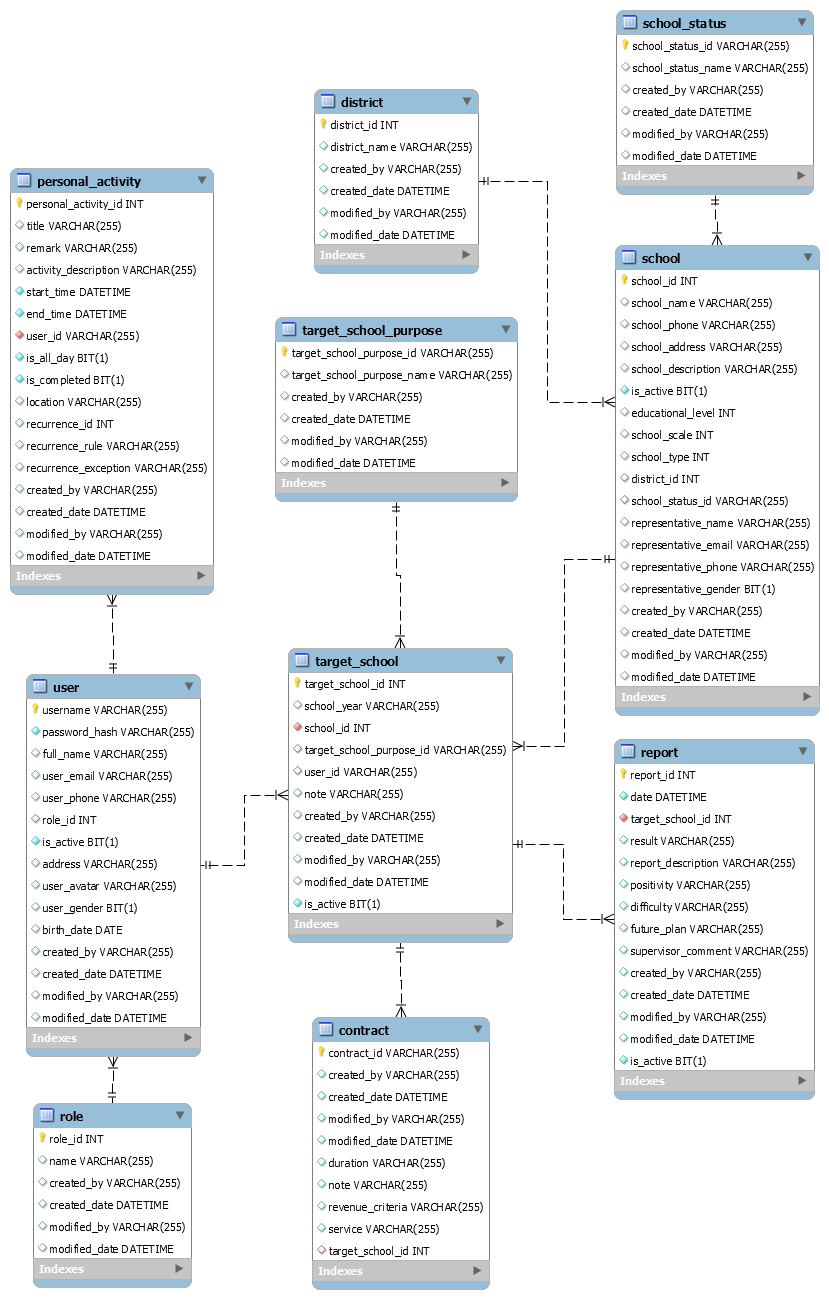
|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| Lớn |  |  | Enum attribute |
| Vừa |  |  | Enum attribute |
| Nhỏ |  |  | Enum attribute |

### 4.13 SchoolType

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **Type** | **Visibility** | **Description** |
| Công lập |  |  | Enum attribute |
| Ngoài công lập |  |  | Enum attribute |
| Bán công |  |  | Enum attribute |

## 5. Data & Database Design

### 5.1 Database Design



#### school

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | school\_id | int |  | X | X | X |  |
| 2 | school\_name | varchar | 255 |  |  |  |  |
| 3 | school\_phone | varchar | 255 |  |  |  |  |
| 4 | school\_address | varchar | 255 |  |  |  |  |
| 5 | school\_description | varchar | 255 |  |  |  |  |
| 6 | is\_active | bit |  |  | X |  |  |
| 7 | education\_level | int | 255 |  |  |  |  |
| 8 | school\_scale | int |  |  |  |  |  |
| 9 | school\_type | int |  |  |  |  |  |
| 10 | district\_id | int |  |  |  |  |  |
| 11 | school\_status\_id | varchar | 255 |  |  |  |  |
| 12 | representative\_name | varchar | 255 |  |  |  |  |
| 13 | representative\_email | varchar | 255 |  |  |  |  |
| 14 | representative\_phone | varchar | 255 |  |  |  |  |
| 15 | representative\_gender | bit |  |  |  |  |  |
| 16 | created\_by | varchar | 255 |  |  |  | Use for auditing |
| 17 | created\_date | datetime |  |  |  |  | Use for auditing |
| 18 | modified\_by | varchar | 255 |  |  |  | Use for auditing |
| 19 | modified\_date | datetime |  |  |  |  | Use for auditing |

