

By: Yong Jie, Jia Yi, Jian Le, Denzyl,
Kaihang

TABLE OF CONTENTS

About the Project

What is EasyUni

01



Major Requirements

Our functional (and non-functional)
requirements

02



Use Cases

Discussion forum and Course view

03



04

Overview of System Design

A broad perspective of how our
website works



05

SE practices applied

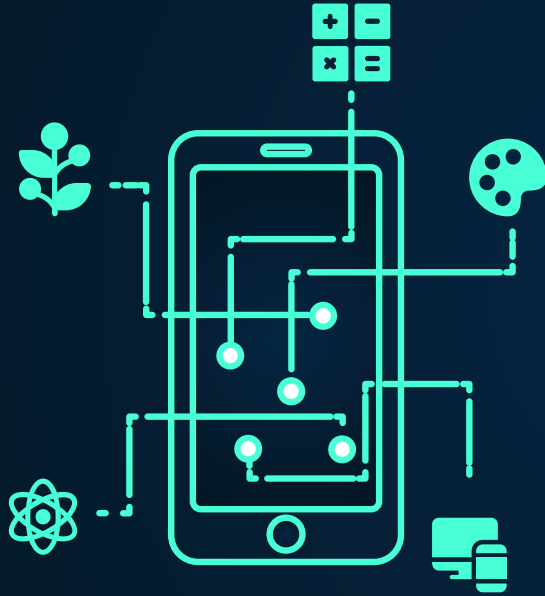
Good SE practices that will enable
us to further extend and build
upon



06

Demo

Live demo



ABOUT EASYUNI

A one-stop site that consolidates all the information for prospective university students, providing them some insight to aid in their university course choice.

