|  |
| --- |
|  |
| Online Book Store For Students  Specification and Design |
|  |

|  |
| --- |
|  |

Document Acceptance and Release Notice

This document is authorised for release once all signatures have been obtained.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| PREPARED: |  | Date: |  | / |  | / |  |
| (for acceptance) | <Name>  Project Manager, Student Group |  |  | | | | |
|  |  |  |  | | | | |
| ACCEPTED: |  | Date: |  | / |  | / |  |
| (for release) | Dr Paul Darbyshire  Project Sponsor, VIT |  |  | | | | |

Table of Contents

[Executive Summary 7](file:///C:\Users\Dell%20E5430\Downloads\Real%20Estate%20Management%20Specification%20&amp;%20Design%20(1).docx#_Toc17800781)

[Specification 8](file:///C:\Users\Dell%20E5430\Downloads\Real%20Estate%20Management%20Specification%20&amp;%20Design%20(1).docx#_Toc17800782)

[System Description 9](file:///C:\Users\Dell%20E5430\Downloads\Real%20Estate%20Management%20Specification%20&amp;%20Design%20(1).docx#_Toc17800783)

[Feasibility Analysis 9](file:///C:\Users\Dell%20E5430\Downloads\Real%20Estate%20Management%20Specification%20&amp;%20Design%20(1).docx#_Toc17800784)

[Technical Feasibility 9](file:///C:\Users\Dell%20E5430\Downloads\Real%20Estate%20Management%20Specification%20&amp;%20Design%20(1).docx#_Toc17800785)

[Economic Feasibility 9](file:///C:\Users\Dell%20E5430\Downloads\Real%20Estate%20Management%20Specification%20&amp;%20Design%20(1).docx#_Toc17800786)

[Organizational Feasibility 10](file:///C:\Users\Dell%20E5430\Downloads\Real%20Estate%20Management%20Specification%20&amp;%20Design%20(1).docx#_Toc17800787)

[Requirements Specification 10](file:///C:\Users\Dell%20E5430\Downloads\Real%20Estate%20Management%20Specification%20&amp;%20Design%20(1).docx#_Toc17800788)

[Functional 10](file:///C:\Users\Dell%20E5430\Downloads\Real%20Estate%20Management%20Specification%20&amp;%20Design%20(1).docx#_Toc17800789)

[Non-functional 11](file:///C:\Users\Dell%20E5430\Downloads\Real%20Estate%20Management%20Specification%20&amp;%20Design%20(1).docx#_Toc17800790)

[Use Cases 12](file:///C:\Users\Dell%20E5430\Downloads\Real%20Estate%20Management%20Specification%20&amp;%20Design%20(1).docx#_Toc17800791)

[Use Case Diagrams 12](file:///C:\Users\Dell%20E5430\Downloads\Real%20Estate%20Management%20Specification%20&amp;%20Design%20(1).docx#_Toc17800792)

[Use Case Descriptions 15](file:///C:\Users\Dell%20E5430\Downloads\Real%20Estate%20Management%20Specification%20&amp;%20Design%20(1).docx#_Toc17800793)

[Context Model 15](file:///C:\Users\Dell%20E5430\Downloads\Real%20Estate%20Management%20Specification%20&amp;%20Design%20(1).docx#_Toc17800794)

[Design 15](file:///C:\Users\Dell%20E5430\Downloads\Real%20Estate%20Management%20Specification%20&amp;%20Design%20(1).docx#_Toc17800795)

[Architectural Design 16](file:///C:\Users\Dell%20E5430\Downloads\Real%20Estate%20Management%20Specification%20&amp;%20Design%20(1).docx#_Toc17800796)

[Database Structure 17](file:///C:\Users\Dell%20E5430\Downloads\Real%20Estate%20Management%20Specification%20&amp;%20Design%20(1).docx#_Toc17800798)

[Sequence Diagrams 19](file:///C:\Users\Dell%20E5430\Downloads\Real%20Estate%20Management%20Specification%20&amp;%20Design%20(1).docx#_Toc17800800)