#### district

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | district\_id | int |  | X | X | X |  |
| 2 | district\_name | varchar | 255 |  |  |  |  |
| 3 | created\_by | varchar | 255 |  |  |  | Use for auditing |
| 4 | created\_date | datetime |  |  |  |  | Use for auditing |
| 5 | modified\_by | varchar | 255 |  |  |  | Use for auditing |
| 6 | modified\_date | datetime |  |  |  |  | Use for auditing |

#### school\_status

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | school\_status\_id | varchar | 255 | X | X | X |  |
| 2 | school\_status\_name | varchar | 255 |  |  |  |  |
| 3 | created\_by | varchar | 255 |  |  |  | Use for auditing |
| 4 | created\_date | datetime |  |  |  |  | Use for auditing |
| 5 | modified\_by | varchar | 255 |  |  |  | Use for auditing |
| 6 | modified\_date | datetime |  |  |  |  | Use for auditing |

#### target\_school

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | target\_school\_id | int |  | X | X | X |  |
| 2 | school\_year | varchar | 255 |  |  |  |  |
| 3 | school\_id | int |  |  | X |  |  |
| 4 | target\_school\_purpose\_id | varchar | 255 |  |  |  |  |
| 5 | user\_id | varchar | 255 |  |  |  |  |
| 6 | note | varchar | 255 |  |  |  |  |
| 7 | created\_by | varchar | 255 |  |  |  | Use for auditing |
| 8 | created\_date | datetime |  |  |  |  | Use for auditing |
| 9 | modified\_by | varchar | 255 |  |  |  | Use for auditing |
| 10 | modified\_date | datetime |  |  |  |  | Use for auditing |

#### report

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | report\_id | int |  | X | X | X |  |
| 2 | date | datetime |  |  | X |  |  |
| 3 | target\_school\_id | int |  |  | X |  |  |
| 4 | result | varchar | 255 |  |  |  |  |
| 5 | report\_description | varchar | 255 |  |  |  |  |
| 6 | positivity | varchar | 255 |  |  |  |  |
| 7 | difficulty | varchar | 255 |  |  |  |  |
| 8 | future\_plan | varchar | 255 |  |  |  |  |
| 9 | supervisor\_comment | varchar | 255 |  |  |  |  |
| 10 | created\_by | varchar | 255 |  |  |  | Use for auditing |
| 11 | created\_date | datetime |  |  |  |  | Use for auditing |
| 12 | modified\_by | varchar | 255 |  |  |  | Use for auditing |
| 13 | modified\_date | datetime |  |  |  |  | Use for auditing |

#### target\_school\_purpose

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | target\_school\_purpose\_id | varchar | 255 | X | X | X |  |
| 2 | target\_school\_purpose\_name | varchar | 255 |  |  |  |  |
| 3 | created\_by | varchar | 255 |  |  |  | Use for auditing |
| 4 | created\_date | datetime |  |  |  |  | Use for auditing |
| 5 | modified\_by | varchar | 255 |  |  |  | Use for auditing |
| 6 | modified\_date | datetime |  |  |  |  | Use for auditing |

#### user

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | username | varchar | 255 | X | X | X |  |
| 2 | password\_hash | varchar | 255 |  | X |  |  |
| 3 | full\_name | varchar | 255 |  |  |  |  |
| 4 | user\_email | varchar | 255 |  |  |  |  |
| 5 | user\_phone | varchar | 255 |  |  |  |  |
| 6 | role\_id | int |  |  |  |  |  |
| 7 | is\_active | bit |  |  | X |  |  |
| 8 | address | varchar | 255 |  |  |  |  |
| 9 | user\_avatar | varchar | 255 |  |  |  |  |
| 10 | user\_gender | bit |  |  |  |  |  |
| 11 | birth\_date | date |  |  |  |  |  |
| 12 | created\_by | varchar | 255 |  |  |  | Use for auditing |
| 13 | created\_date | datetime |  |  |  |  | Use for auditing |
| 14 | modified\_by | varchar | 255 |  |  |  | Use for auditing |
| 15 | modified\_date | datetime |  |  |  |  | Use for auditing |

#### role

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | role\_id | int |  | X | X | X |  |
| 2 | name | varchar | 255 |  |  |  |  |
| 3 | created\_by | varchar | 255 |  |  |  | Use for auditing |
| 4 | created\_date | datetime |  |  |  |  | Use for auditing |
| 5 | modified\_by | varchar | 255 |  |  |  | Use for auditing |
| 6 | modified\_date | datetime |  |  |  |  | Use for auditing |

#### personal\_activity

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | personal\_activity\_id | int |  | X | X | X |  |
| 2 | title | varchar | 255 |  |  |  |  |
| 3 | remark | varchar | 255 |  |  |  |  |
| 4 | activity\_description | varchar | 255 |  |  |  |  |
| 5 | start\_time | datetime |  |  | X |  |  |
| 6 | end\_time | datetime |  |  | X |  |  |
| 7 | user\_id | varchar | 255 |  | X |  |  |
| 8 | is\_all\_day | bit |  |  | X |  |  |
| 9 | is\_completed | bit |  |  | X |  |  |
| 10 | location | varchar | 255 |  |  |  |  |
| 11 | recurrence\_id | int |  |  |  |  |  |
| 12 | recurrence\_rule | varchar | 255 |  |  |  |  |
| 13 | recurrence\_exception | varchar | 255 |  |  |  |  |
| 14 | created\_by | varchar | 255 |  |  |  | Use for auditing |
| 15 | created\_date | datetime |  |  |  |  | Use for auditing |
| 16 | modified\_by | varchar | 255 |  |  |  | Use for auditing |
| 17 | modified\_date | datetime |  |  |  |  | Use for auditing |

