Design and Development of a Real Estate Management System Using Information Technology

N. Dharshinee[1], Dr. M. Vargheese[2], Mrs.S. Valarmathi[3]

[1]*Master Of Technology In, Information Technology, Psn College of Engineering and Technology (Autonomous) Melathediyoor Tirunelveli*

[2]*M. Tech., Ph.D, Professor and Head, Department of CSE, PSN College of Engineering and Technology, Tirunelveli*

[3]*Assitant Professor, Department of CSE, PSN College of Engineering and Technology, Tirunelveli*

Abstract—**This system will provide facility to the user to search Residential and Commercial property and view property.**

This system will provide facility to view the property by admin and user. User will able to upload the property information to the site and able to manage it. This system will provide facility to the user to fill up their requirement and according to their Requirement Admin can add the Requirement property. This system will provide facility to the user to publish advertise to the Site and view. This system will provide facility to the user to feedback to the site. Nowadays many real estate companies still using traditional approach of promotion and selling of their property. This project aims at development of an efficient management system for the real estate industry. Now builders can advertise about their new project online and also sell properties online using internet.

I. INTRODUCTION

1 Background of the Study

The real estate sector is a dynamic and data-intensive domain that involves property transactions, asset management, customer relationship management, and financial operations. Traditional real estate management systems rely heavily on manual documentation, paper- based records, and fragmented data storage, which result in inefficiencies, redundancy, and increased risk of data inconsistency.

With the evolution of Information Technology (IT) and web-based information systems, there is a growing need to digitize and automate real estate operations. A Real Estate Management System (REMS) is an integrated software solution designed to facilitate data management, process automation, and decision support. By leveraging technologies such as client-server architecture, relational database management systems (RDBMS), and cloud computing, REMS enables efficient handling of large-scale property data and user interactions.

Background And Motivation

1.1 Customer Module

- A Customer Signup
- Customer Login
- Search Property
- Search based on city
- Register Property
- Put Ad for requirement
- Registered Ad list
- Send Message
- View Message
- View FAQ
- Personal Details
- Information and Communication Technology play a great role in different fields and areas. Real Estate business is one area which is also affected.
- It is necessary to ensure a technologically appropriate, efficient, affordable, userfriendly system to benefit the Real estate business. For this reason, this system is being built.
- In general, the system is based on computer technology that gives service to users. The system will ensure time-saving, saving resources, easy approach to your requirements, as well as bring on more security.
- The Real Estate Management System is designed in order to eliminate the problem of the current system. This accessibility of the information will be a great advantage as it reduced effort.

1.2 Features

- REMS start managing your reservation as soon as your customer enquiries on from your website from their mobile or desktop or directly at your stand. With a simple and elegant workflow, your quotes and booking requests becomes available to you. This real estate management system will help manage all your business processes to ensure efficiencies. Customers and owners of the property will know exactly what the status is at any point in time.
- If required, the system can have multiple levels of management form the administrative prospective. The REMS can have separated offices having restricted privileges to any other property not belonging to their own office. REMS allows' you to build user-defined property definitions for specialized customers and offices. Track costs. Our manager tool provides scheduled and detailed workshop servicing. So you can track the transaction of any sale against office distribution and owner share inside Rental Car Manager.

1.3 System Controls and Security

- REMS system controls allow you to change and add features on the fly. It's all available from the System Setup menus. Top security REMS allows you to specify users and menu access. Most importantly (being a cloud-based software) you can control how your users access the system. You can allow user access from the office, or a specific location, or from home if required.

1.4 REMS (Real Estate Management System)

- Real Estate Management System (REMS) is an online real estate software application that manages the overall operational activities and processes, starting from the management of the property, to the management of real estate agencies, agents, clients and financial transactions. It provides comprehensive reports for managing the Real Estate agency performance and efficiency, and enables the management for a better decision-making.
- Whether you operate 10 properties or 1000, REMS is affordable for all sizes of sales or for rent, it scales smoothly to your requirements.
- If required, the system can have multiple levels of management form the administrative prospective. The REMS can have separated offices having restricted privileges to any other property not belonging to their own office. REMS allows' you to build user-defined property definitions for specialized customers and offices. Track costs. Our manager tool provides scheduled and detailed workshop servicing. So, you can track the transaction of any sale against office distribution and owner share inside Rental Car Manager.

1.5 System Controls and Security

- REMS system controls allow you to change and add features on the fly. It's all available from the System Setup menus. Top security REMS allows you to specify users and menu access. Most importantly (being a cloud-based software) you can control how your users access the system. You can allow user access from the office, or a specific location, or from home if required.

1.6 REMS (Real Estate Management System)

- Real Estate Management System (REMS) is an online real estate software application that manages the overall operational activities and processes, starting from the management of the property, to the management of real estate agencies, agents, clients and financial transactions. It provides comprehensive reports for managing the Real Estate agency performance and efficiency, and enables the management for a better decision-making.
- Whether you operate 10 properties or 1000, REMS is affordable for all sizes of sales or for rent, it scales smoothly to your requirements.

II LITERATURE SURVEY

This module keeps record of the data and property online. And helps to maintain the record which saves time and makes the search of available property easily. The main application of the software specified is it can be used by brokers, builders, at construction sites etc in an efficient manner. This saves time and power and thus by benefiting user. The objective of this software is that it will contain all information about owner, customers and available property. Customer & property records can be maintained easily. The software is webbased application. This software will contain all records with ease and also save the Data of the customers and owner. This document will lead to a unique interpretation of product.

