|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Software Requirement Specifications**  [Student Portal for Azaan]  Version: [2.0]   |  |  | | --- | --- | | Supervisor | M. Asim Ali | | Co Supervisor (if any) |  | |  |  | | Project Team Members | Amirul Sunesara  Abu Bakar Siddiq  Aamir Shahzad | | Submission Date | 26/03/2016 | |

[Instructions]

* No section of template should be deleted. You can write ‘Not applicable’ if a section is not applicable to your project. But all sections must exist in the final document.
* All comments/examples mentioned in square brackets ([]) are in the template for explanation purposes and must be replaced / removed in final document.
* This’ Instruction’ section should also be removed in final document.

# Document History

[Revision history will be maintained to keep a track of changes done by anyone in the document.]

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Name of Person** | **Date** | **Description of change** |
| 1.0 | Abubakar Siddiq,  Amirul Sunesara,  Aamir Shahzad | 01/09/2015 | Documentation created according to Software engineering I assigned format. |
| 2.0 | Abubakar Siddiq,  Amirul Sunesara,  Aamir Shahzad | 20/03/2016 | Documentation re-created according to Software engineering II assigned format. |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Distribution List

[Following table will contain list of people whom the document will be distributed after every sign-off]

|  |  |  |
| --- | --- | --- |
| **Name** | **Role** | |
| M. Asim Ali | | Supervisor |
| N.A | | Co- Supervisor |
|  | |  |

# Document Sign-Off

[Following table will contain sign-off details of document. Once the document is prepared and revised, this should be signed-off by the sign-off authority.

Any subsequent changes in the document after the first sign-off should again get a formal sign-off by the authorities.]

|  |  |  |
| --- | --- | --- |
| **Version** | **Sign-off Authority** | **Sign-off Date** |
| 1.0 | Faria Jameel | 15/12/2015 |
| 2.0 | M. Asim Ali | 26/03/2016 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Table of Contents

Contents

[[Instructions] 2](#_Toc446510895)

[Document History 3](#_Toc446510896)

[Distribution List 4](#_Toc446510897)

[Document Sign-Off 5](#_Toc446510898)

[Table of Contents 6](#_Toc446510899)

[1. Introduction 8](#_Toc446510900)

[1.1 Purpose of Document 8](#_Toc446510901)

[1.2 Intended Audience 8](#_Toc446510902)

[1.3 Abbreviations 8](#_Toc446510903)

[1.4 Document Convention 9](#_Toc446510904)

[2. Overall System Description 10](#_Toc446510905)

[2.1 Project Background 10](#_Toc446510906)

[2.2 Project Scope 10](#_Toc446510907)

[2.3 Not In Scope 10](#_Toc446510908)

[2.4 Project Objectives 10](#_Toc446510909)

[2.5 Stakeholders 10](#_Toc446510910)

[2.6 Operating Environment 10](#_Toc446510911)

[2.7 System Constraints 10](#_Toc446510912)

[2.8 Assumptions & Dependencies 10](#_Toc446510913)

[3. Project Feasibility Analysis 11](#_Toc446510914)

[3.1 Economic Feasibility 11](#_Toc446510915)

[3.2 Technical Feasibility 11](#_Toc446510916)

[3.3 Operational Feasibility 11](#_Toc446510917)

[3.4 Schedule Feasibility 11](#_Toc446510918)

[3.5 Conclusion of Feasibility Analysis 11](#_Toc446510919)

[4. Primary and Secondary Research 12](#_Toc446510920)

[4.1 Primary Research 12](#_Toc446510921)

[4.1.1 Observation 12](#_Toc446510922)

[4.1.2 Interviews 12](#_Toc446510923)

[4.1.3 Rationale for Interviews 12](#_Toc446510924)

[4.2 Academic Research 12](#_Toc446510925)

[4.2.1 Development Methodology 12](#_Toc446510926)

[4.2.2 Development Tools (font end and Back end) 12](#_Toc446510927)

[4.2.3 Secondary Research 12](#_Toc446510928)

[5. Context Diagram 13](#_Toc446510929)

[6. External Interface Requirements 14](#_Toc446510930)

