Software Requirement Specification

**for**

**ligevan**

**Version 1.0**

**Copied at 18/09/2024**

**Prepared by Group 1 - SE17C.NJS**

**FPT Quy Nhon AI Campus**

# Member

| **#** | **Full name** | **Student ID** | **Account** | **Contributor** |
| --- | --- | --- | --- | --- |
| 1 | Lê Minh Vương | QE170148 | VuongLM | 100% |
| 2 | Đinh Quốc Chương | QE170097 | ChuongDQ | 100% |
| 3 | Nguyễn Thị Thúy | QE170033 | ThuyNT | 100% |
| 4 | Hồ Trọng Nghĩa | QE170173 | NghiaHT | 100% |
| 5 | Lê Mạnh Hùng | QE170213 | HungLM | 100% |
| 6 | Lâm Mỹ Linh | QE170023 | LinhLM | 100% |

**Table of contents**

[**Member 2**](#_heading=h.u77jcw4x8z9n)

[**I. Introduction 5**](#_heading=h.kc5zee65uxgl)

[A. Purpose 5](#_heading=h.fiznoi7b068l)

[B. Scope 5](#_heading=h.d2y8o250e8u0)

[C. Definitions, Acronyms, and Abbreviations 5](#_heading=h.5ftrwfule5kj)

[D. Overview 5](#_heading=h.w2nqa2ediy3z)

[**II. Overall Description 5**](#_heading=h.bg14cxiceo37)

[A. Product Perspective 5](#_heading=h.wq1wc5piba5b)

[B. User Classes and Characteristics 5](#_heading=h.dsmu0nvfv1ku)

[C. Operating Environment 6](#_heading=h.dv10cjmlix8v)

[D. Design and Implementation Constraints 6](#_heading=h.idfupihjfwq0)

[E. Assumptions and Dependencies 6](#_heading=h.10dshuziwl4)

[**III. Specific Requirements 6**](#_heading=h.dsi6yex1mk6g)

[A. Functional Requirements 6](#_heading=h.n8slsi8i2zse)

[1. User Management 6](#_heading=h.8kkrn9nd6doc)

[2. Class Management 6](#_heading=h.kthdjca5m81w)

[3. Attendance Management 6](#_heading=h.pwoujw5cuai3)

[4. Salary and Fee Management 6](#_heading=h.ggm7954d0hnu)

[5. Communication and Notifications 6](#_heading=h.yx9d4ugr5cc1)

[B. Non-Functional Requirements 6](#_heading=h.hu3ajn2m8ks5)

[1. Performance Requirements 7](#_heading=h.vi7yiaiyl9mx)

[2. Security Requirements 7](#_heading=h.iaeoh63mm63o)

[3. Usability Requirements 7](#_heading=h.q89y2ucijspq)

[4. Availability 7](#_heading=h.9isha8nd56ne)

[5. Scalability 7](#_heading=h.j2z2xpa4wezr)

[**IV. External Interface Requirements 7**](#_heading=h.in6q0rya7t8m)

[A. User Interfaces 7](#_heading=h.c5wfj2qjnzmy)

[B. Hardware Interfaces 7](#_heading=h.wzu1xv4ezn4k)

[C. Software Interfaces 7](#_heading=h.g9zbj0b6sw25)

[**V. System Features 7**](#_heading=h.a2gc6sg8y6kz)

[A. Create and Manage Centers 7](#_heading=h.wwpezwl4a9lf)

[B. Teacher Management 7](#_heading=h.8fdllkcr42ic)

[C. Student Management 8](#_heading=h.qhu94pva7s30)

[D. Class Management 8](#_heading=h.iszd8r1zzglg)

[E. Attendance Tracking 8](#_heading=h.1mpq45mz4oje)

[F. Salary Calculation 8](#_heading=h.o18f5l1kbhf5)

[G. Fee Management 8](#_heading=h.6py6nao7o2n9)

[H. Notification System 8](#_heading=h.8n3xsuvdpldi)

[**VI. Other Non-Functional Requirements 8**](#_heading=h.wqz47yhjs3qq)

[A. Maintainability 8](#_heading=h.h574cd1hw5q)

