

BEDAZZLED MARKETING

Handicraft Products Management System

Project Report Submitted by

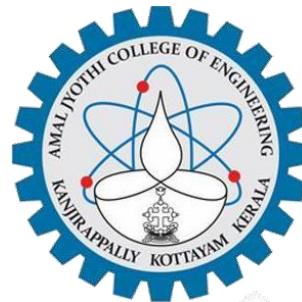
Saran Gopalakrishnan

Reg. No: LAJC18MCA051

In Partial fulfillment for the award of the degree

Of

MASTER OF COMPUTER APPLICATIONS (MCA)
APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY



**AMAL JYOTHI COLLEGE OF ENGINEERING
KANJIRAPPALLY**

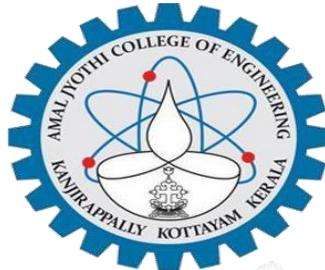
[Affiliated to APJ Abdul Kalam Technological University, Kerala. Approved by AICTE, Accredited by NAAC with 'A' grade. Koovappally, Kanjirappally, Kottayam, Kerala - 686518]

2019-2021

AMAL JYOTHI COLLEGE OF ENGINEERING

[Affiliated to APJ Abdul Kalam Technological University, Kerala. Approved by AICTE,
Accredited by NAAC with 'A' grade. Koovappally, Kanjirappally, Kottayam, Kerala - 686518]

DEPARTMENT OF MASTER OF COMPUTER APPLICATIONS



CERTIFICATE

This is to certify that the project entitled "**Bedazzled Marketing – Handicraft Products Management System**" is a bonafide record of the work done by **Saran Gopalakrishnan LAJC18MCA051**, during the academic year **2019-2021** carried out under our supervision. It is certified that all corrections/suggestions indicated for assessment have been incorporated in the report. This work report has been approved as it satisfies the academic requirements in respect of the project work prescribed by the university for the Master of Computer Applications Degree. Certified further, that to the best of our knowledge the exact work reported herein does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this to any other candidate.

Fr. Dr. Rubin Thottupurathu Jose

Head of the Department

Sr. Elsin Chakkalackal

Project Coordinator

Dr. Juby Mathew

Project Supervisor

External Expert appointed by the university

DECLARATION

I hereby declare that the project report “**Bedazzled Marketing – Handicraft Products Management System**” is a bonafide work done at Amal Jyothi College of Engineering, towards the partial fulfilment of the requirements for the award of the Degree of Master of Computer Applications (MCA) from APJ Abdul Kalam Technological University, during the academic year 2019-2021.

Date.....

Saran Gopalakrishnan

KANJIRAPPALLY

Reg. No: LAJC18MCA051

ACKNOWLEDGEMENT

First and foremost, I thank Almighty God for his gracious guidance through the project. I take this opportunity to express my gratitude to all those who have helped me in completing the project successfully

It has been said that gratitude is the memory of the heart. I acknowledge my deep sense of gratitude to our manager **Rev. Fr. Dr. Mathew Paikatt** for providing all the infrastructural facilities for us, our Principal **Dr. Z V Lakaparampil** for providing good faculty for guidance.

I take the immense pleasure in expressing my thanks to Head of the Department of Master of Computer Applications, **Rev. Fr. Dr. Rubin Thottupurathu Jose**, for his kind patronages in making this project a successful one. I would like to extend my sincere thanks to our project coordinator **Sr. Elsin Chakkalackal** and my project guide **Dr. Juby Mathew** for his guidance and cooperation, without which this would not have been a success.

I am indebted to my beloved teachers whose cooperation and suggestions throughout the project which helped me a lot. I also thank all my friends and classmates for their interest, dedication and encouragement shown towards the project. I convey hearty thanks to parents for the moral support, suggestion and encouragement to make this venture a success.

Saran Gopalakrishnan

ABSTRACT

“BEDAZZLED MARKETING” is a web application project. Most of the Handcrafter could not get desired price for their product and most case a broker or a third party will get most benefit than the actual producer. Also a large area of marketing handicraft product is depending on foreigners but unfortunately our Govt. Could not provide an online sale of such product to a reach of worldwide.

The project entitled ‘BEDAZZLED MARKETING’ is a website which provides a high quality online service and a best place to meet both for the Producers and Consumers. It helps Ordinary Customers to buy high quality product from the Producers. User can request for a product, so they can purchase crafts on their demand. The proposed system is a website in which user can buy handmade creative products as well as handicrafters can sell their products efficiently. We will also provide users to give customization comments to customize their products, they can view the seller or brand details, payment details etc.

CONTENT

Sl. No	Topic	Page No
	Introduction	1
1	Using Git as a Version Control System	2
1.1	Introduction to GitHub	3
1.2	Working With Git	4-6
2	Project Documentation	7
2.1	Introduction	8
2.1.1	Project Overview	8
2.1.2	Project Specification	8
2.2	System Study	9
2.2.1	Introduction	9
2.2.2	Existing system	10
2.2.3	Proposed system	10-11
2.3	Requirement Analysis	12
2.3.1	Feasibility Study	12
2.3.1.1	Economic Feasibility	12
2.3.1.2	Technical Feasibility	12
2.3.1.3	Behavioral Feasibility	13
2.4	Requirement Modeling	13
2.4.1	UML Diagrams	13-26
2.5	System Specification	26
2.5.1	Hardware Specification	26
2.5.2	Software Specification	26
2.6	Software Description	27
2.6.1	PHP	27
2.6.2	MySQL	27-29

2.7	System Design	29
2.7.1	Architectural Design	30
2.7.2	Module Design	30-32
2.7.3	Data Base Design	32-41
2.8	System Testing	42
2.8.1	Introduction	42
2.8.2	Test Plan	43
2.8.2.1	Unit Testing	43
2.8.2.2	Integration Testing	44
2.8.2.3	Validation Testing	44
2.8.2.4	User Acceptance Testing	45
2.8.3	Test Case	45-46
2.9	Implementation	46
2.9.1	Implementation Procedure	47
2.9.2	User Training	48
2.9.3	Operational Document	48
2.9.4	System Maintenance	48
2.10	Conclusion & Future Enhancements	49
2.10.1	Future Enhancement	49
2.10.2	Conclusion	49
2.11	Bibliography	50
2.12	Appendix	50
2.12.1	Sample Code	50-73
2.12.2	Screen Shots	74-84

LIST OF ABBREVIATIONS

IDE -	Integrated Development Environment
HTML -	Hyper Text Markup Language
CSS -	Cascading Style Sheet
SQL -	Structured Query Language
UML -	Unified Modeling Language

INTRODUCTION

The project entitled ‘BEDAZZLED MARKETING’ is a website which provides a high quality online service and a best place to meet both for the Producers and Consumers. It helps Ordinary Customers to buy high quality product from the Producers. Bidding is also possible through this site. User can request for a product, so they can purchase crafts on their demand. “BEDAZZLED MARKETING” is a web application project. Most of the Handcrafter could not get desired price for their product and most case a broker or a third party will get most benefit than the actual producer. Also a large area of marketing handicraft product is depending on foreigners but unfortunately our Govt. Could not provide an online sale of such product to a reach of worldwide.

The proposed system is a website in which user can buy handmade creative products as well as handicrafters can sell their products efficiently. We will also provide users to give customization comments to customize their products, they can view the seller or brand details, payment details etc. The main objective of the ‘BEDAZZLED MARKETING’ is to manage the details of handicrafters, users and agencies .The overall control is done by the administrator and only administrator can guarantee the access. The main purpose of this project is to reduce manual work.

1. USING GIT AS A VERSION CONTROL SYSTEM

1.1 Introduction to GitHub

GitHub is a web-based version-control and collaboration platform for software developers. GitHub, which is delivered through a software-as-a-service (SaaS) business model, was started in 2008 and was founded on Git, an open source code management system created by Linus Torvalds to make software builds faster. And it is used to store the source code for a project and track the complete history of all changes to that code. It allows developers to collaborate on a project more effectively by providing tools for managing possibly conflicting changes from multiple developers. GitHub allows developers to change, adapt and improve software from its public repositories for free, but it charges for private repositories, offering various paid plans. Each public or private repository contains all of a project's files, as well as each file's revision history. Repositories can have multiple collaborators and can be either public or private.

GitHub facilitates social coding by providing a web interface to the Git code repository and management tools for collaboration. GitHub can be thought of as a serious social networking site for software developers. Members can follow each other, rate each other's work, receive updates for specific projects and communicate publicly or privately.

GitHub products and features

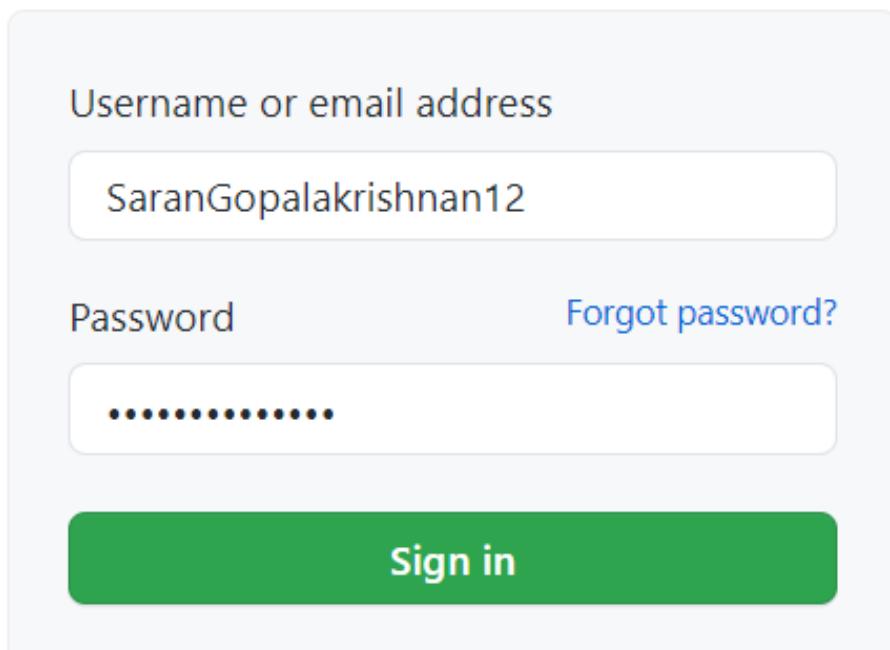
GitHub offers an on-premises version in addition to the well-known SaaS product. GitHub Enterprise supports integrated development environments and continuous integration tool integration, as well as a litany of third-party apps and services. It offers increased security and auditability than the SaaS version.

1.2 Working with Git

- Click on sign up button and create an account in GitHub.com after that sign in to the Github.



Sign in to GitHub



A screenshot of the GitHub sign-in page. It features a large input field for 'Username or email address' containing the text 'SaranGopalakrishnan12'. Below it is a password input field with several dots. To the right of the password field is a blue 'Forgot password?' link. A large green 'Sign in' button is centered below the inputs. At the bottom of the form, there's a link in a blue box that says 'New to GitHub? Create an account.'

Username or email address

SaranGopalakrishnan12

Password

Forgot password?

.....

Sign in

New to GitHub? [Create an account.](#)

□ Create a repository in GitHub

The screenshot shows the GitHub interface for creating a new repository. At the top, there's a search bar and navigation links for 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. Below the header, the main title is 'Create a new repository'. A sub-instruction says 'A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.' There's a 'Repository template' section with a dropdown menu set to 'No template'. The 'Owner' field is populated with 'SaranGopalakrishnan12'. The 'Repository name' field contains 'BEDAZZLED', which is highlighted with a yellow border. A note below says 'Great repository names are short and memorable. Your new repository will be created as BEDAZZLED-nibble?'. The 'Description (optional)' field is empty. Under 'Visibility', the 'Public' radio button is selected, with a note that anyone on the internet can see the repository. The 'Private' radio button is also available. In the 'Initialize this repository with:' section, several options are listed with checkboxes: 'Add a README file' (unchecked), 'Add .gitignore' (unchecked), and 'Choose a license' (unchecked). At the bottom right is a green 'Create repository' button.

□ Once repository is completed, you can setup the repository.

The screenshot displays a series of setup instructions for a new repository. It starts with a 'Quick setup — if you've done this kind of thing before' section, showing options to 'Set up in Desktop' or 'HTTPS / SSH' with the URL 'https://github.com/SaranGopalakrishnan12/BEDAZZLED-MARKETING.git'. It encourages users to 'Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.' Below this, there's a section titled '...or create a new repository on the command line' containing a block of terminal commands:

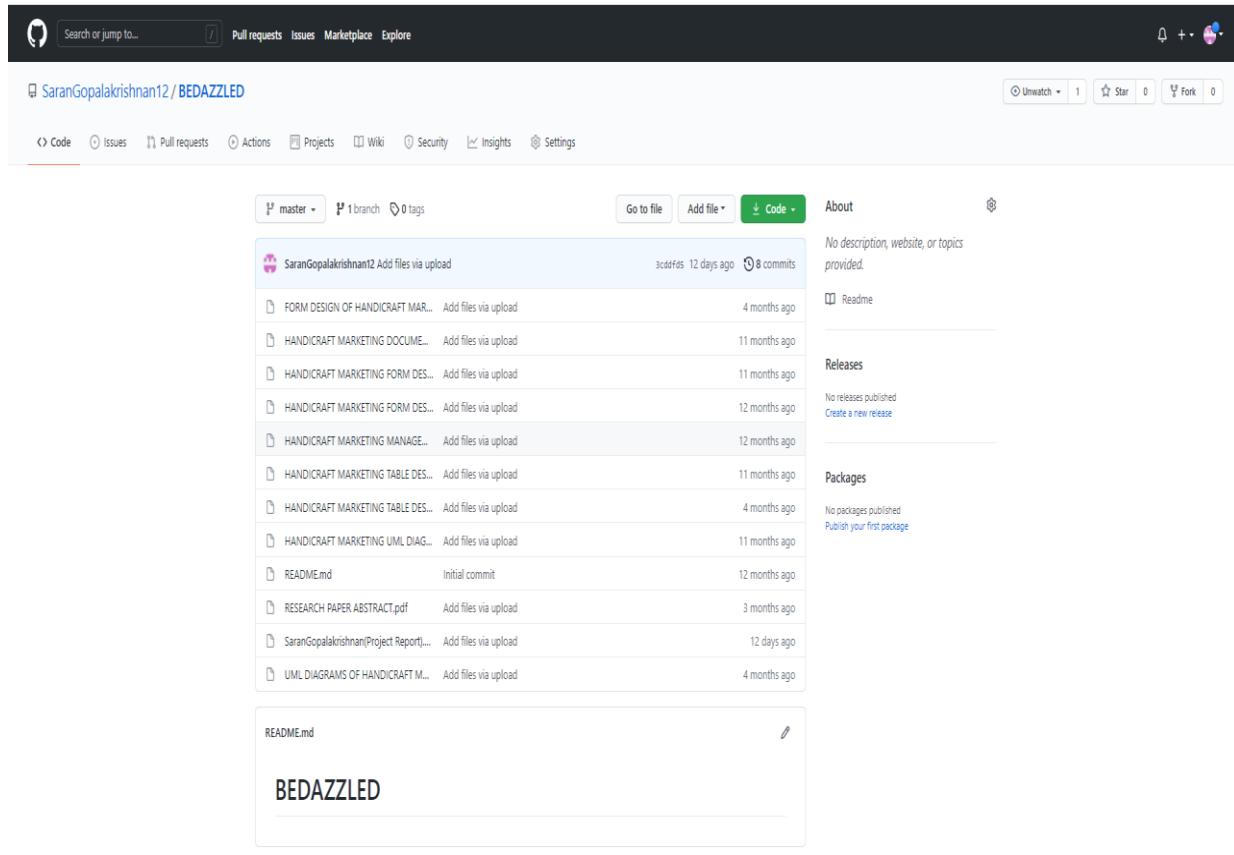
```
echo "# BEDAZZLED-MARKETING" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/SaranGopalakrishnan12/BEDAZZLED-MARKETING.git
git push -u origin main
```

Following this is a section titled '...or push an existing repository from the command line' with the same set of commands:

```
git remote add origin https://github.com/SaranGopalakrishnan12/BEDAZZLED-MARKETING.git
git branch -M main
git push -u origin main
```

Finally, there's a section titled '...or import code from another repository' with the note 'You can initialize this repository with code from a Subversion, Mercurial, or TFS project.' A 'Import code' button is shown. A 'ProTip!' note at the bottom says 'Use the URL for this page when adding GitHub as a remote.'

- Download and install Github desktop application. Once installed Go to Repository>Clone Repository, and select your repository which was created in GitHub.com or select a local system folder.
- Push the local repository to origin.
- Refresh GitHub.com repository to fetch commits.



2. PROJECT DOCUMENTATION

2.1 INTRODUCTION

2.1.1 Project Overview

“BEDAZZLED MARKETING” is a web application project. Most of the Handcrafter could not get desired price for their product and most case a broker or a third party will get most benefit than the actual producer. The project entitled ‘BEDAZZLED MARKETING’ is a website which provides a high quality online service and a best place to meet both for the Producers and Consumers. It helps Ordinary Customers to buy high quality product from the Producers.

2.1.2 Project Specification

The proposed system is a website in which user can buy handmade creative products as well as handicrafters can sell their products efficiently. We will also provide users to give customization comments to customize their products, they can view the seller or brand details, payment details etc. The main objective of the ‘BEDAZZLED MARKETING’ is to manage the details of handicrafters, users and agencies .The overall control is done by the administrator and only administrator can guarantee the access. The main purpose of this project is to reduce manual work. The system includes 4 modules: They are:

1. Admin Module

Admin must have a login into this system. Admin can add or update product categories. Admin can View all the registered Handicrafters, can able to approve or reject Handicrafters, Agencies and Users also can able to view all registered Agencies details. Admin is an authorized person and he is in full control of the overall system.

