|  |
| --- |
|  |
| Real Estate Management & Price Prediction  Specification and Design |
|  |

|  |
| --- |
| Version No: dd-mm-yyyy  Copy: uncontrolled |

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](#_Toc17800781)

[Specification 7](#_Toc17800782)

[System Description 7](#_Toc17800783)

[Feasibility Analysis 7](#_Toc17800784)

[Technical Feasibility 7](#_Toc17800785)

[Economic Feasibility 7](#_Toc17800786)

[Organizational Feasibility 7](#_Toc17800787)

[Requirements Specification 7](#_Toc17800788)

[Functional 7](#_Toc17800789)

[Non-functional 7](#_Toc17800790)

[Use Cases 7](#_Toc17800791)

[Use Case Diagrams 8](#_Toc17800792)

[Use Case Descriptions 8](#_Toc17800793)

[Context Model 8](#_Toc17800794)

[Design 8](#_Toc17800795)

[Architectural Design 8](#_Toc17800796)

[Hardware Specifications 8](#_Toc17800797)

[Database Structure 8](#_Toc17800798)

[Interface Design 8](#_Toc17800799)

[Sequence Diagrams 8](#_Toc17800800)

[State Diagrams 8](#_Toc17800801)

# 

# Executive Summary

The system resolves the issue of searching of Real Estate by bringing this entire traditional system online. People can login as buyer, seller or admin. The system can be acquired by the organization as proprietary and can hire their own admin. Through this system, the sell and purchase of property will become easy and different resources of sellers and buyers will be saved like time, energy, money etc.

The targets are both commercial and residential properties. People can sign up for free, and can use website as per their needs. For selling purpose seller can post pictures, add/update description of property and communicate with buyer. Buyer can search property as per need, bid, communicate and purchase.

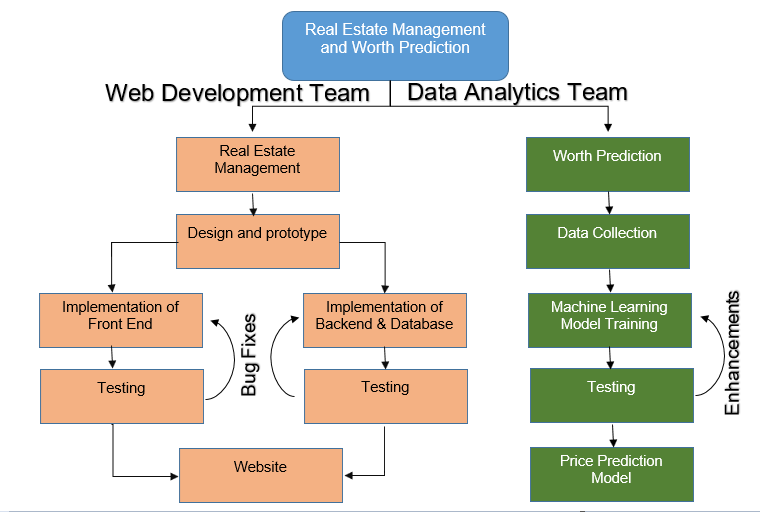
All these activities between seller and buyer will be monitored by the admin. An admin can be an independent entity or part of an organization. The admin is equivalent to real estate agent in the traditional Real Estate purchase process.

The system will be incorporated with Machine Learning model to predict the future price of the property. Initially it will be treated as separate entity, but when enough trusted data will be gathered the, the model will be trained and then it will become part of system. The integration of model when it is not trained contains potential risks.

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 |
| Data Analytics Team | Data Collection | Machine Learning model preparation and its training | | | Enhancements |
| Umbrella Activities | Documentation, Management, Quality Assurance, Reporting | | | | | |

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



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 server-side coding must be scalable and integrate able with ML model in future.
9. The database must represent correct relationships.
10. The database must be normalized (preferable with BCNF, Boyce Codd Normal Form).
11. The database engine must support CCP (Concurrency Control Protocol).
12. The database must be able to store data at least 10 years.
13. The database size must be utilized precisely.

## 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 (particularly when ML is integrated), 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.

### 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 initial cash out-flow is 4500$(Price prediction - 1000$, Website development - 1500$, and deployment - 2000$).

### 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 SELLERS**

1. Sellers can Add property to the website with the details like image, video, demand, description, no of rooms etc.
2. Sellers can register themselves as well as login to the system.
3. Seller can chat with the buyer, if buyer asks something.
4. Seller can check the property of his owns.
5. Seller can check offers to his property.
6. Seller can approve one offer.
7. Seller can contact admin before registration for ay query. Reply will be given through email.

