**1.INTRODUCTION**

* 1. **Introduction To Project:**

**TO-LET FINDER** is widely spreading and revolutionary technology. This project is mainlybased on housing problem. The people involved in housing would be house Owner and a house needy (Tenant). The person who needs would need to roam on roads for the availability of the houses available for rent. This would be a big burden for him and is not cost effective. The house owners who has houses but finds no time to publish their availability would be there. Owner may find it difficult to talk to all the visitors of house and specify the details of the house to whom ever comes in search of house.

This process if made online, it would solve the problems of both house Owner and house needy (Tenant). Our project deals with the problems of both owner and tenant. Owner would specify the address, price, specifications of the house along with the images of the house in the website. The person who needs the house, just specifies his requirements i.e., the address where he/she wants, the price that he/she would bare etc. After the specification of all of them, he/she would be displayed with the related list of houses available. This search of user and the specification of owner can be done in no time. Thus, this provides a greater flexibility. Admin would look after the maintainance of the site and other issues related.

**Scope**:

The worldwide web is one platform where everything and anything is available. So, this project has a very wide scope. Owners of houses all over INDIA can post the houses that are available for rent, so that tenants can get the houses in their specified area. Many advantages that are available for TO-LET FINDER makes it much beneficial than the earlier existing techniques or mechanisms related to housing system.

**2.SYSTEM ANALYSIS**

**2.1 EXISTING SYSTEM:**

People search of houses for rent just by roaming on road. This takes lot of time and the person may or may not get the home of his specified requirements. Someof the house owners may not find free time to tell people about their house which is available for rent. This project deals with the problems of both tenants and house owners.

**Disadvantages of Existing System**:

* It takes alot of time to search houses.
* It is not efficient process.
* It is burden for both owner and tenant.

**2.2 PROPOSED SYSTEM**

The TO-LET FINDER is a software application. Through this, the Owners are registered online.Their information and the posts made by Owners are stored in the database. Here, post refers to the available house at some location. Admin can easily access the details of the posts made by Owners. The Users (house needy) just specifies the State, City and Type of the house he wants and searches the houses. This application keeps the data in a centralized way which is available to all the users simultaneously.

**Advantages of Proposed System:**

* Every process is automated and so, less time consumption.
* Less burden for both owner and tenant.
* It provides an easy way to find Houses.
* No Extra travelling time, cost and effort is involved.
* It is cost and time efficient.

**2.3 FEASIBILITY STUDY**

Feasibility study is an important phase in the software development process. Preliminary investigation examines project feasibility, the likelihood the system will be useful to the organization. The mainobjective of the feasibility study is to test the Technical,Operational and Economical feasibility for adding new modules and debugging old running system. Feasibility study should be performed on the basis of various criteria and parameters. The various feasibility studies are:

* Technical Feasibility
* Operation Feasibility
* Economical Feasibility

**Technical Feasibility:**

The technical issue usually raised during the feasibility stage of the investigation includes the following:

* Does the necessary technology exist to do what is suggested?
* Do the proposed equipments have the technical capacity to hold the data required to use the new system?
* Will the proposed system provide adequate response to inquiries, regardless of the number or location of users?
* Can the system be upgraded if developed?

**Operational Feasibility:**

**User-friendly:** Customer will use the forms for their various transactions i.e. for adding new routes,viewing the routes details. Also the Customer wants the reports to view the various transactions based on the constraints. These forms and reports are generated as user-friendly to the Client.

**Reliability:** The package wills pick-up current transactions on online. Regarding the old transactions, User will enter them in to the system.

**Security:** The web server and database server should be protected from hacking, virus etc.

**Portability:**The application will bedevelopedusing standard open source software like c#(sharp),asp.net etc these software will work both on Windows7,windows8 andwindows8.1 o/s. Hence portability problems will not rise.

**Availability:** This software will be available always.

**Economic Feasibility:**

A system can be developed technically and that will be used if installed must still be a good investment for the organization. In the economical feasibility, the development cost in creating the system is evaluated against the ultimate benefit derived from the new systems. Financial benefits must equal or exceed the costs. The system is economically feasible. It does not require any additional hardware or software.

**2.3 SYSTEM REQUIREMENTS**

**Hardware Requirements:**

Processor : Pentium IV 2.4 GHz

Hard Disk : 40 GB or more

Monitor : 15 VGA color

Mouse : Optical Mouse

Keyboard : 110 keys enhanced

RAM : 512 MB or more

**Software Requirements:**

Operating System : Windows 7 or higher

Language : ASP.Net, C#

Front End : Microsoft Visual Studio 2012

Database : SQL Server Management Studio 2012

**2.4 MODULE DESCRIPTION**

In this project “TO-LET FINDER”, there are three (3) sets of modules. They are:

* Admin,
* Owner,
* User.

**Description of Modules**:The functionalities of these modules are as follows:

The administrator is the ultimate controller of the application with the highest authority.

**Admin Login has the following features:-**

**a) Edit Profile**: It allows modifying profile pic.

**b) Change Password:** It enable admin to change his/her account’s password.

**c) Users Posts:**It enables admin to view/delete posts of users.

**d) Edit:**It enables admin to add/update/delete the information available about States, cities, types of houses.

**Owner Login member has the following features:-**

**a) Edit Profile:**It allows modification of the details of the owner.

**b) New Posts:**It allows the owner to post the details of the house available for rent.

**c) Delete Posts:**It enables owner to delete his older posts.

**User has the following features:-**

User doesn’t need any authentication. Simply, he follow the following steps:

* Enters the Website.
* Selects state, city, type of house he wants and presses search button.
* User then gets the related results in which he finds a house of his requirements and calls the owner to fix the deal.

**3.SYSTEM DESIGN**

**3.1 UML DIAGRAMS(Unified Modeling Language)**

"The Unified Modeling Language (UML) is a language for

* Specifying,
* Visualizing,
* Constructing and
* Documenting

UML is a graphical notation for modeling various aspects of software systems.

UML diagrams represent two different views of a system model:

**Static (or structural) view:**Emphasizes the static structure of the system using objects, attributes, operations and relationships. The structural view includes class diagrams and composite structure diagrams.

**Dynamic (or behavioral) view:**Emphasizes the dynamic behavior of the system by showing collaborations among objects and changes to the internal states of objects. This view includes sequence diagrams, activity diagrams and state machine diagrams

**Structural Diagrams:**

Structural diagrams emphasize the things that must be present in the system being modeled. Sincestructure diagrams represent the structure, they are used extensively in documenting the software architecture of software systems.

**Behavioral Diagrams:**

Behavior diagrams emphasize what must happen in the system being modeled. Since behavior diagrams illustrate the behavior of a system, they are used extensively to describe the functionality of software systems.

**Interaction Diagrams:**

Interaction diagrams, a subset of behavior diagrams, emphasize the flow of control and data among the things in the system being modeled.

**Class diagram:**It describes the structure of a system by showing the system's classes, their attributes, Operations and the relationships among the classes.



**Usecase diagram:**It describes the functionality provided by a system in terms of actors, their goals represented as use cases, and any dependencies among those use cases.