2. Handcrafters Module

The Handcrafter is the one who is registered in the system for selling his product through this application

3. Agency Module

The Agency is the intermediate between handicrafts and users. They can purchase handicrafts from Handcrafters. These products are buying through this application.

4. User Module

The user is the consumer of handicrafts they buy, bid and view products that are uploaded to our network. Also user can view all the events that are organized by the firm.

2.2 SYSTEM STUDY

2.2.1 Introduction

System analysis is a process of gathering and interpreting facts, diagnosing problems and the information to recommend improvements on the system. It is a problem solving activity that requires intensive communication between the system users and system developers. System analysis or study is an important phase of any system development process. The system is studied to the minute's detail and analyzed. The system analyst plays the role of the interrogator and dwells deep into the working of the present system. The system is viewed as a whole and the input to the system are identified. The outputs from the organizations are traced to the various processes. System analysis is concerned with becoming aware of the problem, identifying the relevant and decisional variables, analyzing and synthesizing the various factors and determining an optimal or at least a satisfactory solution or program of action.

A detailed study of the process must be made by various techniques like interviews, questionnaires etc. The data collected by these sources must be scrutinized to arrive to a conclusion. The conclusion is an understanding of how the system functions. This system is called the existing system. Now the existing system is subjected to close study and problem areas are identified. The designer now functions as a problem solver and tries to sort out the difficulties that the enterprise faces. The solutions are given as proposals. The proposal is then weighed with the existing system analytically and the best one is selected. The proposal is presented to the user for an endorsement by the user. The proposal is reviewed on user request and suitable the user is satisfied with proposal.

Preliminary study is the process of gathering and interpreting facts, using the information for further studies on the system. Preliminary study is problem solving activity that requires intensive communication between the system users and system developers. It does various feasibility studies. In these studies, a rough figure of the system activities can be obtained, from which the decision about the strategies to be followed for effective system study and analysis can be taken.

2.2.2 EXISTING SYSTEM

The current system is a manual one where most of the processes are done without using the computer. It is a traditional main sector of craft. The main problem faced by many handcrafters are the unavailability of a centralized market of their product. It also needs extra cost. If there are many stores they act as a mediator, that is they buy products from handcrafter and sell the product in market but most case handicrafter will not get the desired price. Also if there are any foreigners need to buy handcraft they must visit the store. Government supports handcrafters for their welfare and rehabilitation but most of the group will not utilize this type of projects, the right thing is most of them didn't know the favors that Govt. offers. Since, the details are recorded into paper there is huge risk of loss of data.

Drawbacks of existing system

- Very time and effort consuming
- Needs extra cost.
- Lot of paper work required.
- Maintenance is difficult.
- More than one person cannot access the data on the same time.
- Calculations are manually done so there is a chance for occurring errors.
- The main disadvantages of the existing system is that the handicrafter does not get the actual price that they expected.
- Mainly a third party or a broker gets the actual profit and many handcrafters gets cheated.
- There are many stores they act as a mediator. Also if there are any foreigners need to buy handcraft they must visit the store.
- Most of the handcrafters didn't know the favours that government offers to them.

2.2.3 PROPOSED SYSTEM

The proposed system can overcome all the limitations of the existing system. The proposed system is a very flexible online marketing bidding management application for handcrafters. This is a web based application that offers a wide range of operations such as, an online market, the handcraft product upload straight to the online store and ordinary

users can view and buy their desired one. so there is no need for an intermediate to sell their product also handcrafters will never cheated.

The most important advantage of this system is that the handicrafter can submit their products from different location served by internet without using a physical file. The system provide proper security by using the unique username and password and reduce the manual work. It is a paperless work, so there is no loss of data. It is very easy for the users to purchase the products from the store.

Advantages of proposed system

The system is very simple in design and to implement. The system requires very low system resources and the system will work in almost all configurations. It has got following features:

- **Ensure data accuracy:** - The proposed system eliminates the manual errors while entering the details of the users during the registration.
- **You are Able to get creative and innovative handmade products:-** Our customers will get creative handmade products with affordable price and they have the option to get customized product from the sellers. Also sellers can able to do proper business management and can also improve the quality and uniqueness of the products based on the market trends.
- **Better security:** - For data to remain secure measures must be taken to prevent unauthorized access. Security means that data are protected from various forms of destruction. The system security problem can be divided into four related issues: security, integrity, privacy and confidentiality. Username and password requirement to sign in ensures security. It will also provide data security as we are using the secured databases for maintaining the documents.
- **Better service:** - The product will avoid the burden of hard copy storage. We can also conserve the time and human resources for doing the same task. The data can be maintained for longer period with no loss of data.

2.3 REQUIREMENT ANALYSIS

2.3.1 Feasibility study

Feasibility study is made to see if the project on completion will serve the purpose of the organization for the amount of work, effort and the time that spend on it. Feasibility study lets the developer foresee the future of the project and the usefulness. A feasibility study of a system proposal is according to its workability, which is the impact on the organization, ability to meet their user needs and effective use of resources. Thus, when a new application is proposed it normally goes through a feasibility study before it is approved for development.

The document provides the feasibility of the project that is being designed and lists various areas that were considered very carefully during the feasibility study of this project such as Technical, Economic and Operational feasibilities. The following are its features: -

2.3.1.1 Economic Feasibility

The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on project, which will give best, return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require.

The following are some of the important financial questions asked during preliminary investigation:

- The costs conduct a full system investigation.
- The cost of the hardware and software.
- The benefits in the form of reduced costs or fewer costly errors.

The proposed system is developed as part of project work, there is no manual cost to spend for the proposed system. Also all the resources are already available, it give an indication of the system is economically possible for development.

The cost of project, BEDAZZLED MARKETING was divided according to the system used, its development cost and cost for hosting the project. According to all the calculations the project was developed in a low cost. As it is completely developed using open source software.

2.3.1.2 Technical Feasibility

The system must be evaluated from the technical point of view first. The assessment of this feasibility must be based on an outline design of the system requirement in the terms of input,

output, programs and procedures. Having identified an outline system, the investigation must go on to suggest the type of equipment, required method developing the system, of running the system once it has been designed. Technical issues raised during the investigation are:

- Does the existing technology sufficient for the suggested one?
- Can the system expand if developed?

The project should be developed such that the necessary functions and performance are achieved within the constraints. The project requires High Resolution Scanning device and utilizes Cryptographic techniques. Through the technology may become obsolete after some period of time, due to the fact that newer version of same software supports older versions, the system may still be used. So there are minimal constraints involved with this project. The system has been developed using PHP in front end and MySQL in server in back end, the project is technically feasible for development. The system has been developed using PHP in front end and MySQL in server in back end, the project is technically feasible for development. The System used was also of good performance of Processor Intel i3 core; RAM 8GB and, Hard disk 1TB.

2.3.1.3 Behavioral Feasibility

The proposed system includes the following questions:

- Is there sufficient support for the users?
- Will the proposed system cause harm?

The project would be beneficial because it satisfies the objectives when developed and installed. All behavioral aspects are considered carefully and conclude that the project is behaviorally feasible.

BEDAZZLED MARKETING, GUI is simple so that users can easily use it. BEDAZZLED MARKETING is simple enough so that no training is needed.

2.4 Requirement Modeling

2.4.1 UML diagram

UML is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems. UML was created by the Object Management Group (OMG) and UML 1.0 specification draft was proposed to the OMG in January 1997.

UML stands for Unified Modeling Language. UML is different from the other common programming languages such as C++, Java, COBOL, etc. UML is a pictorial language used to make software blueprints. UML can be described as a general-purpose visual modeling language to visualize, specify, construct, and document software system. Although UML is generally used to model software systems, it is not limited within this boundary. It is also used to model non-software systems as well. For example, the process flow in a manufacturing unit, etc. UML is not a programming language but tools can be used to generate code in various languages using UML diagrams. UML has a direct relation with object-oriented analysis and design. After some standardization, UML has become an OMG standard. All the elements, relationships are used to make a complete UML diagram and the diagram represents a system. The visual effect of the UML diagram is the most important part of the entire process. All the other elements are used to make it complete. UML includes the following nine diagrams.

- Activity diagram
- Use case diagram
- Sequence diagram
- Collaboration diagram
- State chart diagram
- Class diagram
- Object diagram
- Component diagram
- Deployment diagram

1. UML Activity Diagram

Activity Diagram describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.

The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc.

Activity diagrams are not only used for visualizing the dynamic nature of a system, but they are also used to construct the executable system by using forward and reverse engineering techniques. The only missing thing in the activity diagram is the message part.

It does not show any message flow from one activity to another. Activity diagram is sometimes considered as the flowchart. Although the diagrams look like a flowchart, they are not.

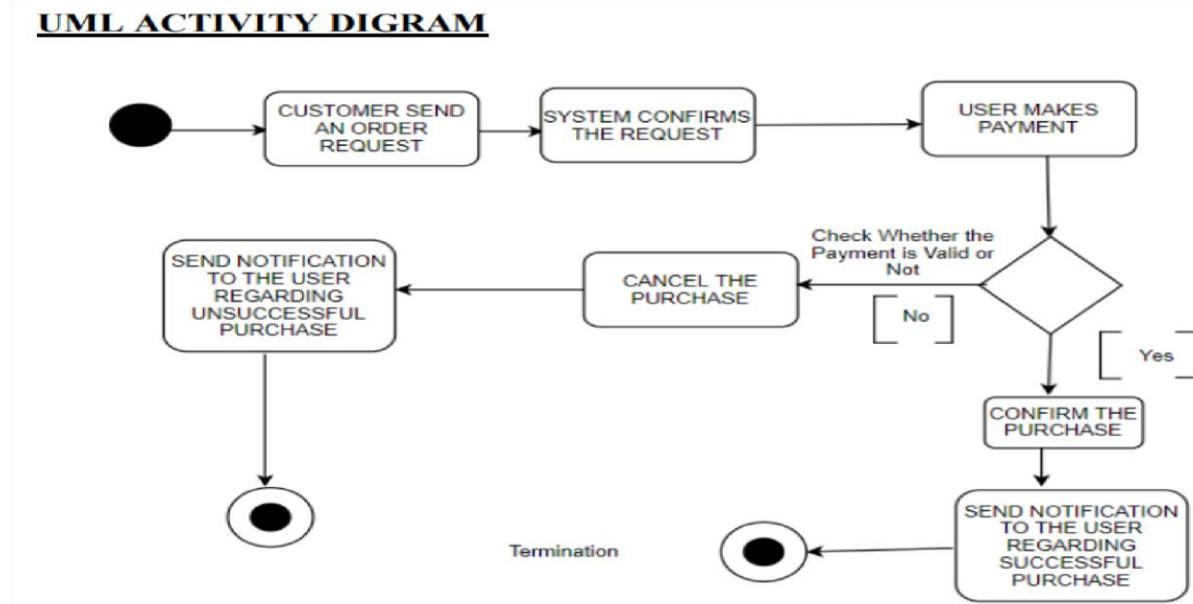


Fig 1: Activity Diagram of Customer Send Order Request

2. UML Use Case Diagram

A use case diagram is a graphic depiction of the interactions among the elements on a system. A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. In this context, the term "system" refers to something being developed or operated, such as a mail-order product sales and service Web site. Use case diagrams are employed in UML (Unified Modeling Language), a standard notation for the modeling of real-world objects and systems. System objectives can include planning overall requirements, validating a hardware design, testing and debugging a software product under development, creating an online help reference, or performing a consumer-service-oriented task. For example, use cases in

a product sales environment would include item ordering, catalog updating, payment processing, and customer relations. A use case diagram contains four components.

- The boundary, which defines the system of interest in relation to the world around it.
- The actors, usually individuals involved with the system defined according to their roles.
- The use cases, which are the specific roles are played by the actors within and around the system.
- The relationships between and among the actors and the use cases.

Use case diagrams are drawn to capture the functional requirements of a system.

After identifying the above items, we have to use the following guidelines to draw an efficient use case diagram.

- The name of a use case is very important. The name should be chosen in such a way so that it can identify the functionalities performed.
- Give a suitable name for actors.
- Show relationships and dependencies clearly in the diagram.
- Do not try to include all types of relationships, as the main purpose of the diagram is to identify the requirements.
- Use notes whenever required to clarify some important points.

