

**A
PROJECT REPORT
ON
“STOCK IMAGE SELLING”**

**SUBMITTED AS A PARTIAL REQUIREMENT
FOR THE DEGREE OF
BACHELOR OF COMPUTER APPLICATION
(B.C.A.)**



**SUBMITTED BY:
PANCHAL MAYUR D. (2322)
SALVI DEVANSHI N. (2350)**

**GUIDED BY:
PROF.RIMPLE DESAI**

**SHREE UTTAR GUJARAT BCA COLLEGE
AFFILIATED TO
VEER NARMAD SOUTH GUJARAT UNIVERSITY, SURAT
YEAR 2022-23**

DECLARATION

It was good eventful session at Shree Uttar Gujrat BCA College working with highly qualified computer teachers and will probably remain the most memorable experience of my life. Hence it is pleasure to acknowledge obligation to all those who helped directly and indirectly in my seminar work.

I am thankful to the **Prof. Rimple Desai** for giving me opportunity to complete the Project and guide me. I'm also thankful to I/c principal **Mr. Pradip H. Rajput** for help and support. I'm also thankful to all professors who helped by giving various suggestions and guidance throughout the project.

Date:
Place : Surat

Panchal Mayur D.(2322)
Salvi Devanshi N.(2350)

ABSTRACT

Selling stock images refers to the process of licensing and distributing images for commercial use. It involves photographers or creators licensing their images to agencies or platforms that sell them to businesses and individuals looking for visual content. Stock images can include photos, illustrations, videos, and other forms of digital content. The use of stock images allows businesses to access a wide variety of visual content without the cost and time required to create custom imagery. The stock image industry has evolved significantly over the years, with the rise of digital platforms and the increasing demand for visual content in various fields, including advertising, marketing, and publishing. The process of selling stock images typically involves photographers or creators submitting their work to stock agencies or platforms, which then review and curate the images before making them available for licensing. The photographer or creator receives a percentage of the licensing fees earned from their images, and the agency or platform typically handles the marketing, distribution, and licensing of the content.

INDEX

SR. NO.	CONTENT	PAGE NO.
1	Introduction	
	1.1 Project Profile	01
2	Environment Description	
	2.1 Hardware & Software Requirements	02
	2.2 Technology Used	03
3	System Requirement Specification	
	3.1 Data Dictionary	14
	3.2 Feasibility Analysis 3.2.1 Economic Feasibility 3.2.2 Technical Feasibility 3.2.3 Behavioural Feasibility	18
	3.3 E-R Diagram	19
	3.4 Data Flow Diagram/ UML Diagrams(Class Diagram, Use Case Diagram, Activity Diagram, Sequence Diagram)	20
4	Software Project Planning	
	4.1 Scope	29
	4.2 Objectives	30
	4.3 Expected Advantages	31
5	System Design	
	5.1 Input Design	32
	5.2 Output Design	44
6	Software Testing	
	6.1 Introduction	57
	6.2 Objectives	57
	6.3 Levels of Testing	57
	6.4 Test Cases	59
7	Limitation and System Enhancement	
	7.1 Limitation	61
	7.2 System Enhancement	62
8	Conclusion	63
9	Bibliography	64

1.Intorduction

- Stock images are defined as the archive of images that can be licensed commercially and it's a vast and vastly useful resource for a wide variety of applications ranging from advertising and social media to graphic design to websites to image-specific uses in sectors like architecture and real estate. Let's take a quick look at the basics of stock images to see how they might become useful to you as a window film dealer.
- It allows both the admin and the user to search for the desired images.
- The system helps both users and image manager to keep of all the images available.

1.1 Project Profile

Development Tools	
Front End	Html, CSS, JavaScript, jQuery, Bootstrap, Ajax
Back End	Python, Django, ORM(Object Relation Method)
Editors	Sublime 3
Application Server	Windows PowerShell
Web Browser	Chromium, Chrome
Platform Tools	HTML5, CSS3
Extra Technologies	JavaScript, Ajax

2.Environment Description

- The efficient hardware and software configuration requires running the system is as suggest below. The configuration suggested is for better performance. Same functionality or higher configuration will always better.

2.1 Hardware & Software Requirements

❖ Hardware Requirements:

Components	Development
Processor:	Intel(R) Core(TM) i5-10300H CPU @ 2.50GHz 2.50 GHz
RAM:	8.00 GB
Hard-Disk:	512 GB SSD

❖ Software Requirements:

Components	Development
Operating System:	Windows 11
Project Coding Application:	Visual Studio Code
Database Server:	MYSQL
Front-End Tool:	Html, CSS, JavaScript, jQuery, Bootstrap, Ajax
Back-End Tool:	Python, Django,ORM

2.2 Technology Used

Core Technology

- Python, Django
- MYSQL (Back-End)
- JavaScript, Ajax (Extra Technology)

Development Tools

- Visual Studio Code
- Windows PowerShell

Documentation Tools

- Microsoft Word
- EDRAW
- MS PowerPoint

Extra Technology

- JavaScript
- jQuery
- Ajax
- Bootstrap

Python:



- Python is a popular high-level programming language that is known for its simplicity, readability, and flexibility. It was created by Guido van Rossum in the late 1980s and released in 1991. Python is an interpreted language, meaning that code is executed directly without the need for compilation. It is used in a wide variety of applications, including web development, scientific computing, data analysis, artificial intelligence, and more. Python is known for its easy-to-learn syntax and powerful libraries, which allow developers to quickly build applications and prototypes. It is also an open-source language, meaning that it is freely available for anyone to use, modify, and distribute.

Python History:

- Python was created by Guido van Rossum in the late 1980s while he was working at the National Research Institute for Mathematics and Computer Science in the Netherlands. Van Rossum was inspired by the ABC programming language, which was developed in the 1980s as a language for teaching programming to beginners.
- Van Rossum wanted to create a language that was easy to learn and use, but also powerful enough to be used for real-world applications. He began working on Python in December 1989, and the first version of Python, version 0.9.0, was released in February 1991.
- Today, Python is one of the most popular programming languages in the world, used by developers in a wide range of fields, including web development, scientific computing, data analysis, artificial intelligence, and more. Its popularity can be attributed to its simplicity, readability, and flexibility, as well as its vast collection of third-party libraries and tools.

Django:



- Django is a popular high-level web framework that is written in Python. It was created in 2003 by Adrian Holovaty and Simon Willison while they were working at a newspaper company, the Lawrence Journal-World, in Kansas, USA. The goal of Django was to create a web framework that made it easier and faster to develop web applications.
- Django is designed to be flexible and modular, with a set of built-in tools and libraries that can be used to quickly build robust web applications. It includes features such as an ORM (Object-Relational Mapping) for working with databases, a templating system for generating dynamic HTML pages, and a powerful admin interface for managing data.
- One of the key features of Django is its "batteries included" philosophy, which means that it comes with many useful features and libraries built-in, so developers don't have to spend time searching for and integrating third-party tools.
- Django is widely used for building complex web applications, including content management systems, social networks, e-commerce sites, and more. Its popularity can be attributed to its ease of use, powerful features, and vibrant community of developers who contribute to its development and maintenance.

Why Django Use?

Django is a popular web framework that is used for building web applications in Python. There are several reasons why Django is a good choice for web development:

1.Rapid Development:

Django provides many built-in features and libraries that can be used to quickly build web applications, without having to write a lot of code from scratch. This can help speed up the development process and reduce the time and cost required to build a web application.

2.Scalability:

Django is designed to be scalable and can handle high traffic volumes and large amounts of data. It can be used to build applications of any size, from small prototypes to large-scale enterprise applications.

3.Security:

Django provides several built-in security features, such as protection against SQL injection attacks, cross-site scripting (XSS) attacks, and cross-site request forgery (CSRF) attacks. It also makes it easy to implement authentication and authorization features to control access to different parts of the application.

4.Versatility:

Django can be used to build a wide range of web applications, including content management systems, e- commerce sites, social networks, and more. It is also compatible with many different databases, including PostgreSQL, MySQL, and SQLite.

5.Community Support:

Django has a large and active community of developers who contribute to its development and maintenance. This means that there are many resources available, including documentation, tutorials, and libraries, to help developers build and maintain Django applications.

Overall, Django is a powerful and flexible web framework that can help developers quickly build secure and scalable web applications. It is a good choice for those who want to build complex web applications without having to write a lot of code from scratch.

ORM (Object Relation Model):

Overview:

- TypeORM is an Object Relational Mapper library running in node.js and written in TypeScript. TypeScript is an improvement to JavaScript with optional typing. TypeScript is a compiled language. It is not interpreted at run-time. The TypeScript compiler takes TypeScript files (.ts) and compiles them in to JavaScript files (.js).
- TypeORM supports multiple databases like MySQL, PostgreSQL, MariaDB, SQLite, MS SQL Server, Oracle, SAP Hana and WebSQL. TypeORM is an easy-to-use ORM to scaffold new apps that connect to databases. TypeORM functionality is RDBMS-specific concepts.
- We can quickly create new projects and micro-services. It is inspired from similar tool like Hibernate, Doctrine, Entity framework, etc., from other programming languages.

Features of ORM:

- Automatically create database table schemes based on your models.
- Easily insert, update, and delete object in the database.
- Create mapping (one-to-one, one-to-many and many-to-many) between tables.
- Provides simple CLI commands.

Benefits of ORM:

- High quality and loosely-coupled applications.
- Scalable applications.
- Easily integrate with other modules.
- Perfectly fits any architecture from small to enterprise apps.

SQLite

- SQLite is a popular lightweight relational database management system (RDBMS) that is used in many different applications. It is a serverless database, meaning that it does not require a separate server process to be running to access and modify the data stored in the database.
- SQLite is widely used because of its simplicity, flexibility, and efficiency. It is easy to set up and use and requires minimal configuration. It is also cross-platform, which means that it can be used on different operating systems, including Windows, Mac, Linux, and mobile platforms.
- SQLite supports standard SQL syntax, making it compatible with a wide range of applications and tools that use SQL. It is also highly efficient, with a small footprint and low overhead, making it ideal for use in embedded systems and other resource-constrained environments.
- Some common use cases for SQLite include mobile app development, desktop application development, web development, and data storage for Internet of Things (IoT) devices. It is also often used as a local cache for data that is stored remotely, such as in a cloud-based database.
- Overall, SQLite is a powerful and versatile database system that is widely used in many different applications and environments. Its ease of use, efficiency, and compatibility with SQL make it a popular choice for developers who need a lightweight and flexible database solution.

3. System Requirement Specification

Tables:

1. Admin:

Table Name: tbladmin

Primary Key: admin_id

Description: This table Stores admin's id, name, password, email, phone no, profile pic.

FIELD_NAME	DATATYPE	CONSTRAINTS	DESCRIPTION
adminid	int(100)	Autoincrement	Indicate Admin Id
adminname	varchar(50)	Not Null	Store the Username
password	varchar(100)	Not Null	Store the password
emailid	varchar(100)	Not Null	Store the Email id
phoneno	int (10)	Not Null	Store the Phone Number
profilepic	varchar (10)	Not Null	Store the Profile Picture

2. User:

Table Name: tbluser

Primary Key: user_id

Description: This table Stores user's id, name, password, email, phone no, profile pic.

FIELD_NAME	DATATYPE	CONSTRAINTS	DESCRIPTION
userid	int(50)	Autoincrement	Indicate User Id
username	varchar(100)	Not Null	Store the Username
password	varchar(15)	Not Null	Store the password
emailid	varchar(20)	Not Null	Store the Email id
phoneno	int(11)	Not Null	Store the Phone Number
profilepic	varchar(20)	Not Null	Store the Profile Picture
cityid	int(20)	Not Null	Store the City Id

3. Album

Table name: tblalbum

Primary key: albumid

Description: This table indicates album id , user id, title, status , description, thumbnail.

FIELD_NAME	DATATYPE	CONSTRAINTS	DESCRIPTION
albumid	int(50)	Autoincrement	Indicate User Id
userid	varchar(100)	Not Null	Store the Username
title	varchar(15)	Not Null	Store the password
status	varchar(20)	Not Null	Store the Email id
description	int(11)	Not Null	Store the Phone Number
thumbnail	varchar(20)	Not Null	Store the Profile Picture

4. Image

Table name: tblimage

Primary key: imageid

Description: This table indicates Image details.