* **FOR BUYERS**

1. Buyer can register and login.
2. Buyer can search the property through search filter.
3. Buyer can view the property details like image, video and all the description, demand etc.
4. Buyer can contact admin before registration for any query. Reply will be given through email.
5. Buyer can make offer to any property.
6. Buyer will be notified by email if his offer approves.

* **FOR ADMIN**

1. Admin can register another admin.
2. Admin can view and delete the property.
3. Admin can delete and view the user.
4. Admin can communicate the buyer after approval of the offer.

### 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 Diagrams

* For Admin:

ADMIN

Another Admin Registration

View and delete property

View and delete user

Monitor chat

Reply the contacts

/

Check Approved offers

REAL ESTATE MANAGEMENT SYSTEM

* For Buyer:

Buyer

Search filter properties

Make offer to the property

See video images

Ask question about property

Register/login

Notified if offer approved by admin

REAL ESTATE MANAGEMENT SYSTEM

* For Seller:

Seller

Upload Property for sale

Check offers

Approve one offer

Reply to buyer

Register/login

Enquiry Property

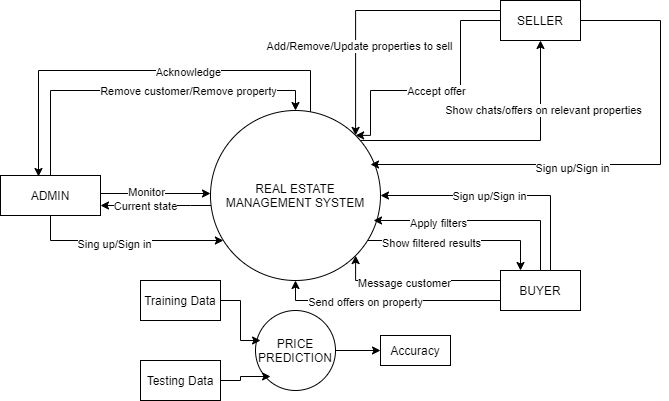
REAL ESTATE MANAGEMENT SYSTEM

### Use Case Descriptions

Admin has privileges. He/she can read communication taking place between buyer and seller so that no unauthorized dealing or activities that are against organization’s C2C business policies take place. If admin find any customer against policy, he/she can remove customer. Admin’s duty also includes to maintain check and balance that which properties are sold out, which are pending and to provide any assistance to customer if needed.

Anyone can explore property without being sing up. But to carry out sell-purchase process both parties must be sign up/sign in. This process is dependent on the activity of seller that is to add property to sell. The seller can add multiple and other information. Any buyer interested in this property can contact seller via chat. He/she can also bid on the project. The seller can receive multiple offers, and its completely choice of customer whom to sell property or not to sell at all. Buyer will be notified if his/her offer on any property is accepted.

## Context Model



# Design

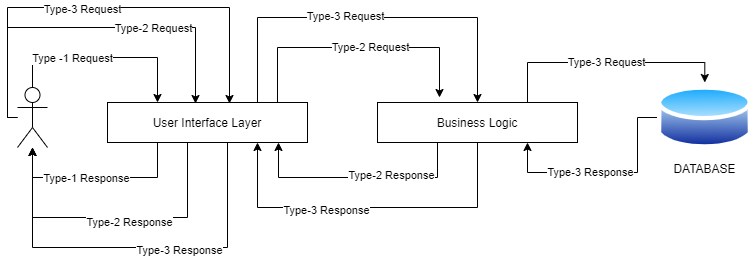
1. The assigned responsibilities to each layer should not mix.
2. No direct communication with database.
3. No unprocessed data pass to database.
4. UI must be server side rendered.
5. Each new request assigned to new thread in server.
6. User can only interact with UI layer.
7. Request-Response protocol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