[Interface Design 18](file:///C:\Users\Dell%20E5430\Downloads\Real%20Estate%20Management%20Specification%20&amp;%20Design%20(1).docx#_Toc17800799)

[State diagram 18](file:///C:\Users\Dell%20E5430\Downloads\Real%20Estate%20Management%20Specification%20&amp;%20Design%20(1).docx#_Toc17800799)

[Database Relational Diagram 18](file:///C:\Users\Dell%20E5430\Downloads\Real%20Estate%20Management%20Specification%20&amp;%20Design%20(1).docx#_Toc17800799)

# Executive Summary

The system resolves the issue of buying and searching books especially for students who can buy their course books from the website. Also, they can sell their books after the semester is ended. The website also offers books on charity basis so that the students who can not afford the books for study can get books from the students who sell the books on charity. There will be the online payment gateway to facilitate the students to pay by sitting at their homes. There will also be the cash on delivery method for the students.

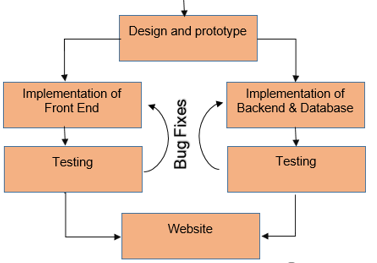
All these activities between seller and buyer will be monitored by the admin. An admin can be an independent entity or part of an organization.

The system will follow agile development methodology, and will be completed in 6 weeks. The activities that will take place are mentioned below:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Weeks | 1 | 2 | 3 | 4 | 5 | 6 |
| Web Development Team | Website design and prototype | | Implementation | | Testing | Bug fixes |
|  |  |  | | |  |
| Umbrella Activities | Documentation, Management, Quality Assurance, Reporting | | | | | |

The activities that will take place during development are demonstrated below:

Online Book Store



This document is mainly concerned with the Design and Prototype Phase.

From the E-Commerce perspective, the business policies will be called C2C (Consumer-to-Consumer). These policies may vary from organization to organization and are true business drives that will be implemented by application.

# Specification

The objectives behind the specifications are listed below.

1. To provide a user friendly and self-eliminatory interface.
2. The GUI must follow industry standards and trends.
3. The SEO should be optimized.
4. No irrelevant information should be provided.
5. Proper validations on front end.
6. The server must perform strict validations to avoid any possibility for application to miss behave.
7. The server must represent correct C2C business model.
8. The database must represent correct relationships.
9. The database must be normalized (preferable with BCNF, Boyce Codd Normal Form).
10. The database engine must support CCP (Concurrency Control Protocol).
11. The database must be able to store data at least for 10 years.
12. The database size must be utilized precisely.
13. The website should be free from bugs.
14. The website should be secure.
15. The website design should be responsive.

## **System Description**

The system is divided into three layers, following three tier architecture. The three layers are user interface layer, business logic layer and database layer. In the objectives discussed above, 1 to 5 are concerned with user interface layer, 6 to 8 with business logic layer and 9 to 13 with database layer.

The UI layer is concerned with implementing all the functionalities that will trigger events to be handled in business logic layer. These controls (buttons, search bar, forms, etc) must be properly validated because any improper action may cause huge business loss. The UI must be easy enough so that naïve person can also use it properly. The UI must be rendered at server to provide good SEO. For this reason, REST APIs will not be used. However, this approach will increase server cost.

The business logic layer or server layer is responsible to implement all the C2C business strategies and input/output processing to more data to/from database. To save time and cost it will be implemented in DotNet using C# programming language and Visual Studio 2017. By using this multi-threaded model users will get response time from servers. The multi-threaded nature of DotNet will cause high CPU cost but in return will give high processing, availability and usability.

The database layer is meant to store all logically related data. The design must be in conformance with C2C business logic and support scalability to meet any change or new requirement. For this reason, relational database (SQL database) will be used. Since multiple users will be using application simultaneously, therefore database must have concurrency, error checking and prioritization mechanism and great support for C# available. This is the reason why SQL Server will be used.