UML DIAGRAMS OF HANDICRAFT MARKETING

UML USECASE DIAGRAM OF ADMIN

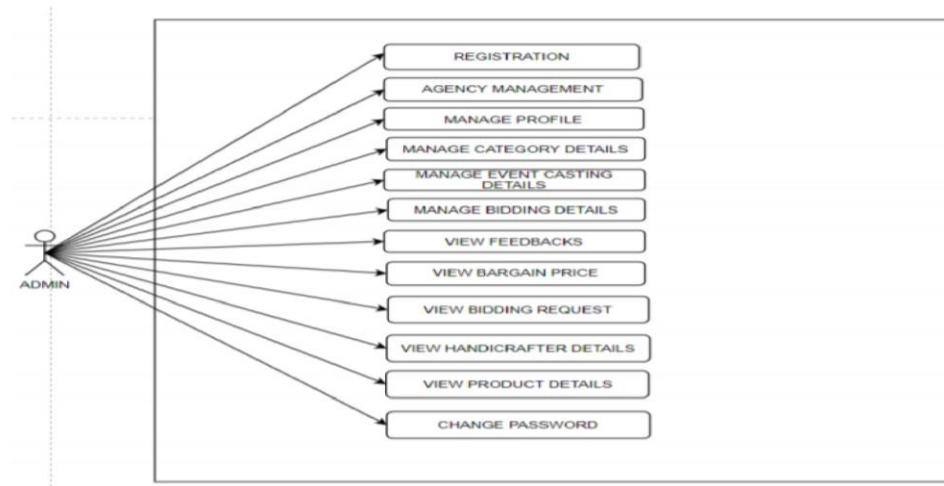


Fig 2: Use Case Diagram of Admin

UML USECASE DIAGRAM OF USER

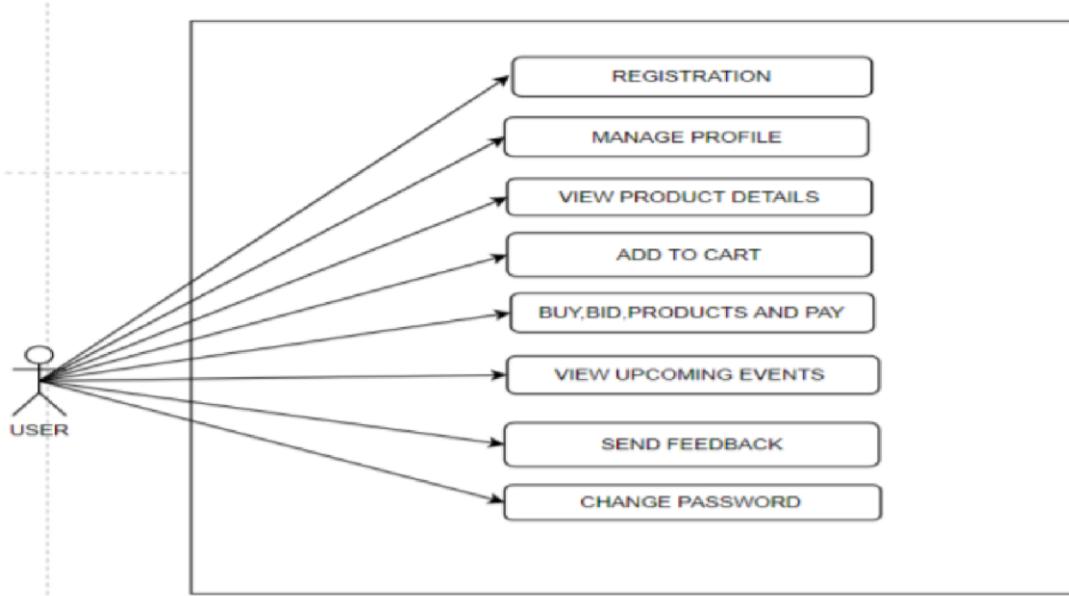


Fig 3: Use Case Diagram of User

UML USECASE DIAGRAM OF AGENCY

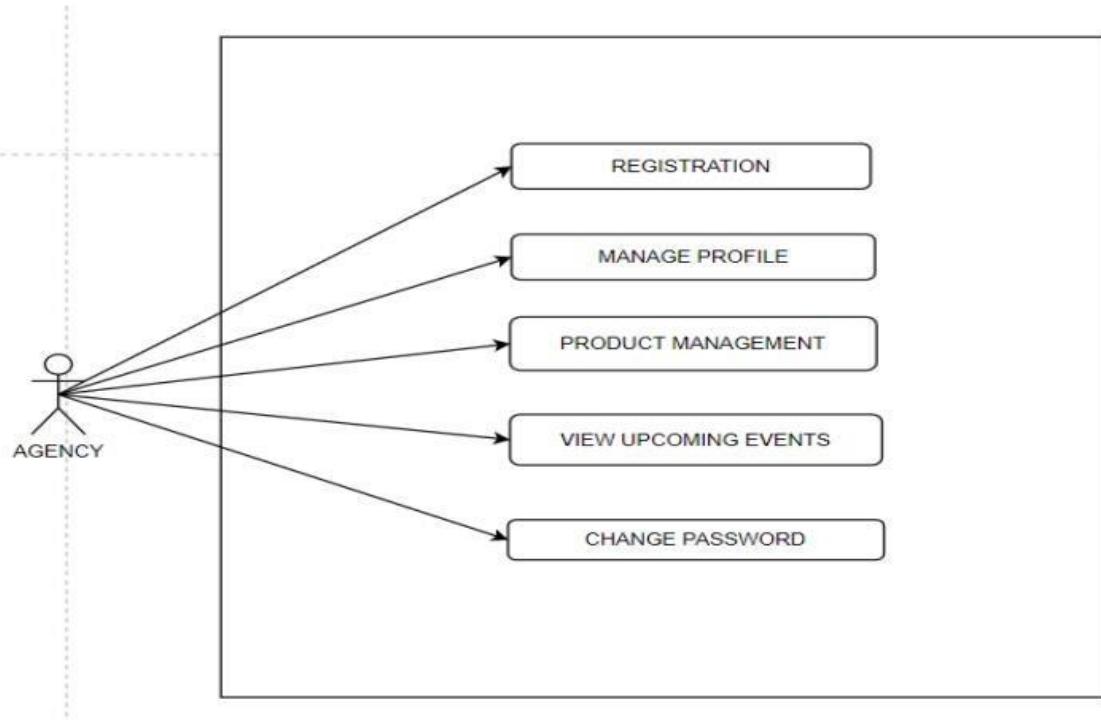


Fig 4: Use Case Diagram of Agency

UML USECASE DIAGRAM OF HANDICRAFTER

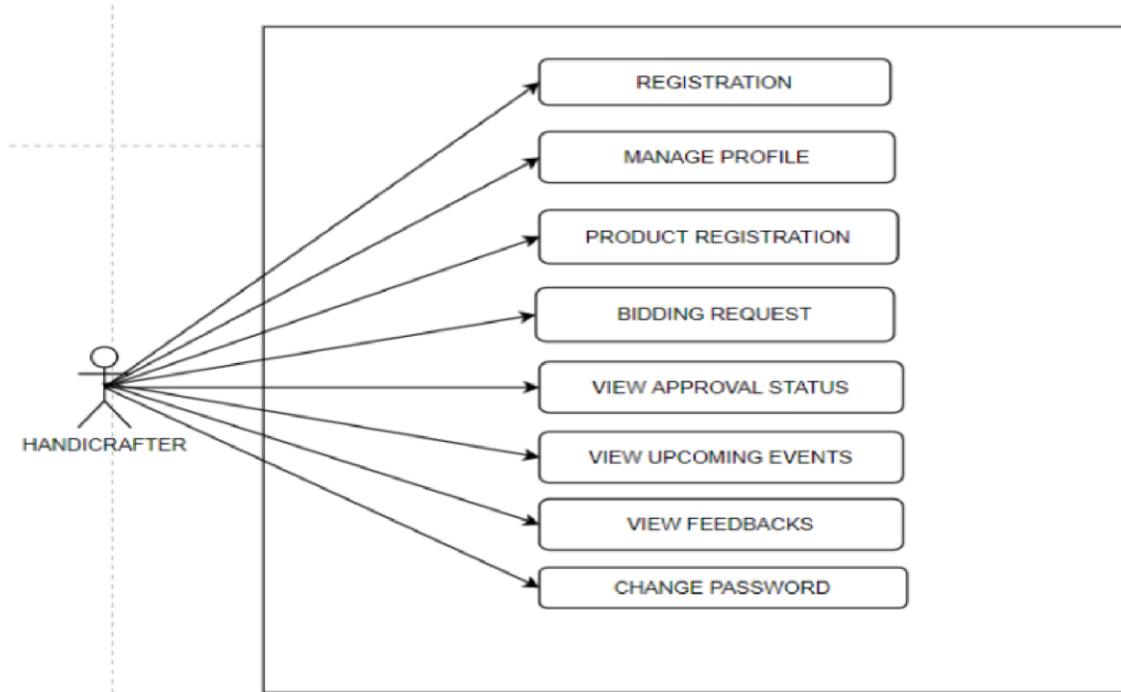


Fig 5: Use Case Diagram of Handicrafter

3. UML Sequence Diagram

A sequence diagram is an interaction diagram that shows how objects operate with one another and in what order. It is a construct of a message sequence chart.

A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called event diagrams or event scenarios. A sequence diagram shows, as parallel vertical lines (*lifelines*), different processes or objects that live simultaneously, and, as horizontal arrows, the messages exchanged between them, in the order in which they occur. This allows the specification of simple runtime scenarios in a graphical manner.

UML SEQUENCE DIAGRAM

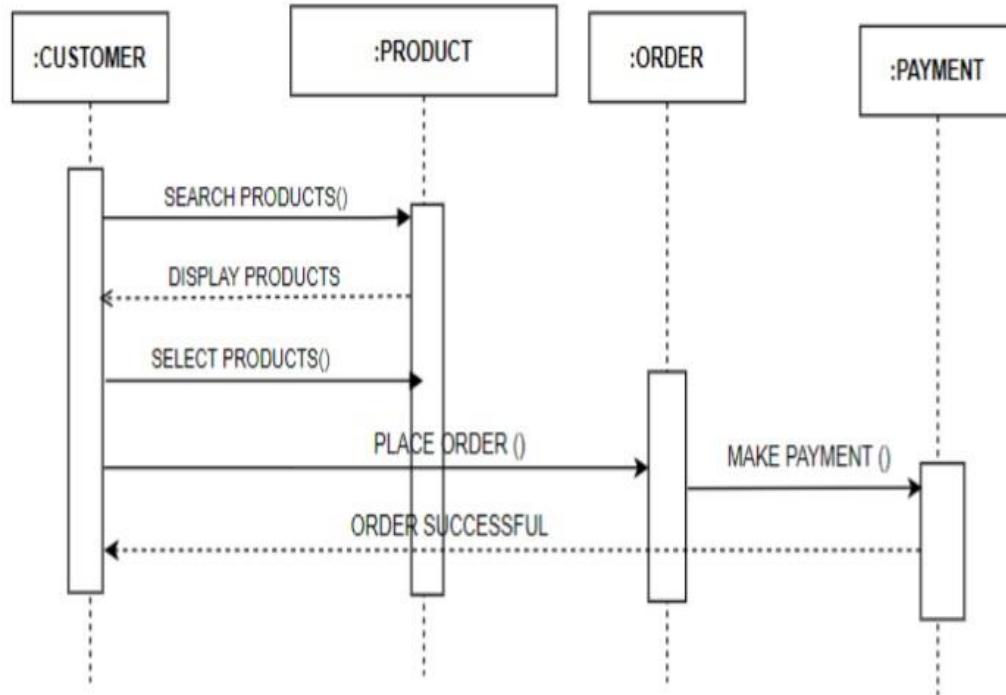


Fig 6: Sequence Diagram of Customer Place Order and Makes Payment

4. UML Collaboration Diagram

The collaboration diagram is used to show the relationship between the objects in a system. Both the sequence and the collaboration diagrams represent the same information but differently. Instead of showing the flow of messages, it depicts the architecture of the object residing in the system as it is based on object-oriented programming. An object consists of several features. Multiple objects present in the system are connected to each other. The collaboration diagram, which is also known as a communication diagram, is used to portray the object's architecture in the system.

Notations of a Collaboration Diagram

Following are the components of a component diagram that are enlisted below:

1. Objects: The representation of an object is done by an object symbol with its name and class underlined, separated by a colon. In the collaboration diagram, objects are utilized in the following ways:

- The object is represented by specifying their name and class.
 - It is not mandatory for every class to appear.
 - A class may constitute more than one object.
- In the collaboration diagram, firstly, the object is created, and then its class is specified.
- To differentiate one object from another object, it is necessary to name them.

2. Actors: In the collaboration diagram, the actor plays the main role as it invokes the interaction. Each actor has its respective role and name. In this, one actor initiates the use case.

3. Links: The link is an instance of association, which associates the objects and actors. It portrays a relationship between the objects through which the messages are sent. It is represented by a solid line. The link helps an object to connect with or navigate to another object, such that the message flows are attached to links.

4. Messages: It is a communication between objects which carries information and includes a sequence number, so that the activity may take place. It is represented by a labeled arrow, which is placed near a link. The messages are sent from the sender to the receiver, and the direction must be navigable in that particular direction. The receiver must understand the message.

The collaborations are used when it is essential to depict the relationship between the object. Both the sequence and collaboration diagrams represent the same information, but the way of portraying it quite different. The collaboration diagrams are best suited for analyzing use cases.

UML COLLABORATION DIAGRAM

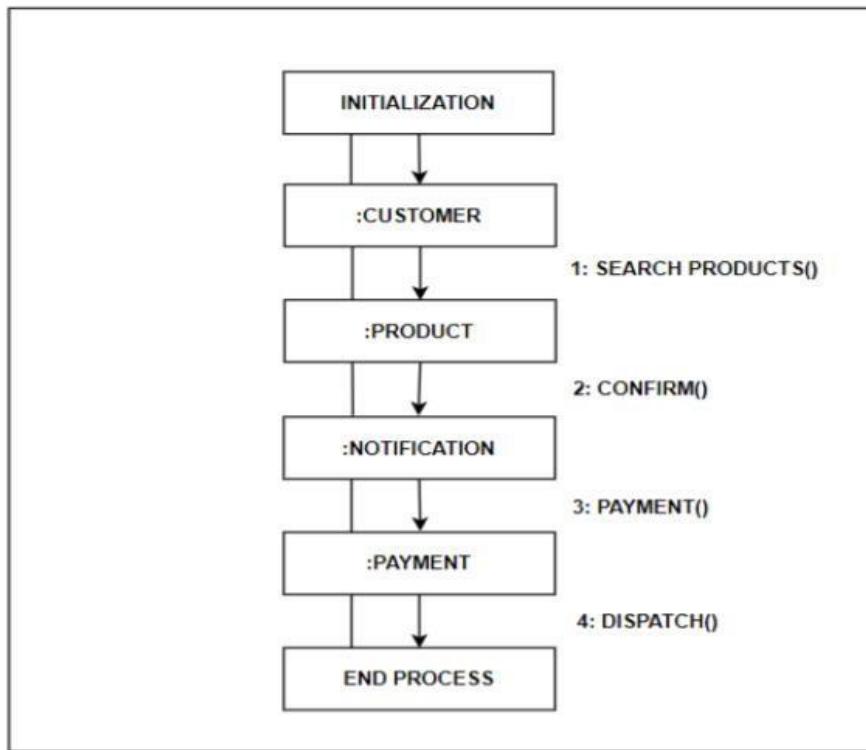


Fig 7: Collaboration Diagram of Customer Search Products and Makes Payment

5. UML State Chart Diagram

It describes different states of a component in a system. The states are specific to a component/object of a system. A Statechart diagram describes a state machine. State machine can be defined as a machine which defines different states of an object and these states are controlled by external or internal events. They define different states of an object during its lifetime and these states are changed by events. Statechart diagrams are useful to model the reactive systems. Reactive systems can be defined as a system that responds to external or internal events.

Statechart diagram describes the flow of control from one state to another state. States are defined as a condition in which an object exists and it changes when some event is triggered. The most important purpose of Statechart diagram is to model

lifetime of an object from creation to termination. Statechart diagrams are also used for forward and reverse engineering of a system.

However, the main purpose is to model the reactive system.

Following are the main purposes of using Statechart diagrams –

- To model the dynamic aspect of a system.
- To model the life time of a reactive system.
- To describe different states of an object during its life time.
- Define a state machine to model the states of an object.

UML STATECHART DIAGRAM

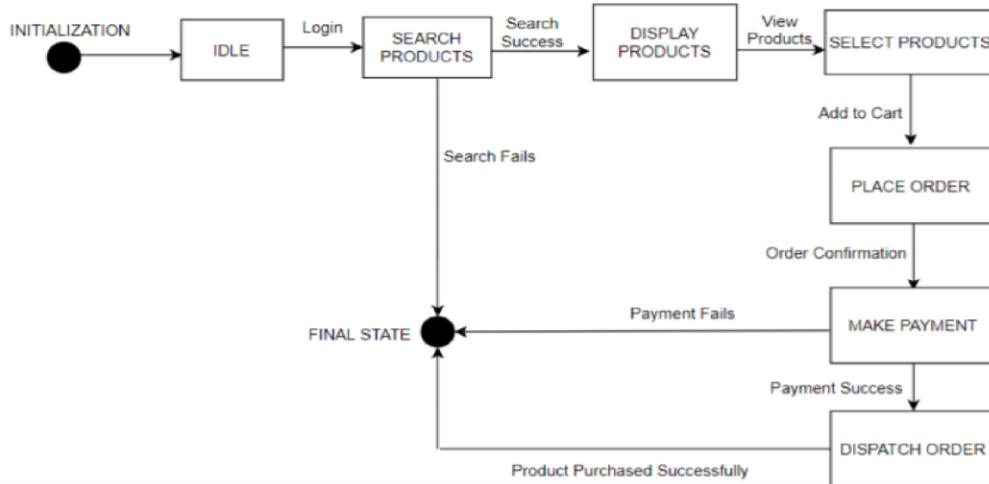


Fig 8: Statechart Diagram of Customer View Products and Makes Payment

6. UML Class Diagram

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application.

Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the

modeling of object-oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages.

Class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints. It is also known as a structural diagram.

The purpose of the class diagram can be summarized as –

- Analysis and design of the static view of an application.
- Describe responsibilities of a system.
- Base for component and deployment diagrams.
- Forward and reverse engineering.

UML CLASS DIAGRAM

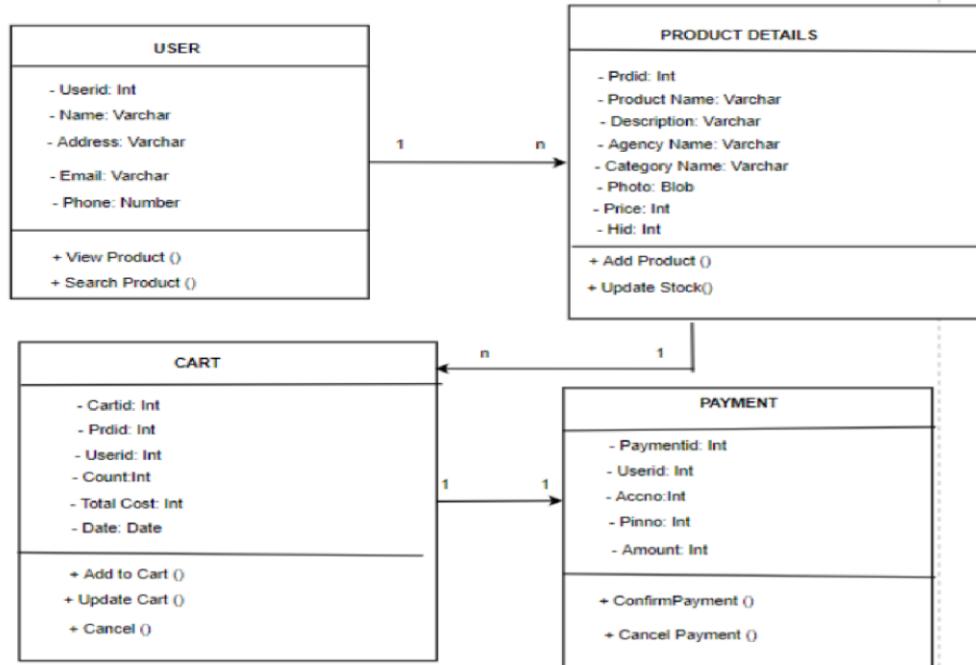


Fig 9: Class Diagram of Customer Add Products to Cart

7. Uml Object Diagram

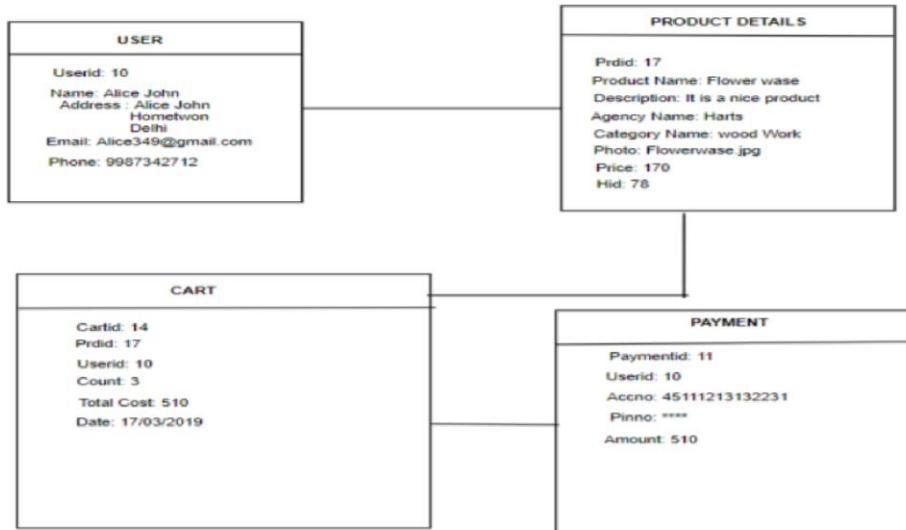
Object diagrams are derived from class diagrams so object diagrams are dependent upon class diagrams. Object diagrams represent an instance of a class diagram. The basic concepts are similar for class diagrams and object diagrams.

Object diagrams also represent the static view of a system but this static view is a snapshot of the system at a particular moment. Object diagrams are used to render a set of objects and their relationships as an instance.