**Admin Usecase diagram** for To-Let Finder:



**User Use case diagram** for To-Let Finder:



**Owner Usecase diagram** for To-Let Finder:



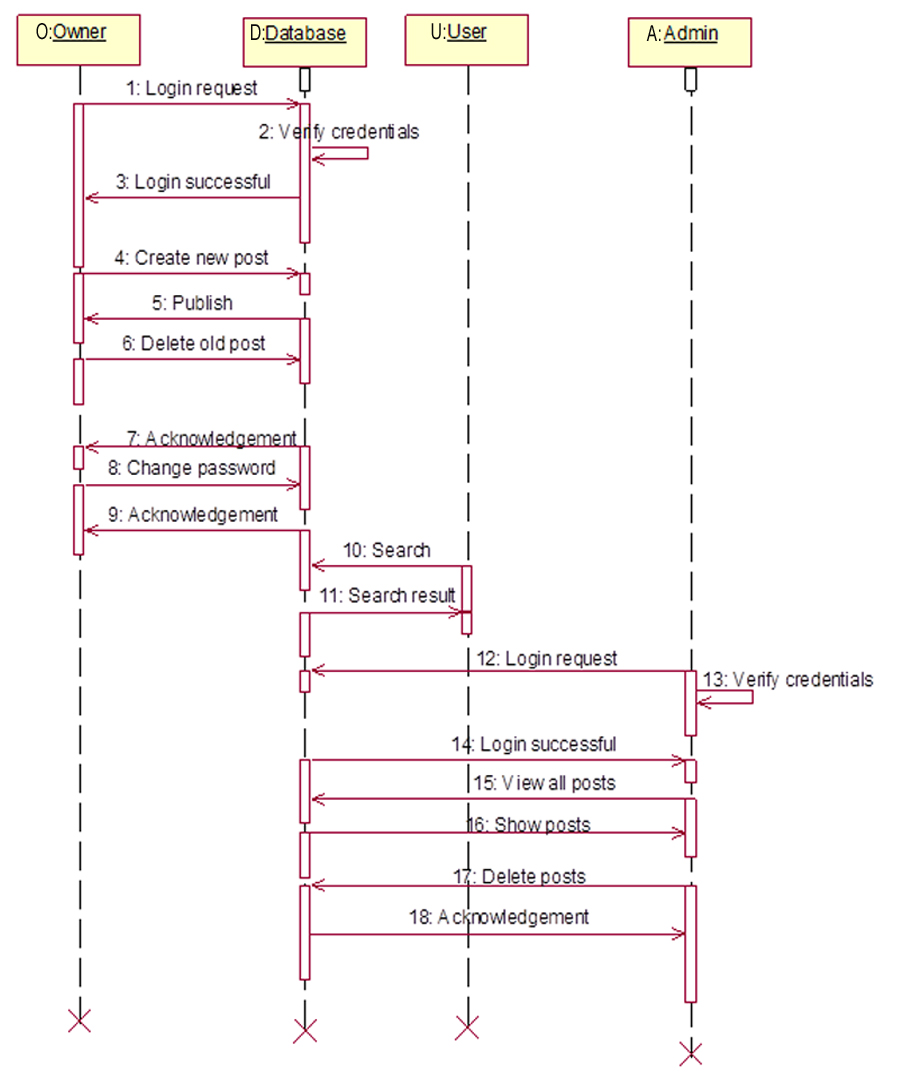
**Activity diagram:**It describes the business and operational step-by-step workflows of components in a systems the overall flow of control.

**Admin** Activity Diagram for To-Let finder:

**User Admin Owner**



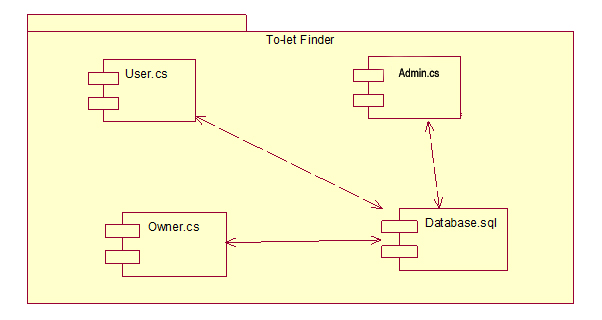
**Sequence diagram:**It shows how objects communicate with each other in terms of a sequence of messages. Also indicates the life spans of objects relative to those messages.



**Collaboration diagram:**It provides an overview in which the nodes represent communication diagrams.



**Component diagram:**It describes how a software system is split up into components and shows the decencies among these diagrams.

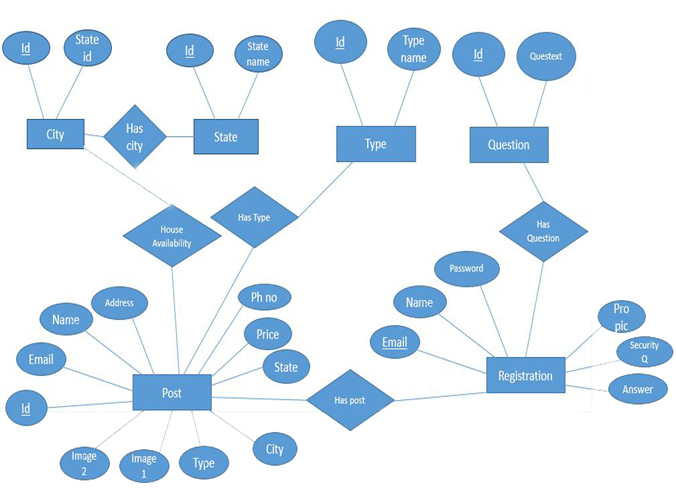


**Deployment diagram:**It describes the hardware used in system implementations and the execution environments and artifacts deployed on the hardware.



**3.3 E-R DIAGRAM (Entity-Relationship)**

E-R Diagram for **To-Let Finder:**



**3.4 DESCRIPTION OF TABLES:**

|  |  |  |  |
| --- | --- | --- | --- |
| **REGISTRATION TABLE** | | | |
| **FIELD NAME** | **DATA TYPE** | **KEY** | **DESCRIPTION** |
| Name | Varchar(50) | - | First name of the student. |
| Phone number | Bigint | - | Number of the owner. |
| Password | Varchar(50) | - | Password of the owner. |
| Email id | Varchar(50) | Primary key | ID of the owner which is unique. |
| Profile Picture | Varchar(50) | - | Profile picture of the owner. |
| Security Question | Varchar(50) | - | Security Question of the Owner. |
| Answer | Varchar(50) | - | Answer of the Security Question. |

|  |  |  |  |
| --- | --- | --- | --- |
| **CITYTABLE** | | | |
| **FIELD NAME** | **DATA TYPE** | **KEY** | **DESCRIPTION** |
| City id | Int | Primary key | ID of the city is to be unique. |
| State id | Int | Foreign key | ID of the state |
| City name | Varchar(15) | - | Name of the city. |

|  |  |  |  |
| --- | --- | --- | --- |
| **STATE TABLE** | | | |
| **FIELD NAME** | **DATA TYPE** | **KEY** | **DESCRIPTION** |
| Id | Int | Primary key | ID of the state is to be unique. |
| State name | Int | Foreign key | Name of the state. |

|  |  |  |  |
| --- | --- | --- | --- |
| **POST TABLE** | | | |
| **FIELD NAME** | **DATATYPE** | **KEY** | **DESCRIPTION** |
| Post id | Int | Primary key | ID of the owner. which is unique |
| Email | Varchar(50) | - | Proof of the owner. |
| Name | Varchar(50) | - | Name of the owner. |
| Phone number | Bigint | - | Phone number of the owner. |
| City | Varchar(50) | - | City of the post. |
| State | Varchar(50) | - | State of the post. |
| Address | Varchar(50) | - | Address of the post. |
| Type | Varchar(50) | - | Type of the house. |
| Price | Int | - | Price of the house. |
| Image1 | Varchar(50) | - | Image1 of the house. |
| Image2 | Varchar(50) | - | Image2 of the house. |
| Image3 | Varchar(50) | - | Image3 of the house. |
| Image4 | Varchar(50) | - | Image4 of the house. |
| Image5 | Varchar(50) | - | Image5 of the house. |
| Image6 | Varchar(50) | - | Image6 of the house. |
| Image7 | Varchar(50) | - | Image7 of the house. |
| Image8 | Varchar(50) | - | Image8 of the house. |
| Image9 | Varchar(50) | - | Image9 of the house. |
| Image10 | Varchar(50) | - | Image10 of the house. |

|  |  |  |  |
| --- | --- | --- | --- |
| **QUESTIONTABLE** | | | |
| **FIELDNAME** | **DATA TYPE** | **KEY** | **DESCRIPTION** |
| Id | Int | Primary key | Unique number for the question. |
| Question text | Varchar(50) | - | Text of the question. |

|  |  |  |  |
| --- | --- | --- | --- |
| **TYPE TABLE** | | | |
| **FIELD NAME** | **DATA TYPE** | **KEY** | **DESCRIPTION** |
| Id | Int | Primary key | Unique identification of type. |
| Type name | Varchar(50) | - | Type of the name. |

**4.IMPLEMENTATION**

**4.1 TECHNOLOGY DESCRIPTION:**

**Microsoft Visual Studio:**

Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs for Microsoft Windows superfamily of operating systems,as well as web sites, web applications and web services. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.

