

**Tribhuvan University**

**Faculty of Humanities and Social Science**

**“GALLERY BIDDING SYSTEM “**

**(GALLER-E)**

**A PROJECT REPORT**

**Submitted to**

**Department of Computer Application**

**Kathmandu Business Campus**

***In partial fulfillment of the requirements for the Bachelors in Computer Application***

**Submitted by:**

Ashok Ranjitkar

BCA 6th Semester

6-2-1219-02-2021

Under the Supervision of

**Shaila Mitra Neupane**



**Tribhuvan University**

**Faculty of Humanities and Social Science**

**Kathmandu Business Campus**

**Supervisor’s** **Recommendation**

I hereby recommend that this project prepared under my supervision by Ashok Ranjitkar entitled “**GALLERY BIDDING SYSTEM**” in partial fulfillment of the requirements for the degree of Bachelor of Computer Application is recommended for the final evaluation.

**SIGNATURE**

Shaila Mitra Neupane

SUPERVISOR

BCA Department

Kathmandu Business Campus



**Tribhuvan University**

**Faculty of Humanities and Social Science**

**Kathmandu Business Campus**

**LETTER OF APPROVAL**

This is to certify that this project prepared by **Ashok Ranjitkar** entitled **“GALLERY BIDDING SYSTEM”** in partial fulfillment of the requirements for the degree of Bachelor in Computer Application has been evaluated. In our opinion it is satisfactory in the scope and quality as a project for the required degree.

Shaila Mitra Neupane Stella Koirala

Supervisor Program Coordinator

BCA Department Kathmandu Business Campus

Kathmandu Business Campus

Internal Examiner External Examiner

# ABSTRACT

The “**Gallery bidding System”** is a user-friendly platform for buying and selling photographs through bidding. It offers a platform for easy browsing, bidding, and purchasing of photographs by eliminating the need to visit physical gallery. This system is designed for both the photographers and buyers and it provides a convenient way to showcase and explore photographs. This system is made using the Apache web server for server hosting MariaDB as database and MySql for database management. Languages such as HTML and CSS and JavaScript for frontend, PHP for backend and Sql for database connection. This website can be accessed using login information stored in the database, where the admin can directly login. Users need to register an account to use the website for the first time. The users can upload their photographs with their details, view the products and bid on other products. The admin can do the same and also manage the users. The main objective of the “Gallery Bidding System” is to provide a platform for the buyers and sellers to trade the photographs with the availability of negotiation through bidding.

**Keywords: Photographs, Bidding, Gallery, Web Application, database, HTML, CSS, JavaScript, PHP, DBMS.**

# ACKNOWLEDGEMENT

I would like to express our special thanks to our supervisor **Mr. Shaila Mitra Neupane** who gave us the golden opportunity to create this wonderful project by the title **“Gallery Bidding System”,** which helped us in gaining more knowledge about various tools and technologies through research.

I would like to express our special thanks to ourBCA Program Coordinator **Ms.** **Stella Koirala** for his support and help for our personal development.

I am very grateful towards our institution Kathmandu Business Campus for their guidance and constant supervision as well as for providing necessary information.

I would also like to thank our parents, friends and teachers who helped us a lot in improving this project.

In the end, I would like to thank Tribhuvan University for giving us this opportunity via the course of Computer Application to help us understand the project ethics at this early stage and helped us to expand our knowledge.

Sincerely,

Ashok Ranjitkar

# LIST OF ABBREVIATIONS

|  |  |
| --- | --- |
| CRUD | Create, Read, Update and Delete |
| CSS | Cascading Style Sheet |
| DFD | Data Flow Diagram |
| ERD | Entity Relation Diagram |
| HTML | Hyper Text Markup Language |
| JS | Java Script |
| MySQL | My Structured Query Language |
| PHP | Hypertext Preprocessor |
| UI | User Interface |

# Table of Figures

Figure 3.1: Waterfall Model

Figure 3.1.2: Use-case Diagram of Gallery Bidding System

Figure 3.1.3: Gantt Chart

Figure 3.1.5: ER Diagram of Gallery Bidding System

Figure 3.1.6: DFD Level 0

Figure 3.1.6: DFD Level 1

Figure 3.2.1: System Flowchart for Admin

Figure 3.2.1: System Flowchart for User

Table of Contents

[ABSTRACT i](#_Toc207803975)

[ACKNOWLEDGEMENT ii](#_Toc207803976)