Scope

The real of World Wide Web has spread across millions of house hold, so naturally, Internet has become by far the best platform for real estate marketing today.

Now a days when everything is online, how is it possible that real estate left web application behind? There are lots of real estate companies who advertise their property online so idea behind developing this application is that their property can also sell, or buy or even rent property using this. These applications are not widely popular but in future, they have large scope of growth.

This website is an online real estate management through which individual agents or buyer can maintain their property document keeping and managing property registration and also access its information and manage all the adding, updating, deleting the ads and some of its tasks. The admin user can inform their agents for regarding to property and update the information regarding property and cancellation of property or changing buyer choice.

The system is very useful for the companies or builders that can post and edit the information of their properties and their personal info and admin can monitor records of all of them. The system is also useful which also keeps track of Account details of buyers and Investors and also RES Industry.

III SYSTEM REQUIREMENTS:

There some System Requirement for creating Online Quiz. Here are some System Requirements

3.1 Windows:

- Microsoft Windows 8/10 64-bit.
- Memory (RAM): Minimum 1 GB; Recommended 4GB or above.
- Hard Drive: Minimum 32 GB; Recommended 64 GB or more.
- Processor: Minimum 1GHz; Recommended 2GHz or more
- Ethernet connection (LAN) or a Wireless adapter (Wi-Fi).

3.2 Visual Studio:

- Requires 1 GB of RAM or more recommended.
- Requires 1 GB of available hard disk space.
- Requires 1024 by 728 or higher display resolution.

3.3 Crome Os:

- Requires 8 GB RAM or more recommended.
- Requires 4 GB of available disk space minimum.
- 1280 x 800 minimum screen resolutions.
- Intel i5 or higher U series or higher recommended
- 3.1.4 MY SQL:
- MY SQL 8.0
- MY SQL 8.0.32

IV. METHODOLOGY PHASE

4.1 Feature Extraction Modules

Figure above shows the methodology that will be used in this research. The iterative model is best thought of as a cyclical process. After an initial planning phase, a small handful of stages are repeated over and over, with each completion of the cycle incrementally improving and iterating on the software. Consider an iterative life cycle model which consists of repeating the following four phases in sequence. The seven stages involved in the iterative model are as follow:

- Initial Planning Phase
- Planning Phase
- Requirement Phase
- Analysis and Design Phase
- Implementation and Testing Phase
- Evaluation Phase
- Deployment Phase

4.1.1 Initial Planning Phase:

In this phase, all the activities are plan according to a given period to complete the system. This phase begins with the brainstorming session with the supervisor to come up with the idea and the title for the project. From the brainstorming session with the supervisor, the title for this project is decided which is REAL ESTATE MANAGEMENT SYSTEM and will implement the rulebased algorithm. This system will develop as a webbased system by using JAVA SCRIPT, HTML and MY SQL. Next, the journals about real estate management have been selected to be a guideline during development.

4.1.2 Planning Phase:

After the initial planning, the planning phase has taken place. In planning, the problem statements have been defined through several study and observation. The main discussion of a problem can be defined based on lecturer need spend more time on grading. Next, the scope also was defined for lecturer to setup a user-friendly real estate management system.

4.1.3 Requirement Phase:

During this phase, existing systems are analyzed and all the requirements that are needed to develop the new system are identified. In this phase, the information regarding REAL ESTATE MANAGEMENT SYSTEM either in the form of journal, articles or research papers are gathered
and studied.

4.1.4 Analysis And Design Phase:

Firstly, we have to identify entities, relationships and attributes in REAL ESTATE MANAGEMENT SYSTEM. Then, we can model the data. The purpose of data modeling is to help us in understanding the data and to manage data as resource. We use MY SQL as a database to store data and bootstrap to design interface of the system. Then, system is created and the development of the prototype based on the functionalities that will be built such as add REAL ESTATE MANAGEMENT.

4.1.5 Implementation And Testing Phase:

This phase is important because all the coding is coded. was built by using JAVASCRIPT, HTML and MY SQL, PHP. Firstly, admin module was developed because admin module is very important where only admin can manage the data of the system. Then finally, a dealer or sellers sell their properties in given categories and their specified price ranges.

4.1.6 Evaluation Phase:

Every property will be evaluated by a mechanism. the users can get a proper report of the property and the evaluation report and a score based on the report the property will get a score by which the property and the seller can be recommended in case of any future sales by the person.

4.1.7 Deployment Phase:

Once the evaluation is done, the real estate management application is deployed in the environment so that the targeted sellers and buyers can use the system. The system will be released to end user to implement whether the application correct functioning or not.

4.2 Application Technologies:

A real estate management system is the complete agenda of real estate
management that has got multiple features and functionalities. The real estate management uses apache server through which the real estate system deals with apartments and properties are displayed, managed and allocated. This type of real estate management system has got multiple benefits few of them is that it eliminates the dependency on paper work and heavy work in the offices, and eliminates any sort of manual workload which is too much in the case of searching for a property with any help of technology.