FIELD_NAME	DATATYPE	CONSTRAINTS	DESCRIPTION
imageid	int(50)	Autoincrement	Indicate Image Id
url	varchar(20)	Not Null	Store the URL of Image
isapproved	varchar(20)	Not Null	Indicate Image is Approved or Not
status	varchar(20)	Not Null	Indicate Image Status
userid	int(50)	Not Null	Store the User Id
description	varchar(20)	Not Null	Indicate Description of Image
price	int(20)	Not Null	Indicates The Price of Image

5. Comment

Table Name: tblcomment

Primary key: commentid

Description: This table indicates comments details.

FIELD_NAME	DATATYPE	CONSTRAINTS	DESCRIPTION
commentid	int(50)	Autoincrement	Indicate Comment Id
userid	int(50)	Not Null	Store the User Id
imageid	int(100)	Not Null	Store the Image Id
comment	varchar(20)	Not Null	Store the Comment
date&time	datetime(6)	Not Null	Store the Profile Picture

6.Album Image

Table Name: tblalbumimage

Primary key: albumimageid

Description: This table indicates Album image details.

FIELD_NAME	DATATYPE	CONSTRAINTS	DESCRIPTION
albumimageid	int(50)	Autoincrement	Indicate Allbum Image Id
allbumid	int(50)	Not Null	Store the Allbum Id
imageid	int(50)	Not Null	Store the Image Id

7.Report User

Table Name: tblreportuser

Primary key: id

Description: This table indicates report user details.

FIELD_NAME	DATATYPE	CONSTRAINTS	DESCRIPTION
id	int(50)	Autoincrement	Indicate Id Which Has Reported
username	varchar(20)	Not Null	Store the Username
report_reason	varchar(20)	Not Null	Store the Report Reason

8.City

Table Name: tblcity

Primary Key: cityid

Description: This table indicates city details.

FIELD_NAME	DATATYPE	CONSTRAINTS	DESCRIPTION
cityid	int(50)	Autoincrement	Indicate City Id
stateid	int(20)	Not Null	Store the State Id
cityname	varchar(20)	Not Null	Store the City Name

9.Image Tag

Table Name: tblimagetag

Primary Key: imagetagid

Description: This table indicates image tag details.

FIELD_NAME	DATATYPE	CONSTRAINTS	DESCRIPTION
imagetagid	int(50)	Autoincrement	Indicate Image Tag Id
imageid	int(50)	Not Null	Indicate the Image Id
tagid	int(50)	Not Null	Store the Tag Id

10.Like

Table Name: tbllike

Primary Key: likeid

Description: This table indicates like details.

FIELD_NAME	DATATYPE	CONSTRAINTS	DESCRIPTION
likeid	int(50)	Autoincrement	Indicate Like Id
userid	int(20)	Not Null	Indicate the User Id
imageid	int(20)	Not Null	Store the Image Id

11.Follow

Table Name: tblfollow

Primary Key: followid

Description: This table indicates follow details.

FIELD_NAME	DATATYPE	CONSTRAINTS	DESCRIPTION
followid	int(50)	Autoincrement	Indicate Follow Id
followerid	int(20)	Not Null	Indicate the Follower Id
userid	int(20)	Not Null	Store the User Id

12.State

Table Name: tblstate

Primary Key: stateid

Description: This table indicates state details.

FIELD_NAME	DATATYPE	CONSTRAINTS	DESCRIPTION
stateid	int(50)	Auto_Increment	Indicate State Id
statename	varchar(50)	Not Null	Indicate the State Name

13. Tags

Table Name: tbltags

Primary Key: tagid

Description: This table indicates tags details.

FIELD_NAME	DATATYPE	CONSTRAINTS	DESCRIPTION
tagid	int(50)	Auto_Increment	Indicate Tag Id
tagcategories	varchar(50)	Not Null	Indicate the Tag Category

3.1 Data Dictionary

1.Admin:

Name	tblAdmin
Alias Name	None
Where and how you used?	Read Login Info (IN) Verify Admin (OUT)
Description	Admin Login with Following Fields: Username + password

2.User

Name	tblUser
Alias Name	None
Where and how you used?	Read Login Info (IN) Verify User (OUT)
Description	User Login with Following Fields: Username + password

3.Country

Name	tblCountry
Alias Name	None
Description	Country Following Fields: Countryid + Countryname

4.State

Name	tblState
Alias Name	None
Description	State Following Fields: stateid + statename + countryid

5.City

Name	tblCity
Alias Name	None
Description	City Following Fields: Cityid +cityname +stateid

6.Category

Name	tblCatagory
Alias Name	None
Description	Category Following Fields: Catagoryid + catagoryname +

7.Image

Name	tblImage
Alias Name	None
Description	Image Following Fields: Imageid + title + url + status +userid + description + price + categoryid

8. Like

Name	tblLike
Alias Name	None
Description	Like Following Fields: Likeid + userid + imageid

9. Comment

Name	tblComment
Alias Name	None
Description	Comment Following Fields: Commentid + imageid + userid +comment

10. Save

Name	tblSave
Alias Name	None
Description	Save Following Fields: Saveid + userid + imageid

11. Album

Name	tblAlbum
Alias Name	None
Description	Album Following Fields: Albumid + userid + title + status +description + thumbnail

12. AlbumImage

Name	tblAlbumImage
Alias Name	None
Description	AlbumImage Following Fields: Albumimageid + imageid + imageurl

13. Tags

Name	tblTags
Alias Name	None
Description	Tags Following Fields: Tagid + tag + Catagoryid

14. Order

Name	tblOrder
Alias Name	None
Description	Order Following Fields: Orederid + imageid + userid +date&time + price + razorpay_payment + razorpay_order_id + razorpay_signature

3.2 Feasibility Analysis

- All projects are feasible, provided that unlimited resources and infinite time are available. However, in the world that can sound like a dream. Especially computer Based systems are likely to be bound by limited resources as well as time.

3.2.1 Economic Feasibility

- The project is economically feasible as the only cost involved is having a computer with the minimum requirements mentioned earlier. For the users to access the application, the only cost involved will be in getting access to the Internet.

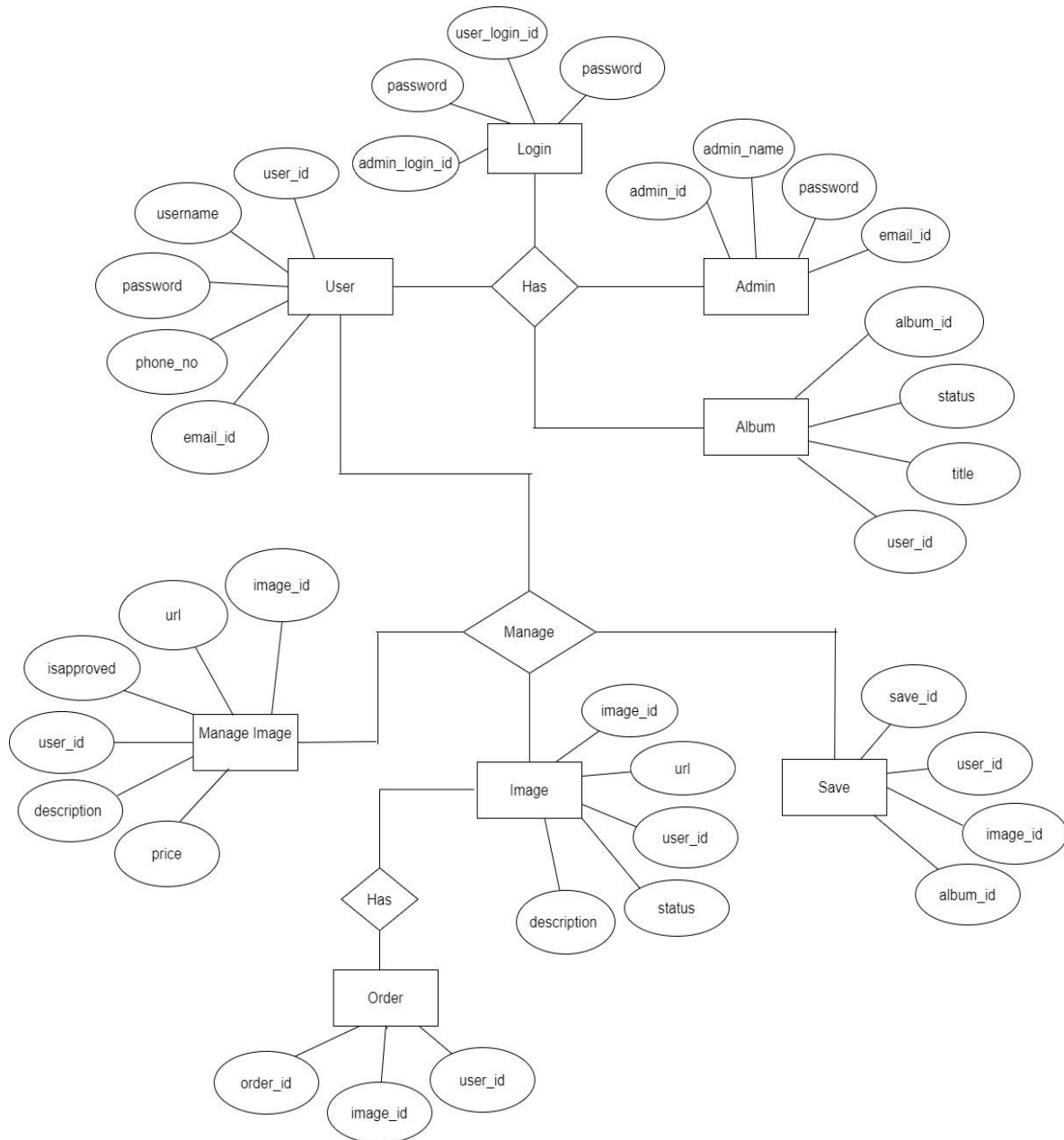
3.2.2 Technical Feasibility

- This technical feasibility study of a system determines whether the technology needed for the proposed system is available and whether this technology can be integrated into the organization. Technical evolution must also access whether the existing system can be upgraded to use the new technology and whether the organization. Technical evolution must also access whether the organization has the expertise to use it. The organization is already well-equipped with the required hardware and software.

3.2.3 Behavioral Feasibility

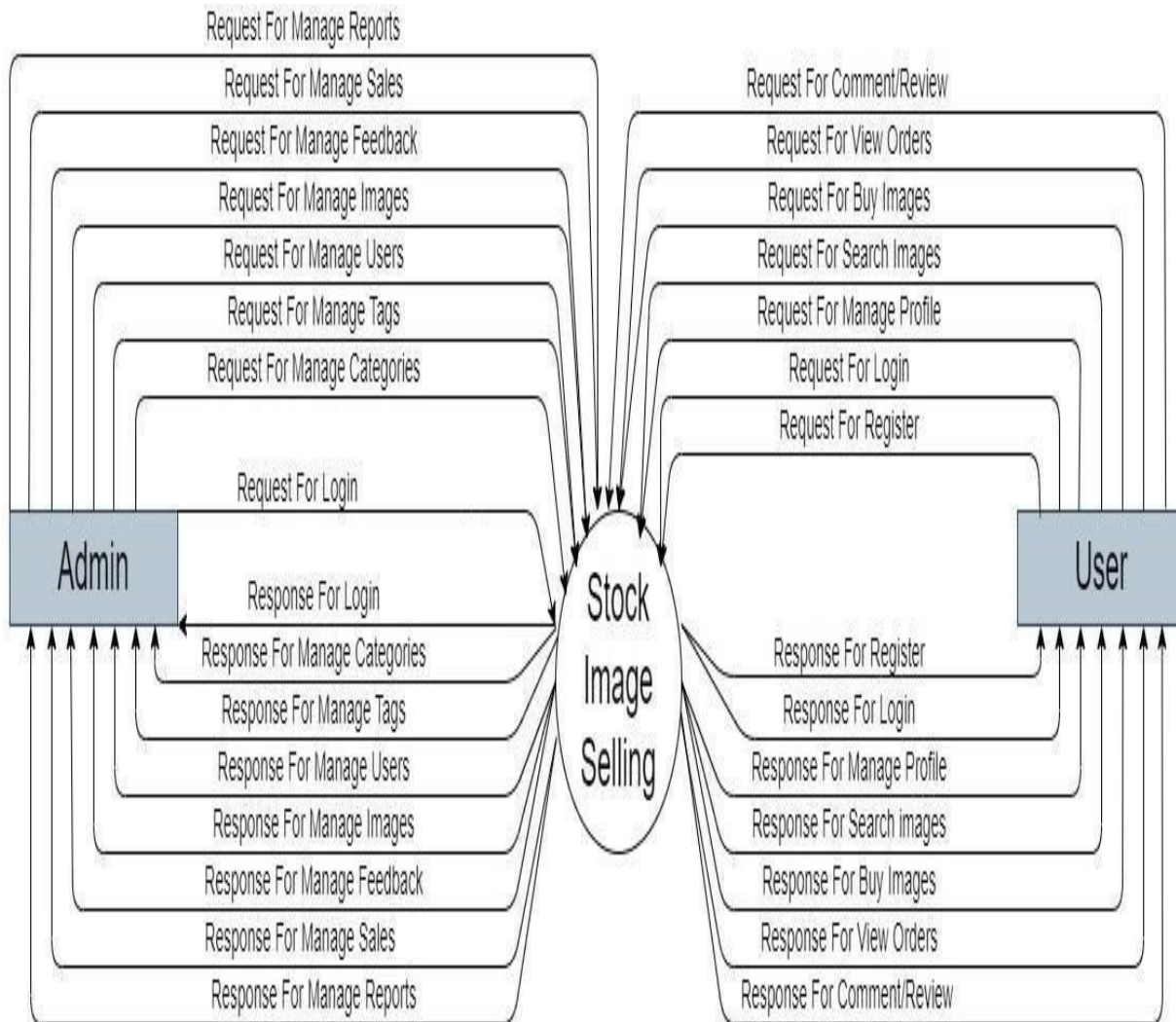
- The application requires no special technical guidance and all the views available in the application are self-explanatory. The users are well guided with warning and failure messages for all the actions taken.