Visual Studio supports different [programming languages](http://en.wikipedia.org/wiki/Programming_language) and allows the code editor and debugger to support (to varying degrees) nearly any programming language, provided a language-specific service exists. Built-in languages include [C](http://en.wikipedia.org/wiki/C_(programming_language))/[C++](http://en.wikipedia.org/wiki/C%2B%2B) (via [Visual C++](http://en.wikipedia.org/wiki/Visual_C%2B%2B)), [VB.NET](http://en.wikipedia.org/wiki/VB.NET) (via [Visual Basic .NET](http://en.wikipedia.org/wiki/Visual_Basic_.NET)), [C#](http://en.wikipedia.org/wiki/C_Sharp_(programming_language)) (via [Visual C#](http://en.wikipedia.org/wiki/Visual_C_Sharp)), and [F#](http://en.wikipedia.org/wiki/F_Sharp_(programming_language)) (as of Visual Studio 2012). Support for other languages such as [M](http://en.wikipedia.org/wiki/M_(programming_language)), [Python](http://en.wikipedia.org/wiki/IronPython), and [Ruby](http://en.wikipedia.org/wiki/IronRuby) among others is available via language services installed separately.

It supports [XML](http://en.wikipedia.org/wiki/XML)/[XSLT](http://en.wikipedia.org/wiki/XSLT), [HTML](http://en.wikipedia.org/wiki/HTML)/[XHTML](http://en.wikipedia.org/wiki/XHTML) , [JavaScript](http://en.wikipedia.org/wiki/JavaScript)  and  [CSS](http://en.wikipedia.org/wiki/Cascading_Style_Sheets). Individual language specific versions of Visual Studio also exist which provide more limited language services to the user: Microsoft Visual Basic, Visual J#, Visual C#, and Visual C++.

|  |
| --- |
| Microsoft Visual Studio |
| [Visual Studio 2012 logo and wordmark.svg](http://en.wikipedia.org/wiki/File:Visual_Studio_2012_logo_and_wordmark.svg) |
|  |
| [Visual Studio 2012 EN.png](http://en.wikipedia.org/wiki/File:Visual_Studio_2012_EN.png)  Screenshot of Visual Studio 2012, editing a program's source code in [Visual Basic.NET](http://en.wikipedia.org/wiki/Visual_Basic.NET) |

|  |  |
| --- | --- |
| [Developer(s)](http://en.wikipedia.org/wiki/Software_developer) | [Microsoft](http://en.wikipedia.org/wiki/Microsoft) |
| [Stable release](http://en.wikipedia.org/wiki/Software_release_life_cycle) | Visual Studio 2013 Update 1 (12.0.30110.00) (January 20, 2014; 58 days ago) [[±]](http://en.wikipedia.org/w/index.php?title=Template:Latest_stable_software_release/Microsoft_Visual_Studio&action=edit)[[1]](http://en.wikipedia.org/wiki/Microsoft_Visual_Studio#cite_note-1) |
| [Preview release](http://en.wikipedia.org/wiki/Software_release_life_cycle) | Visual Studio 2013 Release Candidate(September 9, 2013; 6 months ago) [[±]](http://en.wikipedia.org/w/index.php?title=Template:Latest_preview_software_release/Microsoft_Visual_Studio&action=edit)[[2]](http://en.wikipedia.org/wiki/Microsoft_Visual_Studio#cite_note-2) |
| Written in | [C++](http://en.wikipedia.org/wiki/C%2B%2B) and [C#](http://en.wikipedia.org/wiki/C_Sharp_(programming_language))[[3]](http://en.wikipedia.org/wiki/Microsoft_Visual_Studio#cite_note-3) |
| [Operating system](http://en.wikipedia.org/wiki/Operating_system) | [Microsoft Windows](http://en.wikipedia.org/wiki/Microsoft_Windows) |
| Available in | Chinese, English, French, Portuguese, German, Italian, Japanese, Korean, Spanish and Russian |

|  |  |
| --- | --- |
| [Type](http://en.wikipedia.org/wiki/List_of_software_categories) | [Integrated development environment](http://en.wikipedia.org/wiki/Integrated_development_environment) |
| [License](http://en.wikipedia.org/wiki/Software_license) | [Proprietary software](http://en.wikipedia.org/wiki/Proprietary_software) Express edition: [Registerware](http://en.wikipedia.org/wiki/Registerware) Other editions: [Trialware](http://en.wikipedia.org/wiki/Trialware)[[4]](http://en.wikipedia.org/wiki/Microsoft_Visual_Studio#cite_note-4) |
| Website | [www.visualstudio.com](http://www.visualstudio.com/) |

**C#:**

The correct title of this article is C# (programming language). The substitution or omission of the [#](http://en.wikipedia.org/wiki/Number_sign) is because of [technical restrictions](http://en.wikipedia.org/wiki/Wikipedia:Naming_conventions_(technical_restrictions)).

C# (pronounced as see sharp) is a [multi-paradigm programming language](http://en.wikipedia.org/wiki/Multi-paradigm_programming_language) encompassing [strongtyping](http://en.wikipedia.org/wiki/Strong_typing), [imperative](http://en.wikipedia.org/wiki/Imperative_programming), [declarative](http://en.wikipedia.org/wiki/Declarative_programming), [functional](http://en.wikipedia.org/wiki/Functional_programming), [procedural](http://en.wikipedia.org/wiki/Procedural_programming), [generic](http://en.wikipedia.org/wiki/Generic_programming), [object-oriented](http://en.wikipedia.org/wiki/Object-oriented_programming) ([class](http://en.wikipedia.org/wiki/Class_(computer_science))-based), and [component-oriented](http://en.wikipedia.org/wiki/Component-based_software_engineering) programming disciplines. It was developed by [Microsoft](http://en.wikipedia.org/wiki/Microsoft) within its [.NET](http://en.wikipedia.org/wiki/.NET_Framework) initiative and later approved as a standard by [Ecma](http://en.wikipedia.org/wiki/Ecma_International) (ECMA-334) and [ISO](http://en.wikipedia.org/wiki/International_Organization_for_Standardization) (ISO/IEC 23270:2006). C# is one of the programming languages designed for the [Common Language Infrastructure](http://en.wikipedia.org/wiki/Common_Language_Infrastructure). C# is built on the syntax and semantics of C++, allowing C programmers to take advantage of .NET and the common language runtime.

C# is intended to be a simple, modern, general-purpose, object-oriented programming language. Its development team is led by [Anders Hejlsberg](http://en.wikipedia.org/wiki/Anders_Hejlsberg). The most recent version is C# 5.0, which was released on August 15, 2012.

**Design goals:**

The C# language is intended to be a simple, modern, general-purpose, [object-oriented programming](http://en.wikipedia.org/wiki/Object-oriented_programming) language. The language, and implementations thereof, should provide support for software engineering principles such as [strong type](http://en.wikipedia.org/wiki/Strong_type) checking, array [bounds checking](http://en.wikipedia.org/wiki/Bounds_checking), detection of attempts to use [uninitialized variables](http://en.wikipedia.org/wiki/Uninitialized_variable), and automatic [garbage collection](http://en.wikipedia.org/wiki/Garbage_collection_(computer_science)). Software robustness, durability, and programmer productivity are important.

