## Hardware & Software

### Hardware

As an MVP version system, the hardware supports the development, testing and deployment of MVP systems. We do not consider the hardware required for online services for the time being.

### Hardware parameters for deployment

Cloud platform: Azure

Product： EC2

CPU：2 Cores

Memory：4G

OS: Linux 64 bits

### Hardware parameters for development

Personal computer with the latest version Mac OS or Windows. We use java development language and support cross-platform. The hardware used for development depends on the team members.

### Software

#### Front end

#### Javascript

JavaScript (JS for short) is a lightweight, interpretive or just-in-time programming language with function priority. It is famous as a scripting language for developing web pages and supports object-oriented, imperative, declarative and functional programming paradigms. We use it as the main language of the front end.

#### Thymeleaf

We will use Thymeleaf as our front end framework. Thymeleaf is a template engine natively supported by spring boot. It supports JS to dynamically generate pages, which is conducive to development.

#### Back end

#### Java

Java can "compile once, execute everywhere". Java provides a lot of built-in class libraries. Through these class libraries, it simplifies the programming work of developers, shortens the development time of the project, and provides support for Web application development.

#### Spring

* 1. SpringBoot

Spring Boot is a tool that is closely integrated with the Spring framework to enhance the Spring developer experience. At the same time, it integrates a large number of commonly used third-party library configurations. Most Spring Boot applications require only a very small amount of configuration code (Java-based configuration), and developers can focus more on business logic.

1. SpringMVC

Spring MVC has strong flexibility, non-invasiveness and configuration ability, provides a simple and powerful JSP tag library, supports data binding, and makes it easier to write JSP pages. Spring MVC includes controllers, validators, command objects, model objects, handler mapping view parsers, etc. Each function is implemented by a special object.

#### Tomcat

Tomcat is an open-source web server. It is a small and lightweight application server. It is commonly used in small and medium-sized systems where there are not many concurrent users. It is the first choice for developing and debugging JSP programs. Tomcat occupies small system resources when running, has good scalability, and supports common functions of development application systems such as load balancing and mail services

#### MySQL

MySQL database is the small, fast, low total cost of ownership and is open source. It has a wide range of applications and can form a good development environment.

#### MyBatis

We will use the MyBatis ORM to provide the interface between the model on the application layer and the database layer. This Object-relational mapper (ORM) maps database tables to Java objects, which can be used to perform queries. SpringBoot natively supports MyBatis ORM, which is conducive to development.

#### Maven

Maven is a dependencies management tool that can build and manage Java projects. Our project uses Maven to manage dependencies to ensure uniform versions of system components and CI testing before deployment.

## Materials

#### Git

Git is a free and open-source distributed version control system designed to handle everything from small to very large projects with speed and efficiency. We would use it as a version management tool for branch management and coding collaboration. The repository would be deployed on GitLab.

#### Junit

JUnit can perform unit tests on the code and generate unit test reports based on the unit tests. After JUnit is started, the test is automated, and the execution result is judged without human intervention. We only need to view the final result to know whether the method interface of the entire project runs successfully.

#### Jeter

The Apache JMeter application is open-source software designed to load test functional behaviour and measure performance. We use it to complete stress tests, long-term operation tests and so on to ensure the reliability of the system

#### Shell/Python

We deploy the system on the cloud platform through shell or Python commands. The purpose of the rapid deployment is achieved by writing automated deployment scripts.

#### API document

We need API documents to guide the use of software to complete the development.