

**CAPSTONE PROJECT 1**

**Subjects Schedule And Recommendation System**

**ARCHITECTURE DOCUMENT**

Version 1.0

Date: Oct 30th, 2018

**MENTOR** **: MSc, Nguyen Thi Thanh Tam**

**PROJECT TEAM** : **Gentle Wind**

**TEAM’S MEMBER** : Phan Van Hoi

Nguyen Truong Giang

Trinh Quoc Vuong

Ho Viet Phu

**INTERNATIONAL SCHOOL**

**Project Information**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Project acronym** | SSRS | | | | |
| **Project Tittle** | Subjects Schedule And Recommendation System | | | | |
| **Start Date** | Aug 22, 2018 | | **End Date** | | Dec 10, 2018 |
| **Lead Institution** | International School, Duy Tan University | | | | |
| **Project Mentor & contact details** | Msc. Tam, Nguyen Thi Thanh  Email: [ttamdtu@gmail.com](mailto:ttamdtu@gmail.com)  Tel: 01664130007 | | | | |
| **Scrum Master & contact details** | Hoi, Phan Van  Email: [phanvanhoi.dtu@gmail.com](mailto:phanvanhoi.dtu@gmail.com)  Tel: 01206162700 | | | | |
| **Team members** | **Name** | **Email** | | **Tel** | |
| Giang, Nguyen Truong | [nguyengiangbmt123@gmail.com](mailto:nguyengiangbmt123@gmail.com) | | 01662732866 | |
| Phu, Ho Viet | hovietphu97[@gmail.com](mailto:thainguyenbahuy@gmail.com) | | 0933156407 | |
| Vuong, Trinh Quoc | [vuongltt@gmail.com](mailto:vuongltt@gmail.com) | | 01285044301 | |

|  |  |  |
| --- | --- | --- |
| Document History | | |
| **Version** | **Date** | **Comments** |
| V1.0 | Oct 15, 2018 | Draft for Document |
| V1.1 | Oct 30, 2018 | Update Design |

**Document Approvals**

The following signatures are required for approval of this document.

|  |  |  |
| --- | --- | --- |
| MSc. Tam, Nguyen Thi Thanh  *Mentor & Product Owner* |  | Date: Nov 3, 2018 |
| Hoi, Phan Van  *Scrum Master* |  | Date: Nov 3, 2018 |

Contents

[1. Introduction 6](#_Toc496347405)

[1.1 Purpose 6](#_Toc496347406)

[2. Project Overview 6](#_Toc496347407)

[2.1 Business needs 6](#_Toc496347408)

[2.2 Proposed solution 6](#_Toc496347409)

[2.3 Business drivers 6](#_Toc496347410)

[3. Architectural drivers 7](#_Toc496347411)

[3.1 Functional requirements 7](#_Toc496347412)

[3.2 Business constraints 7](#_Toc496347413)

[3.3 Technical constraints 7](#_Toc496347414)

[3.4 Quality Attribute 8](#_Toc496347415)

[3.5 Contex Diagram 9](#_Toc496347416)

[4. C&C view 10](#_Toc496347417)

[5. Module view 12](#_Toc496347418)

[6. Allocation view 15](#_Toc496347419)

1. **Introduction** 
   1. **Purpose**

This document will cover the following information:

* Brief description of the project (project overview, business goals, general constrains about technical and business problems).
* Architectural drivers (functional requirements, quality attributes and constraints).
* Architectural design (C&C View type, Module View type, Allocation View type).

1. **Project Overview** 
   1. **Business needs**

* View lecturer information.
* View status of each subject more easily.
* Recommend suitable subjects for each student.
* Registration Timetable Assumes: Registration timetable assumes module allows students registered many timetable assumptions. The system gives the right advice for each type of student. Students register for the course quickly.
  1. **Proposed solution**

To solve these problems, The Subjects Schedule And Recommendation System allows students to register multiple timetables assumes. Students can view an overview of their progress.Our team is based on each student object that proposes the subject in a scientific way.

* 1. **Business drivers**

Based on the business needs and business solution our team decides to make an Subjects Schedule And Recommendation System. This system will be developed by PC and Laptop.