The name "C sharp" was inspired by musical notation where a [sharp](http://en.wikipedia.org/wiki/Sharp_(music)) indicates that the written note should be made a semitone higher in [pitch](http://en.wikipedia.org/wiki/Pitch_(music)). This is similar to the language name of [C++](http://en.wikipedia.org/wiki/C%2B%2B), where "++" indicates that a variable should be incremented by 1. The sharp symbol also resembles a [ligature](http://en.wikipedia.org/wiki/Typographic_ligature) of four "+" symbols (in a two-by-two grid), further implying that the language is an increment of C++.

Due to technical limitations of display (standard fonts, browsers, etc.) and the fact that the sharp symbol (U+266F ♯ [music sharp sign](http://en.wikipedia.org/wiki/Sharp_(music)) (HTML: &#9839;)) is not present on the standard keyboard, the [number sign](http://en.wikipedia.org/wiki/Number_sign) (U+0023 # number sign (HTML:&#35;)) was chosen to represent the sharp symbol in the written name of the programming language. This convention is reflected in the ECMA-334 C# Language Specification. However, when it is practical to do so (for example, in advertising or in box art), Microsoft uses the intended musical symbol.

**C# has the following syntax:**

Semicolons are used to denote the end of a statement.

[Curly braces](http://en.wikipedia.org/wiki/Curly_braces) are used to group statements. Statements are commonly grouped into methods (functions), methods into classes, and classes into [namespaces](http://en.wikipedia.org/wiki/Namespaces).

Variables are assigned using an [equal sign](http://en.wikipedia.org/wiki/Equals_sign), but compared using [two consecutive -equals’ signs](http://en.wikipedia.org/wiki/%3D%3D).[Square brackets](http://en.wikipedia.org/wiki/Square_brackets) are used with [arrays](http://en.wikipedia.org/wiki/Array_data_structure), both to declare them and to get a value at a given index in one of them.

**Distinguishing features:**

**Note**: The following description is based on the language standard and other documents listed in the "[External links](http://en.wikipedia.org/wiki/C_Sharp_(programming_language)#External_links)" section.

By design, C# is the programming language that most directly reflects the underlying [Common Language Infrastructure](http://en.wikipedia.org/wiki/Common_Language_Infrastructure) (CLI). Most of its intrinsic types correspond to value-types implemented by the CLI framework. However, the language specification does not state the code generation requirements of the compiler: that is, it does not state that a C# compiler must target a Common Language Runtime, or generate [Common Intermediate Language](http://en.wikipedia.org/wiki/Common_Intermediate_Language) (CIL), or generate any other specific format. Theoretically, a C# compiler could generate machine code like traditional compilers of C++ or [FORTRAN](http://en.wikipedia.org/wiki/Fortran). Some notable features of C# that distinguish it from C and C++ (and Java, where noted) are:

C# supports strongly typed implicit variable declarations with the keyword var, and implicitly typed arrays with the keyword new followed by a collection initializer.

Meta programming via C# attributes is part of the language. Many of these attributes duplicate the functionality of GCC's and Visual C++'s platform-dependent preprocessor directives.

The type dynamic allows for run-time method binding, allowing for JavaScript like method calls and run-time object composition. C# has support for strongly-typed function pointers via the keyword delegate. Like the QT framework's pseudo-C++ signal and slot, C# has semantics specifically surrounding publish-subscribe style events, though C# uses delegates to do so.

The C# languages does not allow for global variables or functions. All methods and members must be declared within classes. Static members of public classes can substitute for global variables and functions. Local variables cannot shadow variables of the enclosing block, unlike C and C++.

**Microsoft SQL Server**

Microsoft SQL Server is a [relational database management system](http://en.wikipedia.org/wiki/Relational_database_management_system) developed by [Microsoft](http://en.wikipedia.org/wiki/Microsoft). As a database, it is a software product whose primary function is to store and retrieve data as requested by other software applications, be it those on the same computer or those running on another computer across a network (including the Internet). There are at least a dozen different editions of Microsoft SQL Server aimed at different audiences and for workloads ranging from small single-machine applications to large Internet-facing applications with many concurrent users. Its primary [query languages](http://en.wikipedia.org/wiki/Query_language) are [T-SQL](http://en.wikipedia.org/wiki/Transact-SQL) and [ANSI SQL](http://en.wikipedia.org/wiki/SQL).

**SQL Server 2012**

SQL Server 2008 (formerly codenamed "Katmai")  was released on August 6, 2008 and aims to make data management [self-tuning](http://en.wikipedia.org/wiki/Self-tuning), self-organizing, and self-maintaining with the development of SQL Server Always On technologies, to provide near-zero downtime. SQL Server 2008 also includes support for [structured](http://en.wikipedia.org/wiki/Structured_data) and semi-structured data, including digital media formats for pictures, audio, video and other multimedia data. In current versions, such multimedia data can be stored as [BLOBs](http://en.wikipedia.org/wiki/Binary_large_object) (binary large objects), but they are generic bit streams. Intrinsic awareness of multimedia data will allow specialized functions to be performed on them. According to [Paul Flessner](http://en.wikipedia.org/w/index.php?title=Paul_Flessner&action=edit&redlink=1), senior Vice President, Server Applications, [Microsoft Corp.](http://en.wikipedia.org/wiki/Microsoft), SQL Server 2008 can be a data storage backend for different varieties of data: XML, email, time/calendar, file, document, spatial, etc. as well as perform search, query, analysis, sharing, and synchronization across all data types.

The [Full-text search](http://en.wikipedia.org/wiki/Full_text_search) functionality has been integrated with the database engine. According to a Microsoft technical article, this simplifies management and improves performance. Spatial data will be stored in two types. A "Flat Earth" (GEOMETRY or planar) data type represents geospatial data which has been projected from its native, spherical, coordinate system into a plane. A "Round Earth" data type (GEOGRAPHY) uses an ellipsoidal model in which the Earth is defined as a single continuous entity which does not suffer from the singularities such as the international dateline, poles, or map projection zone "edges". Approximately 70 methods are available to represent spatial operations for the Open Geospatial Consortium [Simple Features for SQL](http://en.wikipedia.org/wiki/Simple_Features_for_SQL), Version 1.1. SQL Server includes better compression features, which also helps in improving scalability. It enhanced the indexing algorithms and introduced the notion of filtered indexes. It also includes Resource Governor that allows reserving resources for certain users or workflows. It also includes capabilities for [transparent encryption of data](http://en.wikipedia.org/wiki/Transparent_Data_Encryption) (TDE) as well as compression of backups. SQL Server 2008 supports the[ADO.NET Entity Framework](http://en.wikipedia.org/wiki/ADO.NET_Entity_Framework) and the reporting tools, replication, and data definition will be built around the [Entity Data Model](http://en.wikipedia.org/wiki/Entity_Data_Model). [SQL Server Reporting Services](http://en.wikipedia.org/wiki/SQL_Server_Reporting_Services) will gain charting capabilities from the integration of the data visualization products from [Dundas Data Visualization, Inc.](http://en.wikipedia.org/wiki/Dundas_Data_Visualization,_Inc.), which was acquired by Microsoft.

**4.2 SAMPLE CODE**

**Sample code for newpw.aspx.cs**

using System;

usingSystem.Collections.Generic;

usingSystem.Configuration;

usingSystem.Data.SqlClient;

usingSystem.Drawing;

using System.IO;

usingSystem.Linq;

usingSystem.Web;

usingSystem.Web.UI;

usingSystem.Web.UI.WebControls;

public partial class newpw : System.Web.UI.Page

{

SqlConnection con;

String str = ConfigurationManager.ConnectionStrings["tfinderConnectionString"].ConnectionString;

protected void Page\_Load(object sender, EventArgs e)

{

}

protected void btnSubmit\_Click(object sender, EventArgs e)

{

con = new SqlConnection(str);

con.Open();

SqlCommandcmd = new SqlCommand("update registration set password=@pw where email=@email", con);

cmd.Parameters.AddWithValue("@pw", txtN1.Text);

cmd.Parameters.AddWithValue("@email",Session["email"]);

cmd.ExecuteNonQuery();

lblMsg.Text = "success";

Response.Redirect("http://localhost:59423/login.aspx?status=2"); }

}

}

**5.TESTING AND TESTCASE**

**SYSTEM TESTING:**

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. In fact, testing is the one step in the software engineering process that could be viewed as destructive rather than constructive.