4.2.1 Web Server:

Role of a web server is to communicate between the client side and server side by storing, processing and delivering web pages to the client side. Apache HTTP Server Project is a collaborative software development effort aimed at creating a robust, commercial grade, featureful, and freely-available source code implementation of an HTTPS (Web) server. The port number can be assigned programmatically and the web server is accessed through this port.

4.3 Client-Side Programming Language

For the development and designing of web pages HTML, CSS and JavaScript were used. HTML for creating the webpage, CSS for styling and for adding further functionalities JavaScript was used. It is the program that runs on the client machine (browser) and deals with the user interface/display and any other processing that can happen on client machine like reading/writing cookies.

4.3.1 HTML:

Hypertext markup language (HTML) is used for creating web pages and web applications. It describes the structure of the web pages. Information from the HTML documents is sent to the web browser to render or display on the multimedia pages. Html used to describe the structure of a document by presenting a document in a heading, paragraph, image, list, links and other objects like that. Web browser presents the HTML document by using its tags. HTML, URLs play a role in these situations:

- Linking to another document or resource.
- Linking to an external style sheet for inclusion in a page.
- Images objects and applets for inclusion in a page.
- Image maps.
- Form submission
- Frames
- Citing an external reference.
- Referring to metadata conventions describing a document.

HTML has been developed with the vision that all manner of devices should be able to use information on the Web; PCs with graphics displays of varying resolution and color depths, cellular telephones, hand held devices, devices for speech for output and input, computers with high or low bandwidth, and so on. As the Web community grows and its members diversify in their abilities and skills, it is crucial that the underlying technologies be appropriate to their specific needs. HTML has been designed to make Web pages more accessible to those with physical limitations.HTML now offers greater support for diverse human languages within a document. This allows for more effective indexing of documents for search engines, higherquality typography, better text-to-text-speech conversion, correct hyphenation etc. An HTML file can link to a cascading style sheet or JS file is usually at the top of the document with a specified file path which will contain information about which colors to use, which fonts to use and other HTML element rendering information. Instead of using plugins, multimedia can be placed within the HTML code.

4.3.2 CSS

It is important to make HTML pages attractive to the users and for this purpose, developers choose colors, nice fonts, and different layouts. All this work is done by CSS. CSS describes how HTML elements are to be displayed on screen, paper, or in other media. CSS saves a lot of work.CSS (Cascading Style Sheets) is used to style and layout web pages for example, to alter the font, color, size, and spacing of your content, split it into multiple columns, or add animations and other decorative features in short, CSS is used for styling of a HTML document. It is designed in such a way that enables separation of content and presentation so that it makes easier any change of content without interfering with a design. CSS also enables multi web pages to share the single CSS file for styling to reduce repetition and complexity. Style Sheets simplify HTML markup and largely relieve HTML of the responsibilities of presentation. They give both authors and users control over the presentation of documents-font information, alignment, colors, etc. The mechanism for associating a style sheet with a document is independent of the style sheet language. These have been used to separate data form presentation. By using these stylesheets throughout the project, a uniform look and feel can be maintained for all the HTML elements and tags that have been used in the project. If there is any revamp the way the content has been presented in the website, the changes can be made to the appropriate style sheet, which will be reflected across style sheets. With CSS, you can control the color, font, size of text, the spacing between elements, how elements are positioned and laid out, what background images or background colors are to be used, different displays for different devices and screen sizes, and much more! The web gets its name, in fact, from the interdependent network of servers worldwide. The World Wide Web consortium will eventually

phases out many of HTML's presentation elements and attributes. The user's computer is a terminal, or end point, in this network.

4.3.3 PHP

PHP started out as a small opensource project that evolved as more and more people found out how useful it was. Rasmus Lerdorf unleashed the first version of PHP way back in 1994.PHP is a recursive acronym for "PHP: Hypertext Pre- processor". PHP is a serverside scripting language that is embedded in HTML. It is used to manage dynamic content, databases, session tracking, even build entire e-commerce sites. It is integrated with a number of popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server. PHP is pleasingly zippy in its execution, especially when compiled as an Apache module on the Unix side. The MySQL server, once started, executes even very complex queries with huge result sets in record-setting time. PHP supports a large number of major protocols such as POP3, IMAP, and LDAP. PHP4 added support for Java and distributed object architectures (COM and CORBA), making n-tier development a possibility for the first time.PHP is forgiving: PHP language tries to be as forgiving as possible. PHP Syntax is C- Like.

4.4 Server-Side Programming Language:

Server-side programming allows us to instead store the information in a database and dynamically construct and return HTML and other types of files (e.g. PDFs, images, etc.)

4.4.1 MYSQL:

ecosystem. Often called the most popular database and currently enjoying widespread, effective use regardless of industry, it’s clear that anyone involved with enterprise data or general IT should at least aim for a basic familiarity of MySQL. With MySQL, even those new to relational systems can immediately build fast, powerful, and secure data storage systems. MySQL’s programmatic syntax and interfaces are also perfect gateways into the wide world of other popular query languages and structured data stores.

A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or a place to hold the vast amounts of information in a corporate network. In particular, a relational database is a digital store collecting data and organizing it according to the relational model. In this model, tables consist of rows and columns, and relationships between data elements all follow a strict logical structure. An RDBMS is simply the set of software tools used to actually implement, manage, and query such a database.