The purpose of the object diagram can be summarized as –

- Forward and reverse engineering.
- Object relationships of a system.
- Static view of an interaction.
- Understand object behavior and their relationship from practical perspective

UML OBJECT DIAGRAM



- Fig 10: Object Diagram of Customer Add Products to Cart

8. Component Diagram

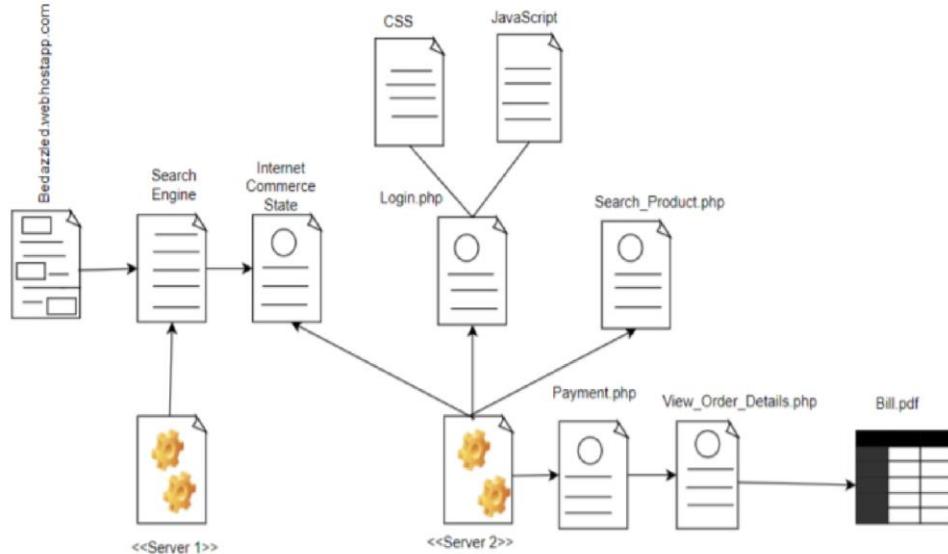
Component diagram is a special kind of diagram in UML. The purpose is also different from all other diagrams discussed so far. It does not describe the functionality of the system but it describes the components used to make those functionalities. Thus from that point of view, component diagrams are used to visualize the physical components in a system. These components are libraries, packages, files, etc.

Component diagrams can also be described as a static implementation view of a system. Static implementation represents the organization of the components at a particular moment. A single component diagram cannot represent the entire system but a collection of diagrams is used to represent the whole.

The purpose of the component diagram can be summarized as –

- Visualize the components of a system.
- Construct executable by using forward and reverse engineering.
- Describe the organization and relationships of the components.

UML COMPONENT DIAGRAM



- Fig 11: Component Diagram of Customer Login to System and Search for Products

9. Deployment Diagram

Deployment diagrams are used to visualize the topology of the physical components of a system, where the software components are deployed. Deployment diagrams are used to describe the static deployment view of a system. Deployment diagrams consist of nodes and their relationships.

It ascertains how software is deployed on the hardware. It maps the software architecture created in design to the physical system architecture, where the software will be executed as a node. Since it involves many nodes, the relationship is shown by utilizing communication paths.

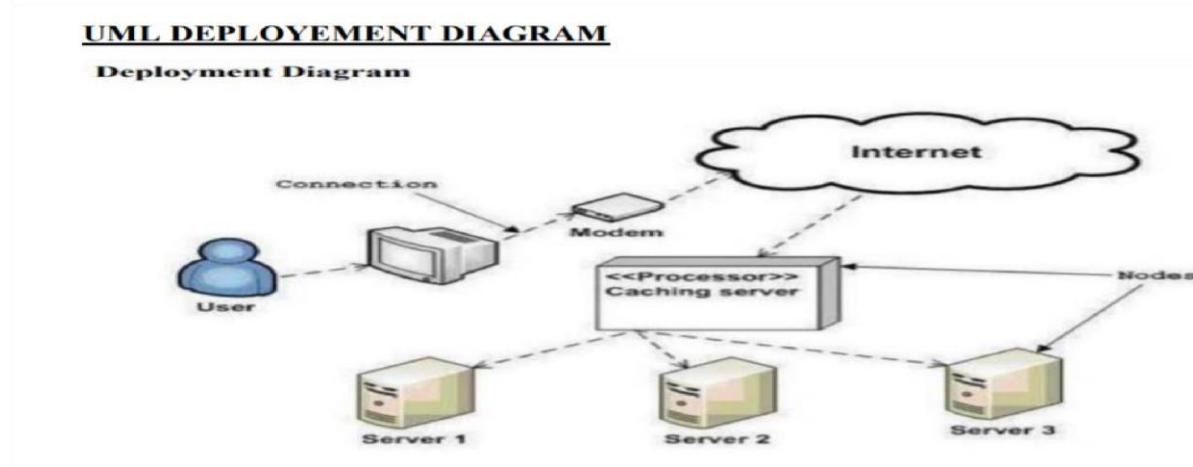


Fig 12: Deployment Diagram of Bedazzled Marketing

2.5 System Specification

2.5.1 Hardware Specification

Processor	- Intel core i3
RAM	- 8 GB
Hard disk	- 1 TB

2.5.2 Software Specification

Front End	- HTML, CSS
Backend	- PHP
Database	- MYSQL
Client on PC	- Windows 7 and above.
Technologies used	- JS, HTML5, J Query, PHP, CSS, Ajax

2.6 Software Description

2.6.1 PHP

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. PHP is now installed on more than 244 million websites and 2.1 million web servers. Originally created by Rasmus Ledorf in 1995, the reference implementation of PHP is now produced by the PHP group. While PHP originally stood for personal Home page, it now stands for PHP: Hypertext Preprocessor, a recursive acronym. PHP code is interpreted by a web server with a PHP processor module which generates the resulting web page. PHP commands can be embedded directly into a HTML source document rather than calling an external file to process data. It has also evolved to include a command-line interface capability and can be used in standalone incompatible with the GNU General Public License (GPL) due to restrictions on the usage of the term PHP. PHP can be deployed on most web servers and also as a standalone shell on almost every operating system and platform, free of charge.

2.6.2 MySQL

MySQL, the most popular Open-Source SQL database management system, is developed, distributed, and supported by Oracle Corporation. The MySQL Web site provides the latest information about MySQL software.

- **MySQL is a database management system.**

A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, you need a database management system such as MySQL Server. Since computers are very good at handling large amounts of data, database management systems play a central role in computing, as standalone utilities, or as parts of other applications.

- **MySQL databases are relational.**

A relational database stores data in separate tables rather than putting all the data in one big storeroom. The database structures are organized into physical files optimized for speed. The logical model, with objects such as databases, tables, views, rows, and columns, offers a flexible programming environment. You set up rules governing the relationships between different data fields, such as one-to-one, one-to-many, unique, required or optional, and “pointers” between different tables. The database enforces these rules, so that with a well-designed database, your application never sees inconsistent, duplicate, orphan, out-of-date, or missing data. The SQL part of “MySQL” stands for “Structured Query Language”. SQL is the most common standardized language used to access databases. Depending on your programming environment, you might enter SQL directly (for example, to generate reports), embed SQL statements into code written in another language, or use a language-specific API that hides the SQL syntax. SQL is defined by the ANSI/ISO SQL Standard. The SQL standard has been evolving since 1986 and several versions exist. In this manual, “SQL92” refers to the standard released in 1992, “SQL: 1999” refers to the standard released in 1999, and “SQL: 2003” refers to the current version of the standard. We use the phrase “the SQL standard” to mean the current version of the SQL Standard at any time.

- **MySQL software is Open Source.**

Open Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. If you wish, you may study the source code and change it to suit your needs. The MySQL software uses the GPL (GNU General Public License), to define what you may and may not do with the software in different situations. If you feel uncomfortable with the GPL or need to embed MySQL code into a commercial application, you can buy a commercially licensed version from us. See the MySQL Licensing Overview for more information.

- **The MySQL Database Server is very fast, reliable, scalable, and easy to use.**

If that is what you are looking for, you should give it a try. MySQL Server can run comfortably on a desktop or laptop, alongside your other applications, web servers, and so on, requiring little or no attention. If you dedicate an entire machine to MySQL, you can adjust the settings to take advantage of all the memory, CPU power, and I/O capacity available.

- **MySQL Server works in client/server or embedded systems.**

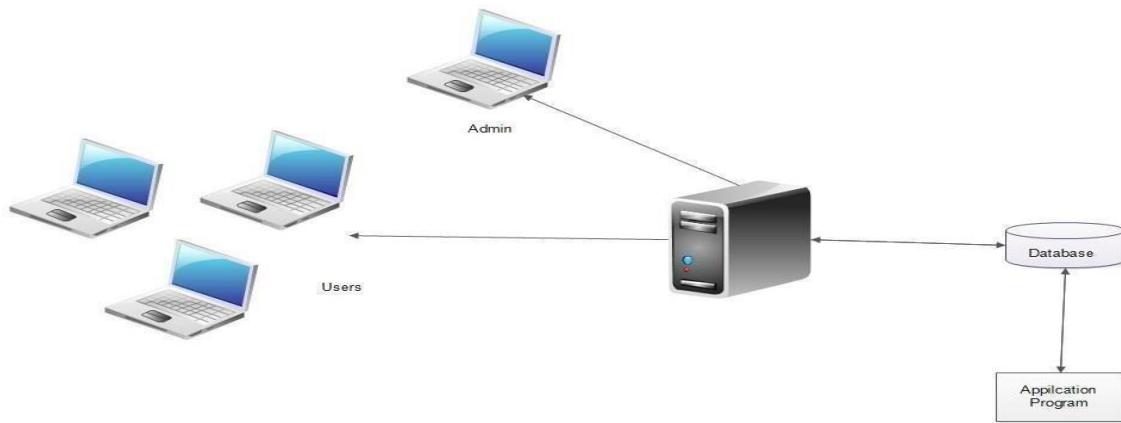
The MySQL Database Software is a client/server system that consists of a multithreaded SQL server that supports different backends, several different client programs and libraries, administrative tools, and a wide range of application programming interfaces (APIs). We also provide MySQL Server as an embedded multi-threaded library that you can link into your application to get a smaller, faster, easier-to-manage standalone product.

2.7 System design

Design is the first step into the development phase for any engineered product or system. Design is a creative process. A good design is the key to effective system. The term “design” is defined as “the process of applying various techniques and principles for the purpose of defining a process or a system in sufficient detail to permit its physical realization”. It may be defined as a process of applying various techniques and principles for the purpose of defining a device, a process or a system in sufficient detail to permit its physical realization. Software design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm that is used. The system design develops the architectural detail required to build a system or product. As in the case of any systematic approach, this software too has undergone the best possible design phase fine tuning all efficiency, performance and accuracy levels. The design phase is a transition from a user-oriented document to a document to the programmers or database personnel.

System design goes through two phases of development: Logical and Physical Design.

2.7.1 Architectural Design



The registered customer, seller, and admin can access the website through internet using their Laptop, Smart Phone, Tablet or Desktop Computer. The System's application program processes the user's request and provides the required services by taking data from the system database.

2.7.2 Module Design

Admin Module

The administrator of the company is allowed to access all the services in the system. Admin has the overall control of the system. Admin can add or update Event categories. Admin can View all the registered Users, Agency, Handicrafter that can able to approve or reject Users, Handicrafter and Agency.

Manage user details, Add Event Categories	Approve/Reject the registered Agencies, Handicrafters and Users.
View order details	View/Replay to messages and requests

Registered User Module

After registration, users can view events, search events, can add events to cart, purchase products through secure online payment.

Customer registration, login	Search events/Add events to cart
Checkout and Order Products	Make payment
View Order Details and download order details	Send complaints and feedbacks

Registered Agency Module

The Agency is the intermediate between handicrafts and users. They can purchase handicrafts from Handcrafters. These products are buying through this application.

Agency Registration, Login	View all available events
View Orders	View added events
Send Request for adding events	View replays of send requests

Registered Handicrafter Module

The Handcrafter is the one who is registered in the system for selling his product through this application.

Handicrafter Registration, Login	View all available events
View Orders	View added events
Send Request for adding products	View replays of send requests

2.7.3 Database Design

A database is an organized mechanism that has the capability of storing information through which a user can retrieve stored information in an effective and efficient manner. The data is the purpose of any database and must be protected.

The database design is a two-level process. In the first step, user requirements are gathered together and a database is designed which will meet these requirements as clearly as possible. This step is called Information Level Design and it is taken independent of any individual DBMS.

In the second step, this Information level design is transferred into a design for the specific DBMS that will be used to implement the system in question. This step is called Physical Level Design, concerned with the characteristics of the specific DBMS that will be used. A database design runs parallel with the system design. The organization of the data in the database is aimed to achieve the following two major objectives.

- Data Integrity
- Data independence

Relational Database Management System (RDBMS)

A relational model represents the database as a collection of relations. Each relation resembles a table of values or file of records. In formal relational model terminology, a row is called a tuple, a column header is called an attribute and the table is called a relation. A relational database consists of a collection of tables, each of which is assigned a unique name. A row in a tale represents a set of related values.

Relations, Domains & Attributes

A table is a relation. The rows in a table are called tuples. A tuple is an ordered set of n elements. Columns are referred to as attributes. Relationships have been set between every table in the database. This ensures both Referential and Entity Relationship Integrity. A domain D is a set of atomic values. A common method of specifying a domain is to specify a data type from which the data values forming the domain are drawn. It is also useful to specify a name for the domain to help in interpreting its values. Every value in a relation is atomic, that is not decomposable.

Relationships

- Table relationships are established using Key. The two main keys of prime importance are Primary Key & Foreign Key. Entity Integrity and Referential Integrity Relationships can be established with these keys.
- Entity Integrity enforces that no Primary Key can have null values.
- Referential Integrity enforces that no Primary Key can have null values.
- Referential Integrity for each distinct Foreign Key value, there must exist a matching Primary Key value in the same domain. Other key are Super Key and Candidate Keys.

Normalization

Data are grouped together in the simplest way so that later changes can be made with minimum impact on data structures. Normalization is formal process of data structures in manners that eliminates redundancy and promotes integrity.

Normalization is a technique of separating redundant fields and breaking up a large table into a smaller one. It is also used to avoid insertion, deletion, and updating anomalies.

Normal form in data modelling use two concepts, keys and relationships. A key uniquely identifies a row in a table.

There are two types of keys, primary key and foreign key. A primary key is an element or a combination of elements in a table whose purpose is to identify records from the same table. A foreign key is a column in a table that uniquely identifies record from a different table. All the tables have been normalized up to the third normal form. As the name implies, it denotes putting things in the normal form. The application developer via normalization tries to achieve a sensible organization of data into proper tables and columns and where names can be easily correlated to the data by the user. Normalization eliminates repeating groups at data and thereby avoids data redundancy which proves to be a great burden on the computer resources. These include:

- ✓ Normalize the data.
- ✓ Choose proper names for the tables and columns.
- ✓ Choose the proper name for the data.

First Normal Form

The First Normal Form states that the domain of an attribute must include only atomic values and that the value of any attribute in a tuple must be a single value from the domain of that attribute.

In other words, 1NF disallows “relations within relations” or “relations as attribute values within tuples”. The only attribute values permitted by 1NF are single atomic or indivisible values. The first step is to put the data into First Normal Form. This can be done by moving data into separate tables where the data is of similar type in each table. Each table is given a Primary Key or Foreign Key as per requirement of the

project. In this we form new relations for each non- atomic attribute or nested relation. This eliminated repeating groups of data. A relation is said to be in first normal form if only if it satisfies the constraints that contain the primary key only.

Second Normal Form

According to Second Normal Form, for relations where primary key contains multiple attributes, no non-key attribute should be functionally dependent on a part of the primary key. In this we decompose and setup a new relation for each partial key with its dependent attributes. Make sure to keep a relation with the original primary key and any attributes that are fully functionally dependent on it. This step helps in taking out data that is only dependent on a part of the key.

A relation is said to be in second normal form if and only if it satisfies all the first normal form conditions for the primary key and every non-primary key attributes of the relation is fully dependent on its primary key alone.

Third Normal Form

According to Third Normal Form, Relation should not have a non-key attribute functionally determined by another non-key attribute or by a set of non-key attributes. That is, there should be no transitive dependency on the primary key. In this we decompose and set up relation that includes the non-key attributes that functionally determines other non-key attributes. This step is taken to get rid of anything that does not depend entirely on the Primary Key. A relation is said to be in third normal form if only if it is in second normal form and more over the non key attributes of the relation should not be depend on other non-key attribute.

TABLE DESIGN

Table No : 01

Table Name :tbl_login

Primary Key :lid

Table Description: To Store Login Information.

Fieldname	Datatype	Size	Constraint	Description
lid	Int	10	Primary Key	Login id
email	Varchar	30		Mailid
password	Varchar	16		Password
status	Int	2		Status
role	Varchar	20		Role of user

Table No : 02

Table Name :tbl_state

Primary Key : stateid

Table Description: To Store State Information.

Fieldname	Datatype	Size	Constrainr	Description
stateid	Int	2	Primary Key	State Id
statename	Varchar	30		State Name
status	Int	2		Active/Inactive

Table No : 03

Table Name :tbl_district

Primary Key :did

Table Description: To Store District Information.

Field Name	Data Type	Size	Constraint	Description
did	Int	2	Primary Key	District Id
dname	varchar	30		District Name
status	Int	2		Active/inactive

Table No : 04

Table Name : tbl_category

Primary Key : catid

Table Description: To Store Category Name of the Products.

Field Name	Data Type	Size	Constraint	Description
catid	Int	10	Primary Key	Category Id
catname	Varchar	30		Category Name
status	Int	2		Active/inactive

Table No : 05

Table Name : tbl_events

Primary Key : eventid

Table Description: To Store Event Casting Details.

Field Name	Data Type	Size	Constraint	Description
eventid	Int	10	Primary Key	Event Id
title	Varchar	20		Name of Event
type	Varchar	20		Type of Event
details	Varchar	50		Details of Event
place	Varchar	50		Place of Event
date	Date	3		Date of Event Conducted
time	Timestamp	5		Time of Event Conducted
status	Int	2		Active/inactive

Table No : 06

Table Name : tbl_feedbacks

Primary Key: fid

Foreign Key : lid

Table Description: To Store Feedbacks Details

Field Name	Data Type	Size	Constraint	Description
fid	Int	10	Primary Key	Feedback Id
feedback	varchar	50		Users Feedback
lid	Int	10	Foreign Key	Login Id referenced from tbl_login
status	Int	2		Active/Inactive

Table No: 07

Table Name: tbl_userreg

Primary Key: userid

Foreign Key: lid, stateid,did

Table Description: To Store User Details.