#### contract

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Field name** | **Type** | **Size** | **Unique** | **Not Null** | **PK/FK** | **Notes** |
| 1 | contract\_id | int |  | X | X | X |  |
| 2 | duration | varchar | 255 |  |  |  |  |
| 3 | note | varchar | 255 |  |  |  |  |
| 4 | revenue\_criteria | varchar | 255 |  |  |  |  |
| 5 | Service | varchar | 255 |  |  |  |  |
| 6 | target\_school\_id | int |  |  |  |  |  |
| 7 | created\_by | varchar | 255 |  |  |  | Use for auditing |
| 8 | created\_date | datetime |  |  |  |  | Use for auditing |
| 9 | modified\_by | varchar | 255 |  |  |  | Use for auditing |
| 10 | modified\_date | datetime |  |  |  |  | Use for auditing |

### 5.2 Data File Design

We use firebase to store images.

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **File Name** | **Type** | **Notes** |
| 1 | avatars | JPG, JPEG, PNG | The images are used for user avatar. |

# V. Software Testing Documentation

## 1. Overall Description

### 1.1 Test Model

Based on the user requirements, team resources, and project conditions, our team has come up with using the prototype model as a software development model. In this model, a prototype is built, tested, and reworked until it is accepted by the user.

Diagram

Description automatically generated

Figure 1 Prototype model

References: <https://www.geeksforgeeks.org/software-engineering-prototyping-model/>

### 1.2 Testing Levels

|  |  |
| --- | --- |
| **Testing Level** | **Description** |
| Unit Testing | Checks the software modules are fit for use |
| Integration Testing | Checks the integrated logic of the software module group |
| System Testing | Checks the whole system if it works as expected or not |
| Acceptance Testing | Checks the software if the user requirements are met or not |

### 1.3 Testing Types

|  |  |
| --- | --- |
| **Testing types** | **Description** |
| Functional Testing | * Unit Testing * Integration Testing * System Testing * Interface Testing * Regression Testing * Beta Testing |
| Non-functional Testing | * Performance Testing |

## 2. Test Plan

### 2.1 Test Stages

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Type of Test** | **Stage of Test** | | | |
| ***Unit*** | ***Integration*** | ***System*** | ***Acceptance*** |
| Interface Testing | X |  | X |  |
| Performance Testing |  | X |  |  |
| Regression Testing | X | X |  |  |
| Beta Testing |  |  |  | X |

### 2.2 Resources

#### Human Resources

|  |  |  |
| --- | --- | --- |
| **Worker/Doer** | **Role** | **Specific Responsibilities/Comments** |
| Phạm Thị Ngọc Hà | Leader | * Functional Testing * Non-functional Testing |
| Lê Gia Nguyên | Member | * Functional Testing * Non-functional Testing |
| Nguyễn Hoàng Gia | Member | * Functional Testing * Non-functional Testing |

#### Environment

|  |  |  |  |
| --- | --- | --- | --- |
| **Purpose** | **Tool** | **Provider** | **Version** |
| Unit Testing | Postman | Postman, Inc | 8.2.1 |
| Unit Testing  Integration Testing  System Testing  Interface Testing  Regression Testing  Performance Testing | Chrome | Google LLC | 89 |

### 2.3 Test Milestones

|  |  |  |  |
| --- | --- | --- | --- |
| **Milestone Task** | **Effort (md)** | **Start Date** | **End Date** |
| Setup the environment | 2 | 09/04/2021 | 09/04/2021 |
| Admin role function unit test | 2 | 09/04/2021 | 09/04/2021 |
| Admin role integration test | 3 | 09/04/2021 | 09/04/2021 |
| Admin role interface test | 2 | 09/04/2021 | 09/04/2021 |
| Salesman role function unit test | 4 | 09/04/2021 | 09/04/2021 |
| Salesman role integration test | 5 | 09/04/2021 | 09/04/2021 |
| Salesman role interface test | 4 | 09/04/2021 | 09/04/2021 |
| Sales manager role function unit test | 4 | 10/04/2021 | 10/04/2021 |
| Sales manager role integration test | 5 | 10/04/2021 | 10/04/2021 |
| Sales manager role interface test | 4 | 10/04/2021 | 10/04/2021 |
| Admin role regression test | 3 | 11/042021 | 11/042021 |
| Admin role performance test | 3 | 11/042021 | 11/042021 |
| Salesman role regression test | 5 |  |  |
| Salesman role performance test | 3 |  |  |
| Sales manager role regression test | 5 |  |  |
| Sales manager role performance test | 3 |  |  |
| Sales supervisor role function unit test | 3 |  |  |
| Sales supervisor role integration test | 3 |  |  |
| Sales supervisor role interface test | 3 |  |  |
| Sales supervisor role regression test | 4 |  |  |
| Sales supervisor role performance test | 3 |  |  |
| System test | 7 |  |  |
| Beta test | 10 |  |  |