MySQL is integral to many of the most popular software stacks for building and maintaining everything from customer-facing web applications to powerful, data- driven B2B services. Its open-source nature, stability, and rich feature set, paired with ongoing development and support from Oracle, have meant that internet- critical organizations such as Facebook, Flickr, Twitter, Wikipedia, and YouTube all employ MySQL backends. Because MySQL enjoys the most widespread use in many industries, business users from new webmasters to experienced managers should strive to understand its main characteristics. Deciding whether to use this technology, and communicating about it effectively, starts with a review of MySQL’s basic availability, structure, philosophy, and usability.

4.4.2 Javascript:

JavaScript is a dynamic programming language that's used for web development, in web applications, for game development, and lots more. It allows you to implement dynamic features on web pages that cannot be done with only HTML and CSS. Many browsers use JavaScript as a scripting language for doing dynamic things on the web. Any time you see a click-to-show dropdown menu, extra content added to a page, and dynamically changing element colors on a page, to name a few features, you're seeing the effects of JavaScript. HTML defines the structure of your web document and the content therein. CSS declares various styles for the contents provided on the web document.HTML and CSS are often called markup languages rather than programming languages, because they, at their core, provide markups for documents with very little dynamism. JavaScript, on the other hand, is a dynamic programming language that supports Math calculations, allows you to dynamically add HTML contents to the DOM, creates dynamic style declarations, fetches contents from another website, and lots more. JavaScript is a high-level, often just-in-time compiled language that conforms to the ECMAScript standard.[10] It has dynamic typing, prototype-based object- orientation, and first-class functions. It is multi-paradigm, supporting event- driven, functional, and imperative programming styles. It has application programming interfaces (APIs) for working with text, dates, regular expressions, standard data structures, and the Document Object Model (DOM).

The ECMAScript standard does not include any input/output (I/O), such as networking, storage, or graphics facilities. In practice, the web browser or other runtime system provides JavaScript APIs for I/O. JavaScript engines were originally used only in web browsers, but are now core components of some servers and a variety of applications. The most popular runtime system for this usage is Node. Although Java and JavaScript are similar in name, syntax, and respective standard libraries, the two languages are distinct and differ greatly in design.

4.5 Methodology:

4.5.1 Admin Module:

- Dashboard: In this section, admin can see all detail in brief like total property type, total country, total state, total city, total agent, total owner, total buyer(user) ant total property listed.
- Property Type: In this section, the admin can manage property type (add/update).

- Country: In this section, admin can manage country (add/update).
- State: In this section, admin can manage state (add/update).
- City: In this section, the admin can manage city (add/update).
- Owner: In this section, admin can view the detail of the owners.
- Agents: In this section, admin can view the detail of agents.
- User: In this section, admin can view the detail of the user.
- List of properties: In this section, admin can view details of the property listed,
- Reviews: In this section admin, can view reviews and also approved, disapproved and delete the reviews which is given by users.
- Pages: In this section, the admin can manage about us and contact us pages.
- Search Property: In this section admin, can search the listed property by its property id, name and mobile number.

4.5.2 User Module:
Property Owner:

- Home Page: Owner can view the home page of the real estate management system
- About: Owner can view about us page.
- Properties: Owner can view own and other owner properties they can view property type wise, Status wise and city wise.
- Contact us: Owner can view contact us page.
- My Account: In this, there is three sections: (A) User Profile
- In this section, the owner will do the following activity
- Edit Profile: Owner can edit his/her own profile.
- Change Password: Owner can change his/her own password.
- Add Property: Owner can add his/her own property
- My properties: Owner can see his/her own listed properties.
- Received Enquiries: Owner can view receive enquiries against his/her own listed properties and also answer the enquiries.
- Answer Enquiries: Owner can view answer enquiries.
- Logout: Owner can logout from own account.

- (B)Change Password: Owner can change his/her own password. (C)Logout: Owner can logout from own account.

4.5.3 Brokers/Agent:

- Home Page: Agents can view the home page of the real estate management system.
- About: Agents can view about us page.
- Properties: Agents can view own and other owner properties they can view property type wise, Status wise and city wise.
- Contact us: Agents can view contact us page.
- My Account: In this, there are three sections: (A)User Profile
- In this section, Agents do the following activity:
- Edit Profile: Agents can edit his/her own profile.
- Change Password: Agents can change his/her own password.
- Add Property: Agents can add his/her own property.
- My Properties: Agents can see his/her own listed properties.
- Received Enquiries: Agents can view receive enquiries against his/her own listed properties and also answer the enquiries.
- Answer Enquiries: Agents can view answer enquiries.
- Logout: Agents can logout from own account.
- (B)Change Password: Agents can change his/her own password.
- (C)Logout: Agents can logout from own account.

4.5.4 Buyer (Simple User):

- Home Page: User can view the home page of the real estate management system
- About: User can view about us page.
- Properties: User can view properties they can view property type wise, Status wise and city wise and put his/her enquiries against any property
- Contact us: User can view contact us page.
- My Account: In this, there is three sections: (A)User Profile
- In this section, Agents do the following activity
- Edit Profile: Agents can edit his/her own profile.
- Change Password: Agents can change his/her own password.

- Logout: Agents can logout from own account.
- (B). Change Password: Agents can change his/her own password.
- (C). Logout: Agents can logout from own account.
- The Website works in This Manner.