[LIST OF ABBREVIATIONS iii](#_Toc207803977)

[Table of Figures iv](#_Toc207803978)

[CHAPTER 1: 1](#_Toc207803979)

[INTRODUCTION 1](#_Toc207803980)

[1.1 Introduction 1](#_Toc207803981)

[1.2 Problem Statement 1](#_Toc207803982)

[1.3 Objectives 2](#_Toc207803983)

[1.4 Scope and Limitations 2](#_Toc207803984)

[1.5 Report Organization 3](#_Toc207803985)

[CHAPTER 2: 4](#_Toc207803986)

[BACKGROUD STUDY AND LITERATURE REVIEW 4](#_Toc207803987)

[2.1 Background Study 4](#_Toc207803988)

[2.2 Literature Review 4](#_Toc207803989)

[CHAPTER 3: 5](#_Toc207803990)

[SYSTEM ANALYSIS AND DESIGN 5](#_Toc207803991)

[3.1 System Analysis 5](#_Toc207803992)

[Technical Feasibility 7](#_Toc207803993)

[Operational Feasibility 7](#_Toc207803994)

[3.1.6 Process Modeling (DFD) 9](#_Toc207803995)

[3.2 System Design 11](#_Toc207803996)

[13](#_Toc207803997)

[3.3 Algorithm Explanation: Alpha Blending for Watermark 16](#_Toc207803998)

[3.3.1 What is Alpha Blending? 16](#_Toc207803999)

[3.3.2 Steps Used in This Project: 16](#_Toc207804000)

[3.3.3 Code: 16](#_Toc207804001)

[3.3.4 Result: 17](#_Toc207804002)

[CHAPTER 4: IMPLEMENTATION AND TESTING 18](#_Toc207804003)

[4.1 IMPLEMENTATION 18](#_Toc207804004)

[4.1.1 TOOLS USED 18](#_Toc207804005)

[4.1.2 IMPLEMENTATION DETAILS OF MODULES 19](#_Toc207804006)

[4.2 TESTING 20](#_Toc207804007)

[CHAPTER 5: 23](#_Toc207804008)

[CONCLUSION AND FUTURE RECOMMENDATION 23](#_Toc207804009)

[5.1 Lesson Learnt/ Outcomes 23](#_Toc207804010)

[5.2 Conclusion 23](#_Toc207804011)

[5.3 Future Recommendations 23](#_Toc207804012)

[References 24](#_Toc207804013)

[Appendix 25](#_Toc207804014)

[25](#_Toc207804015)

# CHAPTER 1:

# INTRODUCTION

## 1.1 Introduction

Photo Bidding System is a web-based application which is used to buy and sell photographs by bidding the price. This application is made for the users for browsing, buying selling and auctioning the photographs. With this project, I aim to provide a lightweight, easy to use platform for the photography enthusiasts without having to physically visit the galleries.

This application is very user-friendly and can be used with less technical knowledge and low hardware resource. This web-based application will help photographers to manage and organize their pictures as well as present them to potential buyers who are looking for such content. The buyers can browse the photographs and pay for the products through the Gallery Bidding System.

## 1.2 Problem Statement

There is an increase in the photographic field but there is a lack of proper online platforms to showcase their talents. Most of the other platforms cover the mainstream photographers only. Due to this the individual photographers struggle to monetize their work.

The buyers also have to physically visit the galleries to buy the photographs. There is little to no availability of negotiation between the buyers and sellers. There are very few well-established websites for the selling photographs through bidding system.

## 1.3 Objectives

The main objectives of Photo Bidding System:

To create an online platform that allows users to showcase their works

To provide photographers the medium to monetize their work

To implement watermark algorithm for digital image security

## 1.4 Scope and Limitations

#### 1.4.1 Scope

This project is a web-based application for photographers to sell their photographs to potential buyers.

To provide a negotiable environment to both buyers and sellers through the bidding system

#### 1.4.2 Limitations

This system has the following limitations:

This system cannot guarantee the originality of the photographs uploaded by the users

This system does not check if the email exists or not

## 1.5 Report Organization

**Introduction**

This chapter deals with the implementation of the system with its objectives and limitations along with the reason why the system is made.

**Background study and Literature Review**

This chapter summarizes the work that has been carried out in the field of auction and ecommerce websites and describes the features of existing applications related to bidding and selling.

**System Analysis and Design**

This chapter deals with the different requirements of the system, which describes the functional and non-functional requirements, feasibility analysis, ER- Diagram, Data Flow Diagram, design of the system with system architecture, database schema and interface design.