3.3 E-R Diagram

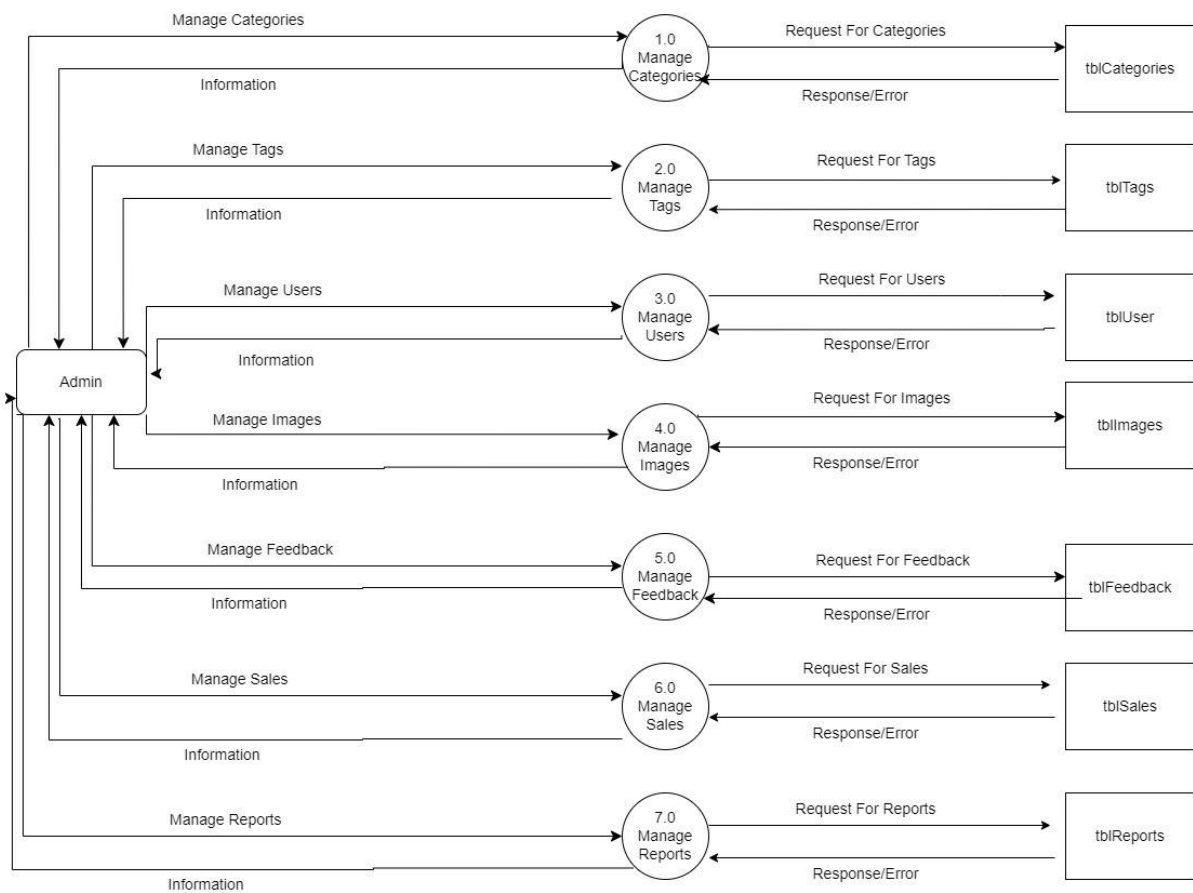


3.4 Data Flow Diagram

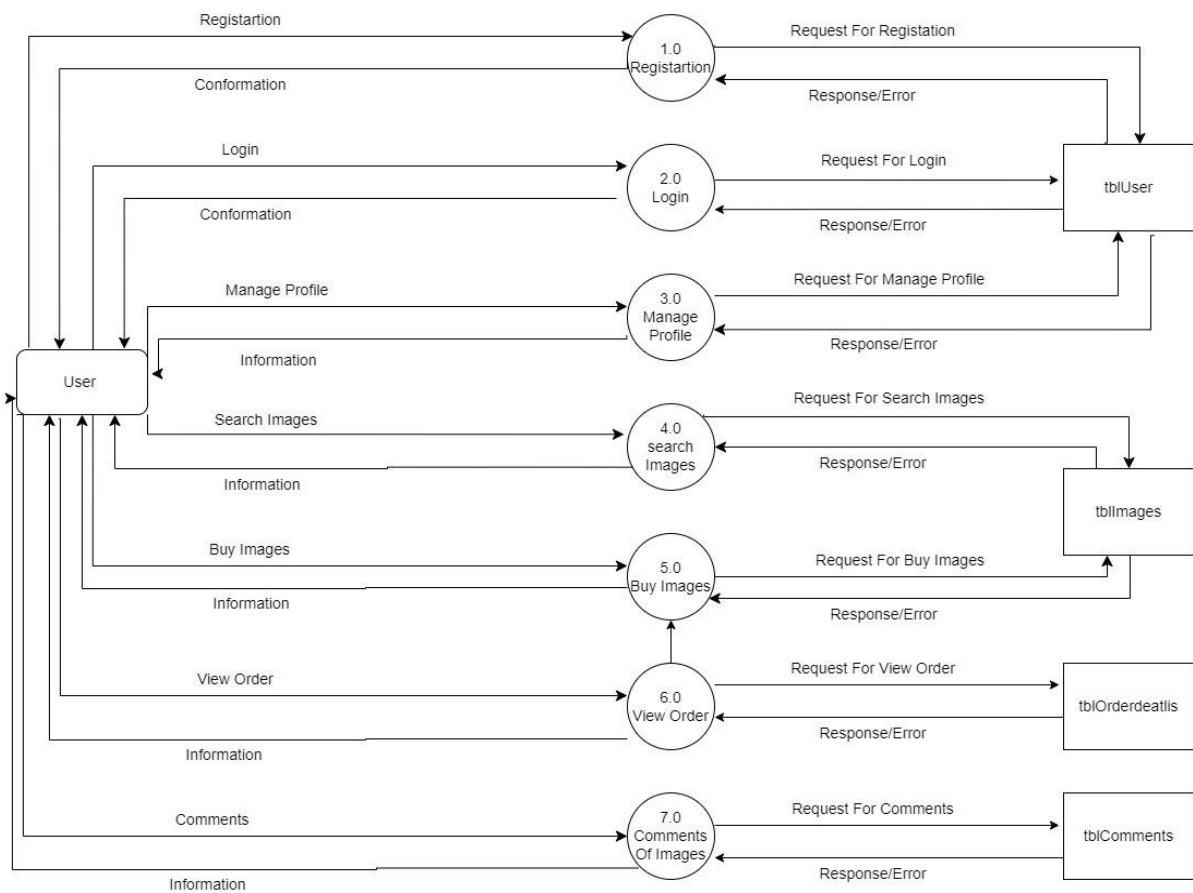
❖ 0 Level DFD:



❖ 1st Level DFD For Admin:



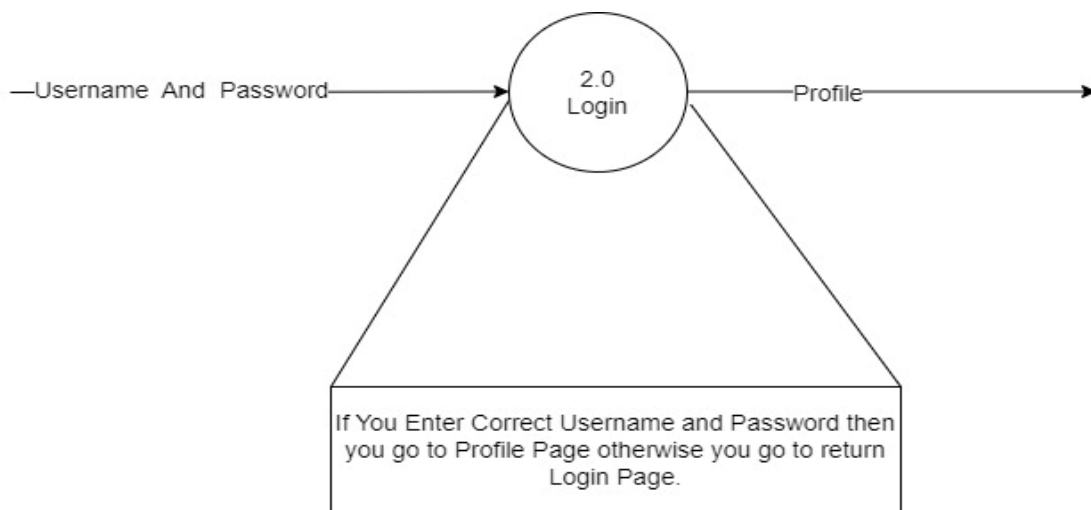
• 1st Level DFD For User:



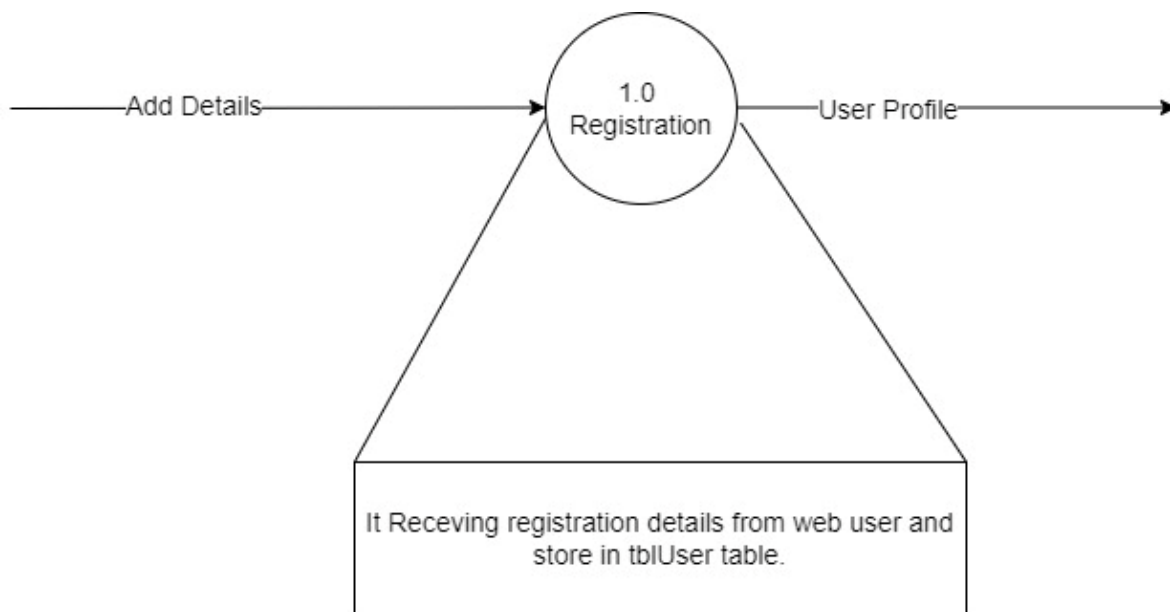
❖ Process

Specification:

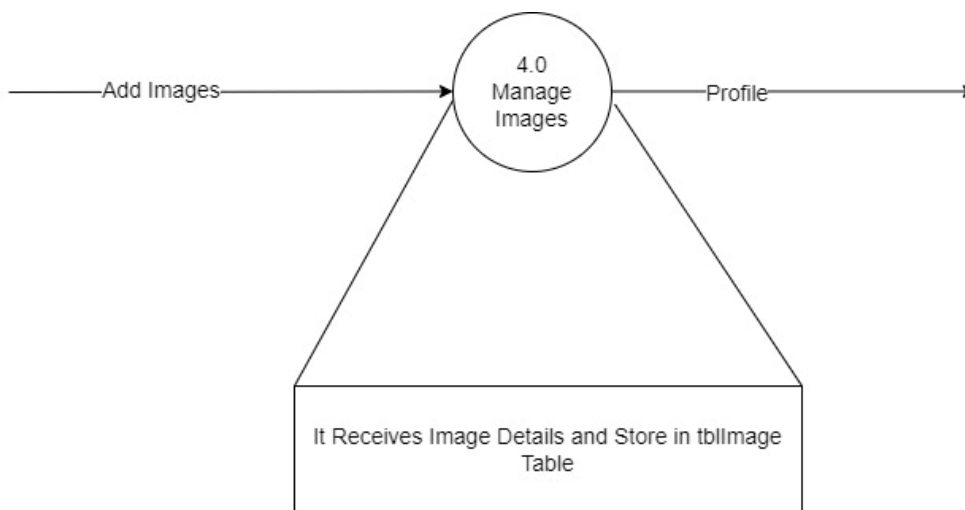
1. Login:



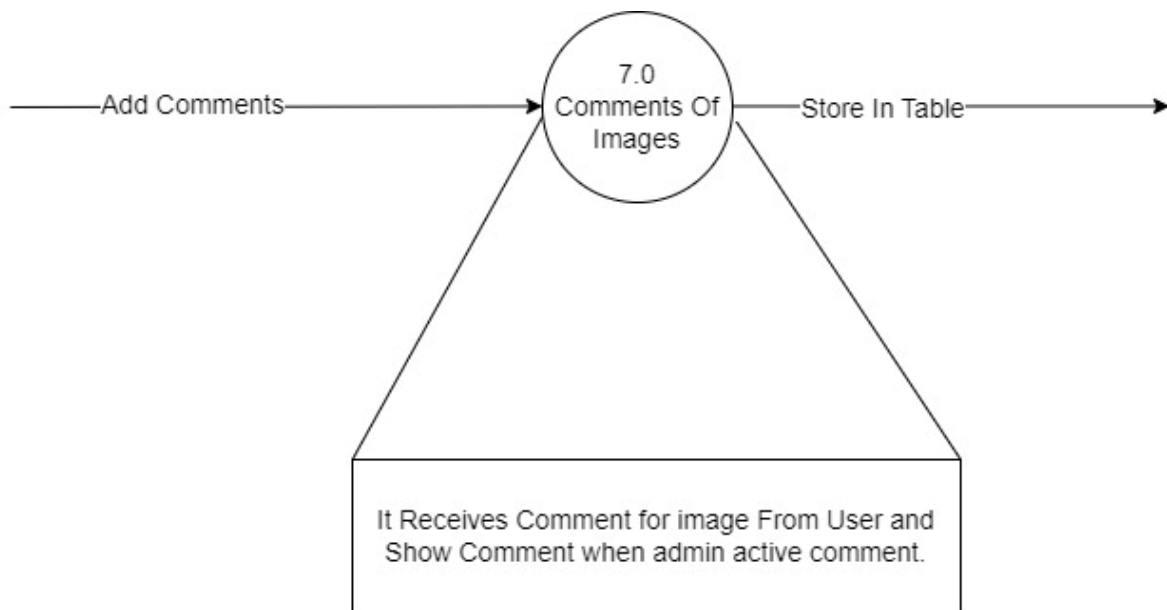
2. Registration:



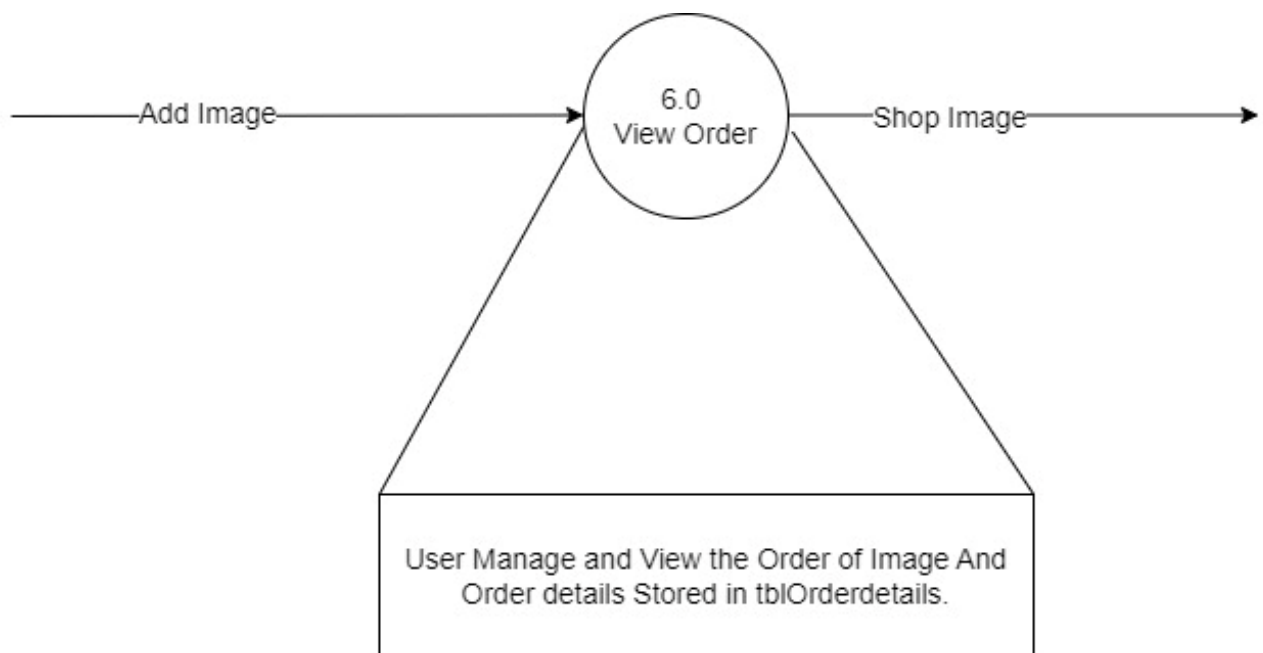
3. Manage Image:



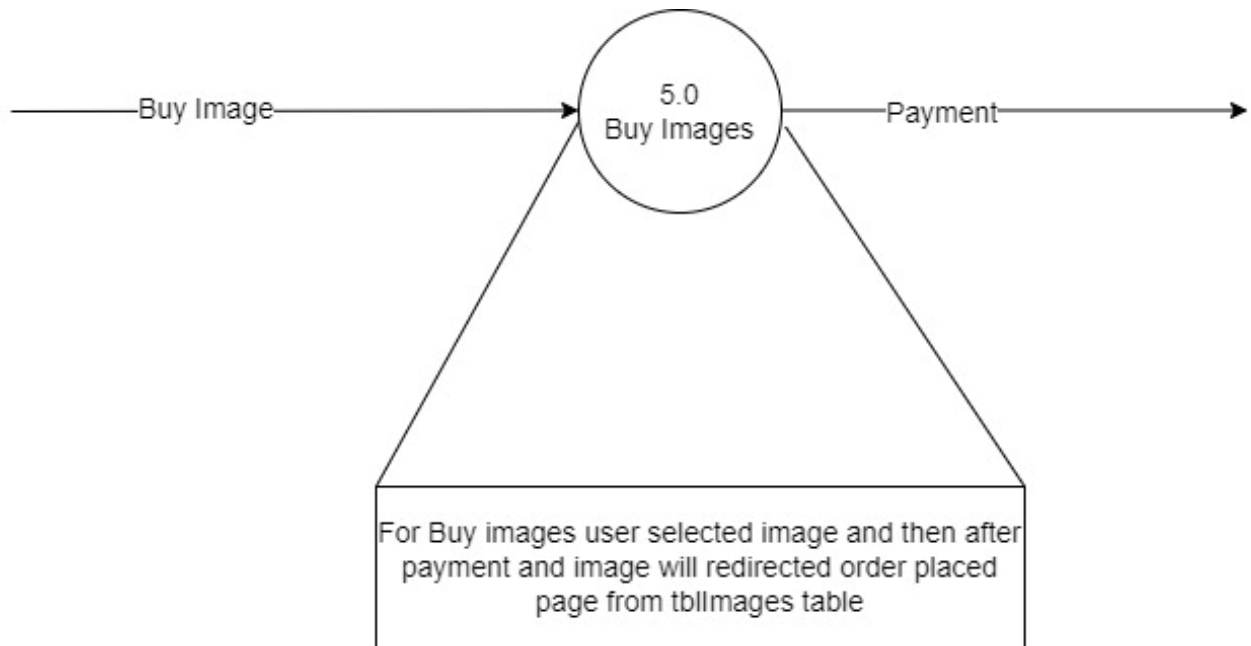
4. Comments of Images:



5.View Order:



6. Buy Images:



4. Software Project Planning

4.1 Scope

A stock image selling website is a platform that allows photographers and artists to upload and sell their digital images to customers who need them for various purposes, such as advertising, web design, editorial content, and more. The scope of a stock image selling website includes the following:

1) User interface:

The website should have an easy-to-use interface that allows users to search for and purchase images quickly and efficiently.

2) Image submission:

The website should allow photographers and artists to upload their images easily and efficiently. This includes features such as image resizing, categorization, and metadata.

3) Image management:

The website should have a robust image management system that allows photographers and artists to manage their image portfolios, track sales, and view analytics.

4) Licensing and pricing:

The website should offer a range of licensing options, such as royalty-free and rights-managed licenses, and pricing options that are competitive with other stock image selling websites.

5) Customer support:

The website should have a customer support system in place to help customers with any issues they may encounter when purchasing and using images.

6) Payment processing:

The website should have a secure payment processing system in place that allows customers to purchase images with ease and confidence.

7) Marketing and promotion:

The website should have a marketing and promotion strategy in place to attract both photographers and artists as well as customers looking for high-quality images.

4.2 Objectives

The objectives of a stock image selling website may include:

1) Providing a platform for photographers and digital artists to sell their work:

A stock image selling website allows photographers and digital artists to showcase and sell their images to a wider audience. This can provide a valuable source of income for creatives, especially those who may not have the resources to market their work independently.

2) Meeting the needs of buyers:

Stock image selling websites typically cater to a diverse range of buyers, including businesses, publishers, marketers, and designers. These buyers require high-quality images that meet their specific needs, whether it's for a website, social media post, or marketing campaign.

3) Offering a wide variety of images:

Stock image selling websites aim to provide a diverse range of images that meet the needs of buyers. This may include images of people, nature, objects, and landscapes, as well as illustrations and vectors.

4) Streamlining the buying process:

Stock image selling websites typically provide an easy-to-use platform that streamlines the buying process. This may include features such as search filters, bulk purchasing options, and instant downloads.

5) Ensuring fair compensation for photographers and artists:

Stock image selling websites typically operate on a revenue-sharing model, which ensures that photographers and artists receive a fair share of the profits generated from their work. This can incentivize creatives to continue producing high-quality images for sale on the platform.

4.3 Expected Advantages

1. Stock Photos Provide a Lot of Choices

Using a stock image source like Designer gives you access to many photographs, regardless of what you are looking for. Since most of the treasure is in one location, whether you are searching for a single picture or a collection of stock images, image search engines and algorithms make it simple to explore the content until you locate the perfect shot.

