<Survey4All>

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[1]

The System Design Document (SDD) is written after the initial system decomposition is done, and updated throughout the development. SDD describes 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.

SDD is used to define interfaces between teams of developers and serve as a reference when architecture-level decisions need to be revisited. The audience for the SDD includes the project management, the system architects (i.e., the developers who participate in the system design), and the developers who design and implement each subsystem.

# Introduction

Provide a brief overview of the software architecture and the design goals. It also provides references to other documents and traceability information (e.g., related requirements analysis document, references to existing systems, constraints impacting the software architecture).

## Purpose of the System

## Design Goals

## Definitions, Acronyms, and Abbreviations

## References

References to existing systems, etc.

# Current Software Architecture

Describe the architecture of the system being replaced**. If there is no previous system**, this section can be replaced by **a survey of current architectures for similar systems**. The purpose of this section is to make explicit the background information that system architects used, their assumptions, and common issues the new system will address.

# 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.**

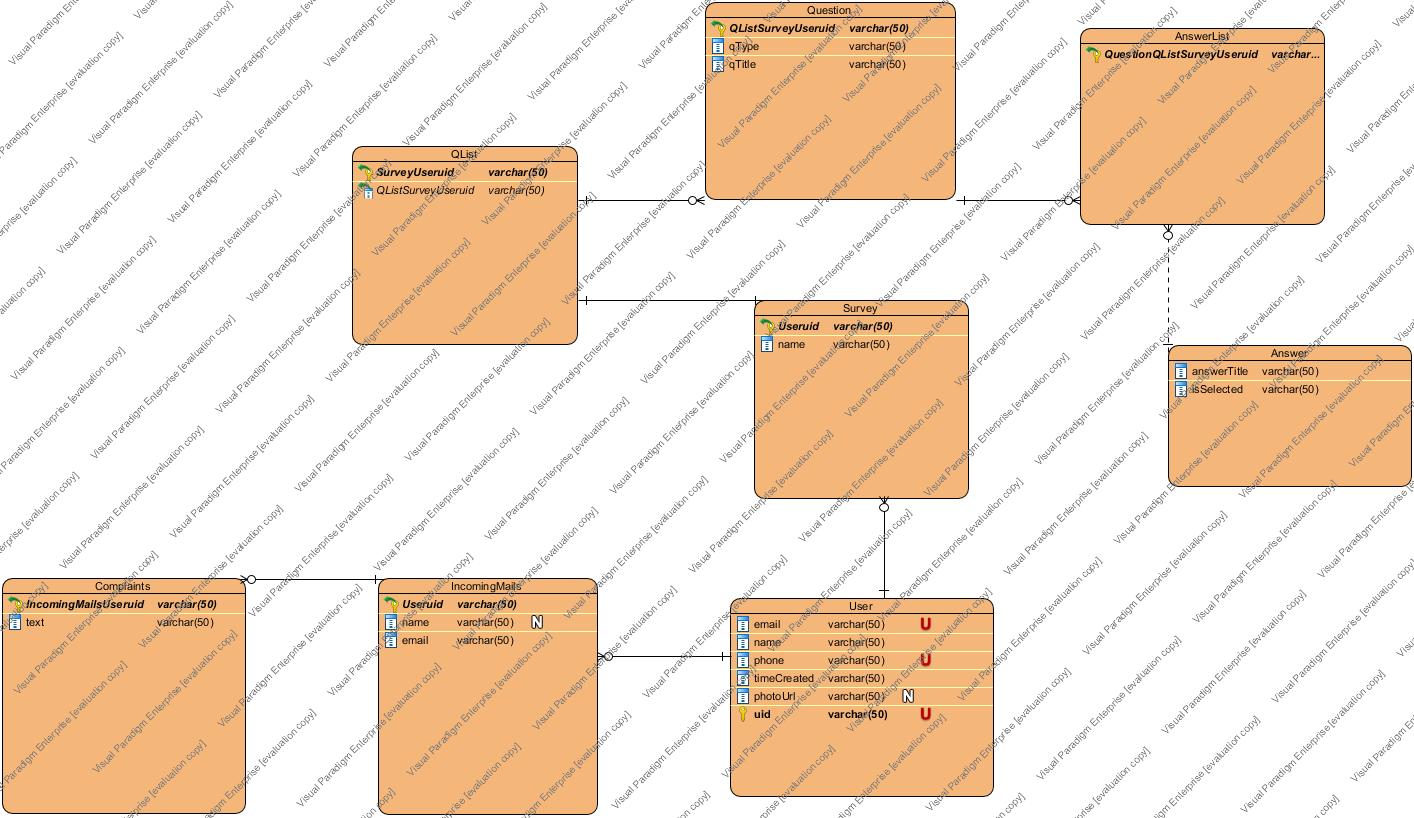
## Hardware Software Mapping

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

The system must needs persistent storage of users and their information , questions which created by user and answers which selected by user, their surveys, their complaints and their relationship each other and the data must be accessible from different users and every request should be returned as fast as possible.Also users can be controlled and we needs Real-Time Database that’s why using a Real-Time Database is must.

Security and growing system storage are the biggest concerns and there should not be much work with the server side.



The data scheme for the database is mainly made up of its users , its survey created by registered users , its questions and its answers and its complaints. The table ‘User’ holds information of user. The table ‘Survey’ holds name of survey and list of questions which created by registered user. The table ‘Question’ holds types of question and title of question and a list of answers for each question.The table ‘IncomingMails’ holds name and email of user who will submit a complaint and a list of complaints which submitted by user.

Database will provide query-independent access to data for the higher layers for encapsulation.

## Access Control and Security

|  |  |  |  |
| --- | --- | --- | --- |