**Implementation and Testing**

This chapter focuses on the tools used in the system development, implementation details and the results of the test performed.

**Conclusion and Future Recommendation**

This chapter highlights brief summary of lesson learnt, outcome and conclusion of the whole project and explain what have been done and what further improvements could be done.

# CHAPTER 2:

# BACKGROUD STUDY AND LITERATURE REVIEW

## 2.1 Background Study

In the present context, I have similar types of websites like Hamro Bazaar, Freepik, and Shutterstock which are websites for browsing and selling the products but they don’t include the bidding system. Likewise, there are bidding websites such as eBay, which is an overall e-commerce website with bidding system but it is not specified towards photographs. They provide a wide range of pictures and illustrations but most of them are only available for premium users.

## 2.2 Literature Review

In recent years, the rapid change in digital technologies has transformed various industries, including the field of photography. There are various types of platforms that includes marketplace or brand dedicated stores with one objective to sell products online. [1] The arrival of online platforms has revolutionized the way photographers showcase their work and how consumers access and purchase visual content. The proposed project aims to contribute to this evolving landscape by developing an online photograph bidding system, providing a novel platform for photographers to exhibit their creations and potential buyers to engage in a dynamic bidding process. [2]

The proposed online photographs bidding system builds upon the existing literature by combining elements of digital transformation, online art marketplaces, auction models, user experience design, and addressing security considerations. [2] Many users are attracted to online shopping and using online services. Many online portals and shopping portals are launched. [3] The combination of these insights will contribute to the development of a robust platform that empowers photographers and enriches the online photography ecosystem.

# CHAPTER 3:

# SYSTEM ANALYSIS AND DESIGN

## 3.1 System Analysis

This system is designed with the use of the waterfall model that consists of requirement analysis, design, implementation, testing and deployment. It is planned to implement waterfall

Figure3. 1: Waterfall Model

time, fixed requirements, and well-understood technology. So, waterfall methodology is suitable to use.

Requirement Analysis

Requirement Analysis

System Design

System Design

Testing

Testing

Implementation

Figure 3.1: Use Case Diagram of Gallery Bidding SystemImplementation

#### 3.1.1 Requirement Analysis

For this system, requirements are basically identified through functional and non-functional requirements.

#### 3.1.2 Functional Requirements

The system should allow user to sell their photographs.

The system should allow user to bid on photographs.

The system should allow user to register and login.

#### Use-Case Diagram

In the Gallery Bidding System, there are two actors: admin and user, where the user can register, login, view, upload, and delete picture and bid. The admin can do most of the same operations and can manage users and manage pictures.

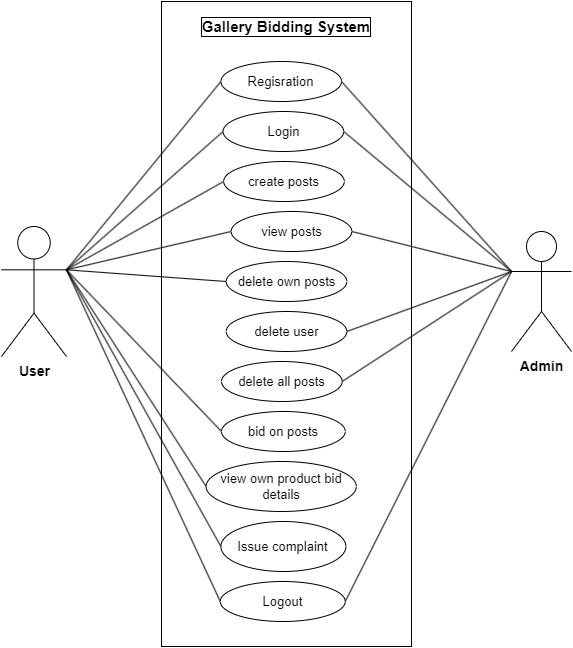


Figure3.1.2: Use-case diagram of Gallery Bidding System

#### 3.1.3 Non-Functional Requirements

**Scalable:** This project will be able to handle a huge amount of data as the project is made using Php.

**Secure:** This project will be secure as there is validation of users. The system requires 8-character password.

#### 3.1.4 Feasibility Study

### Technical Feasibility

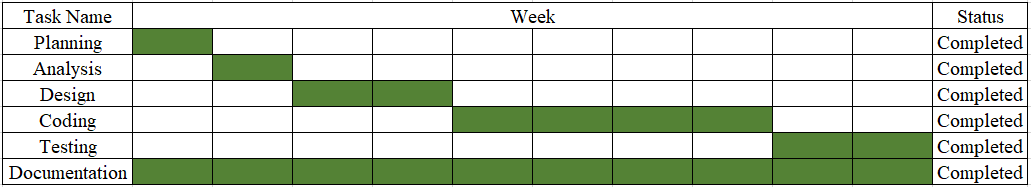
To design this system, off-shelf and existing technologies, software and hardware so there is no technological hurdle to build this system.