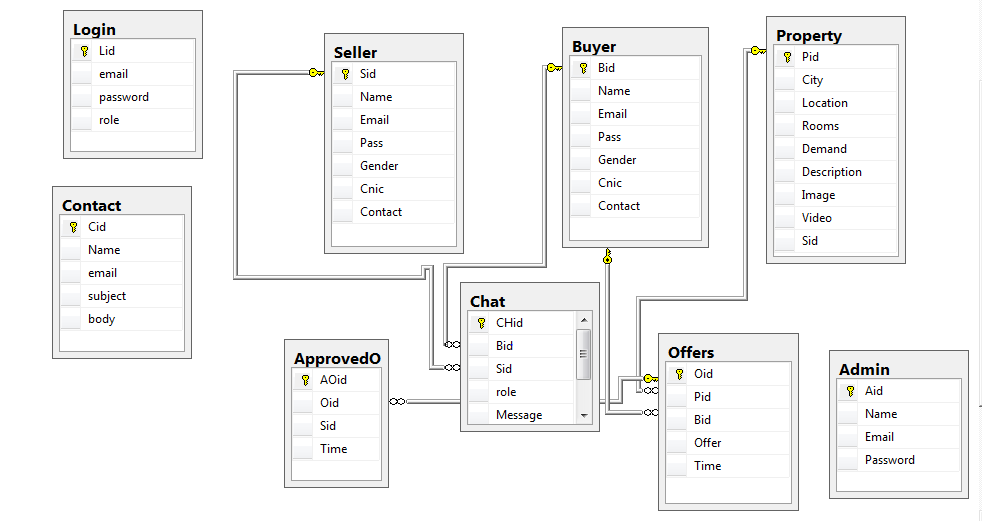
## Architectural Design



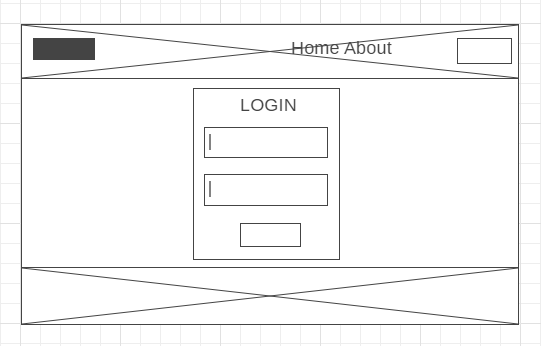
## Hardware Specifications

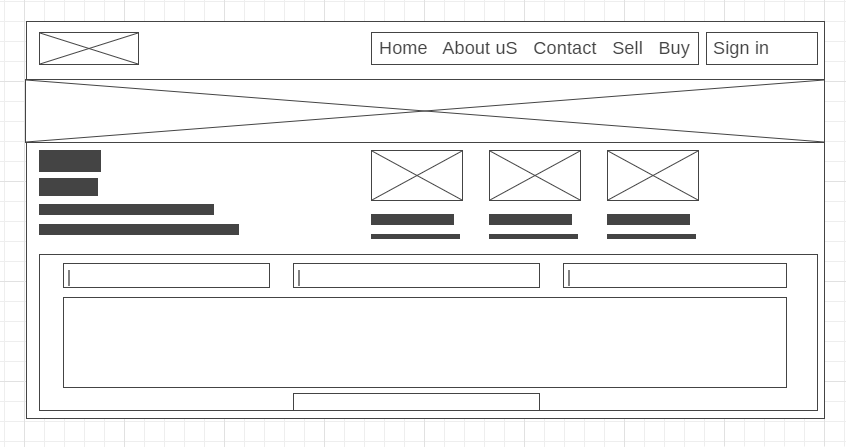
This section is not applicable because all hardware specifications will be maintained by CSP and may vary from time to time

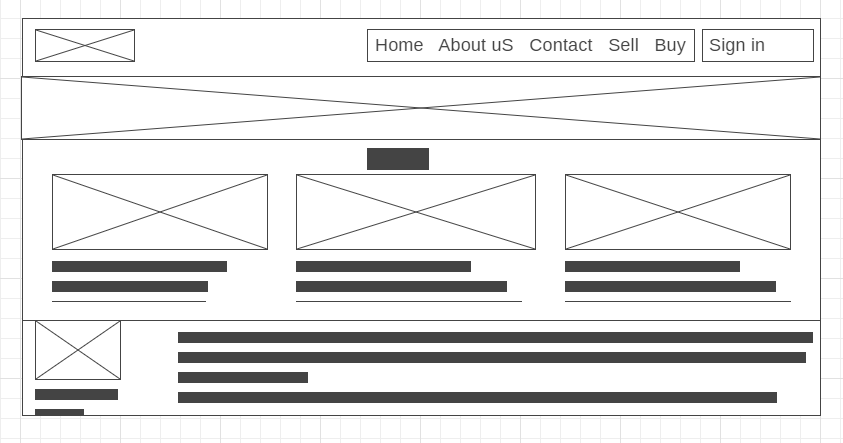
## Database Structure



## Interface Design







## Sequence Diagrams

Interaction Diagrams

## State Diagrams

If applicable