A strategy for software testing integrates software test case design methods into a well-planned series of steps that result in the successful construction of software. Testing is the set of activities that can be planned in advance and conducted systematically. The underlying motivation of program testing is to affirm software quality with methods that can economically and effectively apply to both strategic to both large and small-scale systems.

**Strategic approach to software testing:**

The software engineering process can be viewed as a spiral. Initially system engineering defines the role of software and leads to software requirement analysis where the information domain, functions, behavior, performance, constraints and validation criteria for software are established. Moving inward along the spiral, we come to design and finally to coding. To develop computer software we spiral in along streamlines that decrease the level of abstraction on each turn.

A strategy for software testing may also be viewed in the context of the spiral. Unit testing begins at the vertex of the spiral and concentrates on each unit of the software as implemented in source code. Testing progress by moving outward along the spiral to integration testing, where the focus is on the design and the construction of the software architecture. Talking another turn on outward on the spiral we encounter validation testing where requirements established as part of software requirements analysis are validated against the software that has been constructed. Finally we arrive at system testing, where the software and other system elements are tested as a whole.

**UNIT TESTING:**

Unit testing focuses verification effort on the smallest unit of software design, the module. The unit testing we have is white box oriented and some modules the steps are conducted in parallel.

1. **White box testing:**

This type of testing ensures that

* + All independent paths have been exercised at least once.
  + All logical decisions have been exercised on their true and false sides.
  + All loops are executed at their boundaries and within their operational bounds.
  + All internal data structures have been exercised to assure their validity.

To follow the concept of white box testing we have tested each form .we have created independently to verify that Data flow is correct, All conditions are exercised to check their validity, All loops are executed on their boundaries.

1. **Conditional testing:**

In this part of the testing each of the conditions were tested to both true and false aspects. And all the resulting paths were tested. So that each path that may be generate on particular condition is traced to uncover any possible errors.

1. **Data flow testing:**

This type of testing selects the path of the program according to the location of definition and use of variables. This kind of testing was used only when some local variable were declared. The definition-use chain method was used in this type of testing. These were particularly useful in nested statements.

1. **Loop testing:**

In this Loop testing, all the loops are tested to all the limits possible.

**TESTCASES TABLE**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **DESCRIPTION** | **INPUT** | **Expected Behavior** | **Observed**  **Behavior** | **Status**  **P=passed**  **F=failed** |
| Username | Incorrect | Incorrect Username | Login is unsuccessful | F |
| Email id in Registration | Already Exists | Registration Unsuccessful | Registration Unsuccessful | P |
| Password | Correct | correct password | Successful login | P |
| Password | Incorrect | Incorrect  Password | Unsuccessful login | F |
| Registration | Successful | Successfully Registered | Successfully Registered | P |

**TEST CASES SCREENS**

**Owner Registration email already exists (Test case screen):**



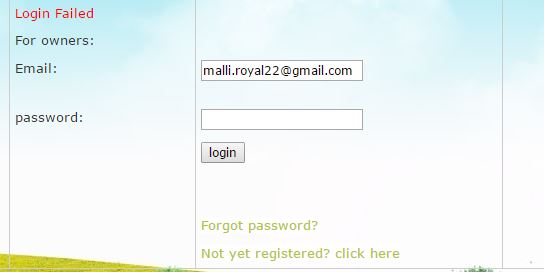
**Email not yet registered (Test case screen):**



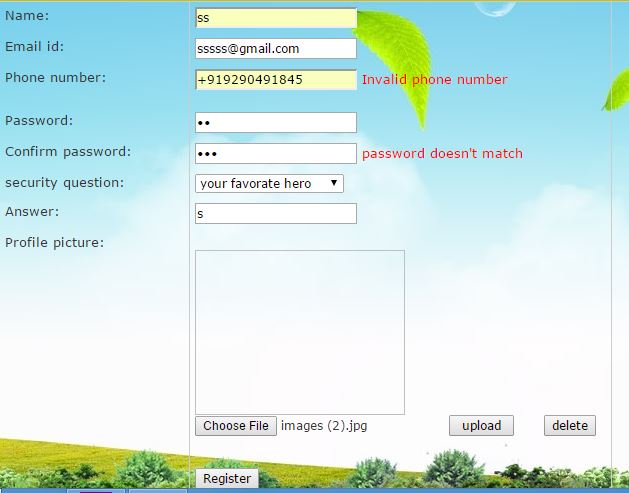
**Password does not match (Test case screen):**