### Operational Feasibility

The system does not require extra software and hardware i.e., it uses open-source technologies. So, there is no recurring cost than just the internet connection.

#### Gantt Chart

Figure3.1.2: Gantt-Chart



#### 3.1.5 Data Modeling (ER-Diagram)

Figure3.1.5: ER-diagram of Gallery Bidding System

The Entity Relationship (ER) diagram of the Gallery Bidding System illustrates the interaction between the main entities of the system: Admin, User, Products, Bids, and Complaints. The admin entity manages the users, handles products, and reviews system activities. The User entity creates products for auction, performs bidding on available items, and can post complaints if issues arise. Each product, identified by a unique product ID, includes details such as name, description, category, starting bid, regular price, images, and bidding end date. The Bids entity records all bidding activities, storing bid amount, bid time, bid status, and linking each bid to both a user and a product. The Complaints entity allows users to raise issues, storing complaint details along with user information, category, and timestamps. Relationships between entities are clearly shown, such as one admin managing many users, users posting multiple complaints, and multiple users bidding on multiple products. This ER model ensures smooth interaction between system components and provides a structured foundation for database design.

## dfd1.drawio (2).png3.1.6 Process Modeling (DFD)

Figure 3.1.6: DFD level 0

The **Level 0 DFD** of the Gallery Bidding System provides a high-level overview of the entire system. It shows two main external entities: **User** and **Admin**, both interacting with the system. Users can upload, delete, and bid on pictures, as well as view bidding and product details. Admins, on the other hand, manage users and pictures while also handling user complaints. This diagram represents the system as a single process interacting with external entities, without going into internal details.

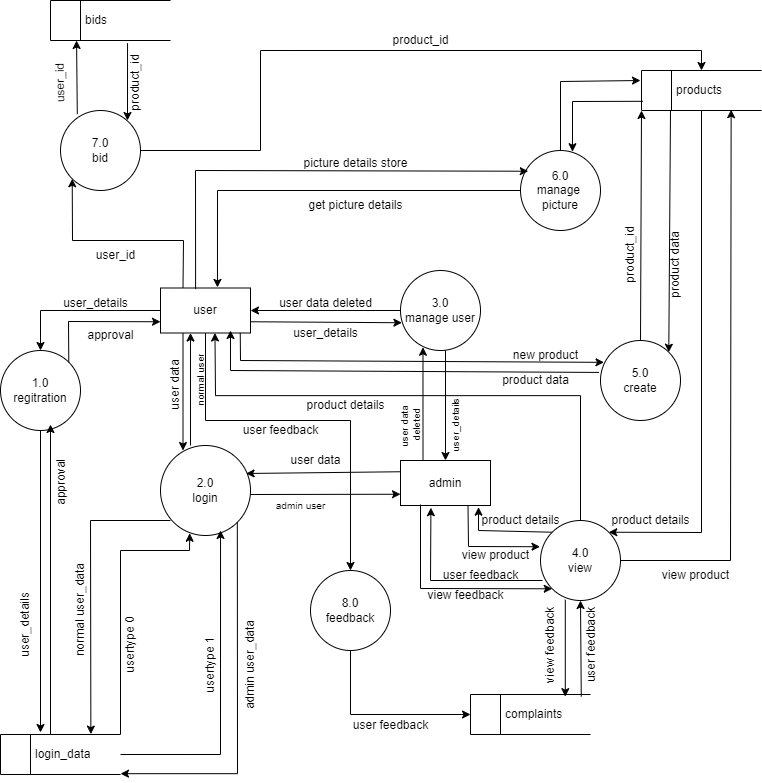


Figure 3.1.6: DFD level 1

The Level 1 DFD breaks down the Gallery Bidding System into detailed processes. It consists of multiple processes such as Registration (1.0), Login (2.0), Manage User (3.0), View (4.0), Create (5.0), Manage Picture (6.0), Bid (7.0), and Feedback (8.0).