2. Give You the Right to Use Intellectual Property

Purchasing stock photography ensures that you have a valid license to use the pictures in any manner you desire. The subscription packages make choosing the valid credentials for your photograph simple.

3. Save Time

If you want a picture immediately and do not have the time to hire and engage a photographer, stock photographs are convenient. Purchasing an image is a quick and easy process. The best part about buying an image online is that you can immediately download it, making it available to be used the moment you pay.

4. High-Resolution Pictures at Your Fingertips

All photographs are subjected to technical inspections to ensure that they match our clients' high expectations for image quality. This implies you will never be out of high-quality photographs because there's lots of high-resolution photography at your disposal. Different file sizes are also available to ensure that the photos you see meet your requirements.

5. Before you purchase, you can test the photograph

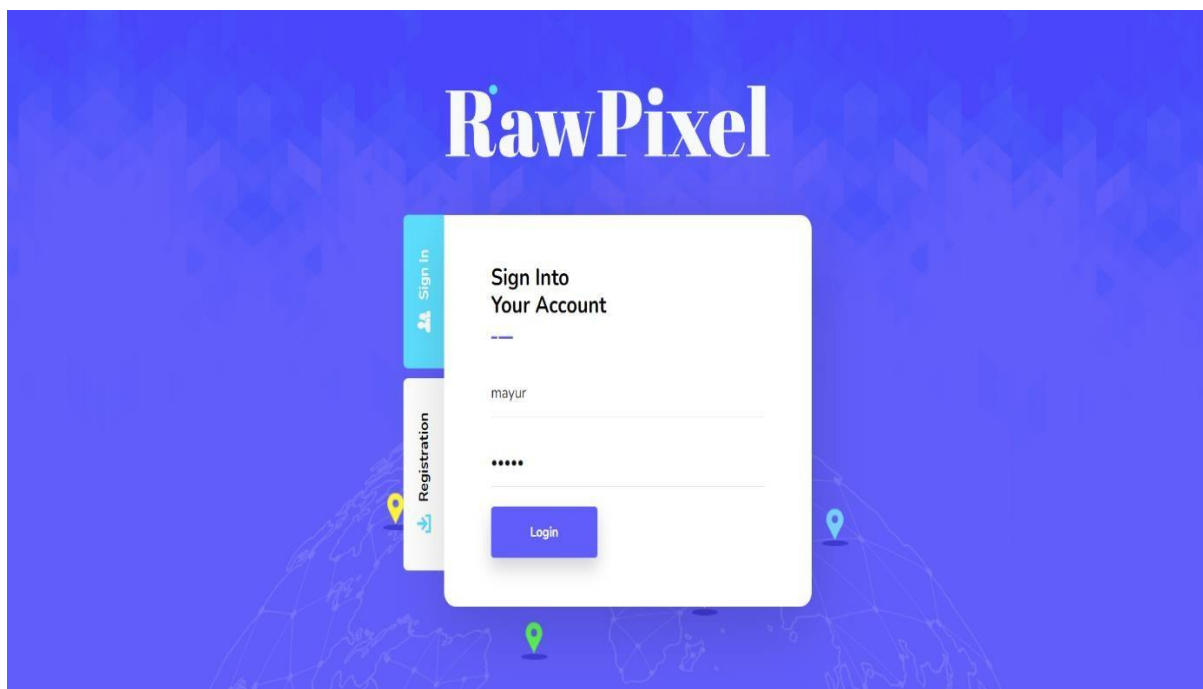
Before purchasing a picture, you can sample it to test whether it will work with your project. This would save you time and expense because you can preview various photos to evaluate how they function in your design and spend on the one that fits best.

5. System Design

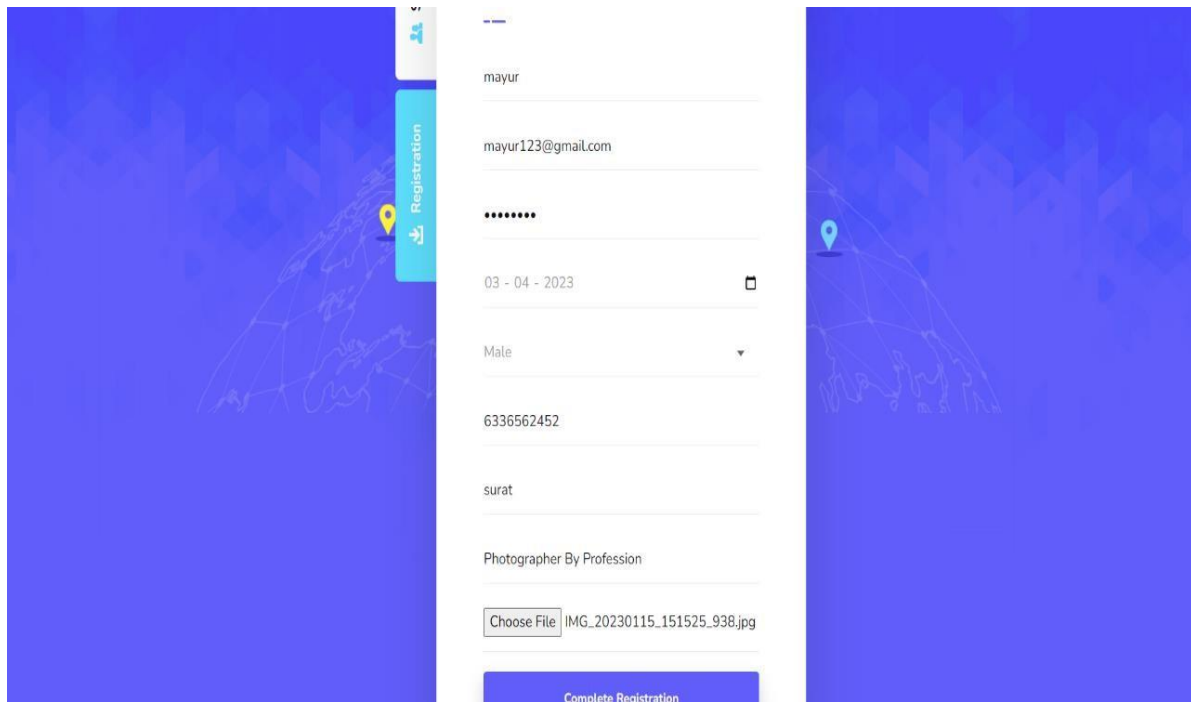
5.1 Input Design

❖ User Side:

- Login Page:



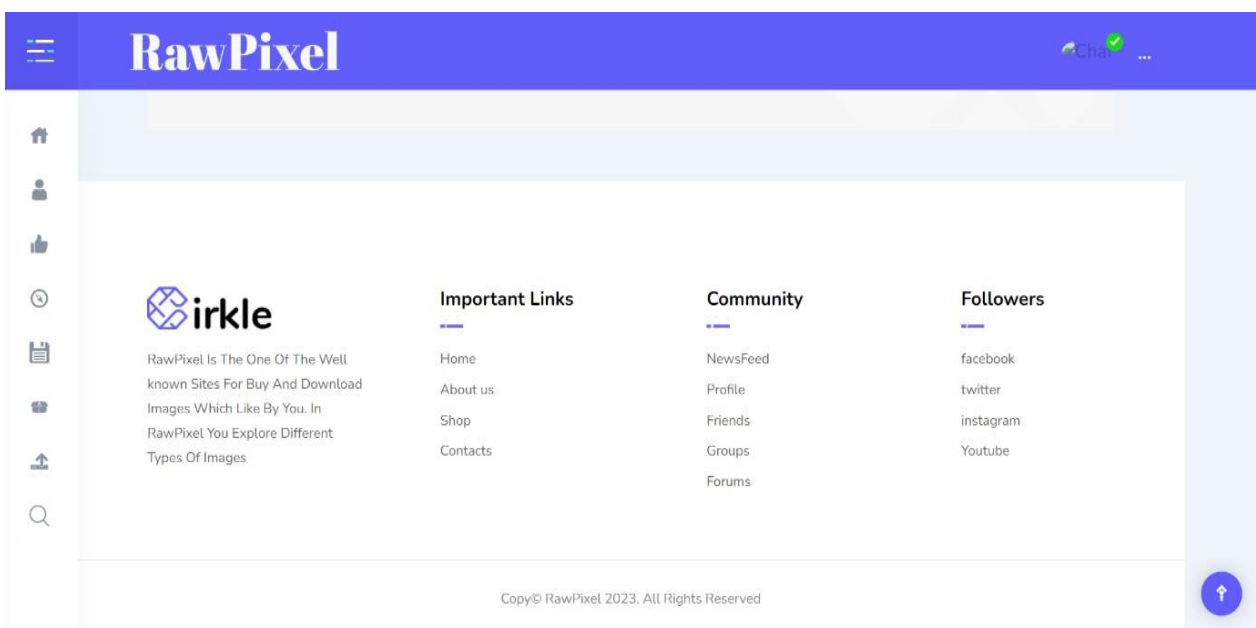
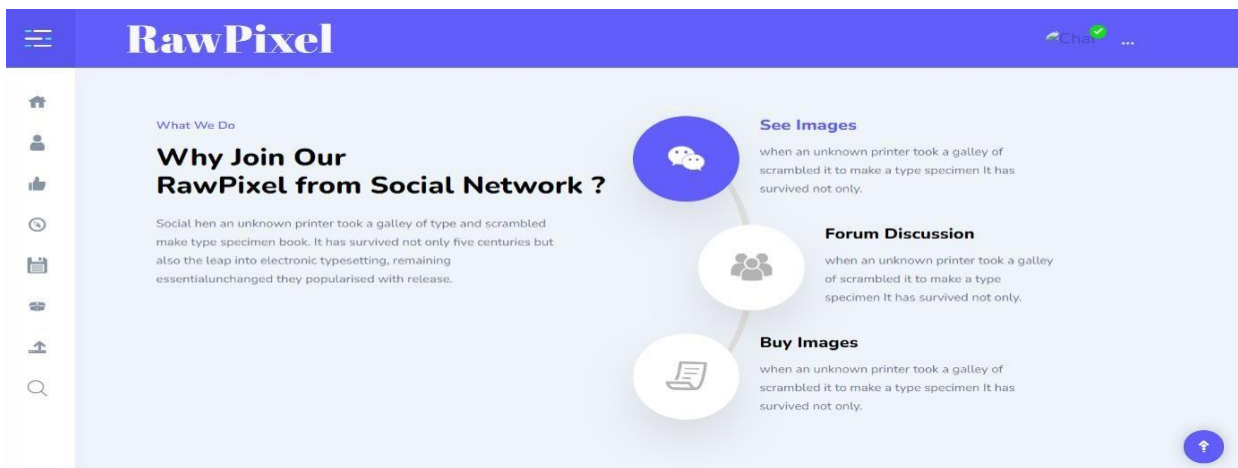
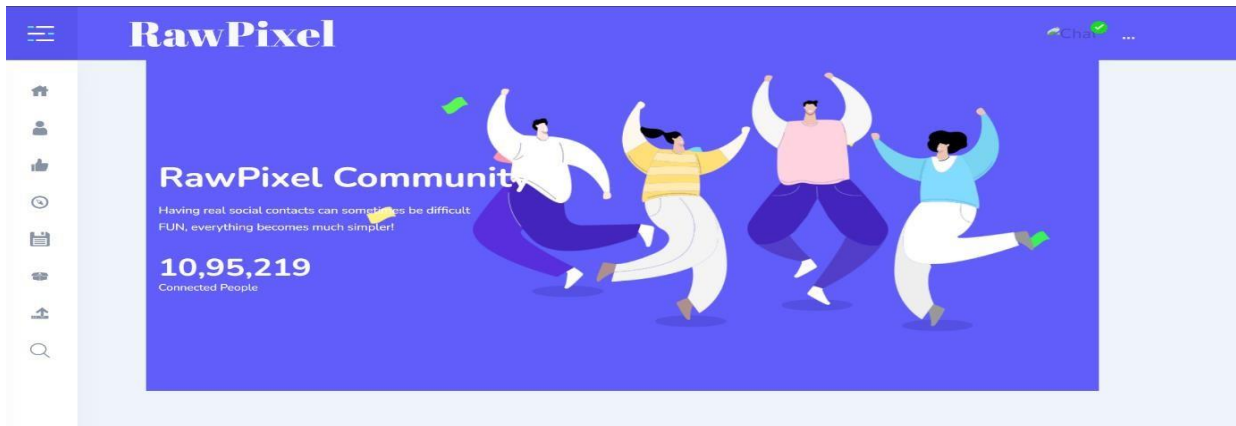
- **Registration Page:**



