**UNIVERSITY OF THE SOUTHERN CARIBBEAN**

**MARACAS ROYAL ROAD, MARACAS, ST. JOSEPH.**

Title of Assignment

SRS Document

An Assignment

Presented in Partial Fulfilment

of the Requirements for the Course

CPR 362: Software Engineering Group Project

INSTRUCTOR: Mr. George Mubita

By

Group: Re;

Abbas Abdul-Aala

Andell Alvarez

Stephan Seepersad

Whitney Shoy

Akeem Walters

Thursday, 19th April 2018

Approval………………...

Table of Contents

[1. Introduction 4](#_Toc511946599)

[1.1 Purpose 4](#_Toc511946600)

[1.2 Scope 4](#_Toc511946601)

[1.3 References 5](#_Toc511946602)

[2. System Description 6](#_Toc511946603)

[2.1 Constraints 6](#_Toc511946604)

[2.2 Assumptions and dependencies 7](#_Toc511946605)

[2.3 Apportioning of requirements 7](#_Toc511946606)

[3. Functional Requirements 7](#_Toc511946607)

[3.1 System Features 7](#_Toc511946608)

[3.1.1 System Feature 1 7](#_Toc511946609)

[3.1.2 System Feature 2 7](#_Toc511946610)

[3.2 Use Cases 8](#_Toc511946611)

[3.2.1 Use case Diagram 8](#_Toc511946612)

[3.2.2 Use Case 1 8](#_Toc511946613)

[3.2.3 Use Case 2 9](#_Toc511946614)

[3.2.4 Use Case 3 9](#_Toc511946615)

[3.3 Class Diagram 10](#_Toc511946616)

[4. External Interface Requirements 10](#_Toc511946617)

[4.1 COMMUNICATION INTERFACES 11](#_Toc511946618)

[5. Technical Requirements (Non-functional) 11](#_Toc511946619)

[5.1 Performance 11](#_Toc511946620)

[5.2 Scalability 12](#_Toc511946621)

[5.3 Security 12](#_Toc511946622)

[5.4 Maintainability 12](#_Toc511946623)

[5.5 Usability 13](#_Toc511946624)

[5.6 Multi Lingual Support 13](#_Toc511946625)

[5.7 Auditing and Lodging 13](#_Toc511946626)

[5.8 Availability 14](#_Toc511946627)

[6. Open Issues 14](#_Toc511946628)

[6.1 Issue 1 14](#_Toc511946629)

[6.2 Issue 2 14](#_Toc511946630)

[6.3 Issue 3 14](#_Toc511946631)

[6.4 Issue 4 14](#_Toc511946632)

# 1. Introduction

This section gives an overview about the SRS document. It also describes the purpose of this document.

## Purpose

The purpose of this document is to give a detailed description of the requirements for the Online/Blended Course Request System. It will illustrate the purpose and declaration for the development of the system. It will also explain system constraints and the interface of the system. This document is intended to be proposed to our customer, Mr. Rajkumar and Mr. Mubita for its approval and a reference for developing the first version of the system for USC (University of the Southern Caribbean.

## Scope

“The Online/Blended Course Request System” is a web base application that allows the process of requesting a course to be blended or online to be less strenuous. For a course to be online or blended, lecturers had to print a request form from online, fill out the from then deliver it to the Dean of the Department they are apart of. After, the Dean reviews the form, the Dean either accepts or denies the request. If the Dean confirms the request, the form is given to the Director of EIT where he signs-off of the request or rejects it. This system wishes to provide a better solution to this process.

Lecturers would be able to sign in using their ID and password, request a course to be online or blended and submit requests. The Dean would be able to receive the request moments after it was submitted by a lecturer and approve or deny request. If the request is approved by the Dean, the Director can approve the request as soon as he gets it thus, reducing the amount of time spent on this process.

The Web page of this system would be available on the University of the Southern Caribbean’s website where administrators, lecturers and the Director of EIT can easily access the system. The users must be connected to the internet to interact with the system. All information would be stored and retrieved from in the system’s database. The system is also able to generate a report showing the list of approved courses and course pending approval.