* Users register and log in, after which they can create products, manage their pictures, and place bids.
* Admins manage user accounts, review product details, and monitor feedback and complaints.
* The system stores essential data like user data, product details, bids, login data, and feedback in relevant data stores.  
  This detailed breakdown ensures a clear understanding of how information flows within the system and how different modules interact to support the bidding process.

## 3.2 System Design

#### 3.2.1 System Flowchart

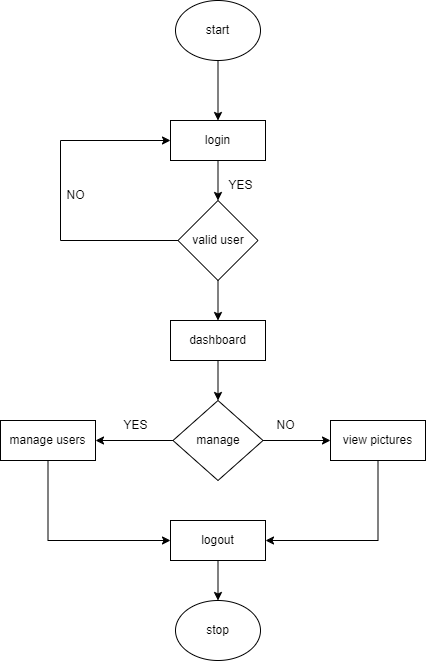
For admin

Figure 3.2.1: System Flowchart for Admin

The flow starts with login. If credentials are invalid, the admin loops back to try again; if valid, they proceed to the dashboard. From there, a manage decision determines the next action: choosing “YES” goes to manage users (add, edit, or remove users), while choosing “NO” goes to view pictures for browsing or reviewing content. From either path, the admin can logout, which ends the process at stop

For user

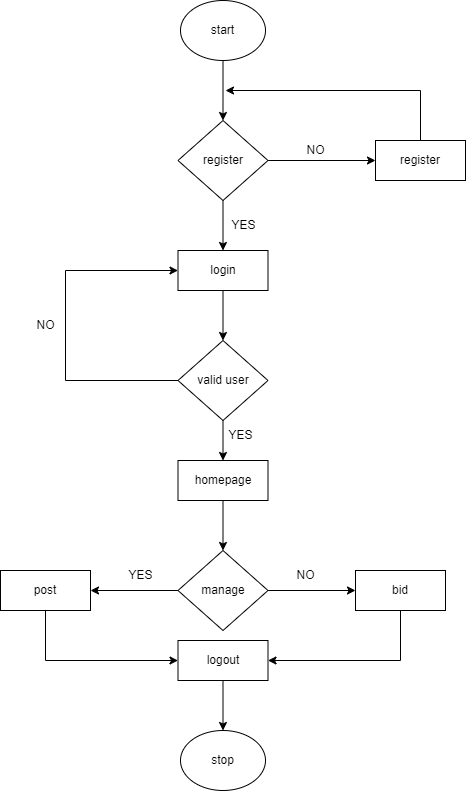
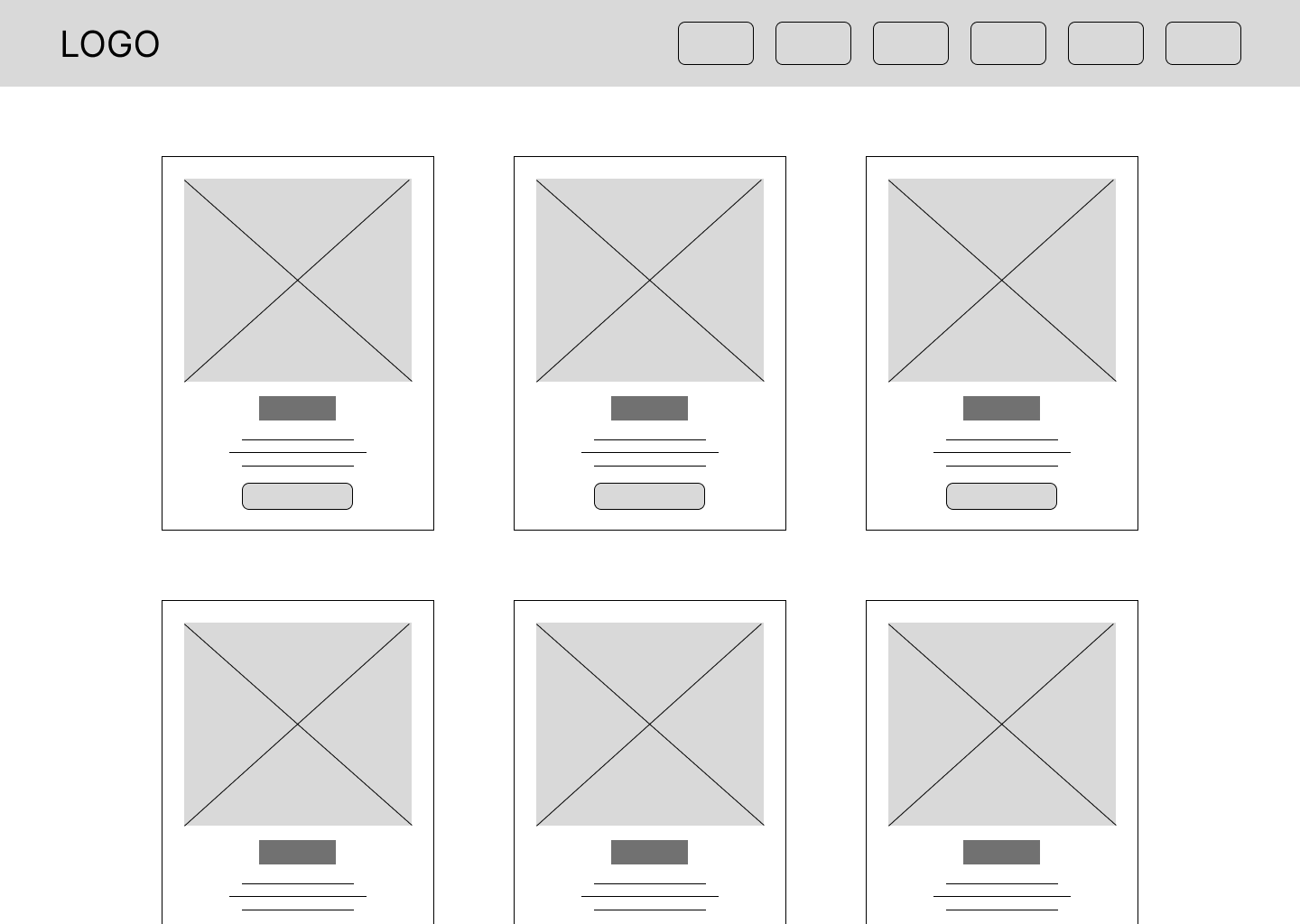


