βExchange Software Design Specification

Matthew Gilligan

…

Section 1: Introduction

* 1. Describe the purpose of this document

Record all the design process and results.

* 1. Describe the scope of this document

System design specification and database design and web design.

* 1. Describe this document's intended audience

Software developer and programmers.

* 1. Identify the system/product using any applicable names and/or version numbers.

none

1.5 Provide references for any other pertinent documents such as:

• Related and/or companion documents

***Software requirement specification***

• Prerequisite documents

***Software requirement specification***

• Documents which provide background and/or context for this document

***Software requirement specification***

• Documents that result from this document (e.g. a test plan or a development plan)

***System implementation - Codes***

* 1. Define any important terms, acronyms, or abbreviations

***βExchange means book exchange in CSULA***

Section 2: System Overview

Provide a general description of the software system including its functionality and matters related to the overall system and its design.

***βExchange system will provide a web-based platform for students to exchange books. Students can register into the system and post the books they want to sell or exchange. Students can also browse available books. The system can display the picture of the books. The administrator can view and delete the users.***

Section 3: Design Considerations

3.1 Assumptions and Dependencies

***βExchange will run on Apache, PHP, MySQL platform. MySQL’s Version is above 5.0***

3.2 General Constraints

***The βExchange system will need Chrome browser. βExchange runs on Windows and Linux machines.***

3.3 Goals and Guidelines

***βExchange must be easy to learn for new users.***

3.4 Development Methods

***Agile method was used to develop the βExchange system.***

Section 4: ArchitecturaI Strategies

4.1 Use of a particular type of product (programming language, database, library, etc.)

***PHP***

***MySQL***

***CSS***

***HTML***

4.2 Reuse of existing software components to implement various parts/features of the system

***n/a***

4.3 Future plans for extending or enhancing the software

***//examples***

* ***Include credit card payment***
* ***Include paypal payment***
* ***Include system chatting, allow users to chat about he books.***

4.4 User interface paradigms (or system input and output models)

* ***Web based system***
  + ***Apache Web Server***
  + ***Chrome browser***
* ***HTML, CSS***

4.5 Hardware and/or software interface paradigms

* ***Web server Apache is used to support βExchange***
* ***HTTP is used as the protocol***
* ***Standard SQL is used to operate the database.***

4.6 Error detection and recovery

***If input user id and password do not match the system, the system will reject the access request.***

4.7 External databases and/or data storage management and persistence

* ***MySQL***

4.8 Distributed data or control over a network

* ***Web based system***

4.9 Concurrency and synchronization

* ***βExchange web based system use database to synchronize books’ and users’ information. This information is updated automatically when user post new books or change book’s information.***

4.10 Communication mechanisms

* ***http protocol***
* ***web based system***

Section 5: System Architecture

This section should provide a high-level overview of how the functionality and responsibilities of the system were partitioned and then assigned to subsystems or components.

5.1 ***User Register Subsystem***

This subsystem allows a new user to register to the system by using email address and password.

5.2 ***User login subsystem***

This subsystem allows an existing user to login to the system by using email address and password. If the user is not registered, the request will be rejected and the system will allow the user to re-try. The re-try times are unlimited

.

.

.

5.n component-n (subsystem-n) name or description

Section 6: DetaiIed System Design

6.1 ***User Register System***

• Definition

***This subsystem allows a new user to register to the system by using email address and password.***

• Responsibilities

***This subsystem allows a new user to provide email address and password to the system and store them into the database***

• Constraints

***This subsystem does not check the format of the email address and password. There is no limitation to the email address and password.***

• Uses/Interactions

***This subsystem provides the email address and password to database***

• Resources

***This subsystem interacts with database.***

• Processing

***This subsystem displays a web based form. After user inputs his/her email address and password, the user can click the submit button to submit the information to backend PHP program UserInsert.php. The UserInsert.php receives all the data: email address and password from the webpage and inserts all of them into database table: user.***

• Interface/Exports

***User information includes:***

* ***email address***
* ***password***

6.2 ***User Login System***

***…***

6.n

Section 7: GraphicaI User Interface Design

7.1 ***User Register System***

[picture of GUI]

7.2 ***User Login System***

…

7.n

Section 8: Data Base Design

8.1 Table User

|  |  |
| --- | --- |
| Email\_address | VarChar(100) |
| Password | VarChar(100) |

8.2 Table Books

Section 9: GIossary

An ordered list of defined terms and concepts used throughout the document.