MAJOR REQUIREMENTS



INFORMATION

Provision of accurate
and up-to-date
information



USER EXPERIENCE

Intuitive for new
users

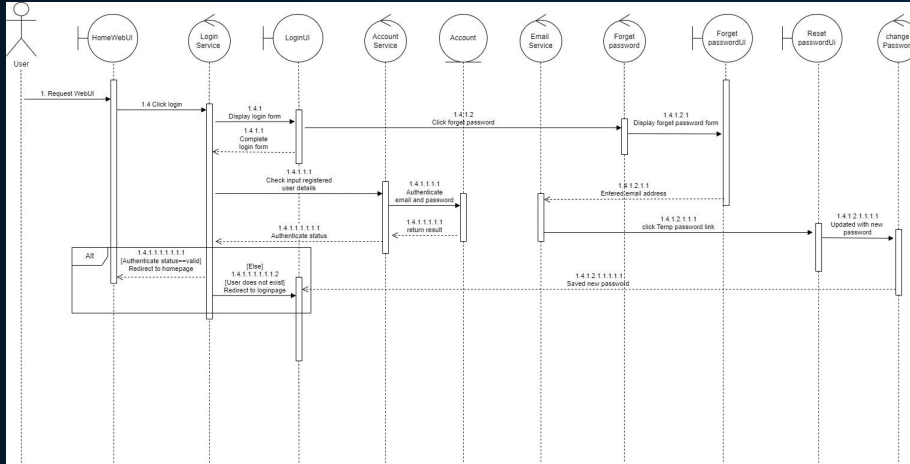


EXTENSIBILITY

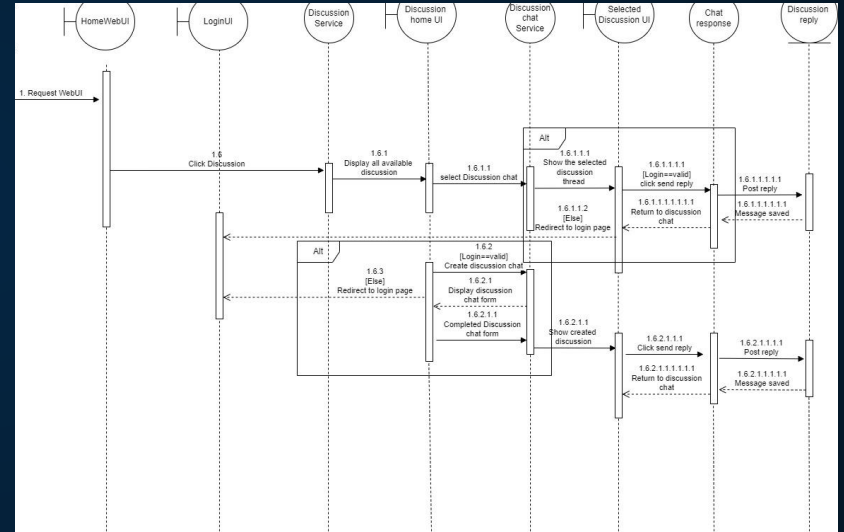
New features can be
easily added

Use Cases

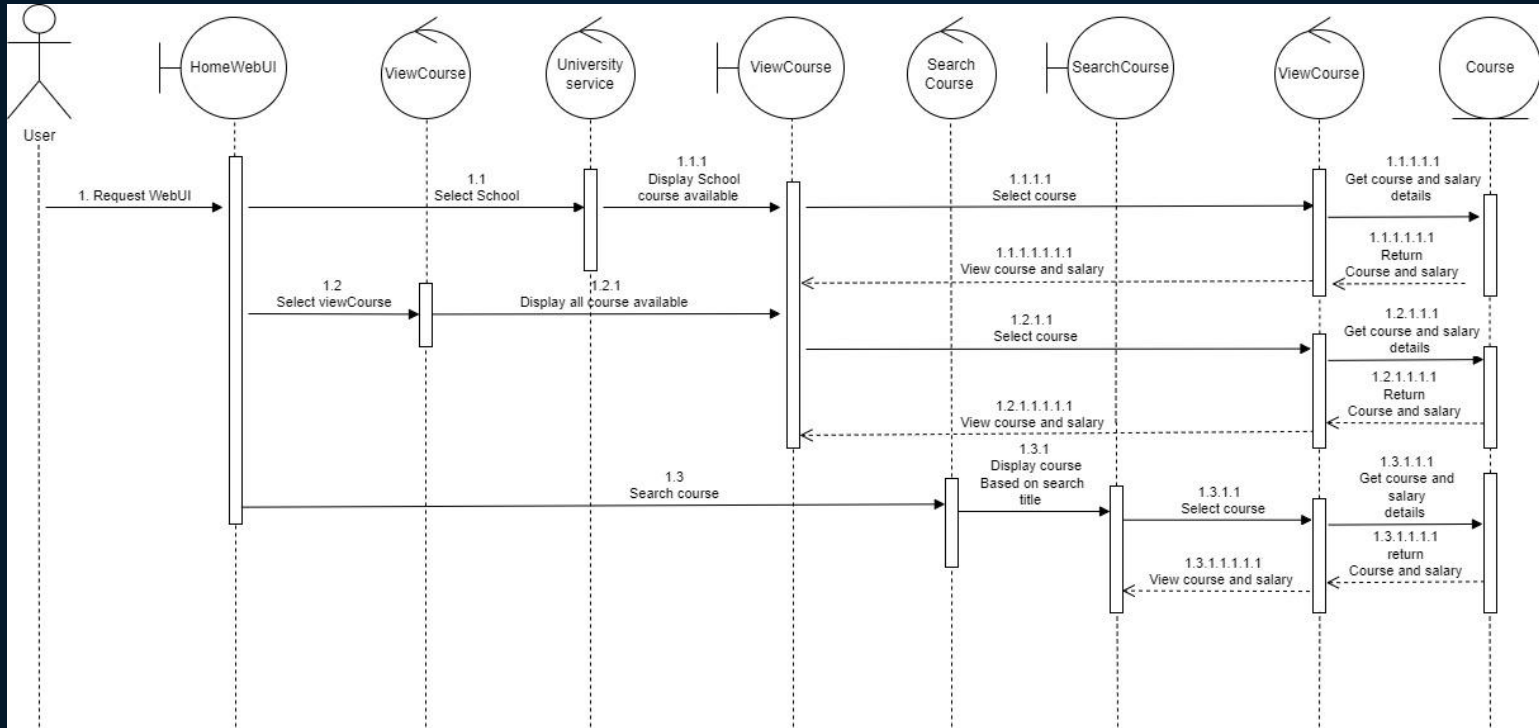
Login SD



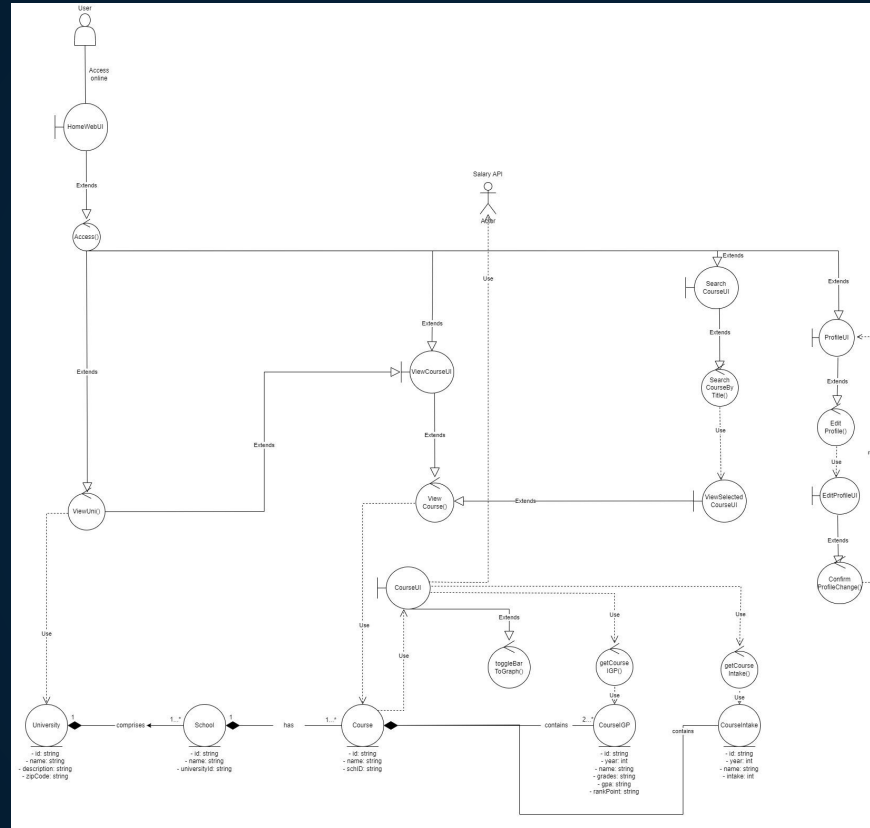
Discussion SD







Course Viewing Sequence Diagram



Course Viewing Class Diagram

System Design - Backend

Backend (RESTful API Server)

- Spring Boot
- Java 17 & Maven
- MongoDB (NoSQL)



System Design - Frontend

Frontend (Stateless)

- Vue3.js
- Bootstrap
- Javascript



Why Split Frontend & Backend?

- **Safeguarding** - Extremely low chances of breaking the entire web app when editing code as frontend and backend are separated.
- **Modularity** - Multiple developers can work on different modules without affecting each other's work.
- **Scalability** - Ease of upgrade/changing frameworks (e.g. changing Vue to React).

Software Eng Practices applied



Scrum Framework



MVC Architecture



Software Refactoring



Integrated Testing





Future Upgrades

Include other forms of Tertiary Education



Course Reviews



Map data



Live Demo