[6.1 Hardware Interfaces 14](#_Toc446510931)

[6.2 Software Interfaces 14](#_Toc446510932)

[6.3 Communications Interfaces 14](#_Toc446510933)

[7. Functional Requirements 15](#_Toc446510934)

[7.1 Functional Hierarchy 15](#_Toc446510935)

[7.2 List of all functional requirements 15](#_Toc446510936)

[7.3 List of Actors and Use Cases 15](#_Toc446510937)

[7.3.1 Use case diagram and use case description 15](#_Toc446510938)

[8. Proposed Objects List from Use Cases 16](#_Toc446510939)

[9. Initial Class Diagram 16](#_Toc446510940)

# Introduction

## 1.1 Purpose of Document

The product is a web portal to be built as a course management system for an institute called Azaan. This product aims to provide solution to institute by making its management of its most precious resources, student easier. Portal is built to bridge the gap between student and instructors and to make interaction between them easier. It also aims to provide the organization with a platform from where it can overlook its resources remotely.

## 1.2 Intended Audience

As its name states software resource specification is a made to clarify the requirements that are demanded by the client. It defines all the requirements of the client in every aspect. This document is made for the software developing team and is intended to be used by following people:

1. Designers

To get correct set of requirements on which they can design the software.

1. Developers(programmers)

It can be used as a reference to requirements.

1. Testing team

Testing team can use it to verify or match the end product with requirements.

1. Client

To verify that the requirement specifications match what he/she demanded.

To get a clear understanding of the document it is recommended to study it in its organized manner.

## 1.3 Abbreviations

[Describe the abbreviations use this document.]

## 1.4 Document Convention

|  |  |
| --- | --- |
| **Name** | **Properties** |
| Main Heading | Bold, Italic, Left aligned, Arial 16pt |
| Sub Heading | Bold, Italic, Left aligned, Arial 12pt |
| Text | Arial 12pt, Justified align |

# Overall System Description

## Project Background

[This section will establish business context in which system is being built. This will describe background information and will mention the actual problem / opportunity in business that triggered the project.]

## Project Scope

[This section will give an overview of project scope. This of project and will mention project boundaries and main functionalities that will be addressed in the system.]

## Not In Scope

[This section will highlight/explicitly mention the functionalities (if any) that are not in the scope of current project.]

## Project Objectives

**[This section will describe the objectives of project that how it is going to address the problem\opportunity identified in business environment and what would be the end result of project.]**

## Stakeholders

[This section will describe stakeholders of the system. This will include different business user classes that are expected to interact with system and similarly the technical people who are going to be involved in software development/management]

## Operating Environment

[Describe the environment in which the software will operate, including the hardware platform, operating system, network environment and other software components or applications with which it must coexist.]

## System Constraints

[Describe the constraints imposed on the system by the external environment. External environment may be caused by the stakeholders, business conditions, technical issues, academic requirements etc and may include the following:

* Software constraints
* Hardware constraints
* Cultural constraints (includes language etc.)
* Legal constraints
* Environmental constraints (e.g., the environment where the software will be installed, It could be a noisy environment, which may require that there is no sound event in the project).
* User constraints (e.g., the project is developed for children, so it may be required that the project has more graphic controls rather than textual controls).
* Off the shelf components that might be used in the project may have their constraints that are consequently transferred to the project.]

## Assumptions & Dependencies

[This section will identify:

* Any assumptions taken regarding the system or environment
* Any dependency of system on any external factor.]

# Project Feasibility Analysis

## Economic Feasibility

## Technical Feasibility

## Operational Feasibility

## Schedule Feasibility

## Conclusion of Feasibility Analysis

Write the feasibility report of your project here. You can get the idea of the feasibility report from book “System Analysis and Design Methods” by Whitten, Bentley and Dittman.

# Primary and Secondary Research

[Here you have mention how you were able to collect the data needed for your project. You will also specify which software methodology you have selected for your project and the reason behind it.]