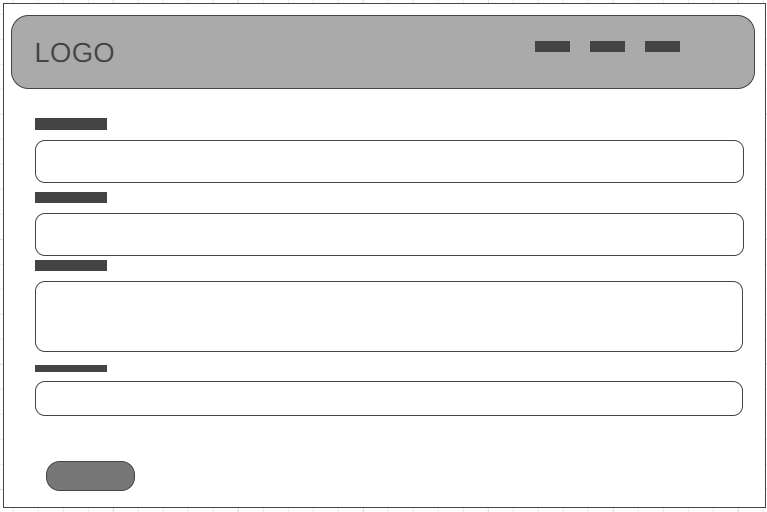
Figure 3.2.1: System Flowchart for user

The process begins at start. If the person isn’t registered, they’re sent to the register step and then return to try again. Registered users proceed to login; invalid credentials loop back to try logging in again. Valid users reach the homepage, where a manage decision determines their action: choosing “YES” leads to post (listing an item), while “NO” leads to bid (bidding on items). From either action, the user can logout, which leads to stop.

#### 1.png3.2.2 Interface Design

### 2.png





## 3.3 Algorithm Explanation: Alpha Blending for Watermark

## 3.3.1 What is Alpha Blending?

Alpha blending is a process that combines two images (original image and watermark) using a transparency factor (alpha) to create a blended image.