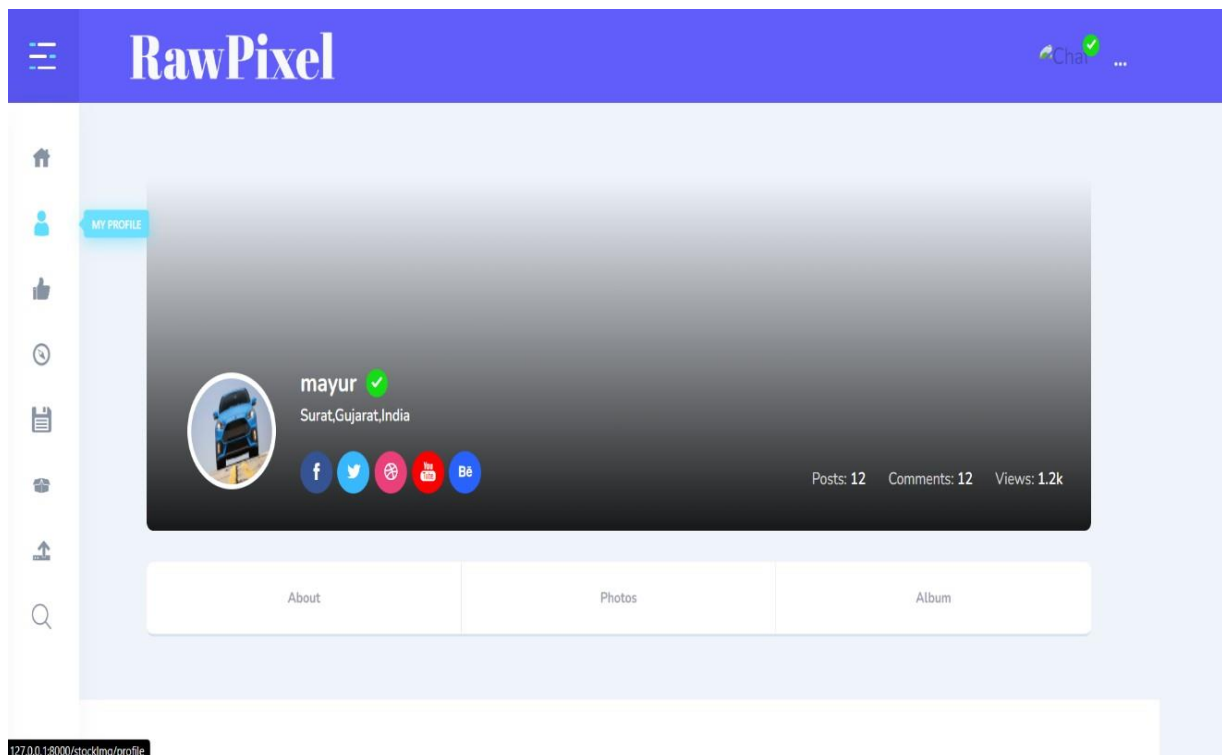
The registration form is displayed on a blue background with a faint map overlay. The form fields are as follows:

- Username:** mayur
- Email:** mayur123@gmail.com
- Password:** (masked with dots)
- Date of Birth:** 03 - 04 - 2023
- Gender:** Male
- Phone Number:** 6336562452
- City:** surat
- Profession:** Photographer By Profession
- Profile Picture:** Choose File IMG_20230115_151525_938.jpg
- Complete Registration:** (button)

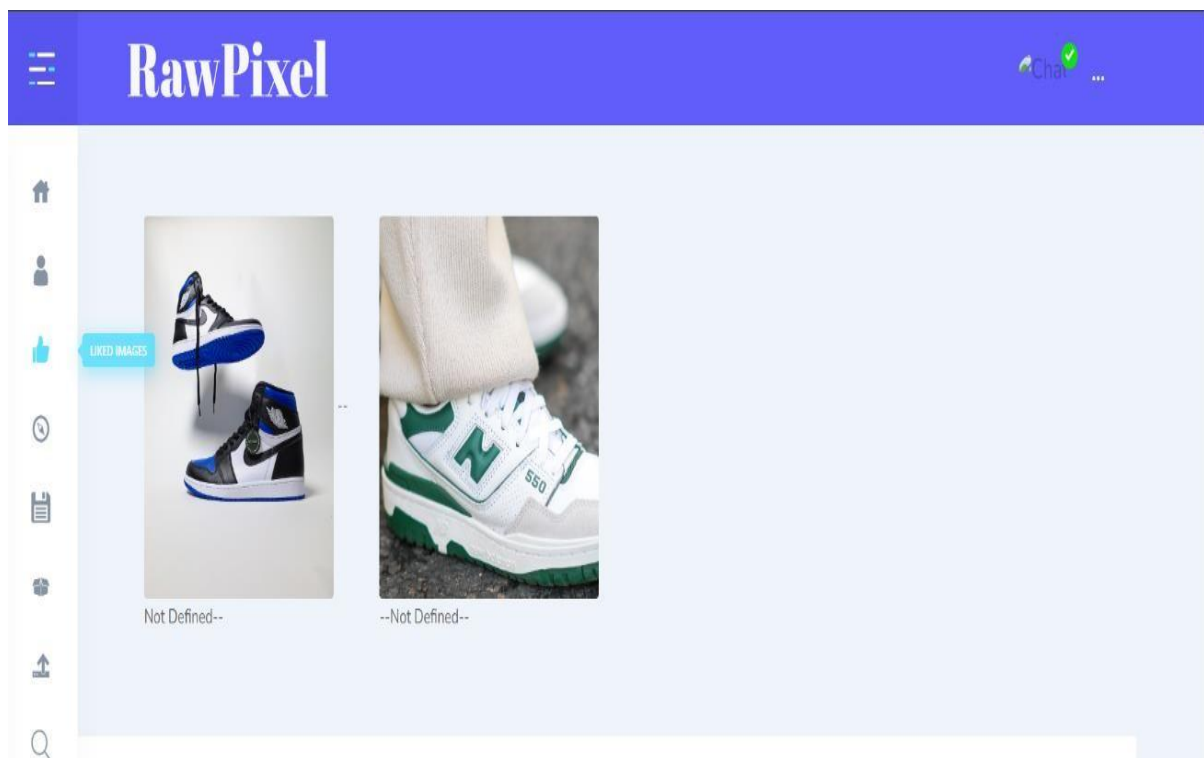
• Home Page:



