<Project Name>

System Design

<Version>

<Date>

<Your Name>

Prepared for

SE301 Software Engineering



Table of Contents

[1. Introduction 1](#_Toc433996772)

[1.1. Purpose of the System 1](#_Toc433996773)

[1.2. Design Goals 1](#_Toc433996774)

[1.3. Definitions, Acronyms, and Abbreviations 1](#_Toc433996775)

[1.4. References 1](#_Toc433996776)

[2. Current Software Architecture 1](#_Toc433996777)

[3. Proposed Software Architecture 1](#_Toc433996778)

[3.1. Overview 1](#_Toc433996779)

[3.2. System Decomposition 1](#_Toc433996780)

[3.3. Hardware Software Mapping 2](#_Toc433996781)

[3.4. Persistent Data Management 2](#_Toc433996782)

[3.5. Access Control and Security 2](#_Toc433996783)

[3.6. Global Software Control 2](#_Toc433996784)

[3.7. Boundary Conditions 2](#_Toc433996785)

[4. Subsystem Services 2](#_Toc433996786)

[5. References 2](#_Toc433996787)

SYSTEM DESIGN DOCUMENT

# Introduction

Design is the abstraction of an answer; it's the common description of the answer to an issue without a details. Design is view sample seen within the evaluation part to be a sample in a design part. After design part we are able to reduce the time required the implementation.

## Purpose of the System

As said within the Requirements Analysis Doc, the aim of the system is to offer system administrator, instructors, student and users with a central location for organizing varied quiz occasions. The aim of this technique supplies the next causes: No physical presence wanted for quizzes, no wastage of time throughout analysis, prompt availability of the outcomes, beginning and completion time are each recorded by the server, a collection of on-line assessments are provided by the software program for the good thing about the scholars. Normally the system is far more preferred as a consequence of these causes across various organizations and the world.

## Design Goals

The design objectives signify the specified qualities of quiz maker system and supply a constant set of standards that should be thought of when making design choices. Based on non-functionality requirements the next design targets must achieved as a way to qualify the system as profitable:

* **Security**

The system security is one of the most important non-functional requirements.

* **Reliability**

The system has to carry out the quiz maker operations with no errors. The application developed needs to be extremely dependable and safe in order that details about any questions and many others shouldn’t be leaked earlier than the precise quiz is held.

* **End User**

Students can take quiz only throughout the previously allotted time slots, however can open application anytime to access different data. System should be able to deal with a number of users. This system should run IOS operating system.

* **Performance**

The system has to be sturdy enough to manage any valid input from the users.

* **Flexibility:**

It requires Internet.

* **Implementation:**

System should be modified and readable. The number of errors must be minimized.

## Definitions, Acronyms, and Abbreviations

**Student**: The system user who will attends to the quiz. At the same time will be able to create the quiz.

**Admin**: The system administrator who will manage all data system data and user controls.

**Instructor**: The user who will prepare and present the questions to the students and calculate their scores by entering the answers of the questions.

**Registered User:** Only public quizzes are allowed. He/ She will be create the public quiz.

**Visitor:** Visitor can only register and than login in the application.

# Current Software Architecture

There are some websites and applications which have similar purposes as our system. These systems are allowing users to create or join some quizzes. The applications mentioned are iSpring QuizMaker[2], Easypromos Quiz[3] and Typeform Quiz[4], and the websites are Quiz-Maker[5] and SurveyMonkey[6]. iSpring QuizMaker is made for measuring the competency levels of staff members. Easypromos Quiz is made for testing the followers’ knowledge of a brand and products or a specific topic using a quiz. Typeform Quiz is based on business growth, personal or professional growth and it is designed to teachers or corporate trainers helping people with new knowledge and skills. The website Quiz-Maker is a site where you can start building your own quizzes at the first page you see, you can prepare your own quizzes and share it with anyone you’d like to share. SurveyMonkey is a website where you can prepare your own surveys and share it with the people you want, it can be used like our system but the main idea is different. There is no system like our QuizMaker System which is based on education and providing instructors to create lecture quizzes to the students which are related with those classes and also to the registered user to create quizzes by their own and to make people join to the quizzes they have prepared.

# Proposed Software Architecture

Documents the system design model of the new system.

## Overview

Present a bird’s-eye view of the software architecture and briefly describes the assignment of functionality to each subsystem.

## System Decomposition

Describe the decomposition into **subsystems and the responsibilities** of each. **This is the main product of system design.**

## Hardware Software Mapping

Describe how subsystems are assigned to hardware and off-the-shelf components. It also lists the issues introduced by multiple nodes and software reuse.

## Persistent Data Management

Describe the persistent data stored by the system and the data management infrastructure required for it. This section typically includes the description of **data schemes, the selection of a database, and the description of the encapsulation of the database**.

## Access Control and Security

Describe the user model of the system in terms of an access matrix. This section also describes security issues, such as the selection of an authentication mechanism, the use of encryption, and the management of keys.

## Global Software Control

Describe how the global software control is implemented. In particular, this section should describe how requests are initiated and how subsystems synchronize. This section should list and address synchronization and concurrency issues.

## Boundary Conditions

Describe the start-up, shutdown, and error behavior of the system. (If new use cases are discovered for system administration, these should be included in the requirements analysis document, not in this section.)

# Subsystem Services

Describe the **services provided by each subsystem**. Although this section is usually empty or incomplete in the first versions of the SDD, this section serves as a reference for teams for the boundaries between their subsystems. The interface of each subsystem is derived from this section and detailed in the Object Design Document.

# References

1. Bruegge B. & Dutoit A.H.. (2010). *Object-Oriented Software Engineering Using UML, Patterns, and Java*, Prentice Hall, 3rd ed.
2. <https://www.ispringsolutions.com/ispring-quizmaker>
3. <https://www.easypromosapp.com/quiz/>
4. <https://www.typeform.com/quizzes/>
5. <https://www.quiz-maker.com/>
6. <https://www.surveymonkey.com/mp/quiz/>