### 2.4 Deliverables

|  |  |  |
| --- | --- | --- |
| No | Deliverables | Due Date |
| 1 | Report5\_Unit Test Case | 12/04/2021 |
| 2 | Report5\_ Test Case Document | 13/04/2021 |

## 3. Test Cases

|  |  |
| --- | --- |
| **No** | **Filename** |
| 1 | Report5\_Unit Test Case.xls |
| 2 | Report5\_Test Case Document.xls |

## 4. Test Reports

|  |  |
| --- | --- |
| **No** | **Filename** |
| 1 | Report5\_Unit Test Case.xls |

# VI. Release Package & User Guides

## 1. Deliverable Package

### 1.1 Source codes & documents

| **No.** | **Items** | **Sub-Items** | **Type** | **Version** |
| --- | --- | --- | --- | --- |
| ***Code Package*** | | | | |
| 1 | Web Apps |  | New | 1.0.0 |
| 2 | API Services |  | New | 1.0.0 |
| ***Database*** | | | | |
| 1 | Tables | contract.sql | New | 1.0.0 |
| district.sql | New | 1.0.0 |
| personal\_activity.sql | New | 1.0.0 |
| report.sql | New | 1.0.0 |
| role.sql | New | 1.0.0 |
| school.sql | New | 1.0.0 |
| school\_status.sql | New | 1.0.0 |
| target\_school.sql | New | 1.0.0 |
| target\_school\_purpose.sql | New | 1.0.0 |
| user.sql | New | 1.0.0 |
| ***Documents*** | | | | |
| 1 | Introduction | Report1\_Project Introduction.docx | New | 1.0.0 |
| 2 | Management Plan | Report2\_Project Management Plan.docx | New | 1.0.0 |
| 3 | Requirement | Report3\_Software Requirement Specification.docx | New | 1.0.0 |
| 4 | Design | Report4\_Software Design Document.docx | New | 1.0.0 |
| 5 | Testing | Report5\_Test Documentation.docx | New | 1.0.0 |
| Report5\_Test Case Document.docx | New | 1.0.0 |
| Report5\_Unit Test Case.docx | New | 1.0.0 |
| 6 | User Guides | Report6\_Software User Guides.docx | New | 1.0.0 |
| 7 | Final | Report7\_Final Project Report.docx | New | 1.0.0 |

### 1.2 Known Issues, Limitations & Restrictions

* The current system does not provide a mobile app version for the user.
* The current system does not support communication channel with customers.
* The web application responsive interface has not stable yet.
* The interface still lacks consistency, may or may not cause uncomfortable for the user.
* The current application only supports English default.
* The system lacks testing, which leads to application performance or bugs that may cause during implementation.
* During the lack of team resources, some main features of the application have not been implemented.

## 2. Installation Guides

### 2.1 System Requirements

* + 1. **Hardware Requirements**
* Scince this system is a web application, all the devices using the system must run over the Internet.
* Our system is deployed on the server, which the minimum requirements services to run in the following:

For Web App service:

|  |  |
| --- | --- |
| **Hardware** | **Minimum Requirements** |
| vCPU | 1 |
| Operation System | Linux Server/Window server |
| Memory (GiB) | 1GiB |

For API Services service:

|  |  |
| --- | --- |
| **Hardware** | **Minimum Requirements** |
| vCPU | 1 |
| Operation System | Linux Server/Window server |
| Memory | 1GiB |

For Database service:

|  |  |
| --- | --- |
| **Hardware** | **Minimum Requirements** |
| vCPU | 1 |
| Operation System | Linux server/Window sẻver |
| Computer Memory | 1GiB |

* + 1. **Software Requirements**

This web application works with: Chrome (version 52 or above), Firefox (version 51 or above), Edge (version 14 or above), Safari (version 10 or above), or Opera (version 38 or above) browser.

### 2.2 Setup Files

N/A

### 2.3 Installation Instruction

N/A

## 3. User Manual

### 3.1 Terms and definitions

* This system is built for Major Education with a project code SMSME, which stands for Sales Management System for Major Education.
* This system is an enterprise application, with the license rules and terms of use belong to Major Education.

### 3.2 System requirements

#### 3.2.1. Hardware Requirements

* Scince this system is a web application, all the devices using the system must run over the Internet.
* Our system is deployed on the server, which the minimum requirements services to run in the following:

For Web App service:

|  |  |
| --- | --- |
| **Hardware** | **Minimum Requirements** |
| vCPU | 1 |
| Operation System | Linux Server/Window server |
| Memory (GiB) | 1GiB |

For API Services service:

|  |  |
| --- | --- |
| **Hardware** | **Minimum Requirements** |
| vCPU | 1 |
| Operation System | Linux Server/Window server |
| Memory (GiB) | 1GiB |

For Database service:

|  |  |
| --- | --- |
| **Hardware** | **Minimum Requirements** |
| vCPU | 1 |
| Operation System | Linux server/Window sẻver |
| Memory (GiB) | 1GiB |

#### 3.2.2. Software Requirements

* This web application works with: Chrome (version 52 or above), Firefox (version 51 or above), Edge (version 14 or above), Safari (version 10 or above), or Opera (version 38 or above) browser.