## Primary Research

### Observation

### Interviews

### Rationale for Interviews

## Academic Research

### Development Methodology

#### Appropriate Methodologies available

#### Selection of Methodology

#### Methodology Reasoning

### Development Tools (font end and Back end)

#### Comparison of Different Tools

#### Selection of Appropriate Tool

* + - 1. **Development Tool Reasoning**

### Secondary Research

#### Books

#### Web URLs

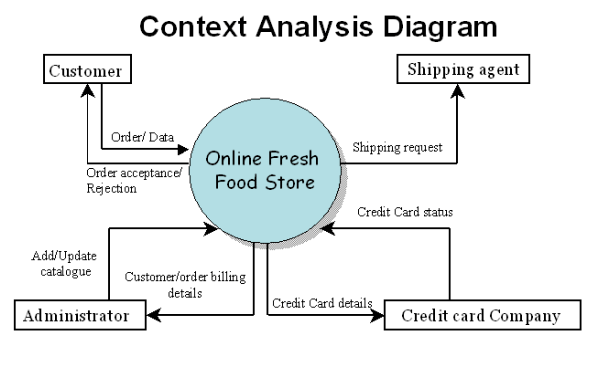
#### Periodicals

#### Journals

# Context Diagram

[Place a system level context diagram here. Below is an example of System Context diagram of an online Food Oder system.

]



# External Interface Requirements

[This section is intended to specify any requirements that ensure that the new system will connect properly to external components.]

## Hardware Interfaces

[Describe the characteristics of each interface between the software and hardware components of the system. This description might include the supported device types, the nature of the data and control interactions between the software and the hardware.]

## Software Interfaces

[Describe the connections between this system and other external software components (identified by name and version), including databases, operating systems, tools, libraries, and integrated commercial components. Identify and describe the purpose of the data items or messages exchanged among the software components. Describe the services needed and the nature of the inter-component communications. Identify data that will be shared across software components. ]

## Communications Interfaces

[Describe the requirements associated with any communication functions the system will use, including e-mail, web browser, network communications standards or protocols, electronic forms, and so on. Define any pertinent message formatting. Specify communication security or encryption issues, data transfer rates, and synchronization mechanisms.]

# Functional Requirements

## Functional Hierarchy

[This section will give a big picture of overall system functionality. The main modules/features of system and their sub-functions will be described here in the form of a functional hierarchy so that, before getting into the use case, audience could grab the idea of overall system functions.]

## List of all functional requirements

## List of Actors and Use Cases

### Use case diagram and use case description

[Use Case Diagram]

[Use Case Description]