[B. Portability 8](#_heading=h.hiaixrnmty54)

[C. Backup and Recovery 8](#_heading=h.ylarog2oia3q)

[**VII. Appendices 8**](#_heading=h.s0v0n8gsjt9k)

[A. Purpose 9](#_heading=h.votno0wbvvox)

[B. Scope 9](#_heading=h.49t3fc5s1e7y)

[C. Definitions, Acronyms, and Abbreviations 9](#_heading=h.38hgmcdoy1sx)

[D. Overview 9](#_heading=h.h2x59pq3yjon)

# Introduction

## Purpose

The purpose of this document is to specify the software requirements for a web application that manages English teaching centers. This system will provide functionality for managing multiple centers, teachers, students, and classes, as well as tracking attendance, calculating teacher salaries, and handling student fees.

## Scope

The system will be a web-based application accessible over the internet. The key users will be administrators, teachers, and students. The application will manage multiple English centers, allowing for the assignment of teachers and students to specific centers and classes. Additional functions include attendance tracking, salary and fee calculations, class reassignments, and notifications via phone messaging.

## Definitions, Acronyms, and Abbreviations

* **Center**: Refers to an English teaching center.
* **Teacher**: An instructor assigned to a center and responsible for handling classes.
* **Student**: A person enrolled in one or more classes at a center.
* **Class**: A group of students assigned to a specific teacher for learning purposes.
* **Attendance**: Record of whether a student or teacher was present for a specific class session.

## Overview

This document outlines the specific software requirements for developing a web-based application to manage English teaching centers. The system is designed to streamline the management of teachers, students, and classes while handling operational aspects like attendance tracking, salary calculation, and communication.

# Overall Description

## Product Perspective

This product is a standalone web-based application for managing English teaching centers. The system will include modules for:

* **User Management** (Teachers and Students)
* **Class Management**
* **Attendance Tracking**
* **Salary and Fee Calculations**
* **Communication and Notifications**

## User Classes and Characteristics

* **Administrators**: Manage centers, teachers, students, and classes. They handle the salary calculations and fee management.
* **Teachers**: Manage the classes assigned to them, track student attendance, and communicate with students.
* **Students**: Attend classes and receive notifications for fees and other communications.

## Operating Environment

* Web-based application accessible from modern web browsers.
* Accessible via desktops, laptops, tablets, and mobile devices.

## Design and Implementation Constraints

* The application must be built as a responsive web application.
* The application will be deployed on a public cloud server.
* Integration with Zalo API or similar messaging platforms for sending notifications.

## Assumptions and Dependencies

* The system assumes each user (teacher or student) has a valid phone number and email address.
* Zalo API or equivalent messaging system will be available for communication.

# Specific Requirements

## Functional Requirements

### User Management

* **Teacher Account Creation**: Administrators can create accounts for teachers and assign them to a specific center.
* **Student Account Creation**: Students can be registered using their phone number, and administrators assign them to a specific class.
* **Teacher Transfer**: Administrators can transfer teachers from one center to another.
* **Student Transfer**: Administrators can move students from one class to another.

### Class Management

* **Create and Assign Classes**: Administrators can create classes and assign teachers and students.
* **Class Enrollment**: Students can be assigned to one or more classes at the same center.

### Attendance Management

* **Track Attendance**: Both teachers and students' attendance will be tracked for each class session.
* **Attendance Reports**: Administrators and teachers can generate attendance reports for any given period.

### Salary and Fee Management

* **Salary Calculation**: The system calculates teacher salaries based on the actual number of classes attended.
* **Fee Calculation**: The system calculates fees for students based on the classes they are enrolled in.
* **Send Fee Notifications**: Email notifications with the fee breakdown will be sent to students.

### Communication and Notifications

* **Send Notifications via Zalo**: The system will send messages and notifications to students and teachers using their Zalo phone numbers.

## Non-Functional Requirements

### Performance Requirements

* The system should handle up to 100 concurrent users (teachers, students, administrators) without performance degradation.
* The system should respond within 2 seconds for all common user interactions.

### Security Requirements

* All sensitive user data, including passwords, should be encrypted.
* Role-based access control should be implemented to ensure that only authorized users (e.g., administrators) can modify critical data (e.g., teacher and student records).