Field Name	Datatype	Size	Constraint	Description
userid	Int	10	Primary Key	User Id
name	Varchar	20		Name of User
housename	Varchar	50		House Name
stateid	Int	10	Foreign Key	State Name
did	Int	10	Foreign Key	District Name
city	Varchar	30		City Name
postoffice	Varchar	50		Post Office
pincode	Int	6		Pincode
gender	Varchar	10		Gender
phoneno	Int	10		Phone Number
lid	Int	10	Foreign Key	Login Id
status	Int	2		Approve/Reject

Table No: 08**Table Name: tbl_agencyreg****Primary Key: agid****Foreign Key: lid, stateid, did****Table Description: To Store Agency Details.**

Field Name	Datatype	Size	Constraint	Description
agid	Int	10	Primary Key	Agency Id
name	Varchar	20		Name of Agency
apartmentname	Varchar	50		Apartment Name
stateid	Int	10	Foreign Key	State Name
did	Int	10	Foreign Key	District Name
city	Varchar	30		City Name
postoffice	Varchar	50		Post Office
pincode	Int	6		Pincode
phoneno	Int	10		Phone Number
lid	Int	10	Foreign Key	Login Id
status	Int	2		Approve/Reject

Table No: 09**Table Name: tbl_customorder****Primary Key: customid****Foreign Key: userid,catid****Table Description: To Store Custom Order Details.**

Field Name	Datatype	Size	Constraint	Description
customorid	Int	10	Primary Key	Custom OrderId
catid	Int	10	Foreign Key	Category Id
productname	Varchar	20		Name of product
description	Varchar	50		Product Description
image	Blob	50		Image of Product
quantity	Int	10		Product Quantity
userid	Int	10	Foreign Key	Name of User
status	Int	2		Approve/Reject

Table No: 10**Table Name: tbl_handireg****Primary Key: hid****Foreign Key: lid, stateid, did, agid, worktypeid****Table Description: To Store Handicrafter Details.**

Field Name	Datatype	Size	Constraint	Description
hid	Int	10	Primary Key	Handicrafter Id
name	Varchar	20		Name of Handicrafter
apartmentname	Varchar	50		Apartment Name
stateid	Int	10	Foreign Key	State Name
did	Int	10	Foreign Key	District Name
agid	Int	10	Foreign Key	Agency Name
city	Varchar	30		City Name
postoffice	Varchar	50		Post Office
pincode	Int	6		Pincode
phoneno	Int	10		Phone Number
worktypeid	Int	10	Foreign Key	Type of Work Known
lid	Int	10	Foreign Key	Login Id
status	Int	2		Approve/Reject

Table No : 11**Table Name:** tbl_cart**Primary Key:** cartid**Foreign Key :** prdid,userid**Table Description:** To Store Cart Details.

Field Name	Data Type	Size	Constraint	Description
cartid	Int	10	Primary Key	Cart Id
prdid	Int	10	Foreign Key	Product Id referenced from tbl_productregistration
userid	Int	10	Foreign Key	userid referenced from tbl_userregistration
count	Int	3		No of Items
totalcost	Int	5		Total Cost
date	Date	3		Date of Purchased
status	Int	2		Active/Inactive

Table No: 12**Table Name:** tbl_payment**Primary Key:** paymentid**Foreign Key :** userid**Table Description:** To Store Payment Details.

Field Name	Data Type	Size	Constraint	Description
paymentid	Int	10	Primary Key	Payment Id
userid	Int	10	Foreign key	User id referenced from tbl_userregistration
accno	Int	14		Account No of user
cvv	Int	3		Card verification Value
amount	Int	5		Bank Balance
status	Int	2		Active/inactive

2.8 System testing

2.8.1 Introduction

Software Testing is the process of executing software in a controlled manner, in order to answer the question - Does the software behave as specified? Software testing is often used in association with the terms verification and validation. Validation is the checking or testing of items, includes software, for conformance and consistency with an associated specification. Software testing is just one kind of verification, which also uses techniques such as reviews, analysis, inspections, and walkthroughs. Validation is the process of checking that what has been specified is what the user actually wanted.

Other activities which are often associated with software testing are static analysis and dynamic analysis. Static analysis investigates the source code of software, looking for problems and gathering metrics without actually executing the code. Dynamic analysis looks at the behavior of software while it is executing, to provide information such as execution traces, timing profiles, and test coverage information.

Testing is a set of activity that can be planned in advanced and conducted systematically. Testing begins at the module level and work towards the integration of entire computers based system. Nothing is complete without testing, as it vital success of the system testing objectives, there are several rules that can serve as testing objectives. They are:

Testing is a process of executing a program with the intent of finding an error.

- A good test case is one that has high possibility of finding an undiscovered error.
- A successful test is one that uncovers an undiscovered error.

If a testing is conducted successfully according to the objectives as stated above, it would uncover errors in the software. Also testing demonstrate that the software function appear to be working according to the specification, that performance requirement appear to have been met.

There are three ways to test program.

- For correctness
- For implementation efficiency
- For computational complexity

Test for correctness are supposed to verify that a program does exactly what it was designed to do. This is much more difficult than it may at first appear, especially for large programs.

2.8.2 Test Plan

A test plan implies a series of desired course of action to be followed in accomplishing various testing methods. The Test Plan acts as a blue print for the action that is to be followed. The software engineers create a computer program, its documentation and related data structures. The software developers is always responsible for testing the individual units of the programs, ensuring that each performs the function for which it was designed. There is an independent test group (ITG) which is to remove the inherent problems associated with letting the builder to test the thing that has been built. The specific objectives of testing should be stated in measurable terms. So that the mean time to failure, the cost to find and fix the defects, remaining defect density or frequency of occurrence and test work-hours per regression test all should be stated within the test plan.

The levels of testing include:

- Unit testing
- Integration Testing
- Data validation Testing
- Output Testing

2.8.2.1 Unit Testing

Unit testing focuses verification effort on the smallest unit of software design – the software component or module. Using the component level design description as a guide, important control paths are tested to uncover errors within the boundary of the module. The relative complexity of tests and uncovered scope established for unit testing. The unit testing is white-box oriented, and step can be conducted in parallel for multiple components. The modular interface is tested to ensure that information properly flows into and out of the program unit under test. The local data structure is examined to ensure that data stored temporarily maintains its integrity during all steps in an algorithm's execution. Boundary conditions are tested to ensure that all statements in a module have been executed at least once. Finally, all error handling paths are tested.

Tests of data flow across a module interface are required before any other test is initiated. If data do not enter and exit properly, all other tests are moot. Selective testing of execution paths is an essential task during the unit test. Good design dictates that error conditions be anticipated and error handling paths set up to reroute or cleanly terminate processing when an error does occur.

Boundary testing is the last task of unit testing step. Software often fails at its boundaries.

2.8.2.2 Integration Testing

Integration testing is systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing. The objective is to take unit tested components and build a program structure that has been dictated by design. The entire program is tested as whole. Correction is difficult because isolation of causes is complicated by vast expanse of entire program. Once these errors are corrected, new ones appear and the process continues in a seemingly endless loop. After performing unit testing in the System all the modules were integrated to test for any inconsistencies in the interfaces. Moreover differences in program structures were removed and a unique program structure was evolved.

2.8.2.3 Validation Testing or System Testing

This is the final step in testing. In this the entire system was tested as a whole with all forms, code, modules and class modules. This form of testing is popularly known as Black Box testing or System tests.

Black Box testing method focuses on the functional requirements of the software. That is, Black Box testing enables the software engineer to derive sets of input conditions that will fully exercise all functional requirements for a program.

Black Box testing attempts to find errors in the following categories; incorrect or missing functions, interface errors, errors in data structures or external data access, performance errors and initialization errors and termination errors.

2.8.2.4 Output Testing or User Acceptance Testing

The system considered is tested for user acceptance; here it should satisfy the firm's need. The software should keep in touch with perspective system; user at the time of developing and making changes whenever required. This done with respect to the following points:

- Input Screen Designs,
- Output Screen Designs,

The above testing is done taking various kinds of test data. Preparation of test data plays a vital role in the system testing. After preparing the test data, the system under study is tested using that test data. While testing the system by which test data errors are again uncovered and corrected by using above testing steps and corrections are also noted for future use.

2.8.3 Test Case

Test Case 1					
Project Name: Bedazzled Marketing-Handicraft Products Management System					
Login Test Case					
Test Case ID: Fun_1	Test Designed By: Saran Gopalakrishnan				
Test Priority(Low/Medium/High): High	Test Designed Date: 18-06-2021				
Module Name: Login Screen	Test Executed By : Dr. Juby Mathew				
Test Title : Verify login with valid username and password	Test Execution Date: 20-06-2021				
Description: Test the Login Page					
Pre-Condition : User has valid user name and password					
Step	Test Step	Test Data	Expected Result	Actual Result	Status(Pass/Fail)
1	Navigation to Login Page		Login Page should be	Login page displayed	Pass
2	Provide Valid User name	User Name: varunpr34@gmail.com	User should be able to Login	User Logged in and navigated to User Dashboard with records	Pass
3	Provide Valid Password	Password: Varu@12			
4	Click on Login button				
5	Provide Invalid User Name or password	User Name: 13@gmail Password: varu	User should not be able to Login	Message for enter valid user or password displayed	Pass
6	Provide Null User Name or Password	User Name: null Password: null			
7	Click on Login button				
Post-Condition: User is validated with database and successfully login into account. The Account session details are logged in database					

Test Case 2					
Project Name: Bedazzled Marketing-Handicraft Products Management System					
User Registration Test Case					
Test Case ID:Fun_2		Test Designed By: Saran Gopalakrishnan			
Test Priority(Low/Medium/High):Medium		Test Designed Date: 18-06-2021			
Module Name: User Registration		Test Executed By : Dr. Juby Mathew			
Test Title : Verify new user registration		Test Execution Date: 20-06-2021			
Description: Test the User registration Page					
Pre-Condition : User should not be already registered					
Step	Test Step	Test Data	Expected Result	Actual Result	Status(Pass/Fail)
1	Navigation to Registration Page		Registration Page should be displayed	Registration page displayed	Pass
2	Provide null information	Name: null	Validation message should be displayed	*Mandatory field message displayed	Pass
3	Provide Valid Details of user	All the mandatory registration details of the user	User should be able to register successfully	Registration successful and home page displayed	Pass
4	Click on Register button				
Post-Condition: User is validated with database and successfully login into account. The Account session details are logged in database					

2.9 Implementation

Implementation is the stage of the project where the theoretical design is turned into a working system. It can be considered to be the most crucial stage in achieving a successful new system gaining the users confidence that the new system will work and will be effective and accurate. It is primarily concerned with user training and documentation. Conversion usually takes place about the same time the user is being trained or later. Implementation

simply means convening a new system design into operation, which is the process of converting a new revised system design into an operational one.

At this stage the main work load, the greatest upheaval and the major impact on the existing system shifts to the user department. If the implementation is not carefully planned or controlled, it can create chaos and confusion.

Implementation includes all those activities that take place to convert from the existing system to the new system. The new system may be a totally new, replacing an existing manual or automated system or it may be a modification to an existing system. Proper implementation is essential to provide a reliable system to meet organization requirements. The process of putting the developed system in actual use is called system implementation. This includes all those activities that take place to convert from the old system to the new system. The system can be implemented only after thorough testing is done and if it is found to be working according to the specifications. The system personnel check the feasibility of the system. The more complex the system being implemented, the more involved will be the system analysis and design effort required to implement the three main aspects: education and training, system testing and changeover.

The implementation state involves the following tasks:

- Careful planning.
- Investigation of system and constraints.
- Design of methods to achieve the changeover.

2.9.1 Implementation Procedures

Implementation of software refers to the final installation of the package in its real environment, to the satisfaction of the intended uses and the operation of the system. In many organizations someone who will not be operating it, will commission the software development project. In the initial stage people doubt about the software but we have to ensure that the resistance does not build up, as one has to make sure that:

- The active user must be aware of the benefits of using the new system.
- Their confidence in the software is built up.
- Proper guidance is imparted to the user so that he is comfortable in using the application.

Before going ahead and viewing the system, the user must know that for viewing the result, the server program should be running in the server. If the server object is not up running on the server, the actual process won't take place.

2.9.2 User Training

User training is designed to prepare the user for testing and converting the system. To achieve the objective and benefits expected from computer based system, it is essential for the people who will be involved to be confident of their role in the new system. As system becomes more complex, the need for training is more important. By user training the user comes to know how to enter data, respond to error messages, interrogate the database and call up routine that will produce reports and perform other necessary functions.

2.9.3 Operational Document

After providing the necessary basic training on computer awareness the user will have to be trained on the new application software. This will give the underlying philosophy of the use of the new system such as the screen flow, screen design type of help on the screen, type of errors while entering the data, the corresponding validation check at each entry and the ways to correct the date entered. It should then cover information needed by the specific user/group to use the system or part of the system while imparting the training of the program on the application. This training may be different across different user groups and across different levels of hierarchy.

2.9.4 System Maintenance

Maintenance is the enigma of system development. The maintenance phase of the software cycle is the time in which a software product performs useful work. After a system is successfully implemented, it should be maintained in a proper manner. System maintenance is an important aspect in the software development life cycle. The need for system maintenance is for it to make adaptable to the changes in the system environment. Software maintenance is of course, far more than "Finding Mistakes".

2.10 Conclusion and Future Enhancements

2.10.1 Future Enhancement

- This application can be easily implemented under various situations.
- We can add new features as and when required.
- Reusability is possible as and when required in this application.
- There is flexibility in all the modules.
- It can be implemented in any server in the internet providing an easy access to the users.
- In future we can add more modules as requirement.
- The proposed system is designed in such a way that the payment of product should be done in online mode.
- Customers can able to do advanced search options
- Customers can able to cancel order return products etc.

2.10.2 Conclusion

Computer has got clear advantage over the manual system. The computerized system is more reliable, efficient and fast at the end of the project, I can say that computer play a very crucial role in the development of the firm. All the daily reports generated by the system are to be checked by the concerned official so as to ensure that all the transactions have been put in appropriate accounts and this is tallied with new vouchers.

Computer does maximum work within minimum time. Because it is used in every field so that it provides comfort and suitability to everyone. Providing maximum facilities and comfort to users is the main goal of the system. To achieve this goal, other modern facilities relating to the computer should have to be provided.

Hence, conclude by stating that,"BEDAZZLED MARKETING" as a computer based application could do many more when compare to the traditional manual method. The application is designed in such a way that any further enhancement can be done with ease. The system has the capability for easy integration with other systems. New modules can be added to the existing system with less effort. In future a new function or procedure can be easily added in the system through existing classes or even a new class can be added.

2.11 Bibliography

Books/References:

- Elias M. Award – System Analysis and Design, Galgotia Publication Pvt Ltd, Year: 2002, Edition 2.
- Pankaj Jalote - An Integral Approach to Software Engineering, Narosa Publishing House Pvt Ltd, Year: 2009, Edition 3.
- Korth, Henry f- Database System Concepts, hill book, year: 2014 Edition.
- Gary B. Shelly, Harry J. Rosenblatt, “System Analysis and Design”, 2009.
- Roger S Pressman, “Software Engineering”, 1994.
- PankajJalote, “Software engineering: a precise approach”, 2006.
- James lee and Brent ware Addison, “Open source web development with LAMP”, 2003.
- IEEE Std 1016 Recommended Practice for Software Design Descriptions.

Websites:

- www.w3schools.com.
- www.jquery.com.
- <http://homepages.dcc.ufmg.br/~rodolfo/es-1-03/IEEE-Std-830-1998.pdf>.
- www.agilemodeling.com/artifacts/useCaseDiagram.html.
- <https://mdbootstrap.com/snippets/jquery/mdbootstrap>.