## **Feasibility Analysis**

The feasibility of project from different perspectives is proved below.

### **Technical Feasibility**

One may argue that why multithreaded technology is used when its cost is high. The reason is that, yes, in the initial state use of single threaded technology (like NodeJs) will cause low cost. But once system grows, the organization may need to build entire system again in multithreaded technology. This will make high consumption of resources in terms of time and money. Therefore, its preferred to choose multithreaded model at the first place.

The SQL server is preferred over no-SQL databases like MongoDB because SQL can adopt future requirements and its deployment cost is much less than MongoDB. Also, SQL Server is Secure than any other databases.

### **Economic Feasibility**

The development cost is very low, that is, 1500$, as justified in project proposal. The deployment cost can be controlled by using public cloud service, where organization will pay on the factor of usage (pay as you use model of cloud).

Once the project starts generating revenue and number of users increases, the project can be scaled up using same CSP (Cloud Service Provider).

The project will give economic benefits to the Admin of the website. The investment is only in development cost and hosting cost.

### **Organizational Feasibility**

The organization adopting this system will be completely feasible as initially it only needs to hire admins who will monitor the selling and purchasing activities and rest of the infrastructure needs will be fulfilled by CSP. Once the website grows then technical person will be needed to manage activities related to cloud.

## **Requirements Specification**

### **Functional**

The functional requirements are:

* **FOR STUDENT**

1. Students can Login and Register to the website.
2. Students can Upload their Books Picture, Title, Price, discount, Condition etc to sell.
3. Students can give their books in charity to needy students or any students.
4. Seller Students can Add discount if they want.
5. Sellers Students can Check charity option if they want.
6. Students can recover password if forgot.
7. Students can contact admin before registration for any query. Reply will be given through email.
8. Can change password and other profile information.
9. Buyer Students can search the Book with title.
10. Buyer Students can View All books.
11. Buyer Students can pay online by PayPal Account.
12. Buyer Students can buy charity books.
13. Buyer Students can get discount on some books.
14. Buyer Students can also avail feature of cash on delivery.
15. Buyer Students can search books by categories.