### 3.3 Application Usage

#### Overview

The target users that SMSME mainly aims to are salesmen and sales managers of the Sales Department of Major Education.

For salesmen, this software would help them manage their work-plan as well as their tasks easier and more visually, submit daily reports more directly and they can also monitor other salesmen’s works.

For sales managers, they would not have to collect and analyze all salesmen’s reports manually. Now they can supervise salesmen or search for reports more easily. Moreover, they can see the overall picture of the company's business process, including what schools are lead or customers, who are the most effective salesmen,…

#### Accounts

Purpose

* The admin manages user accounts

Functions

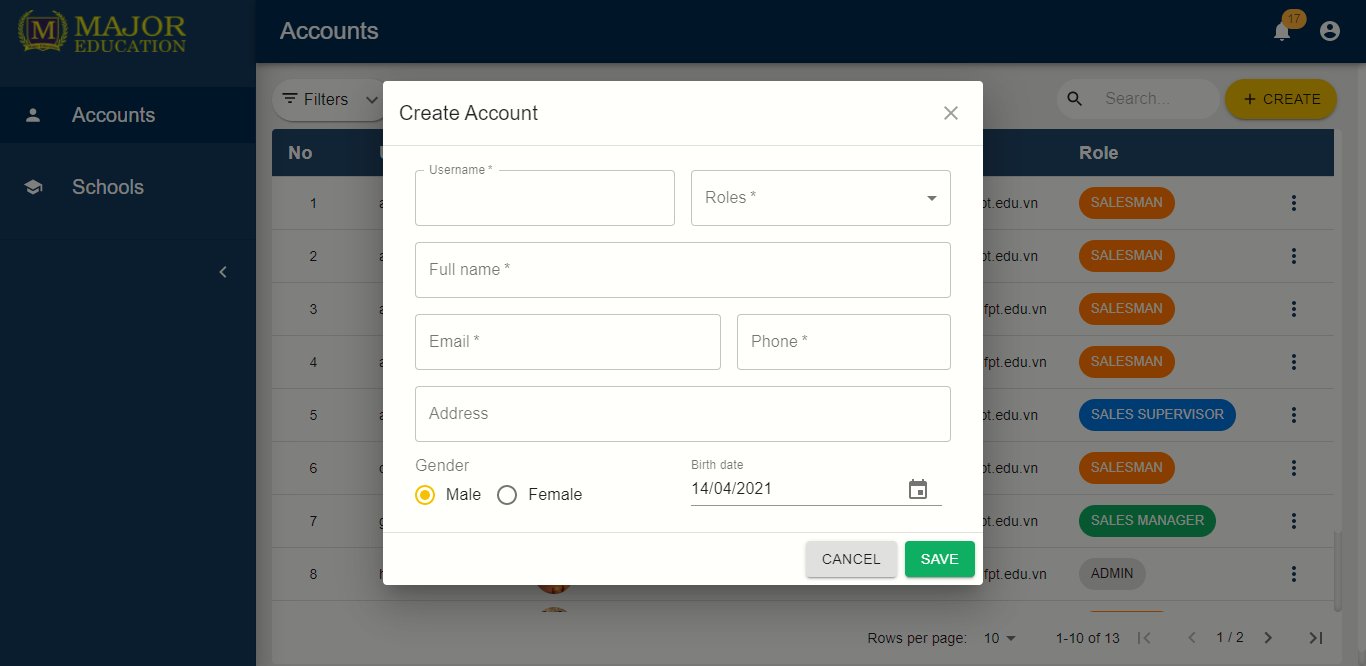
* View list of accounts
* View account’s details
* Search accounts
* Create accounts
* Edit account’s information

User interface(s)

Graphical user interface, application

Description automatically generated

Accounts list



Create Account

Graphical user interface, application

Description automatically generated

Account detail

#### Schools

Purpose

* The admin manages schools info
* The sales manager edits schools info
* The sales supervisor edits schools info
* The salesman edits schools info

Functions

* View list of schools
* View school’s details
* Search schools
* Create schools
* Edit school’s information

User interface(s)

Graphical user interface, application

Description automatically generated

Schools list

Graphical user interface, application

Description automatically generated

School detail

#### Target schools

Purpose

* The sales manager manages target schools
* The sales supervisor manages target schools
* The salesman view assigned of target schools