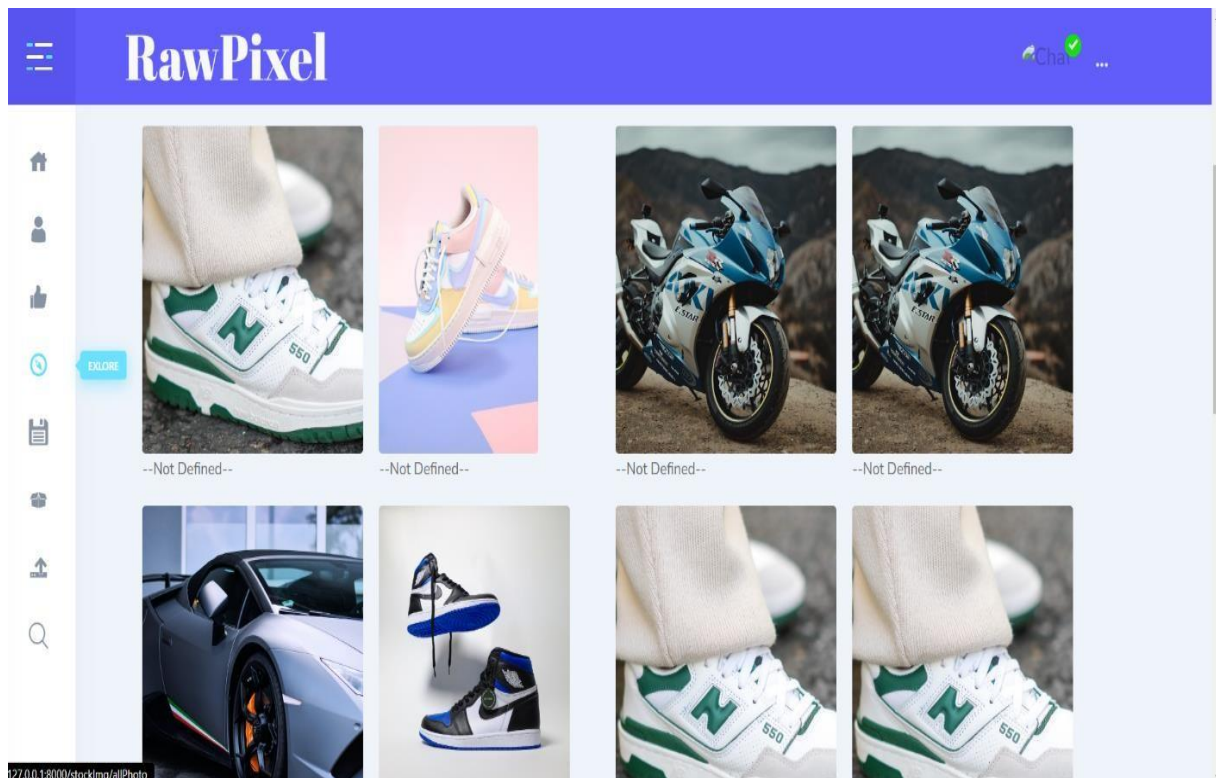
- **My Profile:**



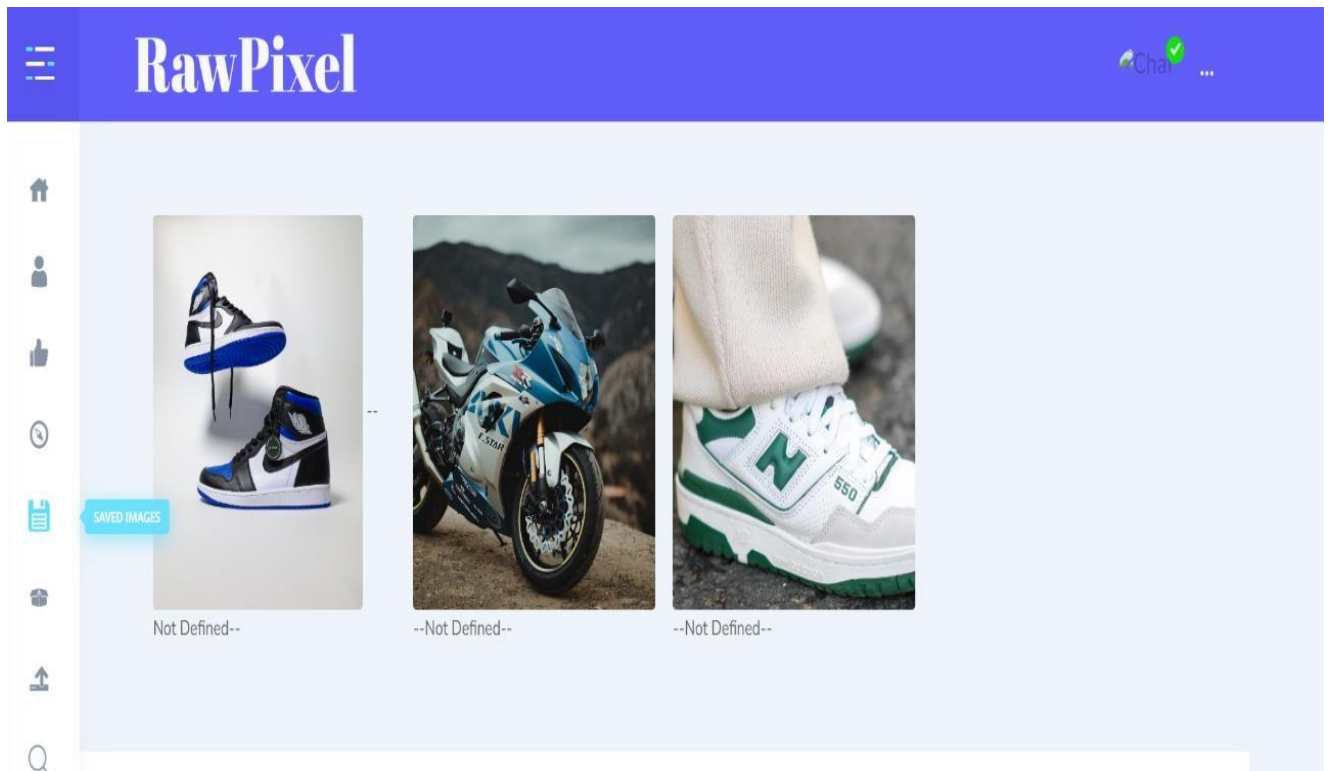
- **Liked Images:**



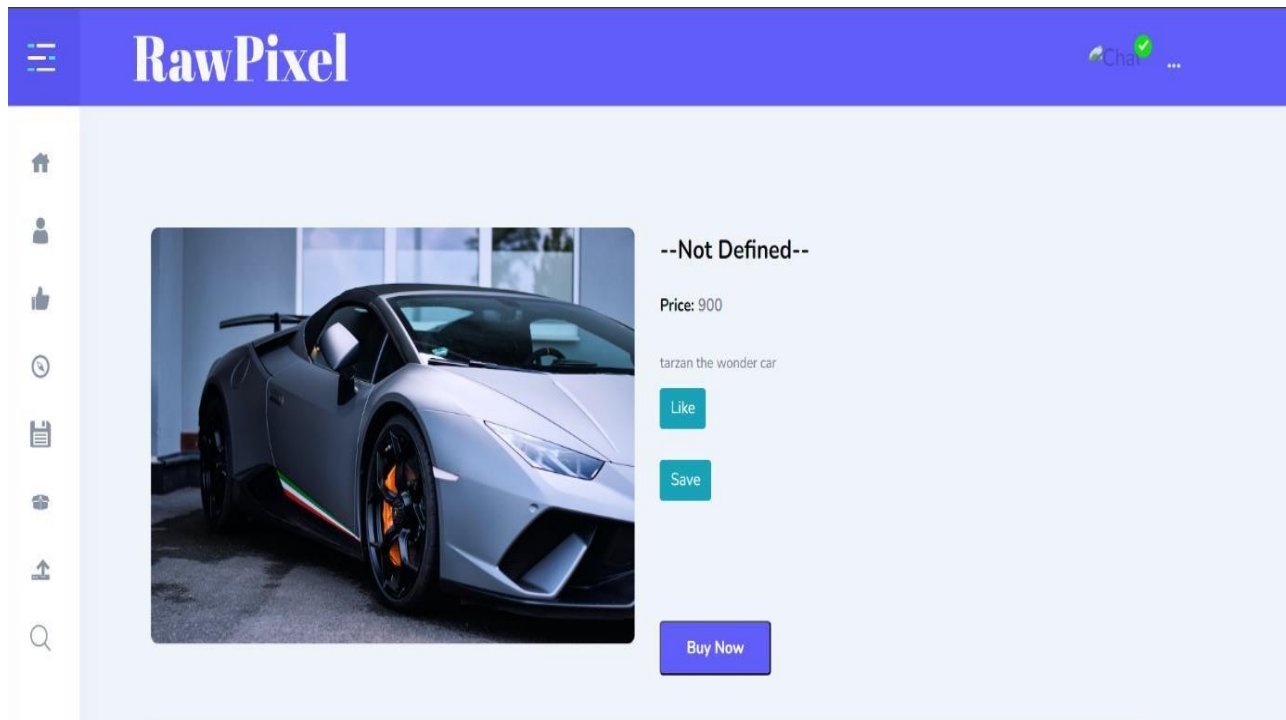
- **Explore Page:**



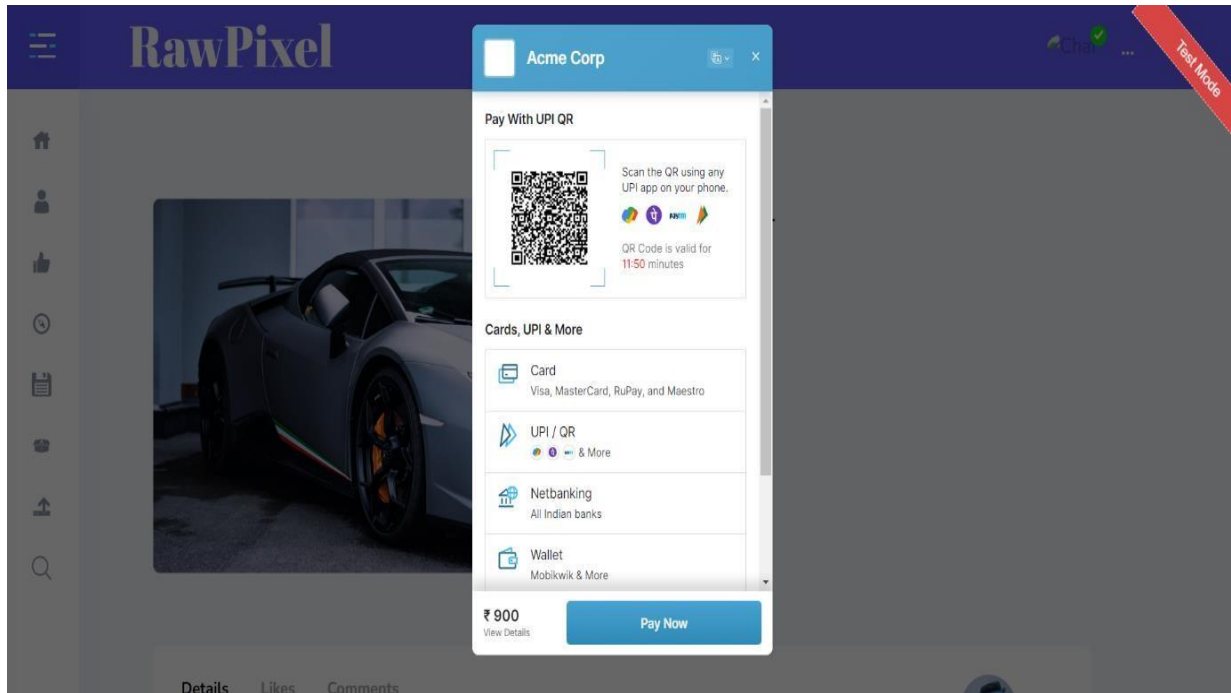
- **Saved Images:**



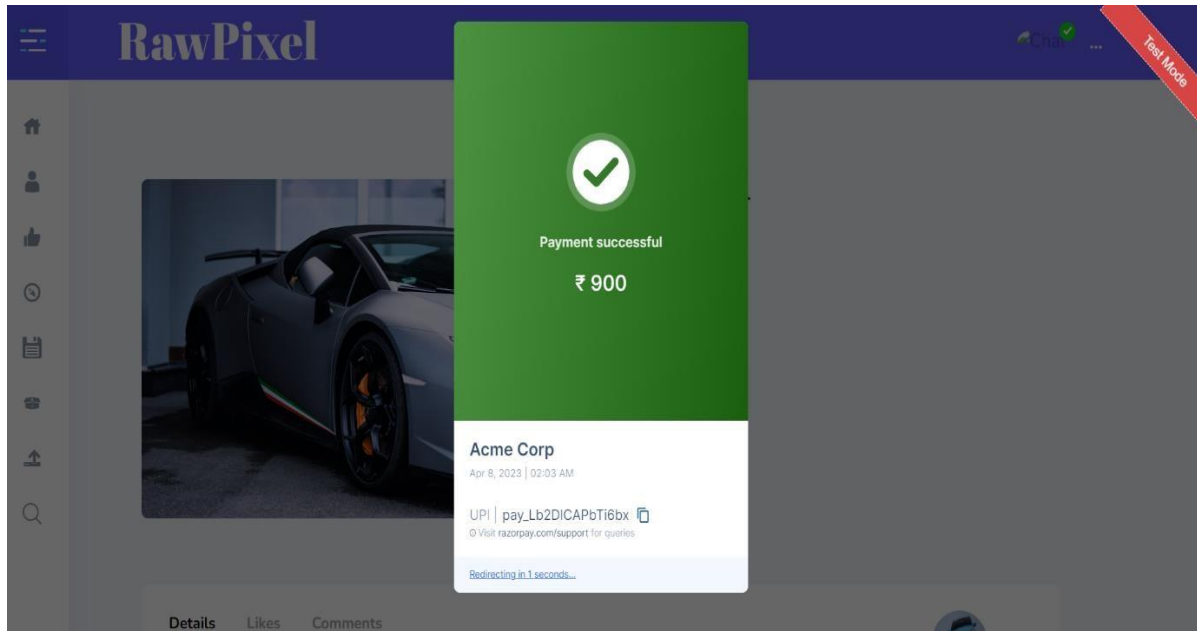
- **Buy Images:**



- **Payment Gateway:**



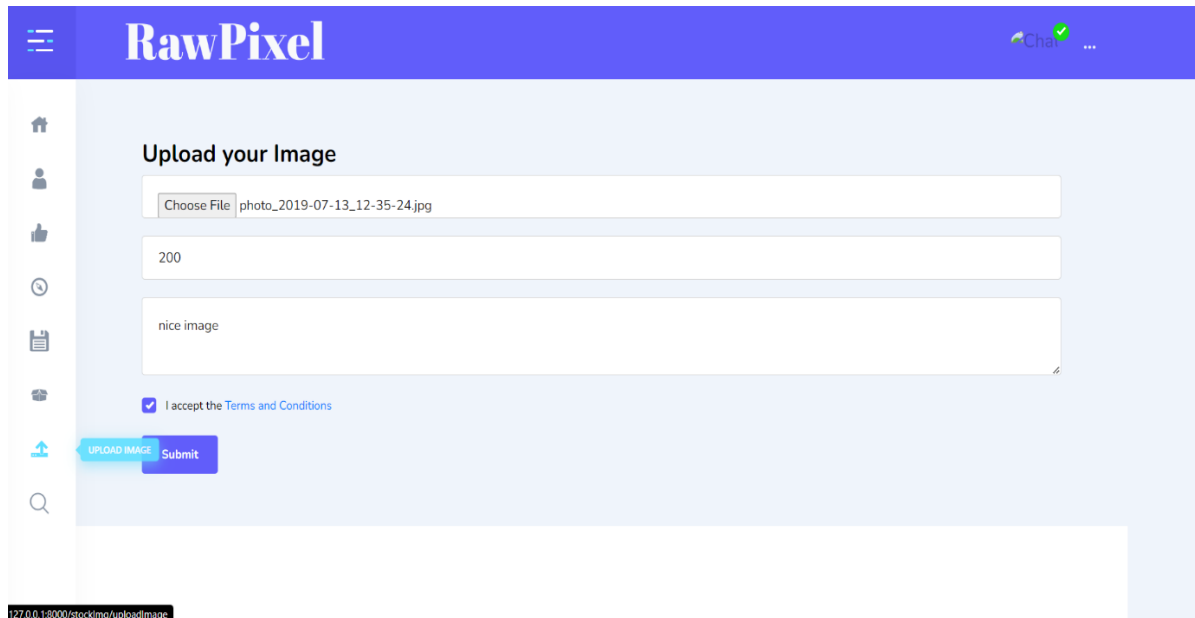
- **Payment Successful:**



- **Order Placed:**

RawPixel			
      	13	April 7, 2023, 12:21 p.m.	900
			
	14	April 7, 2023, 7:33 p.m.	200
			
	15	April 7, 2023, 8:33 p.m.	900
			

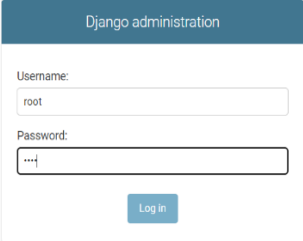
- **Upload Image:**



The screenshot shows the 'RawPixel' website interface for uploading an image. The header is blue with the 'RawPixel' logo and a 'Chat' button. A left sidebar contains navigation icons. The main content area is titled 'Upload your Image' and includes a file selection field (showing 'photo_2019-07-13_12-35-24.jpg'), a price input field (containing '200'), and a description text area (containing 'nice image'). Below these fields is a checked checkbox for 'I accept the Terms and Conditions'. At the bottom of the form are two buttons: 'UPLOAD IMAGE' and 'Submit'. A small status bar at the bottom left of the browser window displays the URL '127.0.0.1:8000/stockimg/uploadImage'.