2.12 APPENDIX

2.12.1 SAMPLE CODE

Connection.php

```
<?php  
$conn = mysqli_connect("localhost","root","","db_bedazzled");  
if(!$conn)  
{  
    ?>  
    <script type="text/javascript"> alert("Failed to establish connection!");</script>
```

```
</script>
<?php
}
?>
```

userregistrationform.php

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>User Registration Form</title>
    <script
src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.4.0/jquery.min.js"></script>    <script
src="https://cdnjs.cloudflare.com/ajax/libs/jqueryvalidate/1.19.1/jquery.validate.min.js">
</script>
<link href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
rel="stylesheet">
    <link href="bootstrap/css/bootstrap.css" rel="stylesheet">
    <link href="userregistrationformstyle.css" rel="stylesheet">
    <script src="USERVALIDATIONCODE/uservalidationcode.js"></script>
</head>
<body>
<header>
    <nav class="navbar navbar-expand-lg">
        <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarcontent" aria-controls="navbar-content" aria-expanded="false" aria-label="toggle-navigation">    <span class="navbar-toggler-icon"></span>
        </button>
        <div class="collapse navbar-collapse" id="navbar-content">
            <ul class="navbar-nav">
                <li class="nav-item">
                    <a class="nav-link" href="index.php">Home</a>
                </li>
                <li class="nav-item">
                    <a class="nav-link" href="loginpage.php">Login</a>
                </li>
            </ul>
        </div>
    </nav>
</header>
<div class="container">
    <div class="row justify-content-center">
        <div class="col-md-6">
            <div class="card border-0 shadow p-4 mb-4" style="border-radius: 15px">
                <div class="card-body">
                    <div class="mb-3">
                        <label for="username" class="form-label">User Name</label>
                        <input type="text" class="form-control" id="username" placeholder="Enter User Name" required>
                    </div>
                    <div class="mb-3">
                        <label for="email" class="form-label">Email Address</label>
                        <input type="email" class="form-control" id="email" placeholder="Enter Email Address" required>
                    </div>
                    <div class="mb-3">
                        <label for="password" class="form-label">Password</label>
                        <input type="password" class="form-control" id="password" placeholder="Enter Password" required>
                    </div>
                    <div class="mb-3">
                        <label for="confirm_password" class="form-label">Confirm Password</label>
                        <input type="password" class="form-control" id="confirm_password" placeholder="Enter Confirm Password" required>
                    </div>
                    <div class="mb-3">
                        <label for="checkbox" class="form-label">Accept Terms and Conditions</label>
                        <input type="checkbox" class="form-control" id="checkbox" required>
                    </div>
                    <div class="mb-3">
                        <button type="submit" class="btn btn-primary w-100" style="border-radius: 15px">Register</button>
                    </div>
                </div>
            </div>
        </div>
    </div>
</div>
</body>
</html>
```

```

    </div>
</nav>
<div class="box">
    <div class="box-1"></div>
    <div class="box-2"></div>
</div>
</header>
<section>
    <h4 class="text-center">Welcome to</h4>
</section>

<h3 class="col-12 col-xl-12 col-lg-12 text-center bg-dark text-white">User Registration Form</h3>
    <form action="action-userregister.php" style="border:1px solid #ccc" method="POST" name="Register" onsubmit="return validate()">
        <div class="container">
            <p col-6 col-xl-6 col-lg-6>Please fill in this form to create an account.</p>
            <br>
            <div class="form-group col-12 col-xl-12 col-lg-12">
                <label for="name"><b>Name</b></label>
                <input type="text" class="form-control" name="name" id="Uname" placeholder="Enter the Name" value="" maxlength="50" />
            </div>
            <div class="form-group col-12 col-xl-12 col-lg-12">
                <label for="housename"><b>House Name</b></label>
                <input type="text" class="form-control" name="housename" id="Uhousename" placeholder="Enter the House Name" value="" maxlength="50" />
            </div>
            <div class="form-group col-12 col-xl-12 col-lg-12">
                <label for="state"><b>State</b></label>
                <select class="form-control" id="UState" name="state" >
                    <option value="" disabled selected>--Please Select your State--</option>
                    <?php include
'connection.php';
$sql="SELECT * FROM `tbl_state`";
$result= mysqli_query($conn, $sql);
while($row = mysqli_fetch_assoc($result))

```

```

{
    echo "<option value=\"".$row['stateid']."'>".$row['statename']."</option>";
} ?>
</select>
</div>
<div class="form-group col-12 col-xl-12 col-lg-12">
    <label for="district"><b>District</b></label>
    <select class="form-control" id="UDistrict" name="district" >
        <option value="" disabled selected>--Please Select your District--</option>
        <?php      include
'connection.php';
$sql="SELECT * FROM `tbl_district`";
$result= mysqli_query($conn, $sql);
while($row = mysqli_fetch_assoc($result))
{
    echo "<option value=\"".$row['did']."'>".$row['dname']."</option>";
} ?>
</select>
</div>

<div class="form-group col-12 col-xl-12 col-lg-12">
    <label for="city"><b>City/Landmark</b></label>
    <input type="text" class="form-control" name="city" id = "UCityname"
placeholder="Enter the City/Landmark" value="" maxlength="50" />
</div>
<div class="form-group col-12 col-xl-12 col-lg-12">
    <label for="postoffice"><b>Post Office</b></label>
    <input type="text" class="form-control" name="postoffice" id="UPostoffice"
placeholder="Enter Postoffice" value="" maxlength="50" />
</div>
<div class="form-group col-12 col-xl-12 col-lg-12">
    <label for="pincode"><b>Pincode</b></label>
    <input type="text" class="form-control" name="pincode" id="UPincode"
placeholder="Enter the Pincode" value="" maxlength="6" />
</div>

<div class="form-group col-12 col-xl-12 col-lg-12">
    <label for="gender"><b>Gender</b></label>
    <input type="radio" id="malec" name="gender" value="male"/> Male

```

```

<input type="radio" id="femalec" name="gender" value="female" /> Female
<input type="radio" id="otherc" name="gender" value="other" /> Other
<label id="Ugender"></label>
</div>
<div class="form-group col-12 col-xl-12 col-lg-12">
<label for="phoneno"><b>Phone Number</b></label>

<input type="text" class="form-control" name="phoneno" id="UPhone" value="" 
maxlength="10" placeholder="Enter the Phone Number" />

</div>
<div class="form-group col-12 col-xl-12 col-lg-12">
<label for="email"><b>Email</b></label>
<input type="text" class="form-control" id="Uemail" placeholder="Enter the EmailId" 
name="email" value="" />

</div>
<div class="form-group col-12 col-xl-12 col-lg-12">
<label for="password"><b>Password</b></label>
<input type="password" class="form-control" id="UPword" 
placeholder="Example@123" name="password" value="" maxlength="8" />

</div>
<div class="form-group col-12 col-xl-12 col-lg-12">
<label for="cpassword"><b>Confirm Password</b></label>
<input type="password" class="form-control" id="UCpword" placeholder="Re-enter the 
Password" name="cpassword" value="" maxlength="8"/> </div>

<label>

<input type="checkbox" name="remember" style="margin-bottom:15px" checked="checked" >
Remember me

</label>
<p>By creating an account you agree to our <a href="#" style="color:dodgerblue">Terms & 
Privacy</a>.</p>

<div class="clearfix">
<button type="submit" name="submit" class="registerbtn" 
value="Register">Register</button>
<button type="reset" class="cancelbtn">Cancel</button>
</div>
</div>
</form>

```

```
</body>
</html>
```

action-userregister.php

```
<?php      include
'connection.php';

$name = $_POST['name'];
$housename = $_POST['housename'];
$city = $_POST['city'];
$postoffice = $_POST['postoffice'];
$pincode = $_POST['pincode'];
$statename = $_POST['state'];
$dname = $_POST['district'];
$gender = $_POST['gender'];
$phoneno = $_POST['phoneno'];
$email = $_POST['email'];
$password = $_POST['password'];
$cpassword = $_POST['cpassword'];
$sql= "INSERT INTO `tbl_login`(`lid`, `email`, `password`, `role`, `status`) VALUES
('$email','$password', 'User', 4)";
mysqli_query($conn, $sql);
$lid=mysqli_insert_id($conn);

$sql= "INSERT INTO `tbl_userreg`(`userid`, `name`, `housename`, `stateid`, `did`, `city`,
`postoffice`, `pincode`, `gender`, `phoneno`, `lid`) VALUES
('$name','$housename','$statename', '$dname','$city','$postoffice','$pincode',
'$gender','$phoneno','$lid')"; if(mysqli_query($conn, $sql)){
echo '<script
type="text/javascript">'; echo 'alert("Registered Successfully")'; echo '</script>';
echo("<script>location.href='http://localhost/BEDAZZLEDMARKETING/index.php'</script>");
}
else{ echo '<script
type="text/javascript">'; echo
'alert("Something Wrong")';
echo '</script>'}
mysqli_close($conn);
?>
```

addproducts.php

```
<?php
    include 'connection.php';
    session_start();
    if(ISSET($_SESSION["lid"])){
        $s_id=$_SESSION["lid"];
    }
    else{
        header("location:index.php");
    }
?>
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1 ">
    <title>Bedazzled Marketing</title>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.4.0/jquery.min.js"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/jquery-validate/1.19.1/jquery.validate.min.js"></script>
    <link href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"
rel="stylesheet">
    <link href="bootstrap/css/bootstrap.css" rel="stylesheet">
    <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"
integrity="sha384-
Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS6JXm"
crossorigin="anonymous">
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.0/jquery.min.js"></script>
    <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/css/bootstrap.min.css"
integrity="sha384-
TX8t27EcRE3e/ihU7zmQxVncDAy5uIKz4rEkgIXeMed4M0jfIDPvg6uqKI2xXr2"
crossorigin="anonymous">
```

```
<!--//latest-->

<script src="https://code.jquery.com/jquery-3.5.1.js" integrity="sha256-QWo7LDvxbWT2tbbQ97B53yJnYU3WhH/C8ycbRAkjPDc="
crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/jquery-validation@1.19.2/dist/jquery.validate.js"></script>

<script src="ADDPROMOTEREGISTRATION/addproductreg.js"></script>

<link href="addproductregstyle.css" rel="stylesheet">

<!-- Custom styles for this page -->

<link href="vendor/datatables/dataTables.bootstrap4.min.css" rel="stylesheet">

</head>

<body >

<?php

    include('includes/header.php');

    include('includes/navbar.php');

    ?>

    <!-- Content Wrapper -->

    <div id="content-wrapper" class="d-flex flex-column">

        <!-- Main Content -->

        <div id="content">

            <!-- Topbar -->

            <nav class="navbar navbar-expand navbar-light bg-white topbar mb-4 static-top shadow">

                <!-- Sidebar Toggle (Topbar) -->

                <form class="form-inline">

                    <button id="sidebarToggleTop" class="btn btn-link d-md-none rounded-circle mr-3">
                        <i class="fa fa-bars"></i>
                    </button>

                </form>

                <!-- Topbar Search -->

                <form class="d-none d-sm-inline-block form-inline mr-auto ml-md-3 my-2 my-md-0 mw-100 navbar-search">

                    <div class="input-group">
```

```
<input type="text" class="form-control bg-light border-0 small" placeholder="Search  
for..." aria-label="Search" aria-describedby="basic-addon2">  
    <div class="input-group-append">  
        <button class="btn btn-primary" type="button">  
            <i class="fas fa-search fa-sm"></i>  
        </button>  
    </div>  
    </div>  
    </form>  
<!-- Topbar Navbar -->  
<ul class="navbar-nav ml-auto">  
    <!-- Nav Item - Search Dropdown (Visible Only XS) -->  
    <li class="nav-item dropdown no-arrow d-sm-none">  
        <a class="nav-link dropdown-toggle" href="#" id="searchDropdown" role="button"  
        data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">  
            <i class="fas fa-search fa-fw"></i>  
        </a>  
        <!-- Dropdown - Messages -->  
        <div class="dropdown-menu dropdown-menu-right p-3 shadow animated--grow-in"  
        aria-labelledby="searchDropdown">  
            <form class="form-inline mr-auto w-100 navbar-search">  
                <div class="input-group">  
                    <input type="text" class="form-control bg-light border-0 small"  
                    placeholder="Search for..." aria-label="Search" aria-describedby="basic-addon2">  
                    <div class="input-group-append">  
                        <button class="btn btn-primary" type="button">  
                            <i class="fas fa-search fa-sm"></i>  
                        </button>  
                    </div>  
                </div>  
            </form>  
        </div>  
    </li>
```

```
<!-- Nav Item - Alerts -->
<li class="nav-item dropdown no-arrow mx-1">
    <a class="nav-link dropdown-toggle" href="#" id="alertsDropdown" role="button"
    data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
        <i class="fas fa-bell fa-fw"></i>
        <!-- Counter - Alerts -->
        <span class="badge badge-danger badge-counter">3+</span>
    </a>
    <!-- Dropdown - Alerts -->
    <div class="dropdown-list dropdown-menu dropdown-menu-right shadow animated--grow-in" aria-labelledby="alertsDropdown">
        <h6 class="dropdown-header">
            Alerts Center
        </h6>
        <a class="dropdown-item d-flex align-items-center" href="#">
            <div class="mr-3">
                <div class="icon-circle bg-primary">
                    <i class="fas fa-file-alt text-white"></i>
                </div>
            </div>
            <div>
                <div class="small text-gray-500">December 12, 2019</div>
                <span class="font-weight-bold">A new monthly report is ready to
                download!</span>
            </div>
        </a>
        <a class="dropdown-item d-flex align-items-center" href="#">
            <div class="mr-3">
                <div class="icon-circle bg-success">
                    <i class="fas fa-donate text-white"></i>
                </div>
            </div>
            <div>

```

```
<div class="small text-gray-500">December 7, 2019</div>
$290.29 has been deposited into your account!
</div>
</a>
<a class="dropdown-item d-flex align-items-center" href="#">
<div class="mr-3">
<div class="icon-circle bg-warning">
<i class="fas fa-exclamation-triangle text-white"></i>
</div>
</div>
<div>
<div class="small text-gray-500">December 2, 2019</div>
Spending Alert: We've noticed unusually high spending for your account.
</div>
</a>
<a class="dropdown-item text-center small text-gray-500" href="#">Show All
Alerts</a>
</div>
</li>
<!-- Nav Item - Messages -->
<li class="nav-item dropdown no-arrow mx-1">
<a class="nav-link dropdown-toggle" href="#" id="messagesDropdown" role="button"
data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
<i class="fas fa-envelope fa-fw"></i>
<!-- Counter - Messages -->
<span class="badge badge-danger badge-counter">7</span>
</a>
<!-- Dropdown - Messages -->
<div class="dropdown-list dropdown-menu dropdown-menu-right shadow animated--grow-in" aria-labelledby="messagesDropdown">
<h6 class="dropdown-header">
Message Center
</h6>
```

```
<a class="dropdown-item d-flex align-items-center" href="#">  
  <div class="dropdown-list-image mr-3">  
      
    <div class="status-indicator bg-success"></div>  
  </div>  
  <div class="font-weight-bold">  
    <div class="text-truncate">Hi there! I am wondering if you can help me with a  
problem I've been having.</div>  
    <div class="small text-gray-500">Emily Fowler · 58m</div>  
  </div>  
</a>  
<a class="dropdown-item d-flex align-items-center" href="#">  
  <div class="dropdown-list-image mr-3">  
      
    <div class="status-indicator"></div>  
  </div>  
  <div>  
    <div class="text-truncate">I have the photos that you ordered last month, how  
would you like them sent to you?</div>  
    <div class="small text-gray-500">Jae Chun · 1d</div>  
  </div>  
</a>  
<a class="dropdown-item d-flex align-items-center" href="#">  
  <div class="dropdown-list-image mr-3">  
      
    <div class="status-indicator bg-warning"></div>  
  </div>  
  <div>  
    <div class="text-truncate">Last month's report looks great, I am very happy with the  
progress so far, keep up the good work!</div>
```

```
<div class="small text-gray-500">Morgan Alvarez · 2d</div>
</div>
</a>
<a class="dropdown-item d-flex align-items-center" href="#">
  <div class="dropdown-list-image mr-3">
    
    <div class="status-indicator bg-success"></div>
  </div>
  <div>
    <div class="text-truncate">Am I a good boy? The reason I ask is because someone told me that people say this to all dogs, even if they aren't good...</div>
    <div class="small text-gray-500">Chicken the Dog · 2w</div>
  </div>
</a>
<a class="dropdown-item text-center small text-gray-500" href="#">Read More Messages</a>
</div>
</li>
<div class="topbar-divider d-none d-sm-block"></div>
<!-- Nav Item - User Information -->
<li class="nav-item dropdown no-arrow">
  <a class="nav-link dropdown-toggle" href="#" id="userDropdown" role="button" data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
    <span class="mr-2 d-none d-lg-inline text-gray-600 small">Valerie Luna</span>
    
  </a>
  <!-- Dropdown - User Information -->
  <div class="dropdown-menu dropdown-menu-right shadow animated--grow-in" aria-labelledby="userDropdown">
    <a class="dropdown-item" href="#">
      <i class="fas fa-user fa-sm fa-fw mr-2 text-gray-400"></i>
```

```
Profile
</a>
<a class="dropdown-item" href="#"></i></i></i>
Logout
</a>
</div>
</li>
</ul>
</nav>
<!-- End of Topbar --&gt;
<!-- Begin Page Content --&gt;
&lt;?php
include 'connection.php';
?&gt;
&lt;div class="modal fade" id="addproducts" tabindex="-1" role="dialog" aria-labelledby="exampleModalLabel" aria-hidden="true"&gt;
&lt;div class="modal-dialog" role="document"&gt;
&lt;div class="modal-content"&gt;
&lt;div class="modal-header"&gt;
&lt;h5 class="modal-title" id="exampleModalLabel"&gt;Add Products&lt;/h5&gt;
&lt;button type="button" class="close" data-dismiss="modal" aria-label="Close"&gt;</pre>
```

```
<span aria-hidden="true">&times;</span>
</button>
</div>

<form action="action-productreg.php" style="border:1px solid #ccc" method="POST"
enctype="multipart/form-data" name="Register" id="Register" onsubmit="return validates()">
<div class="modal-body">
    <div class="form-group">
        <label for="state" style="color:Black;"><b>State</b></label>
        <select name="state" id="sel_state" class="form-control" required>
            <option value="" disabled selected>Select State</option>
            <?php
                $sql="SELECT * FROM `state`";
                $result= mysqli_query($conn, $sql);
                while($row = mysqli_fetch_assoc($result))
                {
                    echo "<option value=\"".$row['StCode']."'>".$row['StateName']."</option>";
                }
            ?>
            </select>
    </div>
    <div class="form-group">
        <label for="district" style="color:Black;"><b>District</b></label>
        <select name="district" id="sel_district" class="form-control" required="required">
            <option value="" disabled selected>--Choose District--</option>
        </select>
    </div>
    <div class="form-group">
        <label for="agencyname" style="color:Black;"><b>Agency Name</b></label>
        <select name="agencyname" id="sel_agency" class="form-control">
            <option value="" disabled selected>--Select Agency--</option>
        </select>
    </div>
```

```
<div class="form-group">
    <label for="productname"><b>Product Name</b></label>
        <input type="text" class="form-control" name="productname"
id="UProductname" placeholder="Enter Product Name" value="" maxlength="50"
required="required" oninput="testExist()">
    </div>
    <div class="form-group">
        <label for="description"><b>Description</b></label><br>
            <textarea class="form-control" name="description" id="Udescription" rows="5"
cols="50" placeholder="Please enter the details" required="required"></textarea>
        </div>
        <div class="form-group">
            <label for="catname"><b>Category Name</b></label>
                <select class="form-control" id="UCategory" name="catname"
required="required">
                    <option value="" disabled selected></option>
                    <?php
                        $sql="SELECT * FROM `tbl_category`";
                        $result= mysqli_query($conn, $sql);
                        while($row = mysqli_fetch_assoc($result))
                        {
                            echo "<option value='".$row['catid']."'>".$row['catname']."</option>";
                        }
                    ?>
                </select>
        </div>
        <div class="form-group">
            <label for="image">Image</label>
            <input type="file" class="form-control-file" name="image" id="image">
        </div>
        <div class="form-group">
            <label for="amount">Amount</label>
```