* 1. **Project goal**

The goal of project is to build the Subjects Schedule And Recommendation System with four main features that meet business need within budget and schedule successfully:

For DTU's student who need a support system to registered and recommended subjects. The

Subjects Schedule And Recommendation System is a website that allows students to view

subject status more easily and students are better prepared for the learning process. Unlike

the current MYDTU system, registration of subject requires a lot of steps and does not have

timetables assumes function. The Subjects Schedule And Recommendation System has the

following functions:

1. ***Recommend Subject*:** The system can help students control and recommend clear and specific subjects.
2. **Statistics Of Subject Status:** The system will sort the subject according to its state: Pass , Fail, not learned
3. **Registration Timetable Assumes**: The system helps student registration multiple timetable assumes***.*** The subjects are recommend for each student
4. **Registration based on Timetable Assumes**: Instead of using the subject code. Based on the timetable assuming students just choose the registration is complete.
5. **Architectural drivers** 
   1. **Functional requirements**

Refer to [SSRS]UserStories v1.1 document of Team

* 1. **Business constraints**

Following are business constraints of the project:

- Project begins from August 22th, 2018 to December 10th, 2018. After delivery, the Gentle Wind team will rectify defects in the deliverables (no additional functionalities or features).

- Resource availability is defined below: 22/08 with 4 members.

- Product follows Mentor's requirement.

* 1. **Technical constraints**

Technical to develop

* Language: JSP/SERVLET, HTML5/CSS3, Javascript,
* Framework: Spring
* Operating system: Multi-platform
* Develop tool: Eclipse.
* Database Management System: SQL Server

Environment

* Web browsers: Internet Explorer, Mozilla Firefox, Apple Safari, Google Chrome, and so on.
* Operation systems: Mi-34rosoft Windows.
  1. **Quality Attribute**

|  |  |
| --- | --- |
| **Quality Attributes** : Availability | **ID** : QA01 |
| **Stimulus** | User turn off the device while server is running. |
| **Source(s) of the stimulus** | Administrator. |
| **Relevant environmental conditions** | Unexpected blackout. |
| **Architectural elements** | The system. |
| **System response** | The system will save the latest information and action that user did the last time |
| **Response measure(s)** | The system |

Table 1: Quality Attributes: Availability

|  |  |
| --- | --- |
| **Quality Attributes** : Performance | **ID** : QA02 |
| **Stimulus** | User want the system to respond their requests quickly |
| **Source(s) of the stimulus** | Administrator |
| **Relevant environmental conditions** | During the using procress. |
| **Architectural elements** | The system. |
| **System response** |  |
| **Response measure(s)** | The system takes only 3-5 seconds to respond Administrator requests. |

Table 2: Quality Attributes: Performance

|  |  |
| --- | --- |
| **Quality Attributes** : Security | **ID** : QA03 |
| **Stimulus** | User whose the contract has expired access the system. |
| **Source(s) of the stimulus** | Administrator |
| **Relevant environmental conditions** | During using process. |
| **Architectural elements** | Network, the system. |
| **System response** |  |
| **Response measure(s)** | User can only see the events but cannot make any changes. |