## References

*ASP Tutorial. (2018). W3schools.com. Retrieved 19 April 2018, from* [*https://www.w3schools.com/asp/default.asp*](https://www.w3schools.com/asp/default.asp)

*Create a web app with ASP.NET Core MVC on Windows with Visual Studio. (2018). Docs.microsoft.com. Retrieved 19 April 2018, from* [*https://docs.microsoft.com/en-us/aspnet/core/tutorials/first-mvc-app/?view=aspnetcore-2.1*](https://docs.microsoft.com/en-us/aspnet/core/tutorials/first-mvc-app/?view=aspnetcore-2.1)

*ASP.NET Web Examples in C# and VB. (2018). W3schools.com. Retrieved 19 April 2018, from* [*https://www.w3schools.com/asp/webpages\_examples.asp*](https://www.w3schools.com/asp/webpages_examples.asp)

*Writing Software Requirements Specifications (SRS) | TechWhirl. (2018). TechWhirl. Retrieved 19 April 2018, from* [*https://techwhirl.com/writing-software-requirements-specifications/*](https://techwhirl.com/writing-software-requirements-specifications/)

*IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.*

# 2. System Description

The system consists of a web portal which is use by lecturers, members of administration and the EIT Director. Lecturers would login and enter the course they wish to be blended or online. Administration would then accept or deny the request. The Director of EIT would then accept or deny the request to have the course online or blended.

While previous methods will require faculty staff and lecturers to document and file many different paper trails, the online system will require no paper formats. Many human errors such misplaced files or late approvals can be avoided with this new and improved online system. Lecturers can start their online/blended course on schedule and maximize their time. This system will be meeting the requirement of using “less paper” in planning and executing online and blended courses.

## 2.1 Constraints

A constraint with is system would be the internet connection. The user would not be able to use the system without internet.

## 2.2 Assumptions and dependencies

One assumption of this system is that it the University of the Southern Caribbean would have a backup database in case the contents of the original database is unintentionally deleted.

## 2.3 Apportioning of requirements

In the case that the project is delayed, or a specific functionality is left out, those requirements could be implemented in the following update and improved upon in later releases.

# 3. Functional Requirements

This section consists of the functional requirements of the System.

## 3.1 System Features

This software system will allow for a practical and improved method of regulating blended or online courses request by lecturers each semester at the University of the Southern Caribbean. It will shorten the time needed to complete the process and make the exchange of information between the lecturers, administration and the director of EIT faster with less hassle.

### 3.1.1 System Feature 1

A user-account based system that handles each lecturer’s information separately and allows them different levels of access.

### 3.1.2 System Feature 2

The simplified ability to send out information of online and blended courses to the public once approved.

## 3.2 Use Cases

This section consists of use cases of the system.

### 3.2.1 Use case Diagram

A close up of a map

Description generated with high confidence

### 3.2.2 Use Case 1

|  |  |
| --- | --- |
| **ID** | *Use Case 1* |
| **Description** | *Process for Requesting a course to be Blended or Online* |
| **Actor** | *Lecture,* |
| **Preconditions** | *Lecturer must be logged in* |
| **Basic Steps** | *Lecturer fills out online request form,* |