V. SIMULATION RESULTS & DISCUSSIONS

This chapter details the experimental setup, software environment, dataset preparation, and the rigorous quantitative analysis performed to validate the effectiveness of the proposed DCNN-LBP fusion model. The discussion focuses on comparing the performance metrics of the hybrid model against standalone DCNN and LBP baselines.

5.1Software Description & Dataset

The quality and diversity of the dataset are paramount for testing the robustness of the model.

- Dataset Source: The experiments utilized the [Name the main dataset, e.g., PlantVillage dataset], which comprises images of various plant species and their common diseases (e.g., Apple Scab, Grape Black Rot, Tomato Early Blight). The dataset was filtered/expanded to include $N$ total images across $C$ distinct disease classes plus one healthy class.
- Data Split: The dataset was divided into three mutually exclusive sets:
  - Training Set (70%): Used to train the model weights.
  - Validation Set (15%): Used for hyperparameter tuning and early stopping during training.
  - Test Set (15%): Used exclusively for final, unbiased performance evaluation.
- Data Augmentation: To prevent overfitting and enhance the model's generalization capability to real-world variations, standard data augmentation techniques were applied to the training set. These included random rotation, zooming, horizontal and vertical flipping, and minor brightness adjustments.

This section outlines the hardware and software used, ensuring the reproducibility of your results.

- Programming Language & Libraries: The entire system was developed using Python (version X.X) as the core programming language. Key deep learning frameworks included TensorFlow/Keras (for DCNN implementation and training) and standard image processing libraries like OpenCV and Scikit- image (for LBP feature extraction and pre-processing).
- Hardware Specifications: All simulation and training runs were executed on a dedicated machine equipped with a GPU (e.g., NVIDIA GeForce RTX 3070) to accelerate training, an Intel Core i7/i9 CPU, and XX GB of RAM. The use of a GPU was essential for efficient back-propagation during the DCNN training phase.
- Operating System: Experiments were conducted on a Linux/Windows operating system environment (e.g., Ubuntu 20.04).

Model Training

- DCNN Training: The base DCNN (e.g., pre-trained ResNet-18) was trained using the Adam optimizer with a learning rate of $10^ {-4}$. Categorical Cross-Entropy was used as the loss function, and training was stopped early if the validation loss did not improve for a predefined number of epochs (patience).
- LBP Feature Extraction: LBP features ($F_{L}$) were extracted for all images using the parameters $P=8$ and $R=1$. These feature vectors were calculated once and stored.
- Fusion Model Training: The concatenated feature vectors ($F_{F} = [F_{D}\ || \ F_{L}]$) were fed into the final classification head. This head was then trained to optimize the weights based on the combined information.

Programe For Example

Source Code:

About:

```
<?php session_start(); error_reporting(0);
include('includes/dbconnection. php');
?>
<!DOCTYPE html>
<html dir="ltr" lang="en-US">

<head>
<! -- Fonts
========================================
======= -->
```

```
  <link
href="https://fonts.googleapis.com/css?family=Open
+Sans:300,300i,400,400i,600,60 0i,700,7
00i,800,800i%7CPoppins:100,100i,200,200i,300,300
i,400,400i,500,500i,600,600i,70
0,700i,8
00,800i,900,900i" rel="stylesheet">
<! -- Stylesheets
========================================
======= -->
<link href="assets/css/external.css" rel="stylesheet">
<link        href="assets/css/bootstrap.min.css"
rel="stylesheet">
 <link href="assets/css/style.css" rel="stylesheet">
<! -- HTML5 shim, for IE6-8 support of HTML5
elements. All other JS at the end of file. --
>
<! --[if lt IE 9]>
<script src="assets/js/html5shiv.js"></script>
<script src="assets/js/respond.min.js"></script>
 <![endif]-->

<! -- Document Title
========================================
======= -->
<title>Real Estate Management System|| About
Us</title>
</head>

<body>
<! -- Document Wrapper
========================================
======= -->
<div id="wrapper" class="wrapper clearfix">
<?php include_once('includes/header.php’) ;>

<! -- Page Title #1
========================================
====== -->
<section id="page-title" class="page-title bg-overlay
bg-overlay-dark2">
<div class="bg-section">
<img        src="assets/images/page-titles/1.jpg"
alt="Background" />
</div>
<div class="container">
<div class="row">
<div class="col-xs-12 col-sm-12 col-md-6 col-md-
offset-3">
```

```
<div class="title title-1 text-center">
<div class="title--content">
<div class="title--heading">
<h1>About</h1>
</div>
<ol class="breadcrumb">
<li><a href="index.php">Home</a></li>
 <li class="active">About</li>
</ol>
</div>
 <div class="clearfix"></div>
</div>
<! --. title end -->
</div>
<!-- . col-md-12 end -->
</div>
<!-- . row end -->
</div>
<! -- .container end -->
</section>
<! -- #page-title end -->

<! -- about #1
========================================
======= -->
<section id="about" class="about bg-white">
<div class="container">
<div class="row">
<div class="col-xs-12 col-sm-5 col-md-5">
                    <div
                                    class="a
bout--img"><imgclass="img-responsive"
src="assets/images/about/real-estate.jpg"
width="350" height="350" alt="about img"></div>
</div>
<! --. col-md-5 -->
<div class="col-xs-12 col-sm-7 col-md-6 col-md-
offset-1">
<?php



$ret=mysqli_query($con,"select * from tblpage where
PageType='aboutus'");
 $cnt=1; while
($row=mysqli_fetch_array($ret)) {



?>
<div class="heading heading-3">
 <h2 class="heading--title"><?php echo
$row['PageTitle’] ;></h2>
</div>
<!-- . heading-title end -->
<div class="about--panel">

<p><?php echo $row['PageDescription'];></p>
</div>
<!-- . about-panel end -->

<!-- . about-panel end -->
</div>
<?php}?>
</div>

</div>
<! -- .container -->
</section>
<! -- #about end -->

<! -- cta #1
========================================
======= -->
<section id="cta" class="cta cta-1 text-center bg-
overlay bg-overlay-dark pt- 90">
<div class="bg-section"><img
 src="assets/images/cta/bg-
1.jpg" alt="Background"></div>
<div class="container">
 <div class="row">
<div class="col-xs-12 col-sm-12 col-md-6 col-md-
offset-3">
<h3>Join our professional team & agents to start
selling your house</h3>
<a href="contact.php" class="btn btn--
primary">Contact</a>
 </div>
<! --. col-md-6 -->
</div>
<!-- . row -->
</div>
<! -- .container -->
</section>
<! -- #cta1 end -->
<! -- Footer #1
========================================
======= -->
<?php include_once('includes/footer.php');?>
</div>
<! -- #wrapper end -->

<! -- Footer Scripts
========================================
======= -->
<script src="assets/js/jquery-2.2.4.min.js"></script>
<script src="assets/js/plugins.js"></script>
<script src="assets/js/functions.js"></script>
</body>

</html>
```