*Table 3: Quality Attributes: Security*

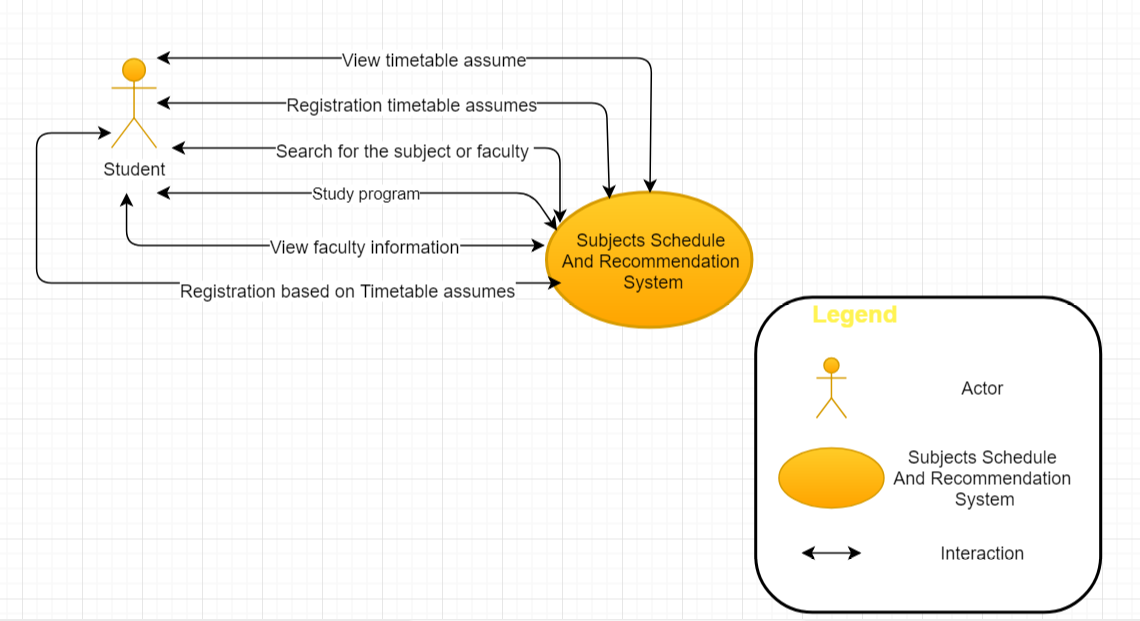
|  |  |
| --- | --- |
| **Quality Attributes** : Compatibility | **ID** : QA04 |
| **Stimulus** | Administrators use the system in many devices. |
| **Source(s) of the stimulus** | Administrator |
| **Relevant environmental conditions** | During the using procress. |
| **Architectural elements** | The system. |
| **System response** | Automatically change the size to fit with the device that administrator is using. |
| **Response measure(s)** | Users can user many different devices to access the system. |

Table 4: Quality Attributes: Capability

|  |  |
| --- | --- |
| **Quality Attributes** : Usability | **ID** : QA05 |
| **Stimulus** | Users can easily use the system at the first time. |
| **Source(s) of the stimulus** | Administrator |
| **Relevant environmental conditions** | During using process |
| **Architectural elements** | User guide, the system. |
| **System response** | Friendly user interface is presented |
| **Response measure(s)** | Users can use any functions of the system with just a few clicks and very easy to start using. |

Table 5:Quality Attributes: Usability

* 1. **Contex Diagram**



1. **C&C view**

We mainly used C&C view to argue and reason about architectural properties, quality attribute requirements, and functional requirements that the system must adhere to.

This view type partitions the system into components that have some runtime presence such as processes, objects, data stores and connectors or that represent pathways of communication such as information flows, and access to shared storage. The diagram below shows the architecture overview including Subjects Schedule And Recommendation System components and other related components. We have representations and behaviors for important components in the following sections.



Figure 4.1: C&C View

**Prose**

|  |  |
| --- | --- |
| **Element** | **Responsibilities** |
| Client/User | Client is a component which manages all of user inputs, and raises appropriate events: login/logout, managing accounts, managing contracts, managing contacts, managing events. |
| Database SQL | Database is a component which contains information of events, accounts, contacts, contracts, event types. All data the system needs. |
| Request/response | Call and return information from Web Service. |
| Query | Query is command for interacting with database. |
| Web Service | Web Service is a technology that allow to transfer data between Database and the system. |
| Web browser | Web browser is the environment to access and use the system. User can access the system on any devices with different sizes by using web browser. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Component** | **Provider** | **Role** | **Responsibility** |
| Web browser | User | Environment that runs the system. | Access the system. |
| Web Service | System | Transfering data between database and applications. | Transfer data. |
| Request/response | System | Communicating between applications and Web Service. | Send and receive information. |
| Query | System | Query command use in database | Return results. |