The formula used is:

Output Pixel = (Alpha × Watermark Pixel) + ((1 - Alpha) × Original Pixel)

## 3.3.2 Steps Used in This Project:

1. Load the original image and watermark image.  
2. Resize the watermark to match or scale proportionally.  
3. Define alpha (e.g., 0.3 for 30% opacity).  
4. Overlay the watermark onto the original image using alpha blending.  
5. Save the new watermarked image to the upload directory.

## 3.3.3 Code:

for ($x = 0; $x < $newWatermarkWidth; $x++) {

        for ($y = 0; $y < $newWatermarkHeight; $y++) {

            $wmColor = imagecolorat($resizedWatermark, $x, $y);

            $alpha = 0.2 ;

            if ($alpha > 0) {

                $wmR = ($wmColor >> 16) & 255;

                $wmG = ($wmColor >> 8) & 255;

                $wmB = $wmColor & 255;

                $imgColor = imagecolorat($image, $xPos + $x, $yPos + $y);

                $imgR = ($imgColor >> 16) & 255;

                $imgG = ($imgColor >> 8) & 255;

                $imgB = $imgColor & 255;

*// Manual blend*

                $newR = (int)(($alpha \* $wmR) + ((1 - $alpha) \* $imgR));

                $newG = (int)(($alpha \* $wmG) + ((1 - $alpha) \* $imgG));

                $newB = (int)(($alpha \* $wmB) + ((1 - $alpha) \* $imgB));

                $newColor = imagecolorallocate($image, $newR, $newG, $newB);

                imagesetpixel($image, $xPos + $x, $yPos + $y, $newColor);

            }

        }

    }

## 3.3.4 Result:

This approach visually marks the image while making it harder to extract the original through browser tools.

# CHAPTER 4: IMPLEMENTATION AND TESTING

## 4.1 IMPLEMENTATION

### 4.1.1 TOOLS USED

Following tools are used for the development of this project

#### FRONTEND

HTML (HyperText Markup Language) is used to structure web pages by defining elements such as headings, paragraphs, images, and links.

CSS (Cascading Style Sheets) is used to style and format HTML elements, making web pages visually appealing and responsive.

JavaScript adds interactivity and dynamic behavior to web pages, enabling features like form validation, animations, and real-time updates.

#### BACKEND

PHP (Hypertext Preprocessor) is a server-side scripting language used to handle backend logic, process user requests, and connect with databases.

MySQL is a relational database system used in the backend to store, manage, and retrieve structured data efficiently.

#### SERVER

#### Apache is an open-source web server that executes PHP scripts and delivers web content to users via HTTP/HTTPS protocols.

#### DATABASE

#### A database stores and organizes data in a structured way, allowing easy access, retrieval, and management of information.

#### DOCUMENTATION TOOLS

MS Office: for overall documentation

Draw.io: for the diagrams

Wireframe.cc: making wireframe

## 4.1.2 IMPLEMENTATION DETAILS OF MODULES

Different modules of this system are described as below:

**Admin module:**

**Admin manage user:**

In this module, admin can perform view and delete actions. The admin can view all the existing users, delete the information of the users.

**Admin manage products:**

In this module, admin can perform view and delete actions. The admin can view all the existing products, delete the products information.

**User module:**

**User product module:**

In this module the user can perform view, create and delete actions. The user can view the product they create, view the bidders of their product, delete their products.

**User bid module:**

In this module the user can bid on the products they prefer. They cannot bid on their own product.

**Login module:**

In this module the users and admin are identified during login. The users and admin log into this system using the valid username and password

