Dream House Project Report

COMP 231

Centennial College

Professor: Hao Lac

Jake Nesovic

Group Members:

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Student No. | Section No. | Signature |
| Abhishek Mekwan | 300-734-015 | 060 |  |
| Gowtham Pinnaka | 300-733-643 | 060 |  |
| Mohammed Ekram Ullah | 300-719-918 | 060 |  |
| Pei Wang | 300-712-924 | 060 |  |
| Sri Yoga | 300-685-658 | 060 |  |
| Santhalingam Sivasambu | 300-546-288 | 060 |  |

Date: September 19th, 2013



Table of Contents

[Project Scope 3](#_Toc367356905)

[Problem statement 3](#_Toc367356906)

[Proposed Solution 3](#_Toc367356907)

[High Level Deliverables 3](#_Toc367356908)

[High Level Architecture 4](#_Toc367356909)

[UML diagrams 5](#_Toc367356910)

[*Use Case Diagram* 5](#_Toc367356911)

[*Class Diagram* 5](#_Toc367356912)

[*Object Diagram* 5](#_Toc367356913)

[*State Chart Diagram* 5](#_Toc367356914)

[*Activity Diagram* 5](#_Toc367356915)

[*Sequence Diagram* 5](#_Toc367356916)

[*Collaboration Component Diagrams* 5](#_Toc367356917)

[*Deployment Diagrams* 5](#_Toc367356918)

[Database Table Design and Analysis 5](#_Toc367356919)

[*Solution to eliminate redundancies and anomalies* 5](#_Toc367356920)

[Customer-Friendly Features 5](#_Toc367356921)

[Security Features 6](#_Toc367356922)

[user documentation 6](#_Toc367356923)

[Team Activities 6](#_Toc367356924)

# Project Scope

## Problem statement

Toronto’s housing market is exploding, and with that market real estate companies are advertising their own product everywhere. Real estate companies use some software/web-application to register customer but there are very few software for agent themselves. Actually, most of real estate software are web based basic application and barely are made for agent. So, our team will build that software for the real estate company agent where all the information of housing market will be there and real estate agent will search out the appropriate housing for the customer through that software. The efficiency of this application will help agent get rid of communication problems, document management problems, media asset problems and so on.

## Proposed Solution

Dream House Project aims at achieving a breakthrough in usability, code portability and performance scalability by providing an improved system which can holds real time data of the real estate market, keep on-sale households information database update and could be used by the real estate company agent to conduct search out the appropriate housing for the customer easily and efficiently.

## High Level Deliverables

o Scope and objectives document

o Analysis and Design

o All Java source code

o Code integration and testing

o An MS-SQL Server database, database schema, etc.

o A .NET application for administrating the data on the server side

o Application Testing

o User manuals/system documentation

# High Level Architecture



**Figure 1:** Architecture drawing of Dream House System

**The description for Figure 1 would be:**

Agent: - is an entity who has access to the system, but cannot make changes in the database.

Administrator: - is an entity who has full access to the system, including authentication to make desirable changes in the database.

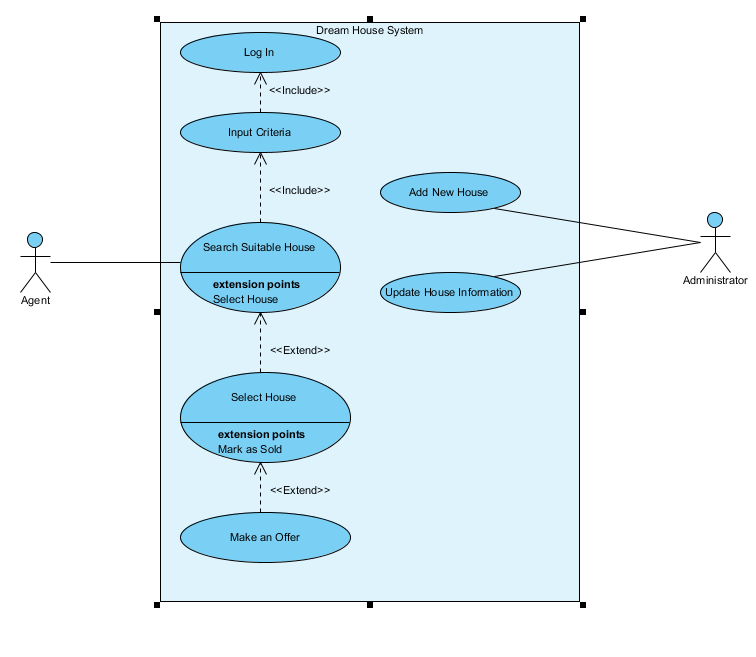
The clients will come to the agents with their particular requirements in mind. The agent will log in the system using his/her credentials. Based on the credentials the system will recognise him/her as agent or administrator, allowing him to access system limited to the credentials. Then the agent will be shown a search page, where the requirements of clients can be configured. After filling required information, the agent will get results matching the client’s requirements. On the search result page agent can browse among the results and show clients available choices.