```
<input type="text" class="form-control" name="amount" id="UAmount"
placeholder="Enter Product Amount" value="" maxlength="50" required="required">
</div>
<div class="form-group">
<label for="quantity">Quantity</label>
<input type="text" class="form-control" name="quantity" id="UAmount"
placeholder="Enter Product Quantity" value="" maxlength="50" required="required">
</div>
<div class="modal-footer">
    <button type="submit" name="Register" class="btn btn-outline-primary"
value="Register">Save</button>
    <button type="reset" name="cancel" class="btn btn-outline-danger">Cancel</button>
</div>
</form>
</div>
</div>
</div>
</div>
</div>
<div class="container-fluid">
<div class="card shadown mb-4">
<div class="card-header py-3">
<center>
    <h3 class="m-0 font-weight-bold text-primary"> ADD PRODUCTS </h3>
    <h6>
        <button type="button" class="btn btn-primary" data-toggle="modal" data-
target="#addproducts">
            Add Products
        </button>
    </h6>
</center>
</div>
</div>
</div>
```

```

<div class="container-fluid">
    <!-- DataTales Example -->
    <div class="card shadow mb-4">
        <div class="card-header py-3">
            <h6 class="m-0 font-weight-bold text-primary">Product Details</h6>
        </div>
        <div class="card-body">
            <div class="table-responsive">
                <?php
                    /* $query = "SELECT * FROM tbl_productreg"; */
                    $query = "SELECT
                        tbl_productreg.*,
                        tbl_agencyreg.agencyname,
                        tbl_category.catname
                    FROM
                        tbl_productreg
                    INNER JOIN `tbl_agencyreg` ON `tbl_agencyreg`.`agid` = `tbl_productreg`.`agid`
                    INNER JOIN `tbl_category` ON `tbl_category`.`catid` = `tbl_productreg`.`catid`
                    INNER JOIN `tbl_login` ON
                        `tbl_login.lid` = `tbl_productreg.lid`
                    WHERE `tbl_productreg`.`status` = 0 AND `tbl_productreg.lid` = $s_id
                    AND `tbl_companyreg.lid` = $s_id";
                    $query_run = mysqli_query($conn, $query);
                ?>
                <table class="table table-bordered" id="dataTable" width="100%" cellspacing="0">
                    <thead>
                        <tr>
                            <th> Sl No </th>
                            <th> Agency Name </th>
                            <th> Product Name </th>
                            <th> Description </th>
                            <th> Category Name </th>
                            <th> Product Image </th>
                            <th> Amount </th>
                            <th> Quantity </th>
                            <th> EDIT </th>
                            <th> DELETE </th>
                        </tr>
                    </thead>

```

```
<tbody>
<?php
if(mysqli_num_rows($query_run) > 0)
{
while($row = mysqli_fetch_assoc($query_run))
{
?>
<tr>
<td><?php echo $row['prdid']; ?></td>
<td><?php echo $row['agencyname']; ?></td>
<td><?php echo $row['productname']; ?></td>
<td><?php echo $row['description']; ?></td>
<td><?php echo $row['catname']; ?></td>
<td> <?php echo ''></td>
<td><?php echo $row['amount']; ?></td>
<td><?php echo $row['quantity']; ?></td>
<td>
<form action="producteditform.php" method="post">
<input type="hidden" name="edit_prdid" value="<?php echo $row['prdid']; ?>">
<button type="submit" name="edit_btn" class="btn btn-success">EDIT</button>
</form>
</td>
<td>
<form action="deleteproductdetails.php" method="post">
<input type="hidden" name="delete_prdid" value="<?php echo $row['prdid']; ?>">
<button type="submit" name="delete_btn" class="btn btn-danger">DELETE</button>
</form>
</td>
```

```
</tr>
<?php
}
}

else {
    echo "No Record Found";
}
?>

</tbody>
</table>
</div>
</div>
</div>
</div>

<?php
include('includes/scripts.php');
?>
<!-- Page level plugins -->
<script src="vendor/datatables/jquery.dataTables.min.js"></script>
<script src="vendor/datatables/dataTables.bootstrap4.min.js"></script>
<!--Page level custom scripts -->
<script src="js/demo/datatables-demo.js"></script>
<script>
$(document).ready(function(){
    $("#sel_state").change(function(){
        var deptid = $(this).val();
        $.ajax({
            url: 'new.php',
            data: {
                action: deptid
            },
        });
    });
});
```

```
dataType: 'json',
type: 'post',
success: function (response) //we got the response
{
    var len = response.length;
    if(len == 0){
        $("#sel_district").empty();
        $("#sel_district").append("<option value=">'NO OPTION AVAILABLE'</option>");
    }
    else
    {
        $("#sel_district").empty();
        for( var i = 0; i<len; i++){
            var id = response[i]['id'];
            var name = response[i]['name'];
            $("#sel_district").append("<option value=\"" + id + "\">" + name + "</option>");
        }
    }
},
error: function (exception) {
    alert('Exception:' + exception);
}
});
$( "#sel_district" ).change(function(){
var deptsid = $(this).val();
$.ajax({
    url: 'newcs.php',
    data: {
        action: deptsid
    },
    dataType: 'json',
```

```

type: 'post',
success: function (response) //we got the response
{
    var len = response.length;
    if(len == 0){
        $("#sel_agency").empty();
        $("#sel_agency").append("<option value='>'NO OPTION AVAILABLE'</option>");
    }
    else
    {
        $("#sel_agency").empty();
        for( var i = 0; i<len; i++){
            var sid = response[i]['sid'];
            var sname = response[i]['sname'];

            $("#sel_agency").append("<option value='"+sid+"'>" +sname+"</option>");
        }
    }
},
error: function (exception) {
    alert('Exception:' + exception);
}
});
});
});
});

</script>
<script>
$(document).ready(function(){
    $('#search').keyup(function(){
        search_table($(this).val());
    });
    function search_table(value){

```

```

$('#employee_table tr').each(function(){
    var found = 'false';
    $(this).each(function(){
        if($(this).text().toLowerCase().indexOf(value.toLowerCase()) >= 0)
        {
            found = 'true';
        }
    });
    if(found == 'true')
    {
        $(this).show();
    }
    else
    {
        $(this).hide();
    }
});
});

</script>
<script src="//ajax.googleapis.com/ajax/libs/jquery/2.0.2/jquery.min.js"></script>
<?php
    include('includes/footer.php');
?
</body>
</html>

```

action-products.php

```

<?php
include 'connection.php';
session_start();
if(ISSET($_SESSION["lid"])){

```

