Software Requirement Specification

Alpha Curr

Members:
Lehlohonolo Motsi (1312548)
Thandi Tshabalala (1472222)
Masingita Rikhotso (1272188)
Tlotlang Sekgetho (1432706)
Muhammad Variawa (755427)

August 31, 2018

Project : Class Venue Allocation.

1 Product Scope

Our system is a Class Venue Allocation (CVA) that allows for Venues to be allocated for bookings and managed. A booking includes a course that will be taught the following year at the university, the diagonal, times, course type and class size. These bookings are submitted by lecturers or school administrators to the CVA. The various bookings that are submitted will then be allocated venues by PIMD utilising the CVA when the deadline has been reached.

The CVA will be a web-application and therefore should run on any supported web browser that has an online access. The database will stores a courses detail's, venue's details, booking's details as well as the allocation of a bookings details. Not only will the CVA allocate venues for the courses but will also

The benefits of the CVA are the following:
-
-
-
-

1.1 Purpose

The purpose of the project is to deliver quality software that will allow users to be able to log bookings. The software will then generate the allotted venue to accommodate the class on the behest of PIMD. The main goal is to provide quality software that is relevant and provides an accurate solution to accommodate the classes and the development towards a timetable/schedule.

1.2 Product Overview

Our product is a software product that will allow a user to log a booking for a class with the relevant class size. Thereafter at a set date PIMD will be able to let the system generate the allocation of venues to classes that have been booked. The product is used to target the current PIMD Admins, Lecturers, Course Coordinators and School Administrators. The product will maintain an ease-of-use interface to allow for scalability and traversal.

1.3 Product Function

1.4 User Classes and Characteristics

Our user class will be Lecturers, Course Coordinators, School Administrators and PIMD. PIMD will be the Super user and be the main focus as they are the major stakeholder.

1.5 Operating Environment

The operating environment will be majority of browsers as we will integrate support for most big browsers, the coding environment will be dependent on the section that is being developed. This means that certain sections can or may be developed in an environment of Visual Studio, NetBeans and Eclipse and the languages could possibly be C-Sharp, Java and Python. Travis are Coveralls will be integrated into our GitHub to allow for ease of testing and constant integration.

1.6 Assumptions and Dependencies

Our assumptions and dependencies include the fact that we will be receiving the databases of items for venues and courses from PIMD and the University. We assume that we are working with a user that is able to use a system and is trustworthy to do the correct thing. All changes that can be made to a Venue, Course, and User will be done by PIMD or the University and we will receive these changes as they do happen. We are utilising the LDAP and LAMP servers to obtain various other information and are restricted in terms of what we are able to do and obtain from these servers. To simulate this system we will create a scaled down database that will contain a few venues, courses and users. We will be scaling our system in terms of Courses and Venues to only be main campus.

2 Intended User and User Goals

2.1 Key High Level Goals and Problems

2.2 User Level Goals

2.3 Environment

Either at home, work, or anywhere with online access.

3 Constraints

3.1 Legal Issues

3.2 Implementation and Software

3.3 Hardware

Since our system is an online web-application, it is constrained by the response time and performance of the server and internet connection.

4 Pricing and Implementation Costs

5 Quality

5.1 Correctness

5.2 Reliability and Efficiency

5.3 Integrity

5.4 Usability and Portability

5.5 Maintainability, Flexibility and Reusability

5.6 Testability

6 External Interface Requirements

6.1 User Interfaces

6.2 Hardware Interfaces

6.3 Software Interfaces