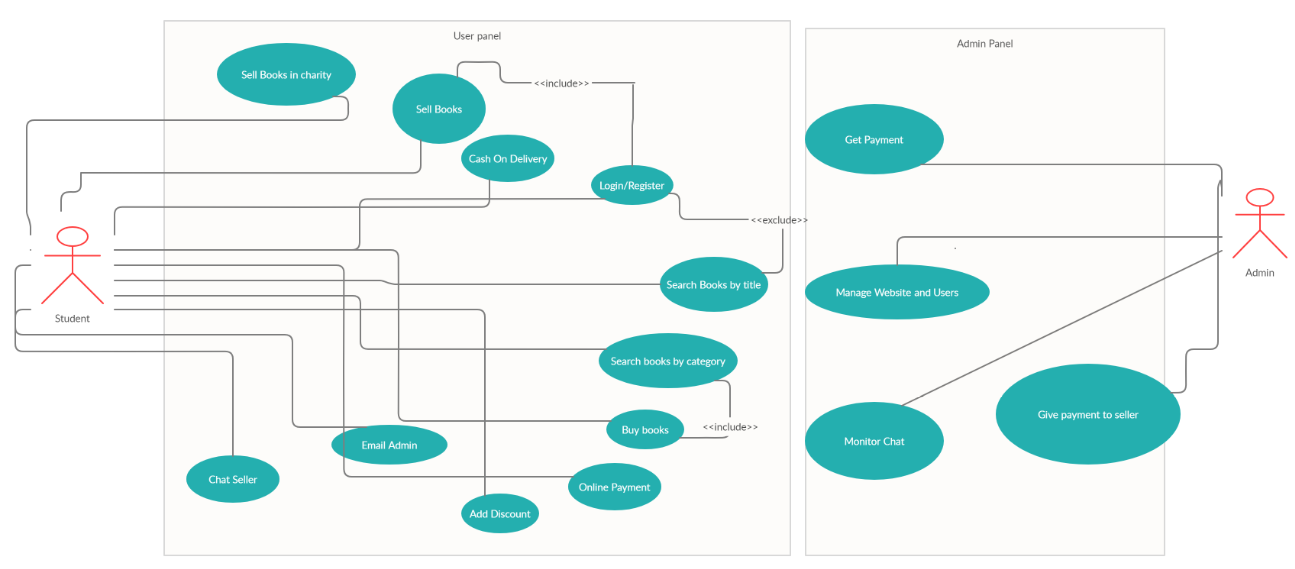
* **FOR ADMIN**

1. Admin can register another admin.
2. Admin can view and delete the Books.
3. Admin can delete and view the users.
4. The Payment will be given to the Admin’s Account then Admin will deduct commission and return the rest price to the sellers.
5. Admin can email users.
6. Admin can manage sales and complete website.

### **Non-functional**

* **Reliability** The system should be reliable and cannot crash during working.
* **Availability** The system should be available 24/7.
* **Security** The system should not allow unauthenticated users to enter into the system.
* **Maintainability** There will be no maintenance required for the software. The database is provided by the end-user and therefore is maintained by this user.
* **Portability** The system should not be limited to some networks. It should be portable we can access it anywhere anytime.

## **Use Cases**



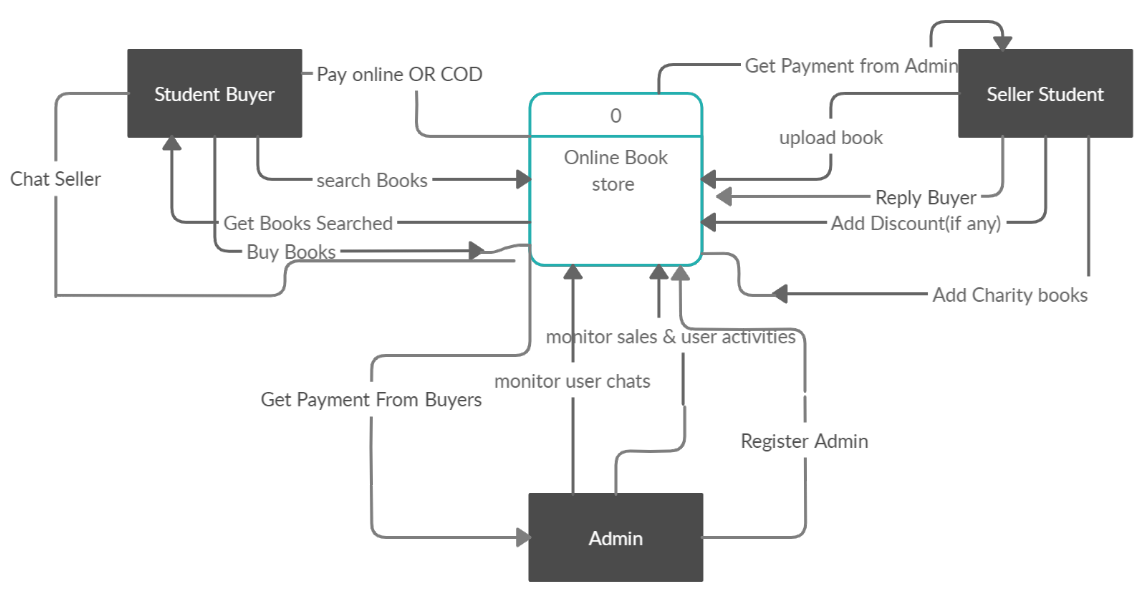
### **Use Case Descriptions**

Admin can monitor users chats as well as sales of the website. The payment of sold books will be transferred to Admin’s Account and then Admin will deduct some commission and transfer rest of the money in sellers Account.

Sellers can upload their property. They can give discounts if they want and also, they can give their books in charity.