## 4.2 TESTING

#### Comprehensive Test Case Table

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test Case ID | Feature/Module | Test Scenario/Description | Input Data / Action | Expected Result | Actual Result | Status |
| TC-ALG-001 | Bidding Algorithm | Place valid bid | User A bids $100 on Product X | Bid accepted, highest bid updated | Bid accepted, highest bid updated | Pass |
| TC-ALG-002 | Bidding Algorithm | Place lower bid than current highest | User B bids $90 on Product X | Bid rejected, error shown | Bid rejected, error shown | Pass |
| TC-ALG-003 | Bidding Algorithm | Tie bids (same amount) | User C bids $100 after User A | First bid remains highest | First bid remains highest | Pass |
| TC-ALG-004 | Winner Selection | Auction ends, highest bidder wins | Auction for Product X ends | User A declared winner | User A declared winner | Pass |
| TC-ALG-005 | Winner Selection | No bids placed | Auction for Product Y ends | No winner, product unsold | No winner, product unsold | Pass |
| TC-USER-001 | User Registration | Register with valid data | Valid username, email, password | User account created, login possible | User account created, login possible | Pass |
| TC-USER-002 | User Registration | Register with existing email | Email already in use | Error: Email already registered | Error: Email already registered | Pass |
| TC-USER-003 | User Login | Login with correct credentials | Valid username & password | Login successful, dashboard shown | Login successful, dashboard shown | Pass |
| TC-USER-004 | User Login | Login with wrong password | Valid username, wrong password | Error: Invalid credentials | Error: Invalid credentials | Pass |
| TC-USER-005 | User Profile Update | Update profile info | Change name, upload new photo | Profile updated, changes visible | Profile updated, changes visible | Pass |
| TC-USER-006 | User Deletion | Delete own account | User requests account deletion | Account deleted, cannot login | Account deleted, cannot login | Pass |
| TC-ADMIN-001 | Admin Login | Admin logs in with valid credentials | Valid admin username & password | Admin dashboard shown | Admin dashboard shown | Pass |
| TC-ADMIN-002 | Admin User CRUD | Admin creates new user | Admin fills user form | New user appears in user list | New user appears in user list | Pass |
| TC-ADMIN-003 | Admin User CRUD | Admin edits user info | Admin updates user email | User info updated in system | User info updated in system | Pass |
| TC-ADMIN-004 | Admin User CRUD | Admin deletes user | Admin deletes user from list | User removed, cannot login | User removed, cannot login | Pass |
| TC-ADMIN-005 | Admin Product CRUD | Admin adds new product | Admin fills product | Product appears in product list | Product appears in product list | Pass |
| TC-ADMIN-006 | Admin Product CRUD | Admin edits product | Admin updates product price | Product info updated | Product info updated | Pass |
| TC-ADMIN-007 | Admin Product CRUD | Admin deletes product | Admin deletes product from list | Product removed from system | Product removed from system | Pass |
| TC-SYS-001 | Access Control | User tries to access admin page | User logged in, visits admin URL | Access denied, redirected to login | Access denied, redirected to login | Pass |
| TC-SYS-002 | Access Control | Admin tries to access user-only page | Admin logged in, visits user page | Access denied or redirected | Access denied or redirected | Pass |
| TC-SYS-003 | File Upload | User uploads valid image | Valid image file | File uploaded, visible in profile | File uploaded, visible in profile | Pass |
| TC-SYS-004 | File Upload | User uploads invalid file type | .exe file | Error: Invalid file type | Error: Invalid file type | Pass |
| TC-SYS-005 | SQL Injection | Attempt SQL injection in login | ' OR 1=1 -- as username | Error: Invalid credentials, safe query | Error: Invalid credentials, safe query | Pass |

# CHAPTER 5:

# CONCLUSION AND FUTURE RECOMMENDATION

## 5.1 Lesson Learnt/ Outcomes

While developing this project, I have been able to gain knowledge regarding the various aspects of planning and organization. I have learned the importance of establishing clear goals and outlining the project’s structure in advance. Moreover, working with HTML and CSS has helped us realize the significance of responsive design and user experience. Furthermore, collaborating on the project highlighted the importance of effective communication and teamwork. By developing this project, I learnt the aspects of web development and continuous learning.

## 5.2 Conclusion

The Gallery Bidding System is being developed by fulfilling the objectives previously defined. The system provides a smooth user interface that makes it easy for the non-technical users to use the system easily. The verification of the user is done during the registration and login to increase security of the system. The system provides an easy platform for users to browse, buy, sell and auction the photographs.

## 5.3 Future Recommendations

The development of this project could have been more efficiently handled in design and development. The system can be updated based on the users’ requirements recommendation. Some of the future recommendations are given below:

Category system can be added

The product can be shown as per the users’ preferences to assure the new products be seen

## References

|  |  |
| --- | --- |
| [1] | B. M. Singh, "Challenges in adoption and implementation of E-Commerce platform: Case study of Nepal," June 2021. |
| [2] | B. Aljaf, "Online Auction System," 2016. |
| [3] | R. Vaidya, "Online Shopping in Nepal: Preferences and Problems," *The Journal of Nepalese Business Studies,* vol. XII, 2019. |

## Appendix

## 