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<1: Sign Up>** | | | | |
| **Use case Id:** | | 1 | | |
| **Actors:**  Student | | | | |
| **Feature:** Student Sign Up Procedure | | | | |
| **Pre-condition:** | | N.A | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | Students fills signup form | | | Successfully sign up, login details enter in database |
| **2.** | Students fills signup form with invalid inputs | | | Validation error |
|  |  | | |  |
| **Alternate Scenarios:** | | | | |
| **1a: N.A**    **2a:** Re-fill signup form with correct inputs | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1** | Student can login after signup process | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross referenced** | | | N.A | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<2: Instructor Sign Up>** | | | | |
| **Use case Id:** | | 2 | | |
| **Actors:**  Administrator | | | | |
| **Feature:** Creation of instructor’s account by Administrator authority | | | | |
| **Pre-condition:** | | Administrator must be logged in | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | Administrator fills correct signup form | | | Successfully sign up, login details enter in database |
| **2.** | Administrator fills signup form with invalid inputs | | | Validation error |
|  |  | | |  |
| **Alternate Scenarios:** | | | | |
| **1a:** N.A    **2a:** Re-fill the form with correct inputs | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1** | Instructor can login after signup process | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross referenced** | | | N.A | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<3: Add Course >** | | | | |
| **Use case Id:** | | 3 | | |
| **Actors:**  Administrator | | | | |
| **Feature:** Add Course and assign an instructor to it | | | | |
| **Pre-condition:** | | Instructor’s account must be created in order to assign course | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | Type course name | | | - |
| **2.** | Select instructor name from dropdown menu | | | Name is selected |
| **3.** | Submit | | | Instructor is successfully assigned with course |
| **Alternate Scenarios:** | | | | |
| **1a:** If invalid or empty course name is entered, validation error occurs.    **2a:** if instructor name is set to default option, validation error occurs. | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1** | Instructor will be able to view student enrollment. | | | |
| **2** | Instructor can upload marks. | | | |
|  |  | | | |
| **Use Case Cross referenced** | | | 2 | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<4: Compose/Delete Messages>** | | | | |
| **Use case Id:** | | 4 | | |
| **Actors:**  Instructor, Administrator, Student | | | | |
| **Feature:** User can send messages to each other and delete messages form inbox | | | | |
| **Pre-condition:** | | User must be logged in | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | Compose message | | | Message sent to receiver |
| **2.** | Delete message | | | Message deleted from inbox |
|  |  | | |  |
| **Alternate Scenarios:** | | | | |
| **1a:**Select a Person which categorize different users.  **1b:**Select name of person  **1c:** Enter Subject, message body  **1d:** Send Message  **1e:** Invalid Form will cause validation error  **2a:**Check Messages you wish to delete  **2b:**Delete Selected Messages | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1** | Inbox must be updated | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross referenced** | | | N.A | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<5: View Messages>** | | | | |
| **Use case Id:** | | 5 | | |
| **Actors:**  Student, Administrator, Instructor | | | | |
| **Feature:** Users can view inbox | | | | |
| **Pre-condition:** | | User must be logged in | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | View Message | | | Display Inbox |
|  |  | | |  |
|  |  | | |  |
| **Alternate Scenarios:** | | | | |
| **1a:** N.A | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **N.A** | N.A | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross referenced** | | | N.A | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<6: Register Course>** | | | | |
| **Use case Id:** | | 6 | | |
| **Actors:**  Student | | | | |
| **Feature:** Student register course that is offered by administration | | | | |
| **Pre-condition:** | | Instructor must be assigned with course | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | Check a course | | | Course checked |
| **2.** | Register course | | | Successful registration |
|  |  | | |  |
| **Alternate Scenarios:** | | | | |
| **1a:** If none is checked, validation error. Repeat step 1    **2a:** N.A | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1** | Students can send feedback after registration | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross referenced** | | | 3 | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<7: Watch Video Lectures>** | | | | |
| **Use case Id:** | | 7 | | |
| **Actors:**  Students | | | | |
| **Feature:** Students can watch lectures that is uploaded by instructors | | | | |
| **Pre-condition:** | | Lectures must be uploaded | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | Watch lectures | | | Display lecture |
|  |  | | |  |
|  |  | | |  |
| **Alternate Scenarios:** | | | | |
| **1a: N.A** | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **N.A** | N.A | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross referenced** | | | <Related use cases, which use or are used by this use case> | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<8:Send Feedback>** | | | | |
| **Use case Id:** | | 8 | | |
| **Actors:**  Students | | | | |
| **Feature:** Students can send feedback to instructors and administrators about course progress | | | | |
| **Pre-condition:** | | Students must be registered | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | Check appropriate remarks | | | Options checked |
| **2.** | Submit feedback | | | Feedback submitted |
|  |  | | |  |
| **Alternate Scenarios:** | | | | |
| **2a:** if an option is unchecked, validation error occurs | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **1** | Student cannot re-submit feedback | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross referenced** | | | N.A | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<9: Upload/View Marks>** | | | | |
| **Use case Id:** | | 9 | | |
| **Actors:**  Instructor, Student | | | | |
| **Feature:** After tests instructors can upload marks and student can view marks | | | | |
| **Pre-condition:** | | User must be logged in , student must be registered , instructor must be assigned with course | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | Instructor Upload the marks | | | Marks uploaded |
| **2.** | Student view secured marks | | | Display marks |
|  |  | | |  |
| **Alternate Scenarios:** | | | | |
| **1a:** Marks must be less than equal to 100 or greater equal to 0  **1b:** Validation error occurs if empty space if left or step 1a is not followed  **2a:** Notify if marks is not uploaded. | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **N.A** | N.A | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross referenced** | | | 3 | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<10: Upload/Delete Video Lectures>** | | | | |
| **Use case Id:** | | 10 | | |
| **Actors:**  Instructors | | | | |
| **Feature:** Instructors can upload video lectures that can be seen by students | | | | |
| **Pre-condition:** | | User must be logged in | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | Upload Lecture | | | Lecture uploaded |
| **2.** | Delete Lecture | | | Lecture Deleted |
|  |  | | |  |
| **Alternate Scenarios:** | | | | |
| **1a:** Enter name and details of video  **1b:**Choose video file path  **1c:** Any format of file is supported  **1d:** Validation error if above is not correctly input  **2a: N.A** | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **N.A** | N.A | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross referenced** | | | N.A | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<11: View Enrolled Students>** | | | | |
| **Use case Id:** | | 11 | | |
| **Actors:**  Instructor | | | | |
| **Feature:** Instructors can view enrolled students data in their course | | | | |
| **Pre-condition:** | | User must be logged in, instructors must be assigned with course | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | View Enrolled Students | | | Display Enrolled Students |
|  |  | | |  |
|  |  | | |  |
| **Alternate Scenarios:** | | | | |
| **1a:** Instructor can search students by their name | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **N.A** | N.A | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross referenced** | | | N.A | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<12: View Feedback>** | | | | |
| **Use case Id:** | | 12 | | |
| **Actors:**  Instructor, Administrators | | | | |
| **Feature:** Users can view statistics of feedback | | | | |
| **Pre-condition:** | | User must be logged in | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | View Feedback | | | Display feedback |
|  |  | | |  |
|  |  | | |  |
| **Alternate Scenarios:** | | | | |
| **1a: N.A** | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **N.A** | N.A | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross referenced** | | | N.A | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **<13: View Overall Enrollment>** | | | | |
| **Use case Id:** | | 13 | | |
| **Actors:**  Administrator | | | | |
| **Feature:** User can view students intended to participate in courses | | | | |
| **Pre-condition:** | | User must be logged in | | |
| **Scenarios** | | | | |
| **Step#** | **Action** | | | **Software Reaction** |
| **1.** | View Enrollment | | | Display enrolled students |
|  |  | | |  |
|  |  | | |  |
| **Alternate Scenarios:** | | | | |
| **1a:** User can also search for student | | | | |
| **Post Conditions** | | | | |
| **Step#** | **Description** | | | |
| **N.A** | N.A | | | |
|  |  | | | |
|  |  | | | |
| **Use Case Cross referenced** | | | N.A | |