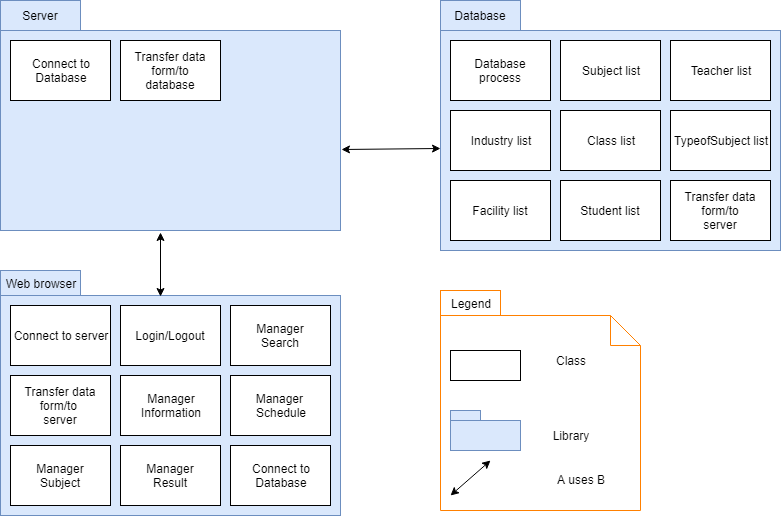
1. **Module view**

This view type partitions the system into a unique non-overlapping set of hierarchically decomposable implementation units (modules). The goal is to show how the source code is decomposed, as well as the dependencies between modules. In other words, it shows the decomposition and the “uses” relation.

The elements in this view type are UML packages or UML classes or interfaces. Packages are hierarchically decomposable; hence, placement of a class or package inside another package in a diagram implicitly indicates an is-part-of relation. This view type also shows dependencies among implementation units. It tells developers what other modules must exist in order for their portion of the system to work correctly.

This view type was selected because it helps the following roles:

* The project manager, who must define work assignments, form teams, and formulate project plans and schedule, knowing which modules are more critical in terms of dependencies.
* Testers who use the modules as their unit of work to create test cases and perform the tests.
* The configuration manager who is in charge of maintaining current and past versions of the units in consistent and functional package assemblies, being able to produce a running version of the system.
* Developers, who are required to implement the elements.
* Maintainers, who are tasked with modifying the software elements.

****

**Prose**

|  |  |
| --- | --- |
| **Element** | **Responsibilities** |
| Database | Database is a module contains some classes that implement the connections with database or processing data input/output of system. |
| Library | Library is a module contains some Interface libraries is used to build background of system |
| Server | Server is a module contains some classes and forms that implement functions to transfer data to *Database*. |
| Web browser | Web browser are modules that contain some classes and forms that implement functions to transfer data to Route. |

1. **Allocation view**

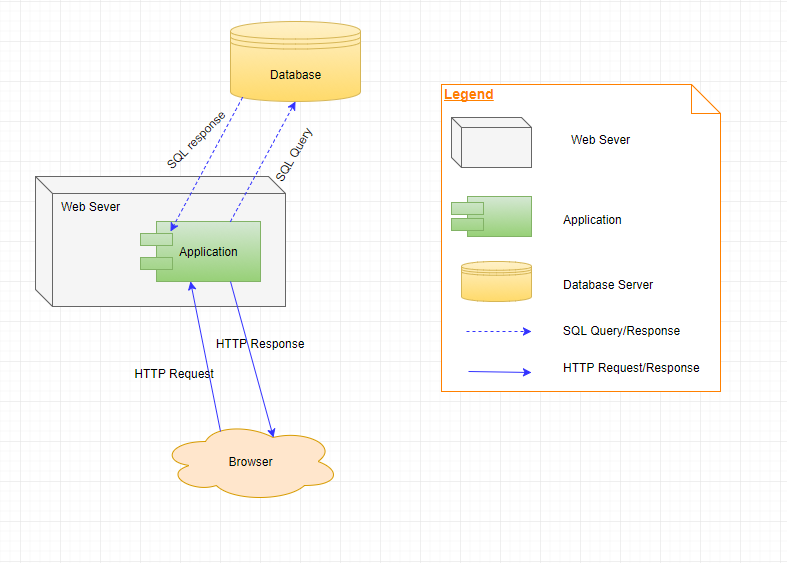


Figure 6.1: Allocation view of Subjects Schedule And Recommendation System

**Prose**

|  |  |
| --- | --- |
| **Element** | **Responsibilities** |
| Database | Database is a component of system that system can save and get data from it. |
| Web browser | Web browser is a component of system that users can view and update some important activities of the system. User must install a web browser to use the system. |
| Web Server | Application is a component that be stored in web server. Application will be called by server and response to Browser. |