**Entered Password is wrong (Test case screen):**

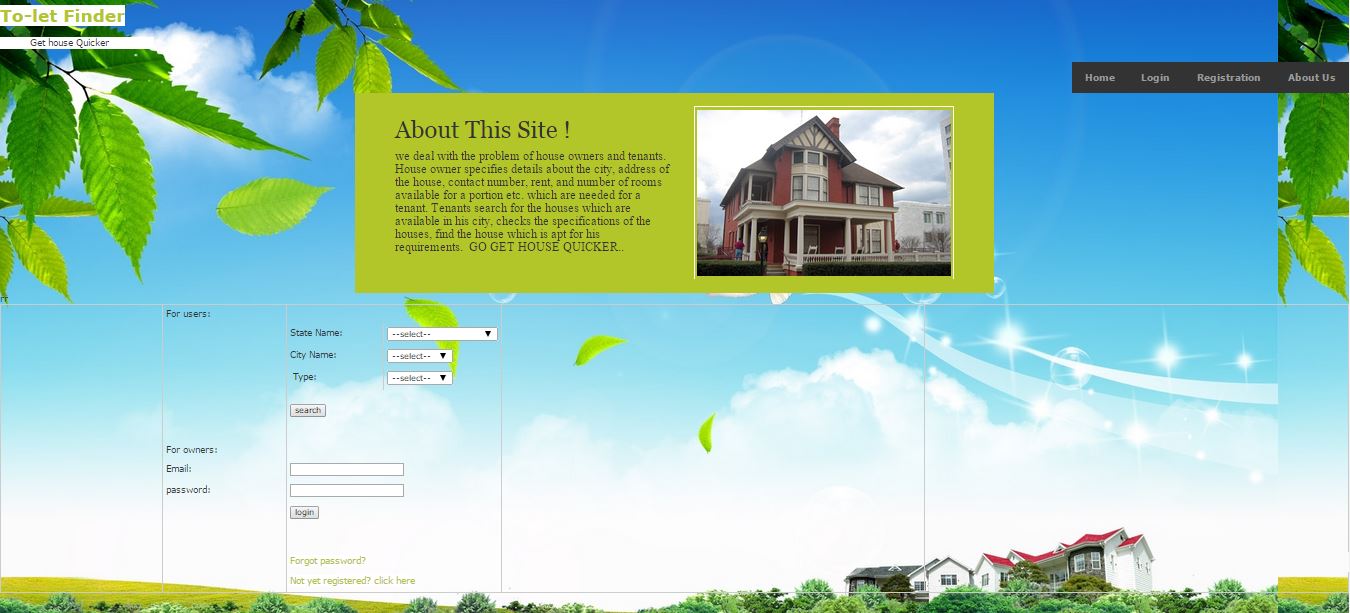


**Registration Validators (Test case screen):**



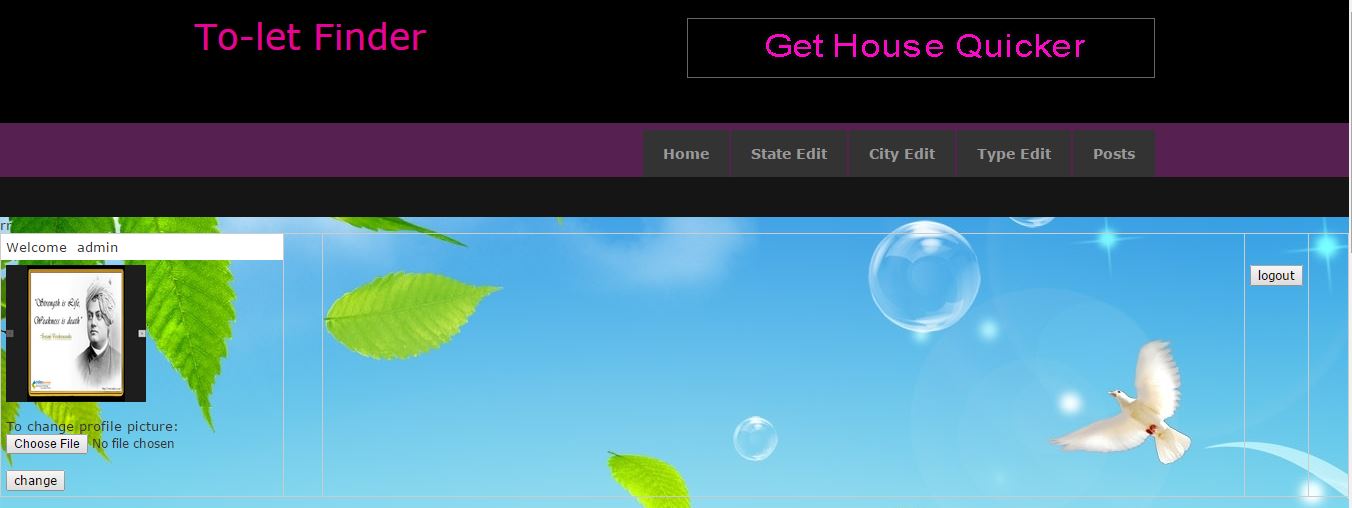
**6.OUTPUT SCREENS**

**Home page:**

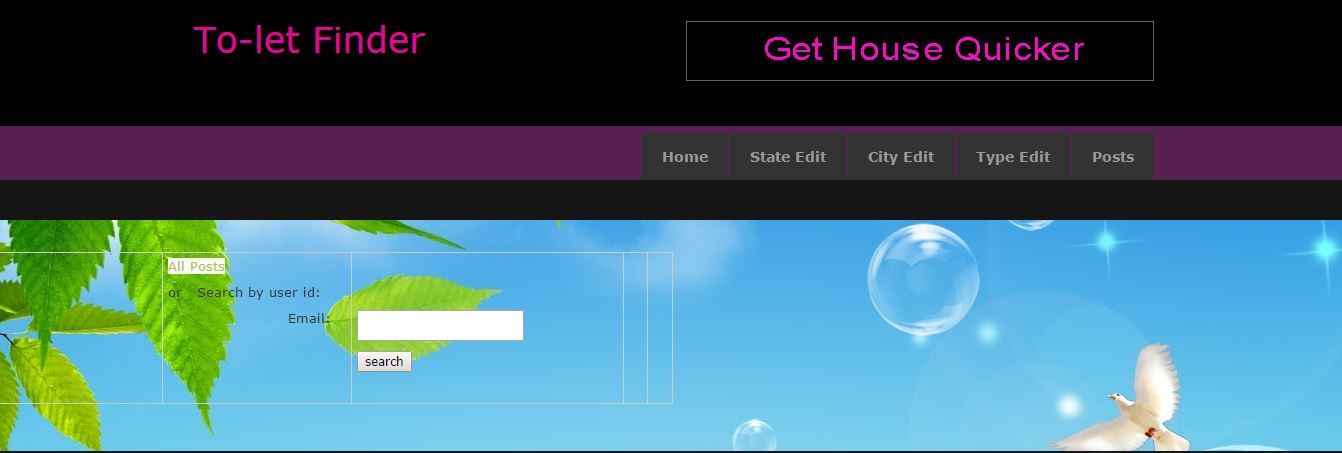


Admin Home page:

**Admin home page:**



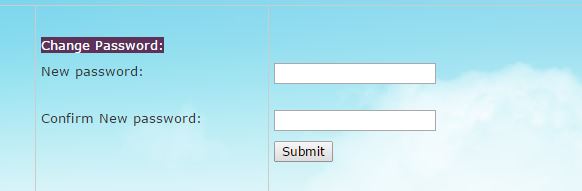
**Admin Posts Search Specification Page:**