CONTACT:

```
<?php session_start(); error_reporting(0); include
('includes/dbconnection. php');
?>


<!DOCTYPE html>
<html dir="ltr" lang="en-US">

<head>

<! -- Fonts
========================================
======= -->
   <link
href="https://fonts.googleapis.com/css?family=Open
+Sans:300,300i,400,400i,600,60 0i,700,7
00i,800,800i%7CPoppins:100,100i,200,200i,300,300
i,400,400i,500,500i,600,600i,70
0,700i,8
00,800i,900,900i" rel="stylesheet">
<! -- Stylesheets
========================================
======= -->
<link href="assets/css/external.css" rel="stylesheet">
<link href="assets/css/bootstrap.min.css"
rel="stylesheet">
<link href="assets/css/style.css" rel="stylesheet">
<! -- HTML5 shim, for IE6-8 support of HTML5
elements. All other JS at the end
 of file. --
>
<!--[if lt IE 9]>
<script src="assets/js/html5shiv.js"></script>
```

```
<script src="assets/js/respond.min.js"></script>
<![endif]-->
 <!-- Document Title
===============================================
<title>Real Estate Management System||Contact Us</title>
</head>

<body>
<!-- Document Wrapper
===============================================
<div id="wrapper" class="wrapper clearfix">
<?php include_once('includes/header.php');?>

<!-- Page Title #1
============================================== -->
<section id="page-title" class="page-title bg-overlay bg-overlay-dark2">
<div class="bg-section">
<img src="assets/images/page-titles/1.jpg" alt="Background" />
</div>
<div class="container">
<div class="row">
<div class="col-xs-12 col-sm-12 col-md-6 col-md-offset-3">
<div class="title title-1 text-center">
<div class="title--content">
<div class="title--heading">
<h1>Contact</h1>
</div>
<ol class="breadcrumb">
<li><a href="index.php">Home</a></li>
<li class="active">Contact</li>
</ol>
 </div>
<div class="clearfix"></div>
</div>
 <!-- .title end -->
</div>
<!-- .col-md-12 end -->
</div>
<!-- .row end -->
</div>
<!-- .container end -->
</section>
<!-- #page-title end -->

<!-- Contact #1
=============================================== -->
<section id="contact" class="contact contact-1">
<div class="container">
<div class="row">
<?php



$ret=mysqli_query($con,"select * from tblpage where PageType='contactus'");
$cnt=1; while
($row=mysqli_fetch_array($ret)) {


?>
<div class="col-xs-12 col-sm-12 col-md-3">
<div class="heading heading-2 mb-55">
<h2 class="heading--title">Get In Touch</h2>
</div>
<div class="contact-panel">
<h3>Address</h3>
 <p><?php echo $row['PageDescription'];?></p>
</div>
<!-- . contact-panel -->
 <div class="contact-panel">
<h3>Phone:</h3>

<p>+<?php echo $row['MobileNumber'];?></p>
</div>
<!-- . contact-panel -->
<div class="contact-panel">
<h3>Email:</h3>
<p><?php echo $row['Email'];?></p>
</div>
<!-- .contact-panel -->
</div>
<!-- .col-md-3 end -->

bgwhite">
<div class="col-xs-12 col-sm-12 col-md-8 col- md-offset-1 bg-white p-30

<div class="about--img"><imgclass="img-responsive" src="assets/images/about/images.jpg" width="2000" height="2000" alt="about img"></div>
</div>
<!-- . col-md-8 end -->
</div>
<?php } ?>
</div>
</section>
<!-- #contact end -->

<!-- cta #1
========================================
======= -->
 <section id="cta" class="cta cta-1 text-center bg-overlay bg-overlay-dark pt- 90">
<div class="bg-section"><img src="assets/images/cta/bg-1.jpg" alt="Background"></div>
<div class="container">
<div class="row">
<div class="col-xs-12 col-sm-12 col-md-6 col-md-offset-3">
<h3>Join our professional team & agents to start selling your house</h3>
<a href="contact.php" class="btn btn--primary">Contact</a>
</div>
<!-- .col-md-6 -->
</div>
<!-- .row -->
</div>
<!-- .container -->
</section>
<!-- #cta1 end -->

<! -- Footer #1
========================================
====== -->
<?php include_once('includes/footer.php');?>
</div>
<! -- #wrapper end -->

<! -- Footer Scripts
========================================
======= -->
<script src="assets/js/jquery-2.2.4.min.js"></script>
 <script src="assets/js/plugins.js"></script>
<script src="assets/js/functions.js"></script>
    <script
src="http://maps.google.com/maps/api/js?sensor=true
&amp;key=AIzaSyCiRALrXFl 5vovX0hAkccXXBFh7zP8AOW8"></script>
 <script
src="assets/js/plugins/jquery.gmap.min.js"></script>
<script>
$('#googleMap'). gMap({address: "121 King St,Melbourne, Australia",zoom: 12, maptype: 'ROADMAP', markers: [{address: "Melbourne, Australia", maptype: 'ROADMAP',
icon: {image:
"assets/images/gmap/marker1.png",
iconsize: [52, 75], iconanchor: [52, 75]}}]})
</script>
<script src="assets/js/map-custom.js"></script>
</body>
</html>
```