### Usability Requirements

* The system must be user-friendly with intuitive navigation for teachers, students, and administrators.
* The system must be accessible on various devices (mobile, desktop, tablet).

### Availability

* The system must have 99.9% uptime and should be accessible at all times except during scheduled maintenance.

### Scalability

* The system must be scalable to support adding more centers, classes, teachers, and students without significant re-engineering.

# External Interface Requirements

## User Interfaces

* **Admin Dashboard**: For managing centers, teachers, students, and classes.
* **Teacher Portal**: For managing their assigned classes, viewing attendance, and communicating with students.
* **Student Portal**: For viewing assigned classes, attendance records, and fee details.

## Hardware Interfaces

* The system will be hosted on cloud infrastructure and accessed through client devices (PCs, tablets, smartphones) over the internet.

## Software Interfaces

* Integration with Zalo or other messaging APIs to send notifications.
* Integration with email services for fee notifications.

# System Features

## Create and Manage Centers

* Admins can create new centers and manage existing centers.
* Each center can have multiple teachers and classes.

## Teacher Management

* Admins can create teacher accounts, assign them to a center, and transfer them between centers.

## Student Management

* Admins can create student accounts using phone numbers and assign them to one or more classes.
* Admins can transfer students from one class to another.

## Class Management

* Admins can create and assign classes to specific teachers and students.

## Attendance Tracking

* Teachers and students' attendance will be tracked and reported.

## Salary Calculation

* The system will calculate teacher salaries based on the actual number of classes attended.

## Fee Management

* The system will calculate student fees and send email notifications to students.

## Notification System

* Notifications can be sent to students and teachers via Zalo or other supported messaging platforms.

# Other Non-Functional Requirements

## Maintainability

* The system must be built using modular and scalable architecture to allow future enhancements.

## Portability

* The system must be compatible with all modern web browsers.

## Backup and Recovery

* Regular backups should be performed to ensure data integrity, and a recovery plan should be in place in case of system failure.

# Appendices

* Detailed workflow diagrams for teacher and student management.
* User interface mockups for Admin, Teacher, and Student portals.

## 

## Purpose

* The purpose of this document is to specify the software requirements for a web application that manages English teaching centers. This system will provide functionality for managing multiple centers, teachers, students, and classes, as well as tracking attendance, calculating teacher salaries, and handling student fees.

## Scope

* The system will be a web-based application accessible over the internet. The key users will be administrators, teachers, and students. The application will manage multiple English centers, allowing for the assignment of teachers and students to specific centers and classes. Additional functions include attendance tracking, salary and fee calculations, class reassignments, and notifications via phone messaging.

## Definitions, Acronyms, and Abbreviations

## 

* **Center**: Refers to an English teaching center.
* **Teacher**: An instructor assigned to a center and responsible for handling classes.
* **Student**: A person enrolled in one or more classes at a center.
* **Class**: A group of students assigned to a specific teacher for learning purposes.
* **Attendance**: Record of whether a student or teacher was present for a specific class session.

## Overview

This document outlines the specific software requirements for developing a web-based application to manage English teaching centers. The system is designed to streamline the management of teachers, students, and classes while handling operational aspects like attendance tracking, salary calculation, and communication.



**2. Overall Description**

**2.1 Product Perspective**

This product is a standalone web-based application for managing English teaching centers. The system will include modules for:

* User Management (Teachers and Students)
* Class Management
* Attendance Tracking
* Salary and Fee Calculations
* Communication and Notifications

**2.2 User Classes and Characteristics**

* **Administrators**: Manage centers, teachers, students, and classes. They handle the salary calculations and fee management.
* **Teachers**: Manage the classes assigned to them, track student attendance, and communicate with students.
* **Students**: Attend classes and receive notifications for fees and other communications.

**2.3 Operating Environment**

* Web-based application accessible from modern web browsers.
* Accessible via desktops, laptops, tablets, and mobile devices.

**2.4 Design and Implementation Constraints**

* The application must be built as a responsive web application.
* The application will be deployed on a public cloud server.
* Integration with Zalo API or similar messaging platforms for sending notifications.