```
$s_id=$_SESSION["lid"];
$StateName = $_POST['state'];
$DistrictName = $_POST['district'];
$agencyname = $_POST['agencyname'];
$productname = $_POST['productname'];
getDescription = $_POST['description'];
$catname= $_POST['catname'];
$image= $_FILES["image"]["name"];
$amount = $_POST['amount'];
$quantity = $_POST['quantity'];
$sql= "INSERT INTO `tbl_productreg`(`prdid`, `lid`, `StCode`, `DistCode`, `agid`, `productname`, `description`, `catid`, `image`, `amount`, `quantity`, `status`) VALUES
('$s_id','$StateName','$DistrictName','$agencyname','$productname','$description','$catname','$image','$amount','$quantity')";
if(mysqli_query($conn, $sql))
{
    move_uploaded_file($_FILES["image"]["tmp_name"],"upload/".$_FILES["image"]["name"]);
    echo '<script type="text/javascript">';
    echo 'alert("Registered Successfully")';
    echo '</script>';
    echo
    ("<script>location.href='http://localhost/BEDAZZLEDMARKETING/HANDICRAFTERDASH
BOARD/addproducts.php'</script>");
} else
{
    echo '<script type="text/javascript">';
    echo 'alert("Something Wrong")';
    echo '</script>';
}
mysqli_close($conn);
?>
```

2.12.2 SCREENSHOTS

Home Page

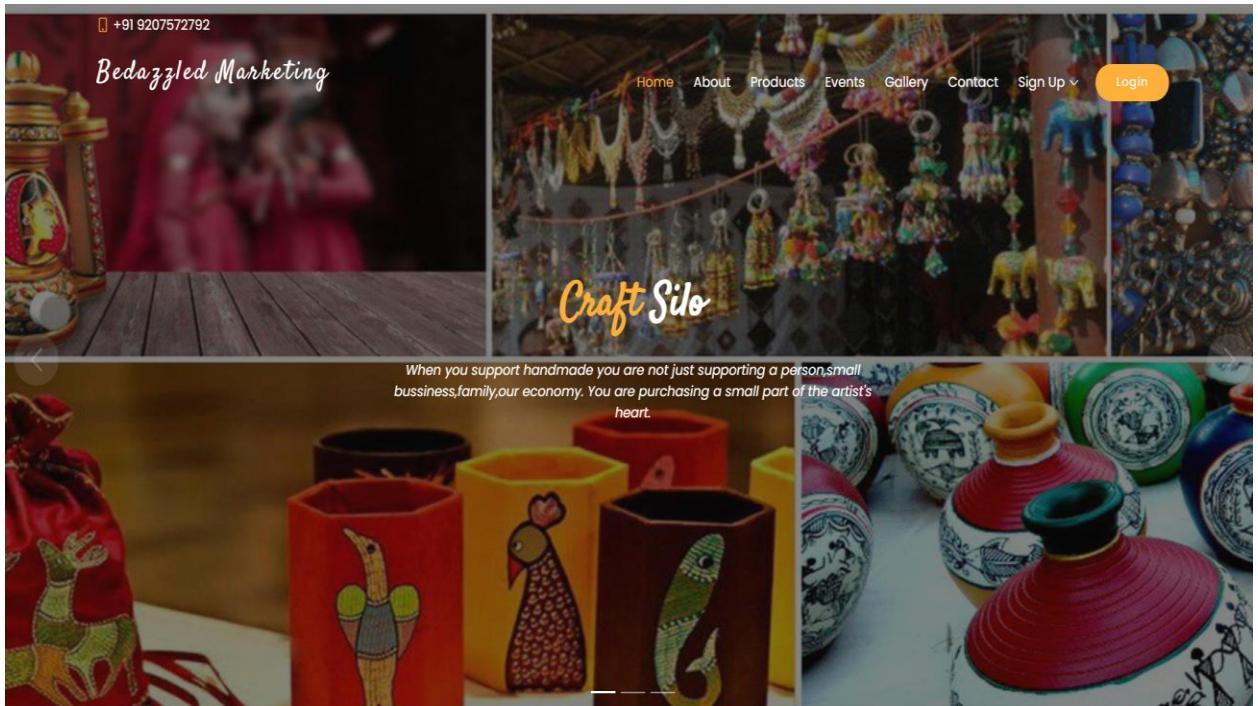


Fig 1: Home Page

LOGIN PAGE

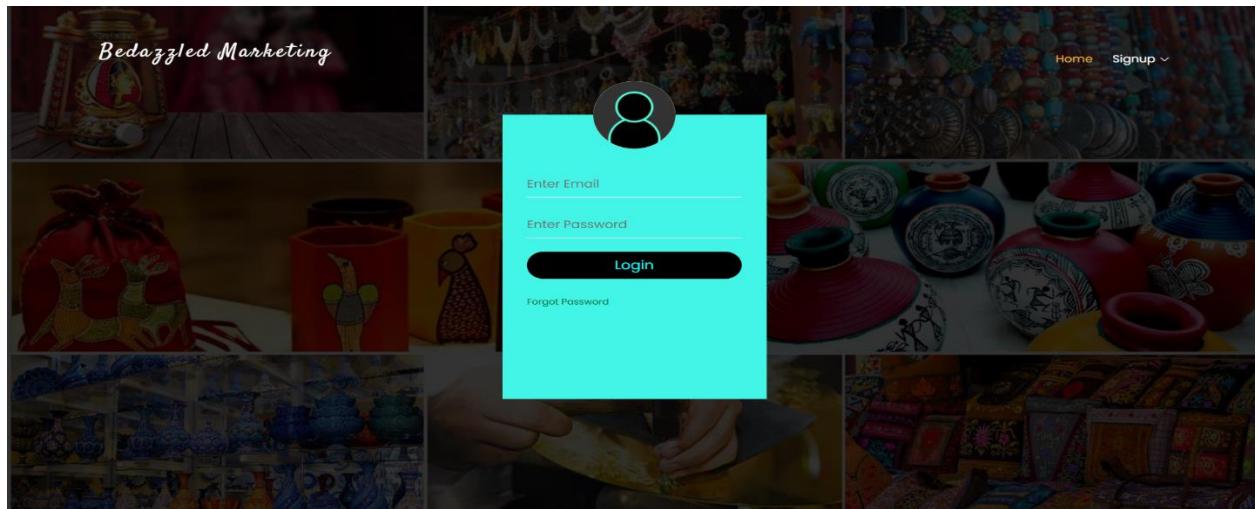


Fig 2: Login Page

USER REGISTRATION PAGE

User Registration Form

Please fill in this form to create an account.

Name

House Name

State

District

City/Landmark

Post Office

Pincode

Gender Male Female Other

Phone Number

Email

Password

Confirm Password

Remember me

By creating an account you agree to our [Terms & Privacy](#).

Fig 3: User Registration Form

HANDICRAFTER REGISTRATION PAGE

Handicrafter Registration Form

Please fill in this form to create an account.

Handicrafter Name

House Name/Building

State

District

City/Landmark

Post Office

Pincode

Gender Male Female Other

Phone Number

Email

Password

Confirm Password

Remember me

By creating an account you agree to our [Terms & Privacy](#).

Fig 4: Handicrafter Registration Form

AGENCY REGISTRATION PAGE

Agency Registration Form

Please fill in this form to create an account.

Agency Name

Apartment Name

State

District

City/Landmark

Post Office

Pincode

Phone Number

Email

Password

Confirm Password

Remember me

By creating an account you agree to our [Terms & Privacy](#).

Fig 5: Agency Registration Form

ADMIN PAGES

Admin Home

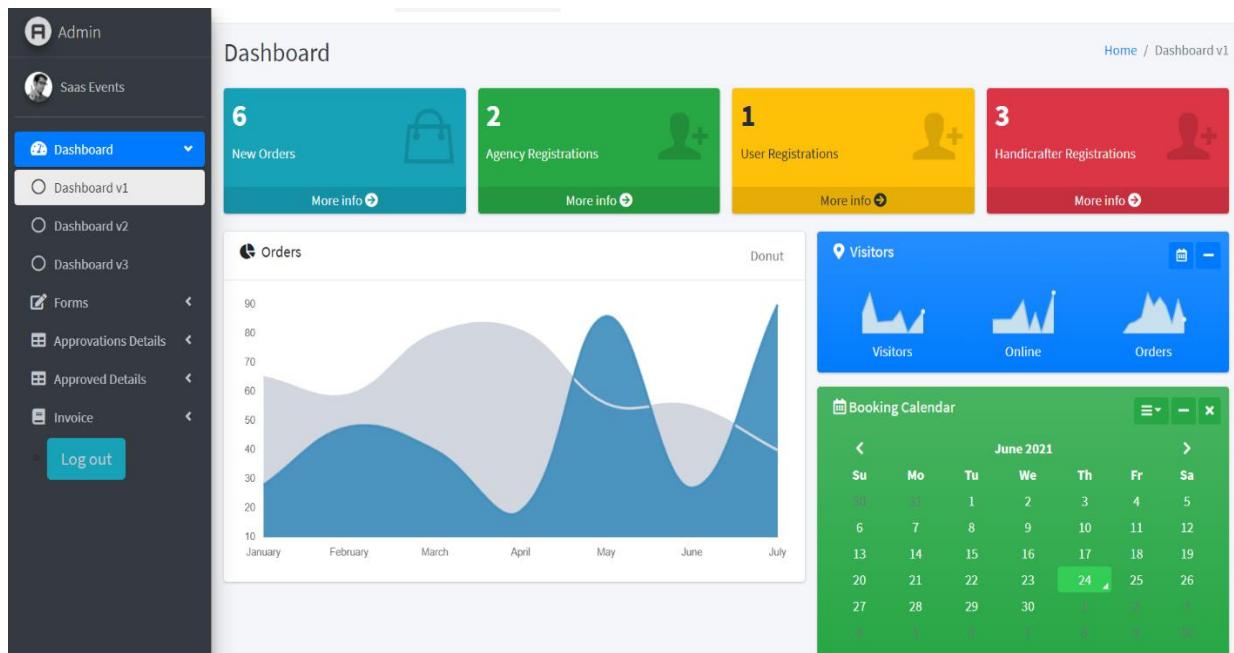


Fig 6: Admin Home

Add Event Details

The screenshot shows the "Event Details" page with the following structure:

- Left Sidebar:** Includes links for Admin, Alexander Pierce, Dashboard (selected), Dashboard v1, Forms, Approvals Details, Approved Details, and Log out.
- Header:** Shows "Home", "Contact", "Search", and navigation icons.
- Section Header:** "Event Details" with a "Home" link.
- Form Header:** "ADD EVENT DETAILS" with a "Add Event Details" button.
- Table Header:** "Event Details" with search and filter options.
- Data Table:** A table listing two events:

Sl No	Event Name	Type Name	Event Details	Place	Date	Start Time	End Time	Current Status	Action	Edit	Delete
1	Mahala	Art Works	It is a nice Art Works	Kottayam	2021-04-22	08:00am	07:00pm	Inactive	--Action--	<button>EDIT</button>	<button>DELETE</button>
2	Mahadhi	Wood Works	It is a nice wood works	Mammod	2021-04-24	9:00am	10:00am	Inactive	--Action--	<button>EDIT</button>	<button>DELETE</button>
- Pagination:** Shows "Showing 1 to 2 of 2 entries" and navigation buttons for "Previous", "1", and "Next".

Fig 7: Event Details

AGENCY PAGES

Agency Home

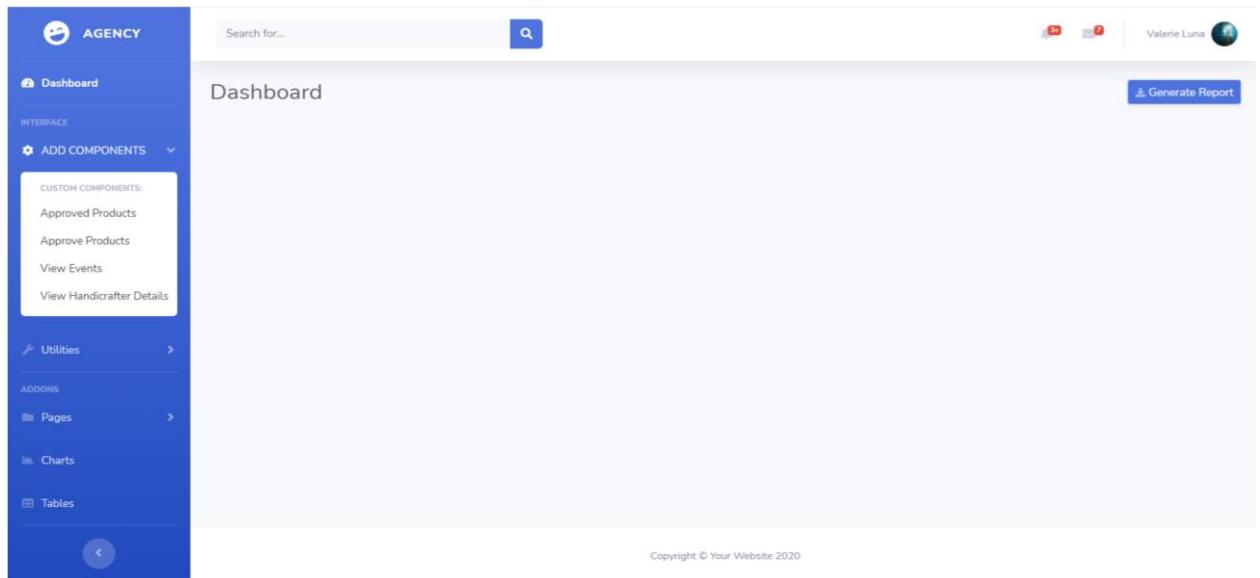


Fig 8: Agency Home

View Event Casting Details

Sl No	Event Name	Type Name	Event Details	Place	Date	Time
1	Mahala	Paper Works	It is a paper crafts	Kochi	2020-11-20	15:00:00
2	Mahala	Paper Works	It is of paper works	Mumbai	2020-11-20	17:35:00

Fig 9: View Event Casting Details

HANDICRAFTER PAGES

Handicrafter Home

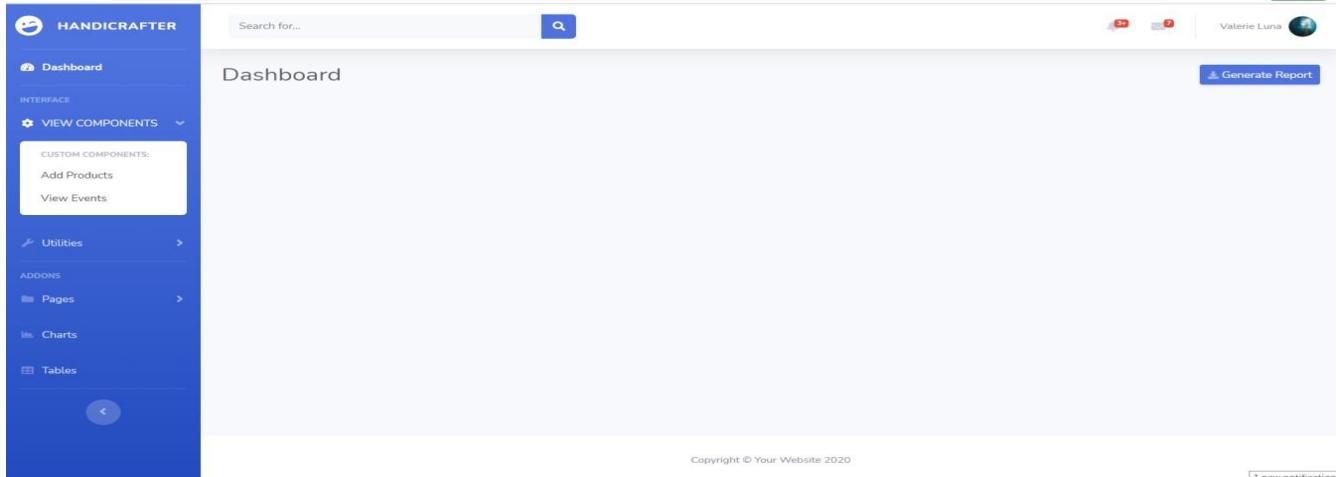


Fig 10: Handicrafter Home

Add Products Page

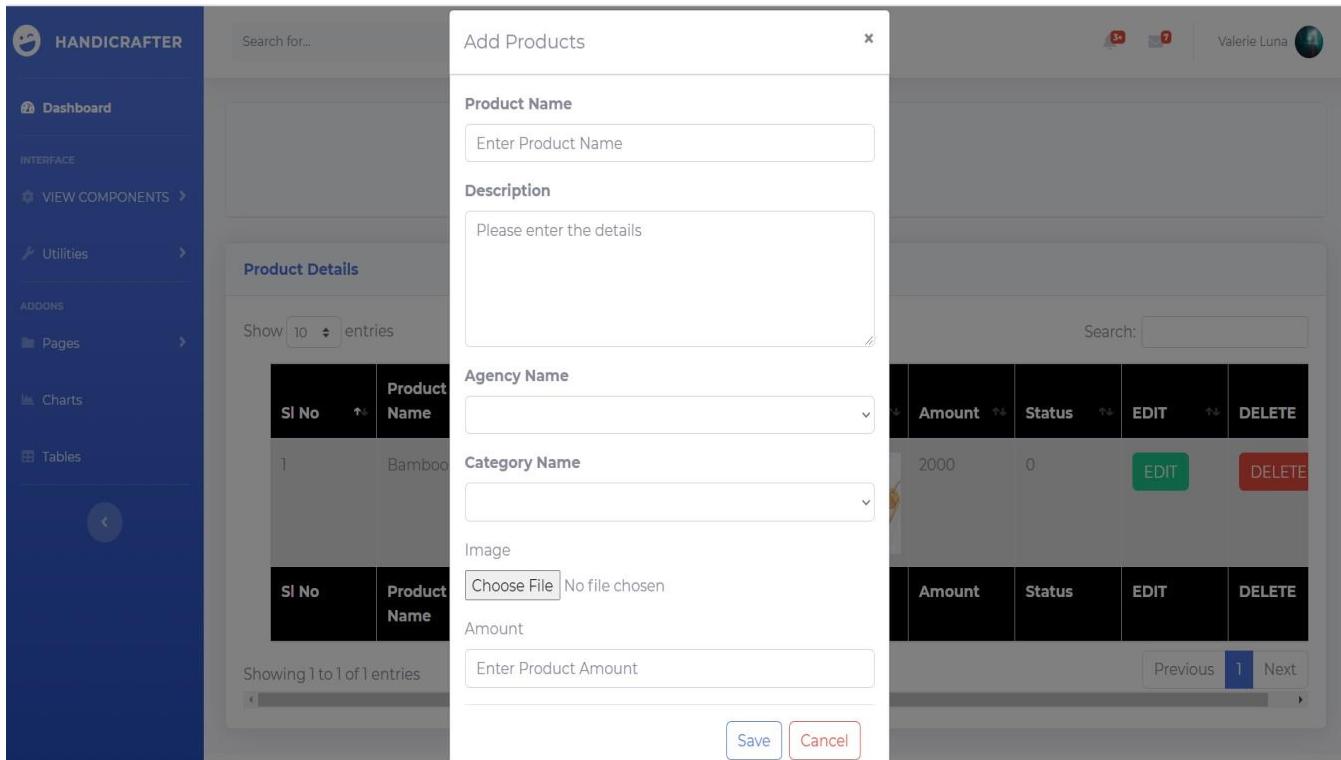


Fig 11: Add Products Page

Custom Order Details

Customer Order Details												Home
Order Details												
Show <input type="button" value="10"/> entries <input type="text" value="Search:"/>												
SL NO	Order Number	Item Name	Image	Item Quantity	Amount	Total Amount	Order Date	Order Time	Status	Action		
1	1	Teak Wood Models		2	1430	5975	14-06-2021	09:30 AM	Approved	--Action--		
2	2	Arabian Landscape		1	2840	2840	18-09-2021	08:40 AM	Approved	--Action--		

Showing 1 to 2 of 2 entries

Previous 1 Next

Fig 12: Custom Order Details

USER PAGES

User Home

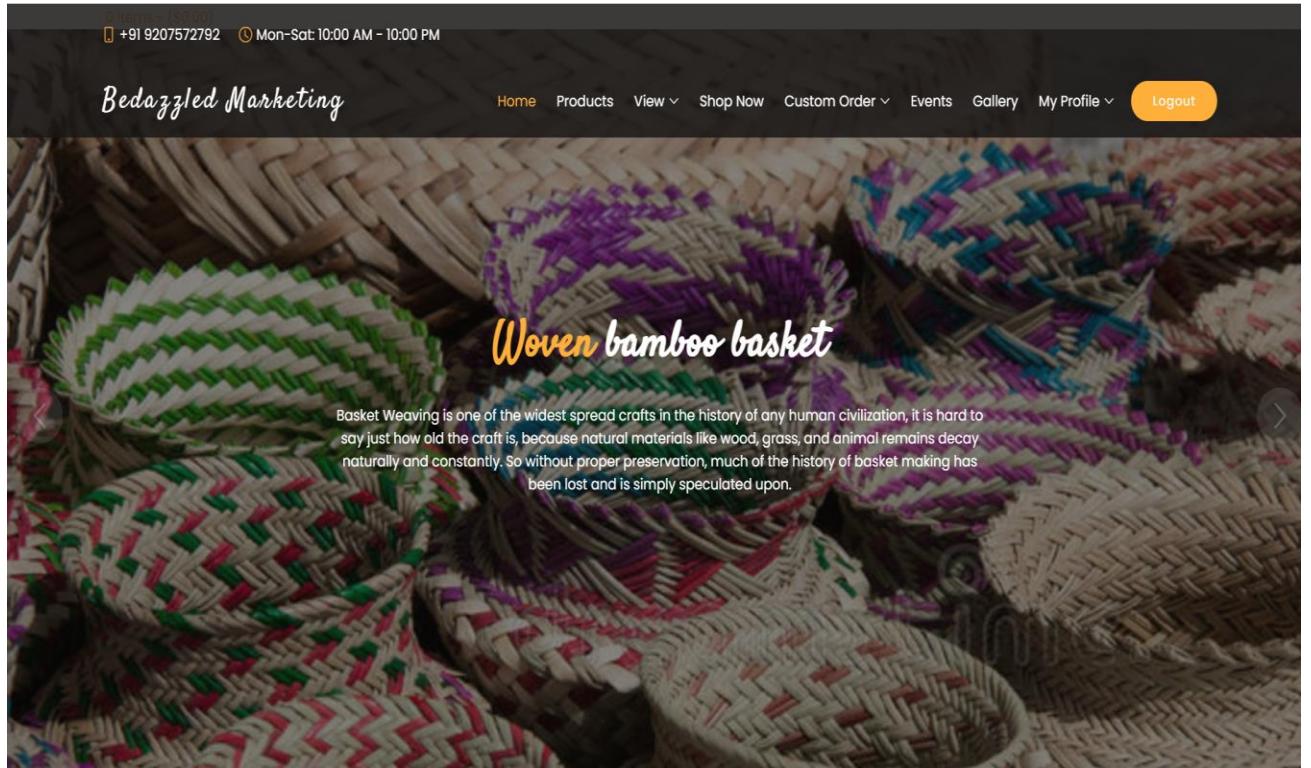


Fig 13: User Home

Products Page

Popular Products

WOODEN HANDICRAFTS

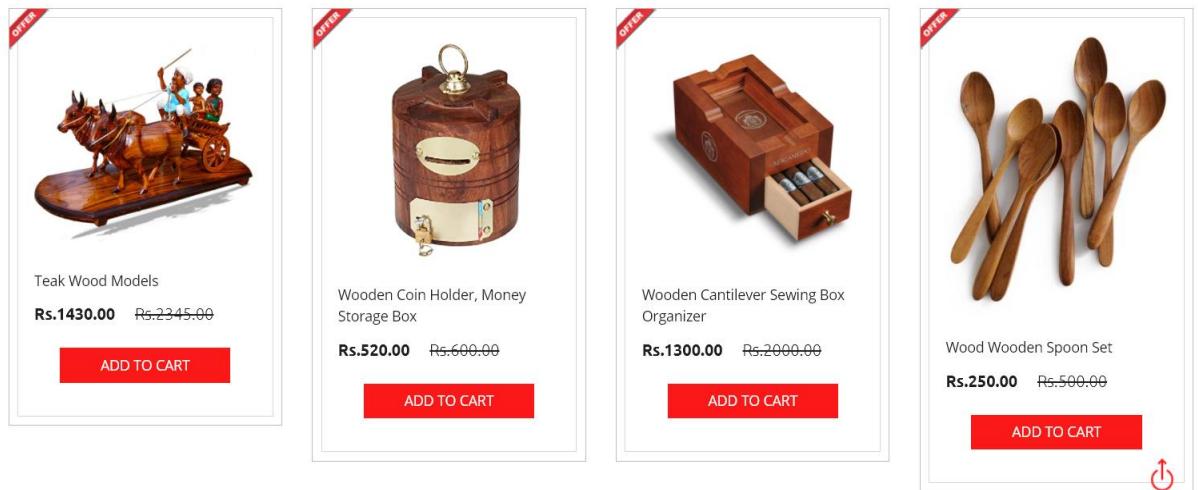


Fig 14: Product Booking

Single Page

TEAK WOOD MODELS

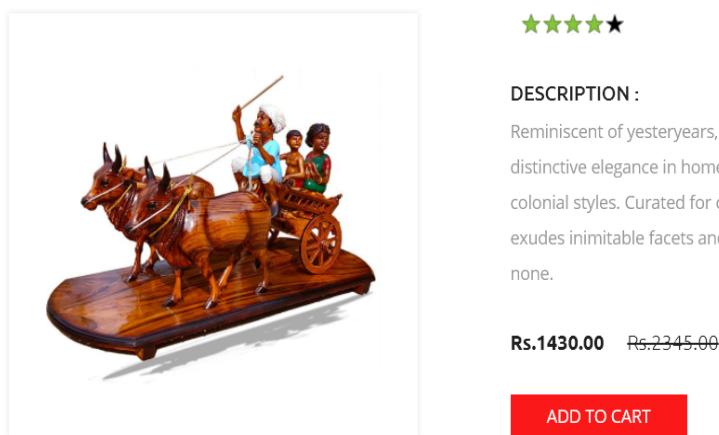


Fig 15: Single Page

Checkout Form

Checkout

YOUR SHOPPING CART CONTAINS: PRODUCTS

SL No.	Product	Quantity	Product Name	Price	Remove
1		2	Teak Wood Models	Rs.5975.00	
2		1	Arabian Landscape	Rs.2840.00	

Fig 16: Checkout Page

Bill Form

CONTINUE TO BASKET

ADD A NEW DETAILS

Item1 -	Rs. 13,500.00
Item2 -	Rs. 4000.00
Item3 -	Rs. 12000.00
Total Service Charges -	Rs. 100.00
Total -	Rs. 29,600.00

Delivery Address:

Full name:	<input type="text"/>
Mobile number:	<input type="text"/>
Landmark:	<input type="text"/>
Town/City:	<input type="text"/>
Address type:	<input type="text"/>

DELIVERY TO THIS ADDRESS

Make a Payment ➔


Fig 17: Bill Form

Payment Page

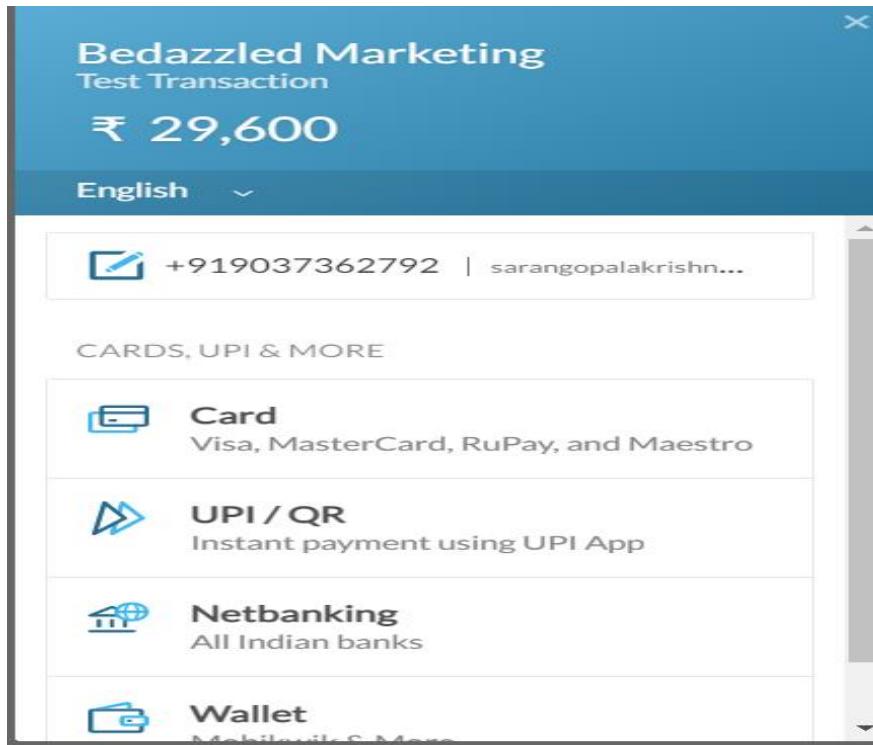


Fig 18: Payment Form

Feedback Form

Fig 19: Feedback Form