Functions

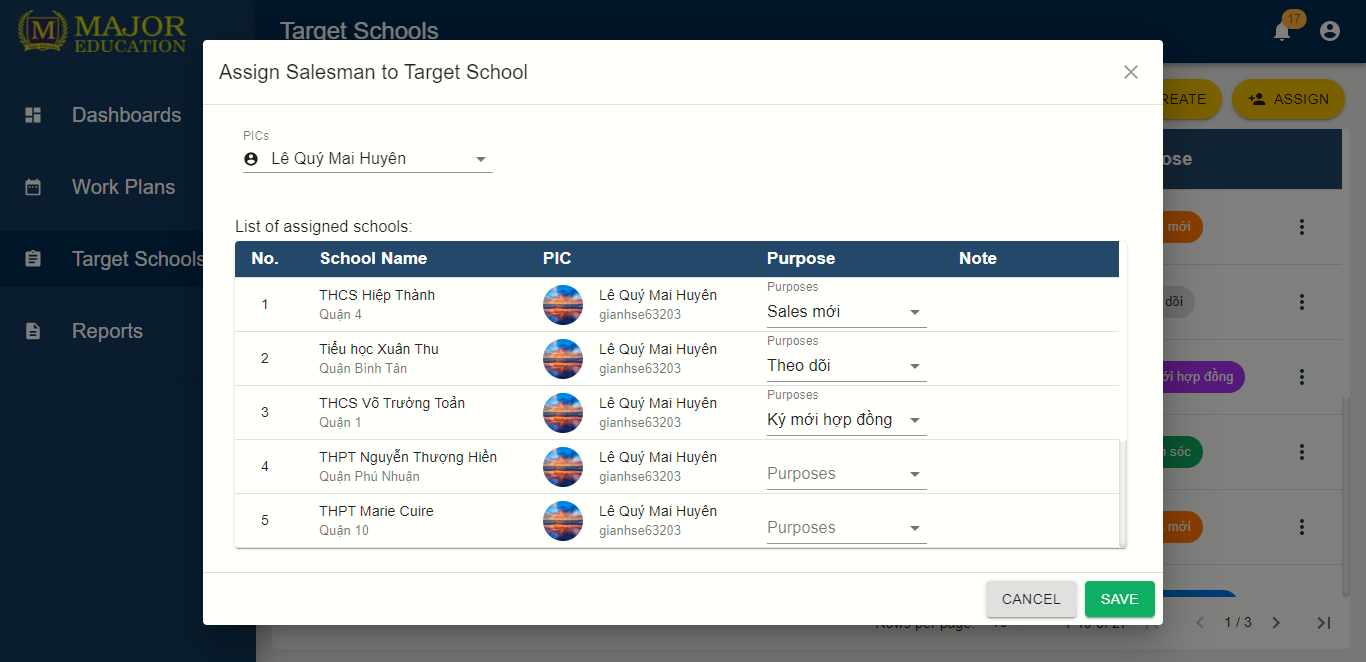
* View list of target shools
* View target school’s details
* Search target schools
* Create target schools
* Edit targets information
* Assign salesmen to target schools
* Remove target schools

User interface(s)

Graphical user interface, application

Description automatically generated

Target Schools list



Assign Salesmen to target schools

#### Reports

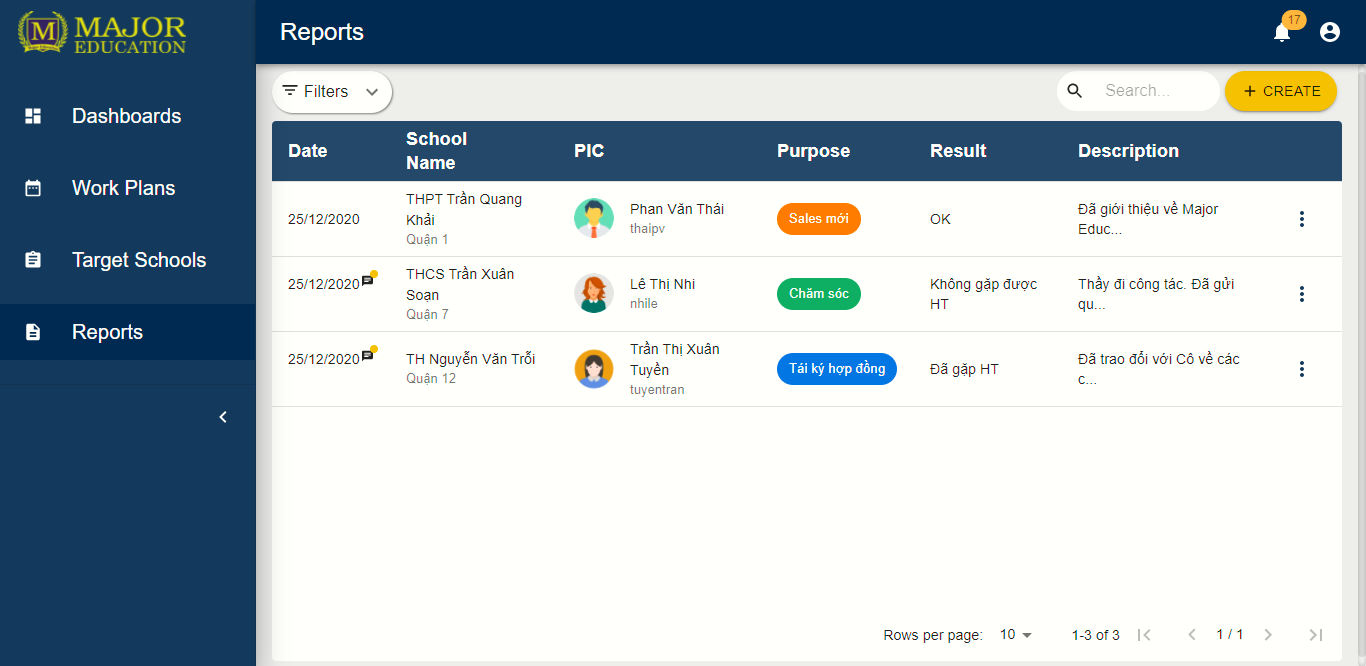
Purpose

* The sales manager manages personal daily reports
* The sales supervisor manages personal daily reports
* The salesman manages personal daily reports
* The sales manager comments on salesman’s report
* The sales department monitors their reports