Buyers can search the book by title as well as filter the books on the basis of books categories. They can give online payment through PayPal or they can give cash on delivery.

## **Context Model**



# Design

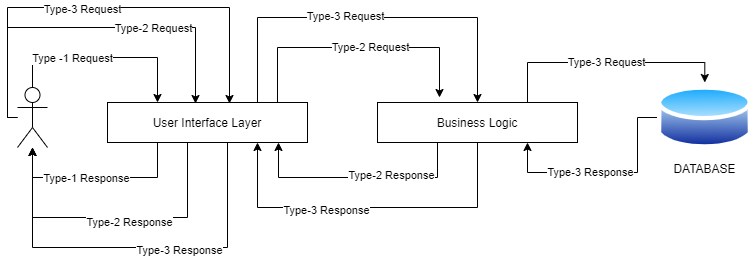
1. The assigned responsibilities to each layer should not mix.
2. No direct communication with database.
3. No unprocessed data pass to database.
4. UI must be server side rendered.
5. Each new request assigned to new thread in server.
6. User can only interact with UI layer.
7. Request-Response protocol will be used
8. User can generate 3 types of quires:

Type-1 that will be handled in UI layer like moving back, visiting pages from cache memory etc

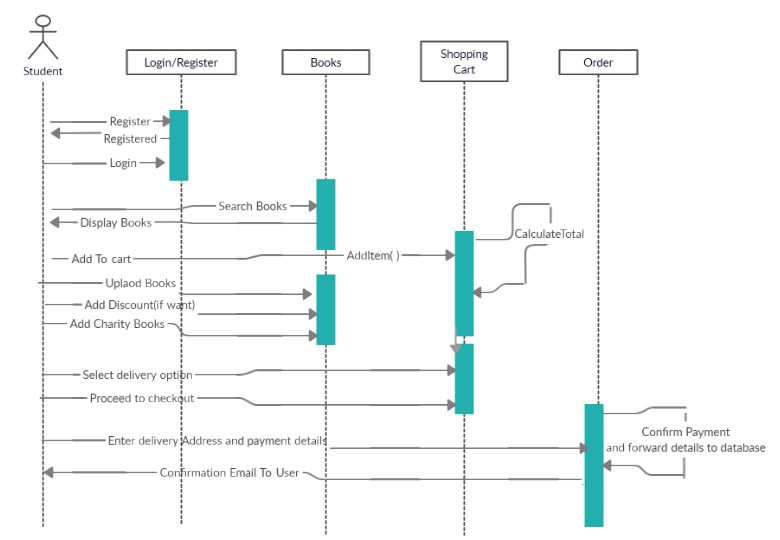
Type-2 that will be passed to and handled by business logic layer like visiting a static page.

Type-3 that will require database and will be processed in business logic before and after passing them to database and UI layer.