**2.5 Assumptions and Dependencies**

* The system assumes each user (teacher or student) has a valid phone number and email address.
* Zalo API or equivalent messaging system will be available for communication.

**3. Specific Requirements**

**3.1 Functional Requirements**

**3.1.1 User Management**

* **Teacher Account Creation**: Administrators can create accounts for teachers and assign them to a specific center.
* **Student Account Creation**: Students can be registered using their phone number, and administrators assign them to a specific class.
* **Teacher Transfer**: Administrators can transfer teachers from one center to another.
* **Student Transfer**: Administrators can move students from one class to another.

**3.1.2 Class Management**

* **Create and Assign Classes**: Administrators can create classes and assign teachers and students.
* **Class Enrollment**: Students can be assigned to one or more classes at the same center.

**3.1.3 Attendance Management**

* **Track Attendance**: Both teachers and students' attendance will be tracked for each class session.
* **Attendance Reports**: Administrators and teachers can generate attendance reports for any given period.

**3.1.4 Salary and Fee Management**

* **Salary Calculation**: The system calculates teacher salaries based on the actual number of classes attended.
* **Fee Calculation**: The system calculates fees for students based on the classes they are enrolled in.
* **Send Fee Notifications**: Email notifications with the fee breakdown will be sent to students.

**3.1.5 Communication and Notifications**

* **Send Notifications via Zalo**: The system will send messages and notifications to students and teachers using their Zalo phone numbers.

**3.2 Non-Functional Requirements**

**3.2.1 Performance Requirements**

* The system should handle up to 100 concurrent users (teachers, students, administrators) without performance degradation.
* The system should respond within 2 seconds for all common user interactions.

**3.2.2 Security Requirements**

* All sensitive user data, including passwords, should be encrypted.
* Role-based access control should be implemented to ensure that only authorized users (e.g., administrators) can modify critical data (e.g., teacher and student records).

**3.2.3 Usability Requirements**

* The system must be user-friendly with intuitive navigation for teachers, students, and administrators.
* The system must be accessible on various devices (mobile, desktop, tablet).

**3.2.4 Availability**

* The system must have 99.9% uptime and should be accessible at all times except during scheduled maintenance.

**3.2.5 Scalability**

* The system must be scalable to support adding more centers, classes, teachers, and students without significant re-engineering.

**4. External Interface Requirements**

**4.1 User Interfaces**

* **Admin Dashboard**: For managing centers, teachers, students, and classes.
* **Teacher Portal**: For managing their assigned classes, viewing attendance, and communicating with students.
* **Student Portal**: For viewing assigned classes, attendance records, and fee details.

**4.2 Hardware Interfaces**

* The system will be hosted on cloud infrastructure and accessed through client devices (PCs, tablets, smartphones) over the internet.

**4.3 Software Interfaces**

* Integration with Zalo or other messaging APIs to send notifications.
* Integration with email services for fee notifications.

**5. System Features**

**5.1 Create and Manage Centers**

* Admins can create new centers and manage existing centers.
* Each center can have multiple teachers and classes.

**5.2 Teacher Management**

* Admins can create teacher accounts, assign them to a center, and transfer them between centers.

**5.3 Student Management**

* Admins can create student accounts using phone numbers and assign them to one or more classes.
* Admins can transfer students from one class to another.

**5.4 Class Management**

* Admins can create and assign classes to specific teachers and students.

**5.5 Attendance Tracking**

* Teachers and students' attendance will be tracked and reported.

**5.6 Salary Calculation**

* The system will calculate teacher salaries based on the actual number of classes attended.

**5.7 Fee Management**

* The system will calculate student fees and send email notifications to students.

**5.8 Notification System**

* Notifications can be sent to students and teachers via Zalo or other supported messaging platforms.



**6. Other Non-Functional Requirements**

**6.1 Maintainability**

* The system must be built using modular and scalable architecture to allow future enhancements.

**6.2 Portability**

* The system must be compatible with all modern web browsers.

**6.3 Backup and Recovery**

* Regular backups should be performed to ensure data integrity, and a recovery plan should be in place in case of system failure.



**7. Appendices**

* Detailed workflow diagrams for teacher and student management.
* User interface mockups for Admin, Teacher, and Student portals.