# Proposed Objects List from Use Cases

1. Registration
2. Sign up
3. View Video lectures
4. Check marks
5. Give Feedback
6. Interact/message
7. Add/Delete instructor
8. Add new course
9. Review feedback
10. Upload marks
11. Upload lectures
12. View students enrolled
13. Communicate

# Initial Class Diagram

[With the help of identified objects develop an initial class diagram here which will be the base of your next design phase.]

1. List of Non-functional Requirements

[Here list all the non-functional requirements that you will implement in your project. For each non-functional requirement mention that how will implement that in coding or how will you provide that non-functional requirement in your project.]

* 1. Performance Requirements

[The performance characteristics of the system that are required by the business should be outlined in this section. Performance characteristics include the speed, precision, concurrency, capacity, safety, and reliability of the software. These characteristics define the performance of the project.]

* 1. Safety Requirements

[Specify the requirements that are concerned with possible loss, damage, or harm that could result from the use of the system. Define any safeguards or actions that must be taken, as well as potentially dangerous actions that must be prevented. Identify any safety certifications, policies, or regulations to which the system must conform.]

* 1. Security Requirements

[Specify any requirements regarding security, integrity, or privacy issues that affect the use of the system and protection of the data used or created by the system. Define all user authentication or authorization requirements, if any. Identify any security or privacy policies or certifications the system must satisfy.]

* 1. User Documentation

[List the user documentation components that will be delivered along with the software, such as user manuals, online help, context-sensitive help and tutorials.]

1. References

[This section should provide a complete list of all documents referenced at specific point in time. Each document should be identified by title, report number (if applicable), date, and publishing organization. Specify the sources from which the references can be obtained. (This section is like the bibliography in a published book).]

1. Appendices

[This section should include supporting detail that would be too distracting to include in the main body of the document.]