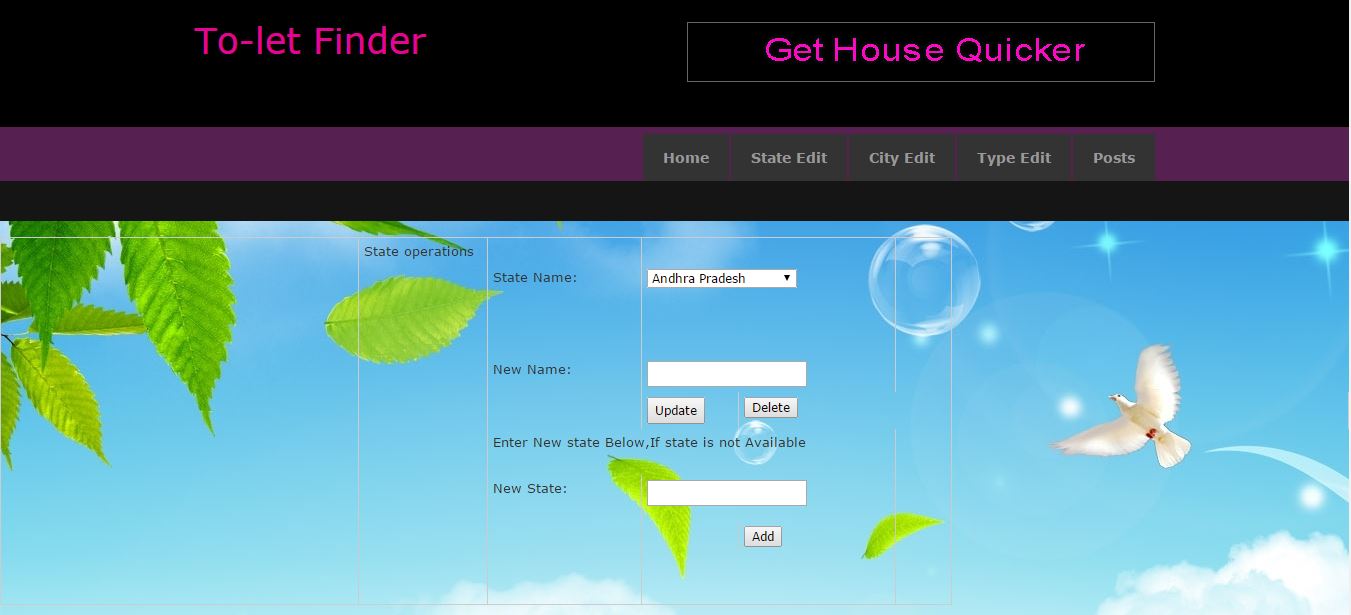
**Admin All Posts:**



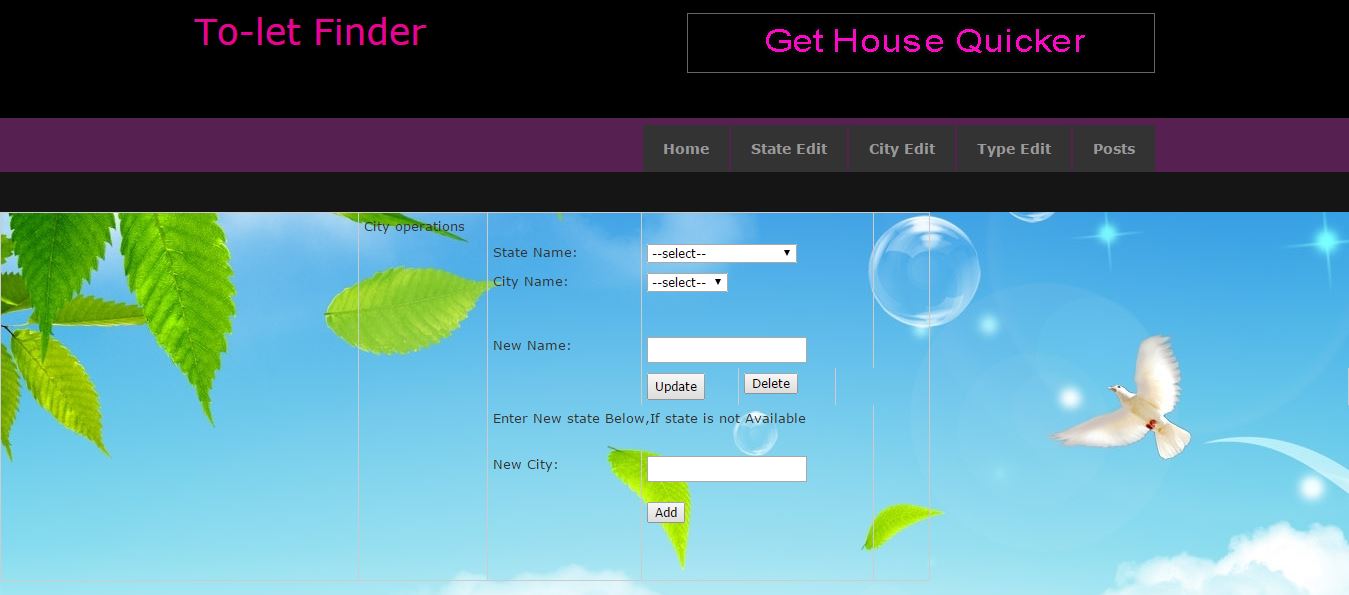
**Admin Password Change:**

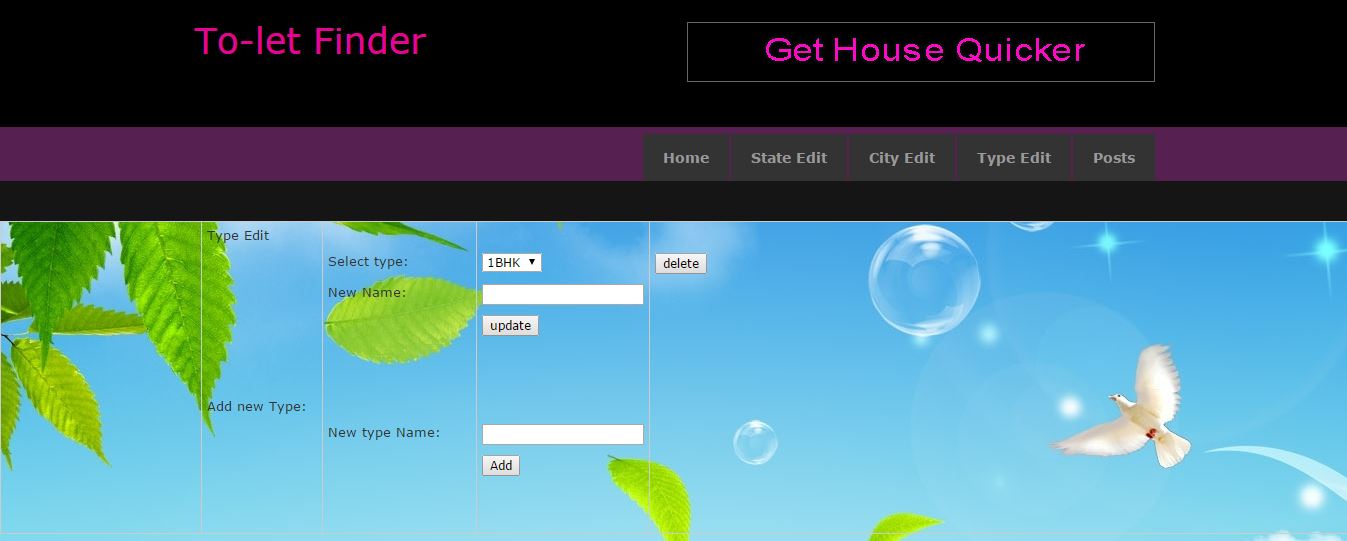


**Admin State Edit:**

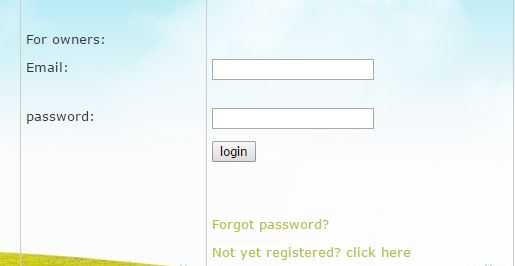


**Admin City Edit:**



**Admin Type Edit:**

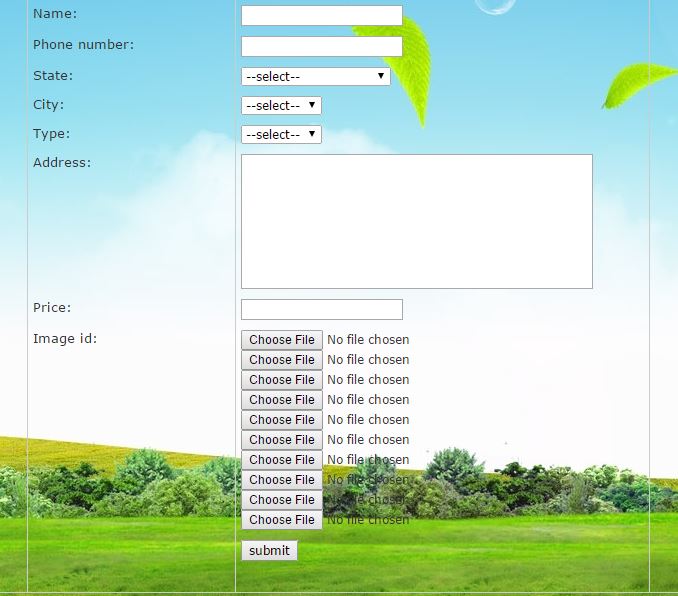
**Owner Login:**



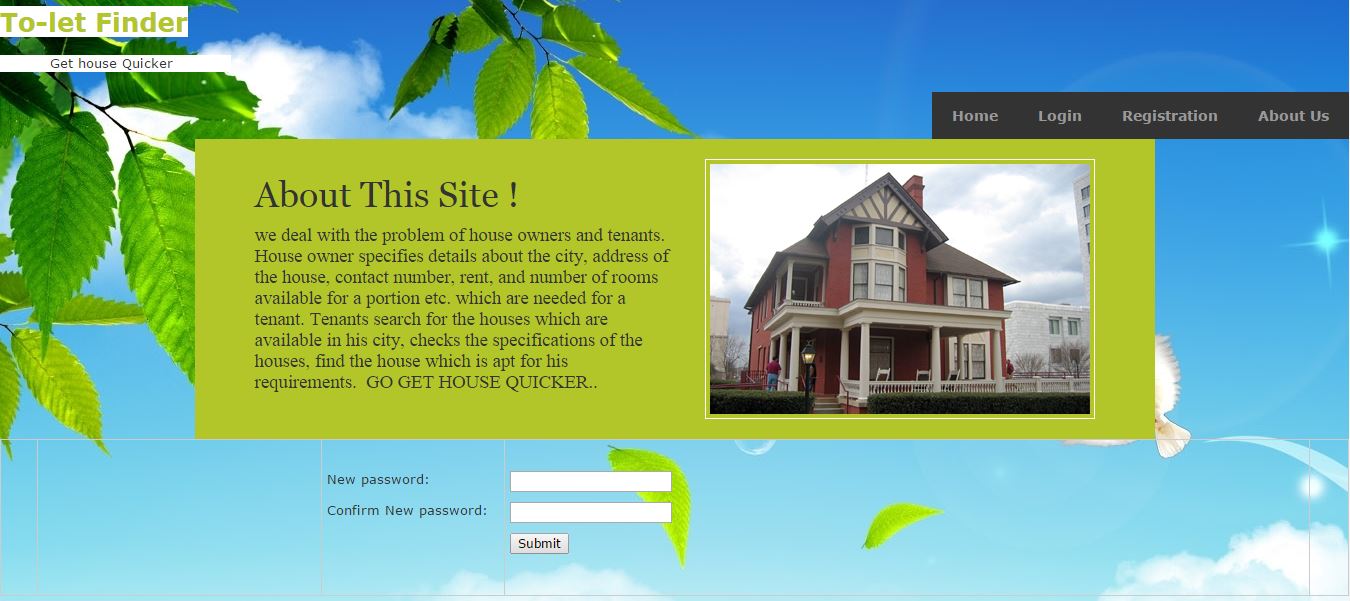
**Owner Home Page:**



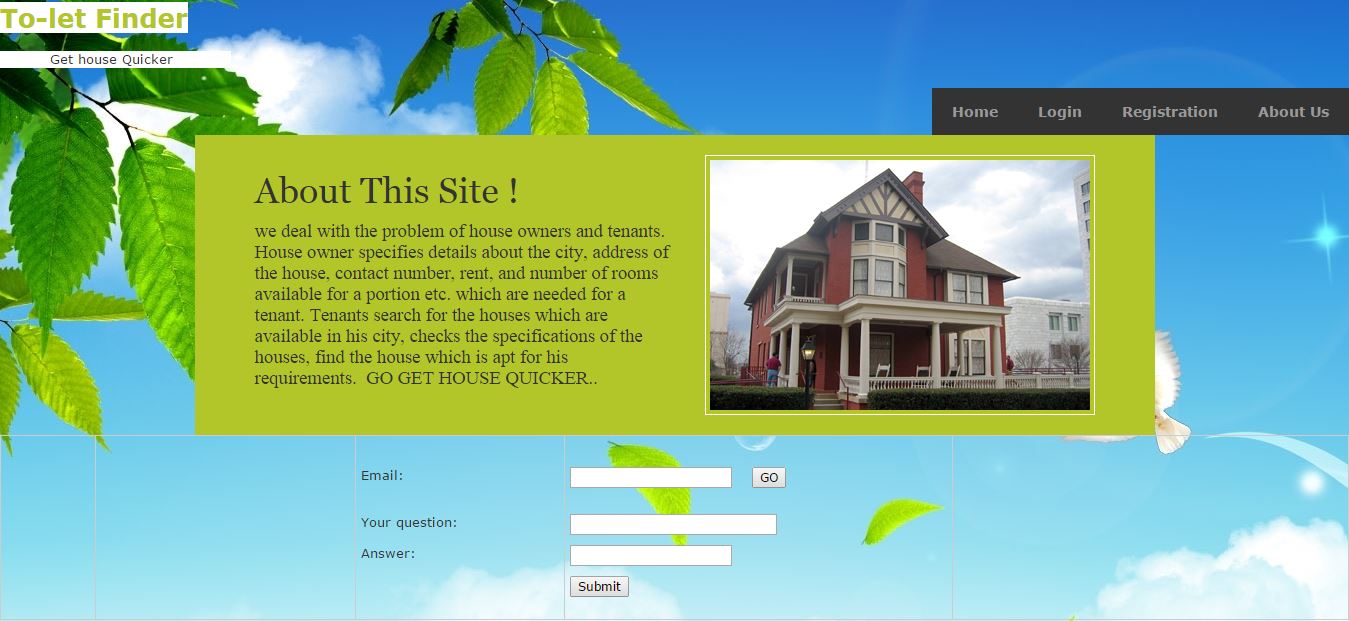
**Owner New Post:**

:

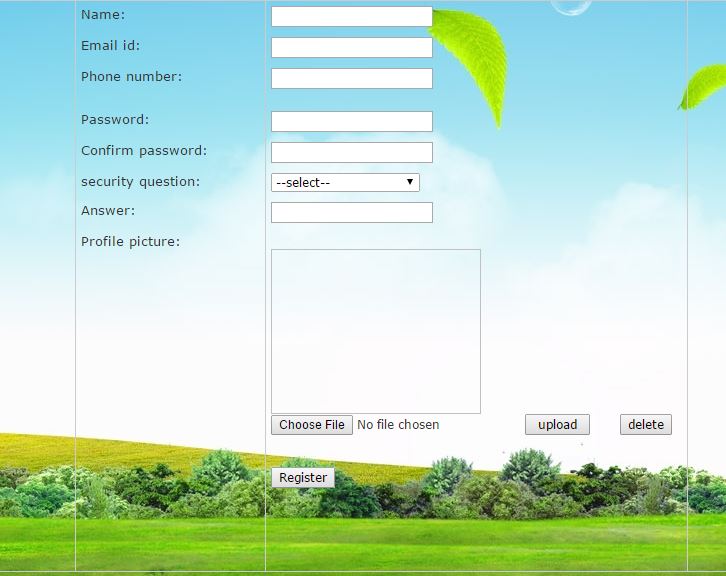
**Owner New Password Page:**



**Owner Password Recovery:**



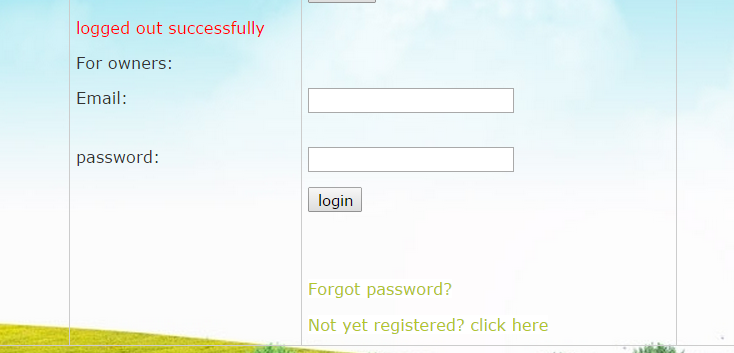
**Owner Registration:**



**Users Search:**



**Logout page:**



**7**.**CONCLUSION**

Through this project, we can overcome the problemsof both tenants and owners.Owners specifies the information of their house like address,contact number, number of portions present in the house etc. The tenants search the houses easily.

This website automates the communication between the owner and the tenant. Moreover it is a reliable source as the information provided is mostly genuine. If any fake data is present, admin takes care of that and does necessary actions to prevent them. In this project, we have advantage as well as disadvantage. Online connectivity (Internet) would be the disadvantage. As, this project is a web based application, the owner or the tenant must be connected online and must have a minimum knowledge of computer and internet. The data in this can be refered any time. It will useful for further reference also.

**8. FUTURE ENHANCEMENT**

The scope of this project “TO-LET FINDER” in the nearby future will be very predominant. The TO-LET FINDER is a web based application. At present, this is restricted to houses only i.e., we’ve developed this only to reduce housing problem. In future, there is great scope for this i.e., it can be applied to lands, vehicles, articles which are available for **rent or sale**. We can also look forward for an online auction center as this project can be easily moulded to that. So, we can easily say that this project has a very good scope in many fields.

**9.BIBLIOGRAPHY**

MICROSOFT .NET WITH C# : Microsoft .net series

ASP .NET 2.0 PROFESSIONAL : Wrox Publishers

ASP .NET WITH C# 2005 : Apress Publications

MICROSOFT ASP .NET 2.0 APP : Wrox Professional Guide

**WEBSITES**

[www.google.com](http://www.google.com)

[www.microsoft.com](http://www.microsoft.com)