Map: System Architecture/Ideation Map • Ideation

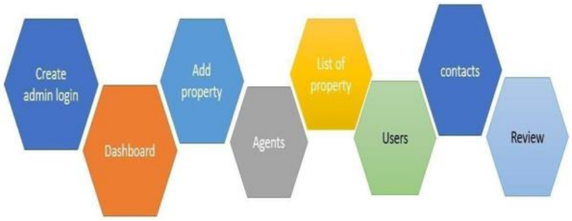
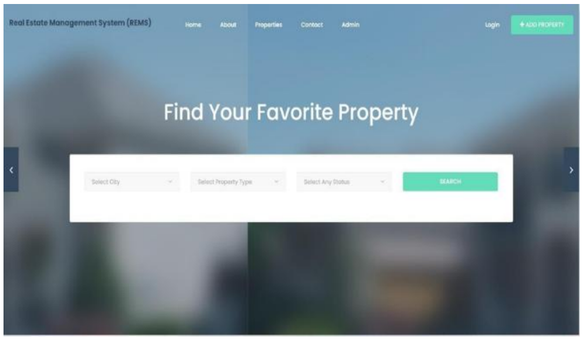
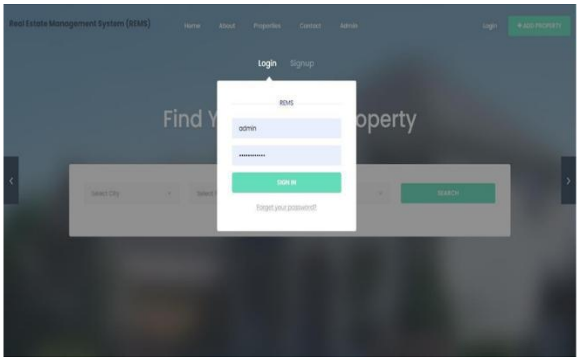
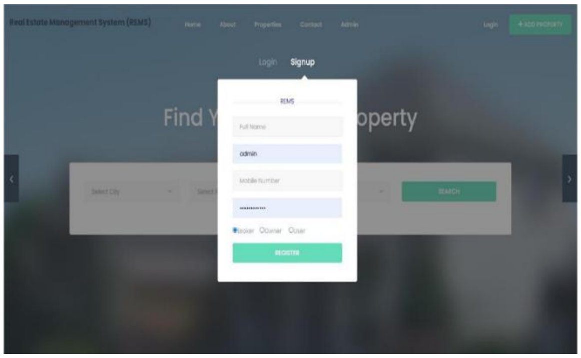


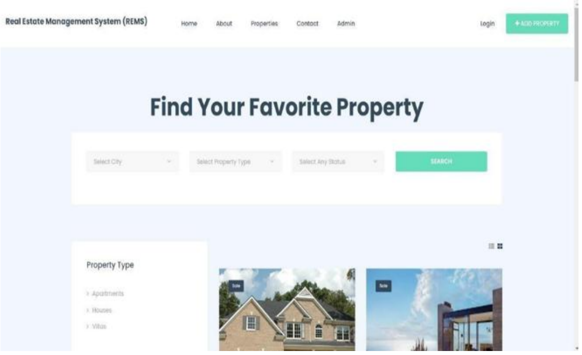
Fig 5.1 ideation map

VI. RESULTS









VII CONCLUSION

Real Estate CONCLUSION The project has given lot of practice in a very wide subject like Web Design. This project has given us a great opportunity to explore our skills, to think towards problem, to accept them and solved them efficiently. It made us implement everything in practice that we had learned and also other more useful parts.