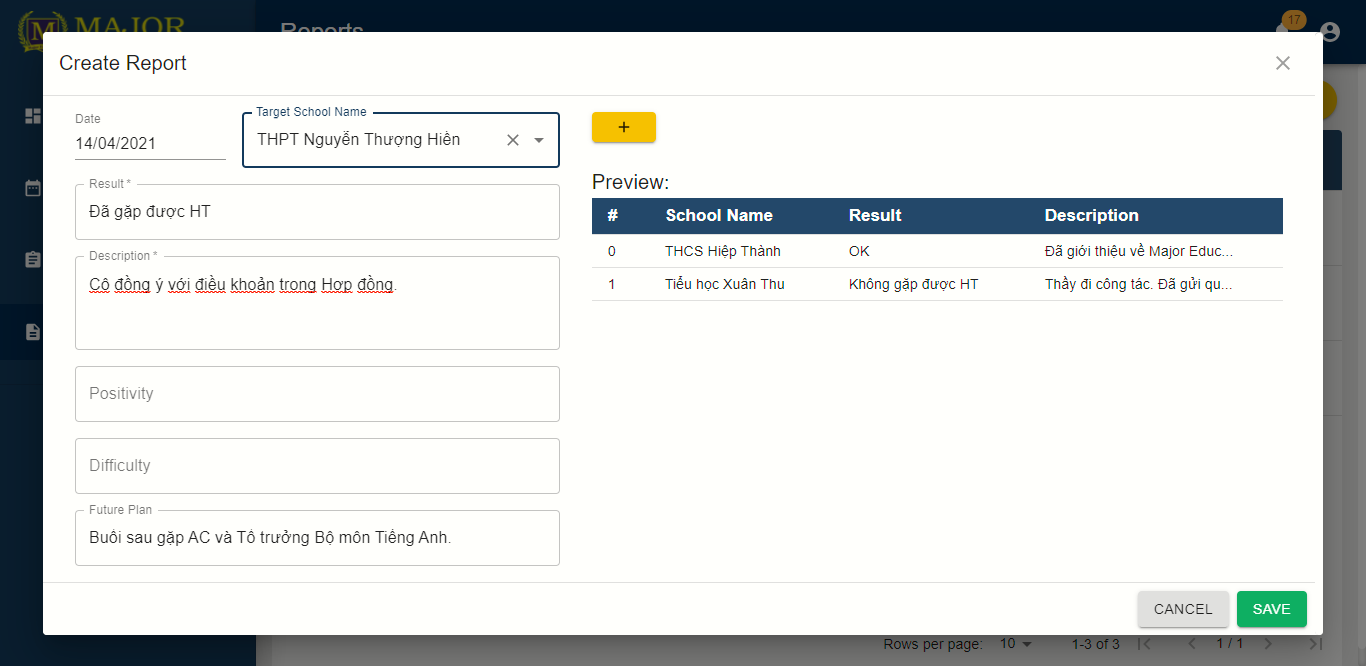
Functions

* View list of reports
* View report’s details
* Search reports
* Create reports
* Edit report's information
* Comment reports
* Remove reports

User interface(s)



Report list



Create Reports

#### Profiles

Purpose

* The user edits personal profile

Functions

* View profile
* Edit profile

User interface(s)