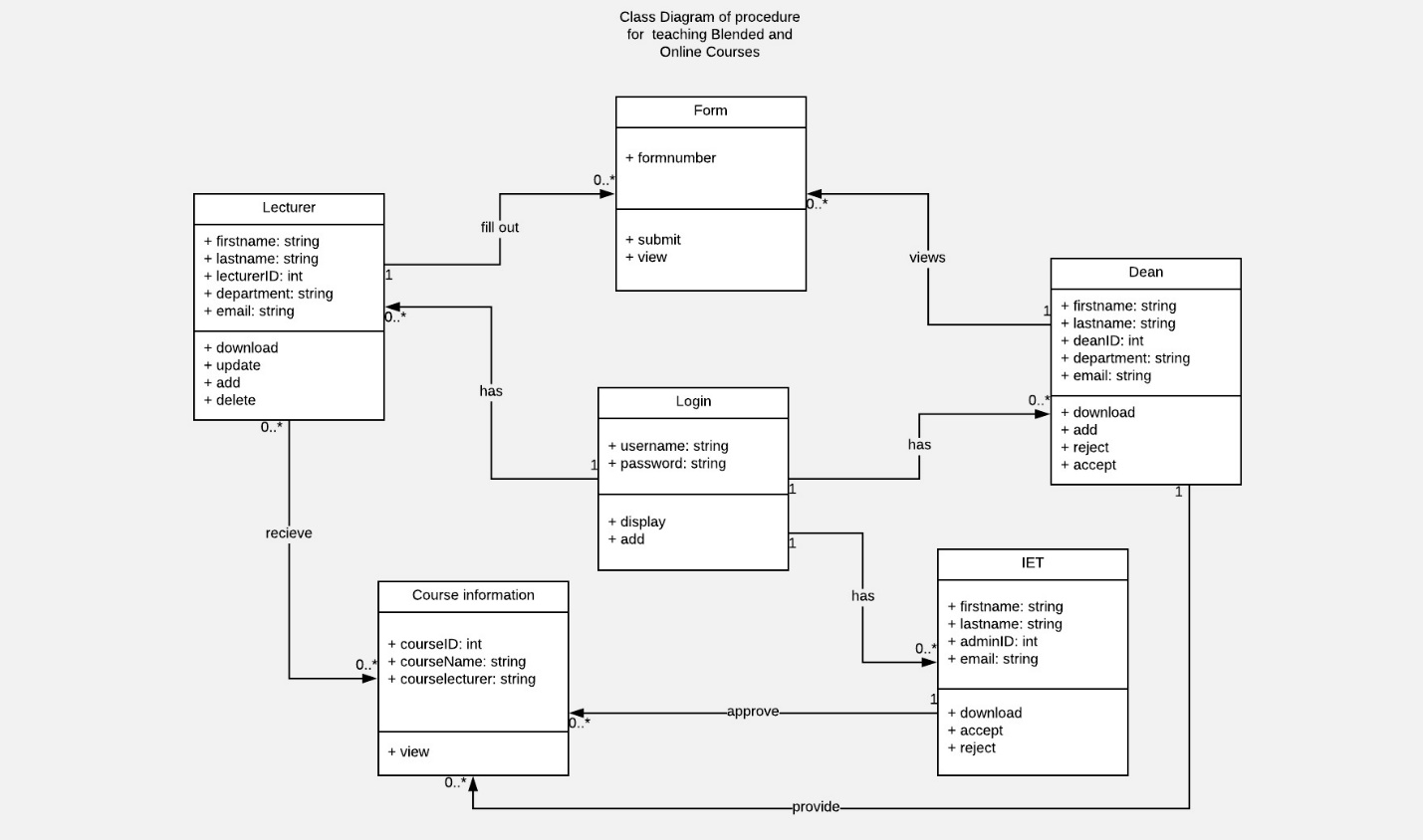
### 3.2.3 Use Case 2

|  |  |
| --- | --- |
| **ID** | *Use Case 2* |
| **Description** | *Admin accepting or denying request* |
| **Actor** | *Administration* |
| **Preconditions** | *Administration must be logged in* |
| **Basic Steps** | *Administration review’s request and accepts or denies request* |

### 3.2.4 Use Case 3

|  |  |
| --- | --- |
| **ID** | *Use Case 3* |
| **Description** | *EIT Accepting or denying request* |
| **Actor** | *EIT* |
| **Preconditions** | *Director must be logged in* |
| **Basic Steps** | *Director review’s request and accepts or denies request* |

## 3.3 Class Diagram



# 4. External Interface Requirements

USER INTERFACES

* Front-end software: Visual studio asp.net
* Back-end software: MySQL Database, C#

HARDWARE INTERFACES

* Windows.