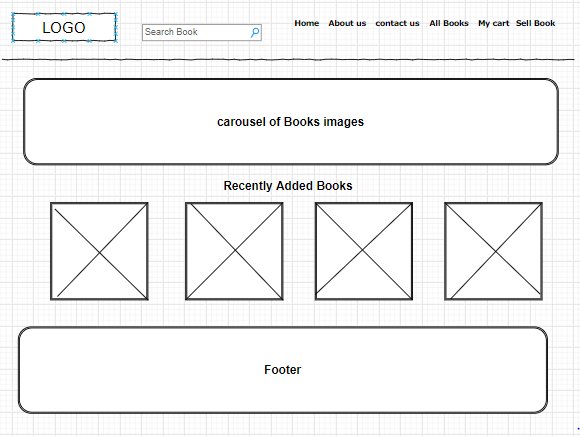
## **Architectural Design**

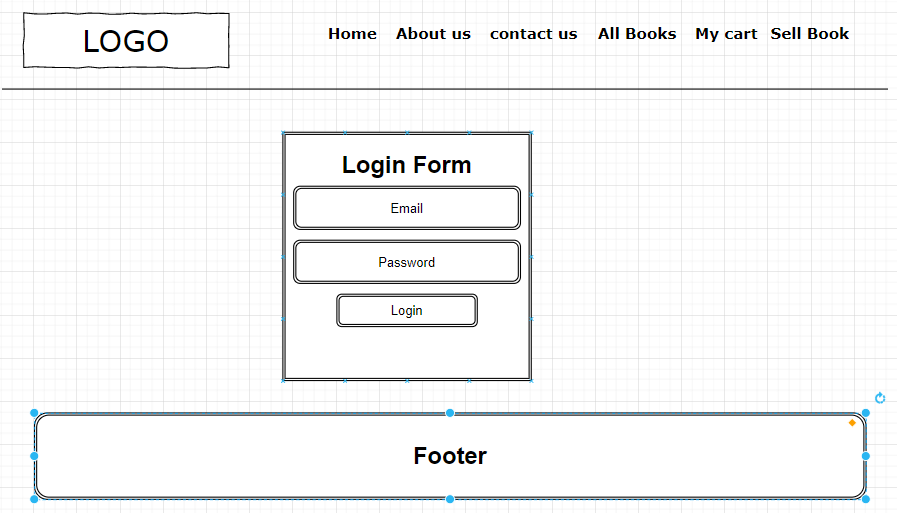


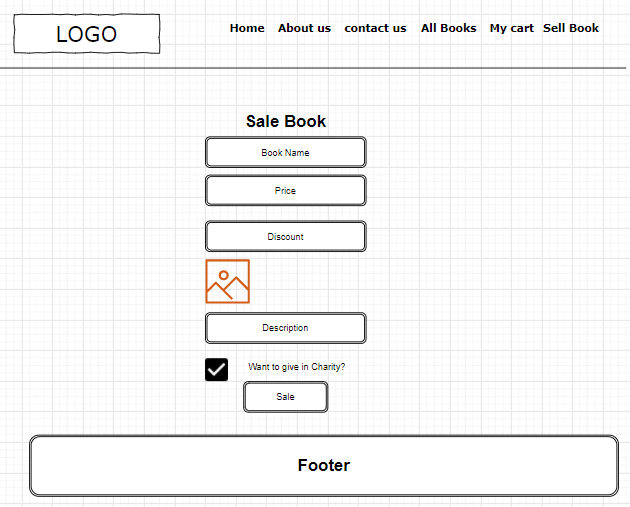
## **Sequence Diagram**



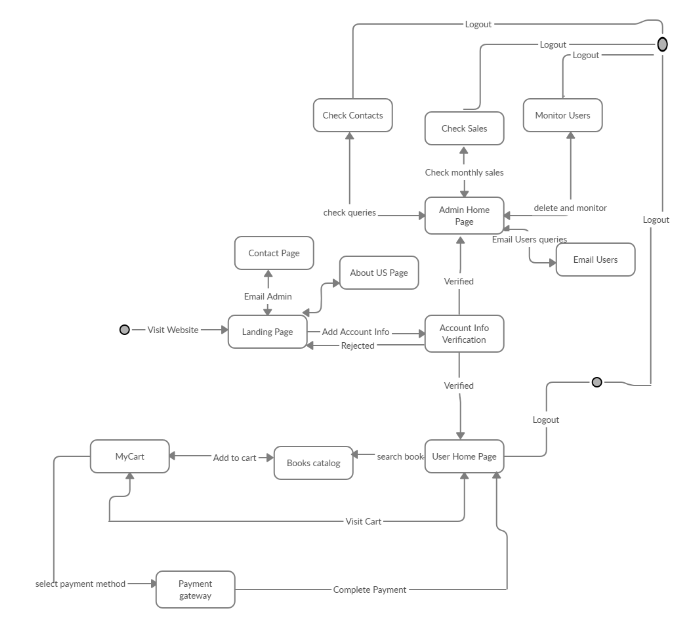
## **Interface Design**







**State Diagram**



**Database Relational Diagram**