Graphical user interface, website

Description automatically generated

Profile

#### Workplans

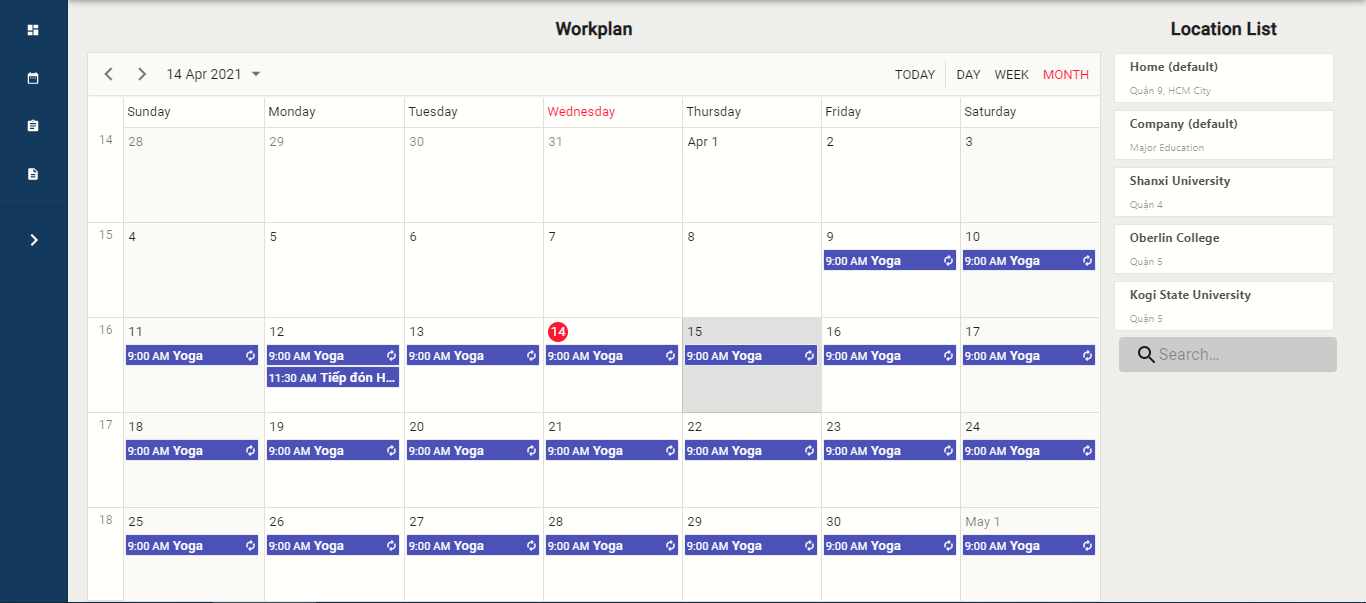
Purpose

* The user manages personal workplans

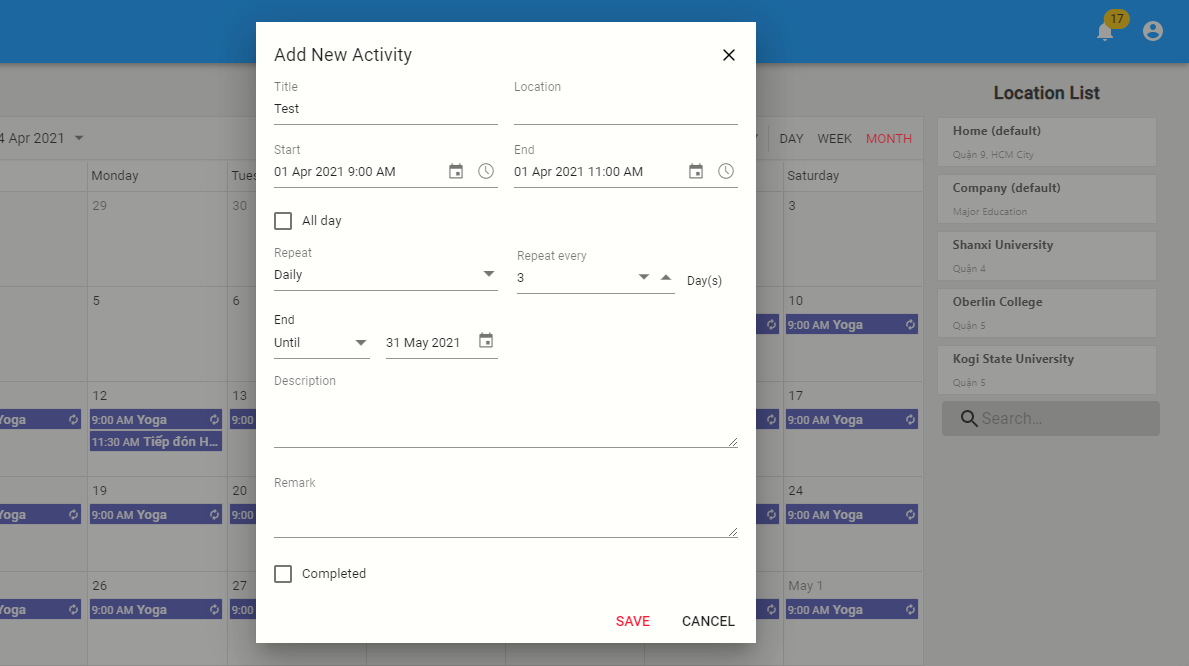
Functions

* View workplans
* View personal activities’s details
* Create personal activities
* Edit personal activities
* Remove personal activities

User interface(s)

******

Workplan

******

Add New Activity

#### Dashboards

Purpose

* The sales department tracks their work process

Functions

* View dashboards

User interface(s)

Graphical user interface, application

Description automatically generated

Dashboards

#### Contarcts

Purpose

* The sales manager manages contracts ?
* The salesman tracks assigned contarcts

Functions

* View list of contracts ?
* View contract’s details
* Search contracts ?
* Create contracts ?
* Edit contacrts’s information ?

User interface(s)

#### Logins

Purpose

* The unauthorized user logins the system
* The user retrieve their password

Functions

* Login with username and password
* Forget password

User interface(s)

Graphical user interface, application

Description automatically generated

Login

#### Logout

Purpose

* The authorized user logouts the system

Functions

* Logout

User interface(s)

Graphical user interface, application

Description automatically generated

Log out

### 3.4 Troubleshooting

1. ***Web App layout broke during cross-device use***

Problems:

* The application responsive interface has not supported for all devices

Workaround:

* Refresh the web app page
* Contacts the system developers for more supports

1. ***Avatar or images loading time is long***

Problems:

* The Internet connection is slow or unstable

Workaround:

* Refresh the web app page
* Connect with a better Internet connection

1. ***Data record loading time is long***

Problems:

* The Internet connection is slow or unstable

Workaround:

* Refresh the web app page
* Connect with a better Internet connection

# VII. Appendix

## 1. Glossary [Optional]

*[Place all definitions or abbreviation used in this document]*

## 2. References [Optional]

*[Place all referenced materials used in this document]*

## 3. Others [Optional]

*[Provide any other information here…]*