|  | Administration | User | Survey |
| Admin | banUser()  viewUserInformation()  viewUserList()  viewComplaints() | login()  logout() | deleteSurvey()  shareSurvey() |
| RegisteredUser |  | login()  logout()  forgotPassword()  searchUser() | favoriteSurvey()  deleteSurvey()  unfavoriteSurvey()  searchSurvey()  fillSurvey()  shareSurvey()  createSurvey()  editSurvey()  checkSurveyStatictics()  checkSurvey() |
| UnRegisteredUser |  | register()  searchUser()  forgotPassword() | fillSurvey()  shareSurvey()  searchSurvey()  checkSurvey() |

Users must provide an at least 8 characters long password to register. The database shall hold the password information hashed. As the system is not designed to hold sensitive information for users, no other encryption is required.

## 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

Startup:

- First of all , admin is declared with Angular.

Register:

* Email is already used.
* Phone is already used.
* Password is less than 8 characters long.
* Password and password again does not match.
* There are blank fields.

Logging in:

* Email or password are wrong.
* Email or password fields are blank.
* Email or password don’t match each other.
* Password is less than 8 characters long.
* After the logged in , the main page does not appear.
* ForgotPassword button don’t send link.

Profile Management:

* There are blank fields.
* System crashes when user edits its information.
* User uploads wrong file type.

Survey Management:

* Title field is blank.
* User does not select question type.
* Answer field is blank.
* User save blank survey sketch.

Searching:

* User or survey which is searching are not exist .

# Subsystem Services

**4.1 Presentation Layer Subsystem Services**

**-** Takes user input from Interface Layer

**-** Passes user input from Interface Layer to Application Layer

**-** Displays Application Layer Results

**4.2 User Subsystem Services**

**-** Registration Service : User Registration.

**-** Login Service: User Login.

**-** Search Service : Searches users and their survey.

**4.3 Storage Subsystem Services**

**-** IncomingMail Service : sending complaints which user has to admin .

**-** SurveysData Service : Service which store, edit , delete survey information.

**-** RegisteredUser Data Service: Service which store, edit , delete user information.

**-** Firestore Service: Service of Firebase.

**4.4 Survey Subsystem Services**

**-** Create Survey Service: enables to create a survey by user.

- Delete Survey Service : enables to delete a survey by user.

- Edit Survey Service : enables to edit a survey by user.

# References

The following is an example of listing a book in this section. Check the text to see how it is cross referenced (The whole document is based on [1]).

1. Bruegge B. & Dutoit A.H.. (2010). *Object-Oriented Software Engineering Using UML, Patterns, and Java*, Prentice Hall, 3rd ed.