SOFTWARE INTERFACES

|  |  |
| --- | --- |
| **Software used** | **Description** |
| Operating system | We have chosen Windows operating system for its best support and user-friendliness. |
| Database | To save the flight records, passengers records we have chosen MySQL database. |
| asp.net | To implement the project, we have chosen asp.net language for its more interactive support. |
| Visual Studio | Used to develop website. |

### 4.1 COMMUNICATION INTERFACES

This project supports all types of web browsers. We are using simple electronic forms for the user information data form, lecturer and course information etc.

# 5. Technical Requirements (Non-functional)

This section consists of the technical requirements of the system.

## 5.1 Performance

The performance of this database requires very little response time, as it uses very little resources of the CPU. This system puts very little strain on the system, therefore high-performance computer systems will not be required when running or accessing our website. Our system will have at most, two seconds of delays when a user tries to login or register. However, there will be no delays when dealing with form applications. When a user enters data, that data is saved and stored in less than seconds, making actual interaction and use of our system very quick and efficient. Delays may also occur when there are errors in data input or logging out the system. With our system, we try to aim at efficiency, we want users to have a quick and easy time inputting data and saving it.

## 5.2 Scalability

Our system should be able to handle a great number of users, including lecturers, administrators, various staff and directors. If ever need be our system can hold even more users, accommodating for all lecturers, admins, staff and directors. Our system is very simple as well and has room to expand and handle multiple transactions at once. Adjustments can be made to make our system even larger scaled. Our system can also run the most multiple lines or advanced SQL lines and give results.

## 5.3 Security

Data encryption is required for all data that is to be entered. Passwords will be required for all users. Very strict security measures will be implemented. Passwords will require 8 characters including at least one word, letter or symbol upon registering for an account or logging into an account. Data encryption will be implemented once the required data has been obtained, or data in general is added.

## 5.4 Maintainability

Maintaining this system should be very simple. No more than two people should be required to maintain the servers and database. Since we use SQL software such as my SQL and ASP.net, it will be very easy for anyone to easily access and maintain the system, whether it be through the code or through the servers. Maintenance will be required for the servers mostly, to keep them running. Our system is easily maintainable as it does not require the creator or original programmer to be around, allowing for our system to be used for a very long period of time before being outdated.

## 5.5 Usability

In terms of usability, our system aims to make our webpages look simply, as well as simple to use. Our system will take advantage of text fields, making it very clear to the users what information is required in each row, as well as utilizing drop boxes and check boxes where required.

## 5.6 Multi Lingual Support

In terms of lingual support, currently it does not have any other languages attached to it other than English. In the near future languages like Spanish and French would be added to our system with plans of adding more languages if necessary.

## 5.7 Auditing and Lodging

When it comes to auditing and logging, we plan to take logs of most if not all activity on our systems. The reason for this is so that we can avoid hacking attempts and it will be easier to detect suspicious activity. We can let users know of suspicious login’s or uploading of weird data.

## 5.8 Availability

Our system shall be available for access at any time. With twenty-four-hour availability, late night logins will be acceptable, even if there may not be a need to access our systems at that hour. Users can login at any time, fill out the forms at any time and search for data at any time. Different users from lecturers to directors can access our systems at any time of day. The only time our systems may be down is if we need to maintain the servers or if there is an extreme overflow of Big Data. In such instances of a system error, the user can simply refresh the system, and start over again, losing at most 30 characters of input.

# 6. Open Issues

This section consists of the issues with the current version of the system.

### 6.1 Issue 1

In its current state the system has a couple of issues. The check box in the Administration page is currently not working due to an error in the code thus; the code for the check box to work is commented out of the system.

### 6.2 Issue 2

Security needs to be updated before this system is implemented. Users can still see the password they entered in the login screen. Also, if the password is entered in one computer it shows the possible passwords in future logins

### 6.3 Issue 3

The director currently cannot enter any info. Any attempt to do so would result in an error.

### 6.4 Issue 4

Menu buttons are currently not working.