Real Estate The project has given a professional touch towards our learning. It has made us implements not only WEB DESIGNING but also our knowledge on different subject like ERP and CRM and others, It also increases our communication skill, made us handle different types of clients. It has played a major role in our academic career. In technical term to say it has acted as an "interface" between our academic and industry. It has made us more productive, respecting a need of user, understanding the problems and making them understand the project, and it has acted as "gateway" towards the corporate world. It made us understand different factors which are required to build fully fledged project.

On personal level it has flourished our skills made us more courage to tackle the problems, made us given the best in us. After completing this project we felt, we could becall as "True I.T professionals"

Future Work

Based on the successes and observations from this project, several avenues for future research and system enhancement are recommended:

- Real-Time Mobile Deployment: The system could be optimized for latency and ported to a mobile application using frameworks like TensorFlow Lite, enabling farmers to perform instant, on-site disease diagnosis using smartphone cameras.
- Semantic Segmentation Integration: Currently, the system classifies the entire image. Future work could integrate a semantic segmentation network to precisely localize and outline the diseased area on the leaf, providing more detailed information about the infection severity.
- Multi-Modal Fusion: Explore fusing the DCNN and LBP features with other modalities, such as hyperspectral data or environmental metadata (temperature, humidity), to further increase diagnostic accuracy and early detection capability.
- Architecture Optimization: Investigate and test different deep learning backbones (e.g., EfficientNet, Vision Transformers) within the fusion framework to see if further marginal performance gains can be achieved.

VIII. LIST OF ABBREVIATION

Abbreviation	Full Form
REMS	Real Estate Management System
IT	Information Technology
UI	User Interface
UX	User Experience
DBMS	Database Management System
RDBMS	Relational Database Management System
SQL	Structured Query Language
API	Application Programming Interface
REST	Representational State Transfer
HTTP	HyperText Transfer Protocol
HTTPS	HyperText Transfer Protocol Secure
CRUD	Create, Read, Update, Delete
RBAC	Role-Based Access Control
ER	Entity Relationship
UML	Unified Modeling Language
SDLC	Software Development Life Cycle
MVC	Model-View-Controller
OOP	Object-Oriented Programming

IX. ACKNOWLEDGEMENT

Our first and foremost thanks to THE ALMIGHITY for the presenting this opportunity to complete this project.

I thank our Chairman Dr. P. SUYAMBU, Ph.D., Founder and Chairman, PSN GROUP OF INSTITUTIONS for permitting us to do this project work and for his kind patronage.

I thank our Vice Chairmen ER.P.S. JEGANNATH & ER.P.S. JAYARAM, for permitting us to do this project work and for his kind patronage.

I thank our Executive Director Dr. P. SELVA KUMAR, M.E, Ph.D., for supporting and permitting us to do this project work and for his kind patronage.

I thank our Principal Dr.AMIRDHA SHER GILL, Ph.D., for permitting us to do this project work and for her kind patronage.

It gives us immense pleasure to thank Dr. M. VARGHEESE, M. Tech, Ph.D., and Head of the department of Computer Science and Engineering for his invaluable and unstringing co-operation in making the project success.

I have unique privilege to acknowledge our sincere thanks to our project guide Mrs.S. Valarmathi, M.E, Assistant Professor of the department of Computer Science Engineering for her esteemed and resourceful suggestion in the making of this project a success.

I thank all of our staff member for their valuable cooperation, guidance and suggestion given to do this project work.

REFERENCES

Creating the References chapter requires listing the actual sources (papers, books, websites) you consulted. Since I don't know the exact papers, you used for your literature survey and implementation, I will provide a template using highly relevant and foundational academic papers and resources for a project titled "Botanic Guardian: Elevating Leaf Disease Detection with DCNN-LBP Fusion."

[A] Core Deep Learning and CNN Architectures

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770–778.

[2] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep convolutional neural networks for identifying plant diseases," Frontiers in Plant Science, vol. 7, p. 1419, 2016.

[3] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and F.-F. Li, "ImageNet: A large-scale hierarchical image database," in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), Miami, FL, USA, 2009, pp. 248–255.

[B] Local Binary Pattern (LBP) and Texture Analysis

[4] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on feature distributions," Pattern Recognition, vol. 29, no. 1, pp. 51–59, Jan. 1996.

[5] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 7, pp. 971–987, Jul. 2002.

[C] Feature Fusion and Hybrid Methods

[6] M. M. S. El-Shaer, A. M. H. El-Mahlawy, and S. M. El-Kenawy, "Hybrid deep convolutional neural network with local binary pattern (DCNN-LBP) for enhanced image classification," International Journal of Computer Applications, vol. 182, no. 4, pp. 31–36, 2019.

[7] Kamilaris and F. Prenafeta-Boldú, "Deep learning in agriculture: A survey," Computers and Electronics in Agriculture, vol. 147, pp. 293–301, Apr. 2018.

[D] [D] Tools and Datasets

[8] F. Chollet et al., "Keras: The Python deep learning API," GitHub, 2015. [Online]. Available: https://github.com/keras-team/keras

[9] D. P. Hughes and M. Salathé, "An open access repository of images on plant health to enable the development of mobile disease diagnostics," in Proc. IEEE Int. Conf. Computer Vision Workshops (ICCVW), 2015.