The administrator has privileges to add new data in the database or update or delete the existing one.

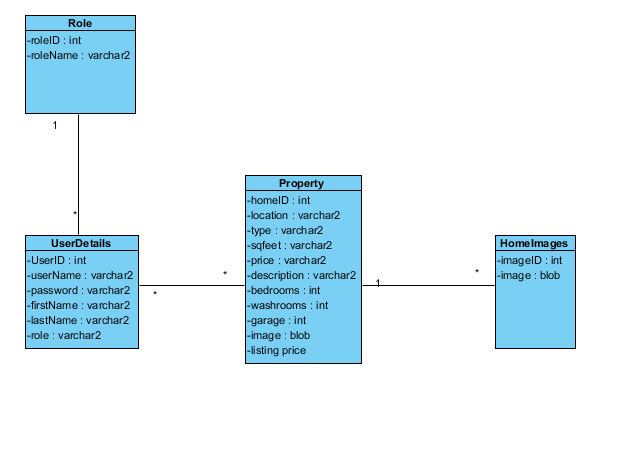
The agent on other hand cannot add or delete records from the database. There are two layers of security in the system.

# UML diagrams

## *Use Case Diagram*



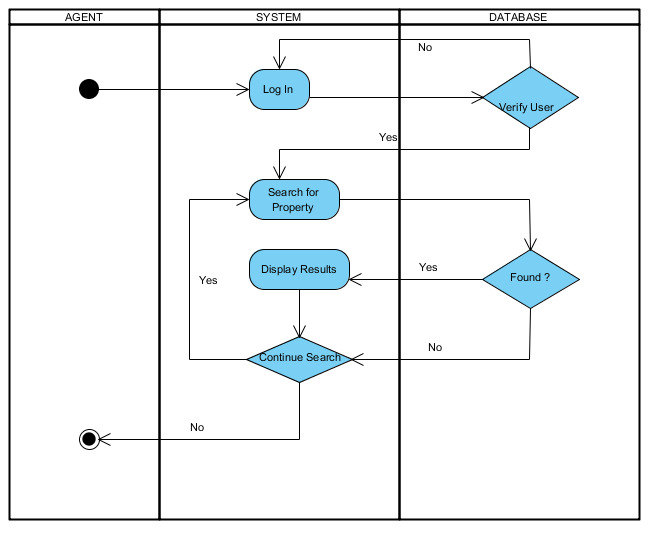
## *Class Diagram*

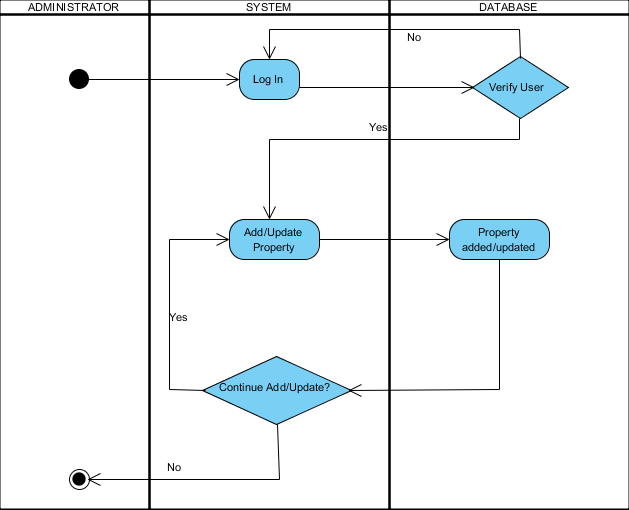


## *Object Diagram*

## *State Chart Diagram*

## *Activity Diagram*





## *Sequence Diagram*

## *Collaboration Component Diagrams*

## *Deployment Diagrams*

# Database Table Design and Analysis

## *Solution to eliminate redundancies and anomalies*

# Customer-Friendly Features

# Security Features

# user documentation

.

# Team Activities