Admin Side:

- **Login Page:**



The image shows a screenshot of the Django administration login page. The page has a light gray background. In the center, there is a white rectangular box with a dark blue header that says "Django administration". Below the header, there are two input fields: "Username:" with the text "root" entered, and "Password:" with three asterisks "***" entered. Below these fields is a blue button labeled "Log in".

• Site Administration:

Django administration
WELCOME, **root**. [VIEW SITE](#) / [CHANGE PASSWORD](#) / [LOG OUT](#)

Site administration

AUTHENTICATION AND AUTHORIZATION

Groups	+ Add	Change
Users	+ Add	Change

STOCKING

Albumimages	+ Add	Change
Albums	+ Add	Change
Catagorys	+ Add	Change
Comments	+ Add	Change
Countrys	+ Add	Change
Imgs	+ Add	Change
Likes	+ Add	Change
Orders	+ Add	Change
Saves	+ Add	Change
States	+ Add	Change
Tagss	+ Add	Change
Ucitys	+ Add	Change
Users	+ Add	Change

Recent actions

My actions

- [amaan](#)
User
- [+ wertyul](#)
Albumimage
- [+ mayur/11](#)
Img
- [+ wertyul](#)
Albumimage
- [+ mayur/10](#)
Img
- [+ wertyul](#)
Albumimage
- [+ mayur/9](#)
Img
- [✗ mayur](#)
User
- [✗ mayur](#)
User
- [+ wertyul](#)
Albumimage

- **Change User:**

Django administration

WELCOME, [ROOT](#) / [VIEW SITE](#) / [CHANGE PASSWORD](#) / [LOG OUT](#)

Home / Stockimg / Users

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

Groups [+ Add](#)

Users [+ Add](#)

STOCKIMG

Albumimages [+ Add](#)

Albums [+ Add](#)

Catagorys [+ Add](#)

Comments [+ Add](#)

Countrys [+ Add](#)

Imgs [+ Add](#)

Likes [+ Add](#)

Orders [+ Add](#)

Saves [+ Add](#)

States [+ Add](#)

Tagss [+ Add](#)

Ucitys [+ Add](#)

Users [+ Add](#)

Select user to change

ADD USER +

Action: Go 0 of 7 selected

☐ USER

☐ ayaan

☐ purva

☐ devi

☐ mayur

☐ Harsh

☐ Harsh

☐ amaan

7 users

- **City :**

Django administration WELCOME, root. VIEW SITE / CHANGE PASSWORD / LOG OUT

Home > Stocking > Uclys

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

- Groups [+ Add](#)
- Users [+ Add](#)

STOCKING

- Albumimages [+ Add](#)
- Albums [+ Add](#)
- Catagorys [+ Add](#)
- Comments [+ Add](#)
- Countrys [+ Add](#)
- Imgs [+ Add](#)
- Likes [+ Add](#)
- Orders [+ Add](#)
- Saves [+ Add](#)
- States [+ Add](#)
- Tagss [+ Add](#)
- Uclys [+ Add](#)**
- Users [+ Add](#)

Select ucly to change [ADD UCly +](#)

Action: 0 of 2 selected

- ☐ ucly
- ☐ mumbai
- ☐ Surat

2 uclys

- **States:**

Django administration

WELCOME, ROOT / VIEW SITE / CHANGE PASSWORD / LOG OUT

Home / Stocking / States

Start typing to filter...

STOCKING

- Albumimages + Add
- Albums + Add
- Categories + Add
- Comments + Add
- Countries + Add
- Imgs + Add
- Likes + Add
- Orders + Add
- Saves + Add
- States + Add**
- Tags + Add
- Uchlys + Add
- Users + Add

AUTHENTICATION AND AUTHORIZATION

- Groups + Add
- Users + Add

Select state to change

ADD STATE +

Action: Go 0 of 3 selected

STATE
<input type="checkbox"/> New York
<input type="checkbox"/> Maharashtra
<input type="checkbox"/> Gujarat

3 states

- **Save Image:**

Django administration WELCOME. root. VIEW SITE / CHANGE PASSWORD / LOG OUT

Home > Stocking > Saves

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

- Groups [+ Add](#)
- Users [+ Add](#)

STOCKING

- Albumimages [+ Add](#)
- Albums [+ Add](#)
- Categories [+ Add](#)
- Comments [+ Add](#)
- Countries [+ Add](#)
- Imgs [+ Add](#)
- Likes [+ Add](#)
- Orders [+ Add](#)
- Saves [+ Add](#)**
- States [+ Add](#)
- Tags [+ Add](#)
- Uctys [+ Add](#)
- Users [+ Add](#)

Select save to change ADD SAVE +

Action: 0 of 4 selected

- ☐ SAVE
- ☐ amaan
- ☐ mayur
- ☐ mayur
- ☐ mayur

4 saves

- **Category:**

Django administration WELCOME, ROOT / VIEW SITE / CHANGE PASSWORD / LOG OUT

Home / Stocking / Catagorys

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

- Groups [+ Add](#)
- Users [+ Add](#)

STOCKING

- Albumimages [+ Add](#)
- Albums [+ Add](#)
- Catagorys [+ Add](#)**
- Comments [+ Add](#)
- Countrys [+ Add](#)
- Imgs [+ Add](#)
- Likes [+ Add](#)
- Orders [+ Add](#)
- Saves [+ Add](#)
- States [+ Add](#)
- Tags [+ Add](#)
- Ucitys [+ Add](#)
- Users [+ Add](#)

Select catagory to change ADD CATAGORY +

Action: 0 of 7 selected

- ☐ CATAGORY
- ☐ Mobile
- ☐ Shoes
- ☐ Bike
- ☐ Car
- ☐ demo3
- ☐ demo2
- ☐ demo1

7 catagorys

- **Orders:**

The screenshot displays the Django administration interface for a stock image selling application. The left sidebar shows the navigation menu with the following sections:

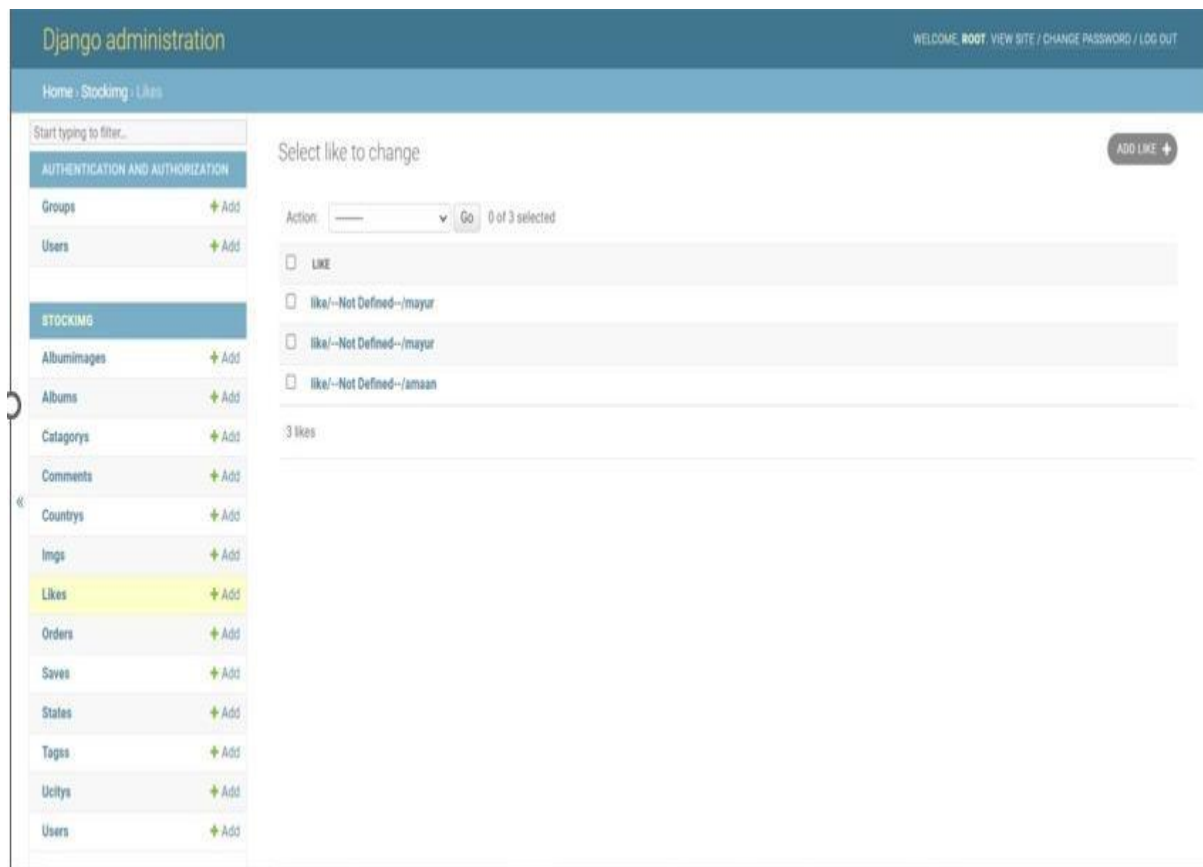
- Authentication and Authorization:** Groups, Users.
- Stocking:** Albumimages, Albums, Categorys, Comments, Countrys, Imgs, Likes, **Orders** (highlighted), Saves, States, Tags, Ucitys, Users.

The main content area is titled "Select order to change" and shows a list of 13 orders. Each order is represented by a row with a checkbox and a number. The "ADD ORDER" button is located in the top right corner of the main content area.

Order ID	Action
ORDER	<input type="checkbox"/>
13	<input type="checkbox"/>
12	<input type="checkbox"/>
11	<input type="checkbox"/>
10	<input type="checkbox"/>
9	<input type="checkbox"/>
8	<input type="checkbox"/>
7	<input type="checkbox"/>
6	<input type="checkbox"/>
5	<input type="checkbox"/>
4	<input type="checkbox"/>
3	<input type="checkbox"/>
2	<input type="checkbox"/>
1	<input type="checkbox"/>

13 orders

- **Likes:**



- Images:

Django administration WELCOME: root. VIEW SITE / CHANGE PASSWORD / LOG OUT

Home » Stockimg » imgs

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

- Groups [+ Add](#)
- Users [+ Add](#)

STOCKIMG

- Albumimages [+ Add](#)
- Albums [+ Add](#)
- Catagorys [+ Add](#)
- Comments [+ Add](#)
- Countrys [+ Add](#)
- « **imgs** [+ Add](#)
- Likes [+ Add](#)
- Orders [+ Add](#)
- Saves [+ Add](#)
- States [+ Add](#)
- Tagss [+ Add](#)
- Ucitys [+ Add](#)
- Users [+ Add](#)

Select img to change ADD IMG +

Action: Go 0 of 14 selected

<input type="checkbox"/>	img
<input type="checkbox"/>	John/19
<input type="checkbox"/>	John/18
<input type="checkbox"/>	amaan/17
<input type="checkbox"/>	mayur/16
<input type="checkbox"/>	mayur/15
<input type="checkbox"/>	mayur/14
<input type="checkbox"/>	mayur/13
<input type="checkbox"/>	mayur/12
<input type="checkbox"/>	mayur/11
<input type="checkbox"/>	mayur/10
<input type="checkbox"/>	mayur/9
<input type="checkbox"/>	amaan/6
<input type="checkbox"/>	mayur/5
<input type="checkbox"/>	mayur/1

- **Country:**

Django administration WELCOME, ROOT. [VIEW SITE](#) / [CHANGE PASSWORD](#) / [LOG OUT](#)

Home > Stocking > Countrys

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

Groups [+ Add](#)

Users [+ Add](#)

STOCKING

Albumimages [+ Add](#)

Albums [+ Add](#)

Catagorys [+ Add](#)

Comments [+ Add](#)

« Countrys [+ Add](#)

Imgs [+ Add](#)

Likes [+ Add](#)

Orders [+ Add](#)

Saves [+ Add](#)

States [+ Add](#)

Tags [+ Add](#)

Ucitys [+ Add](#)

Users [+ Add](#)

Select country to change ADD COUNTRY +

Action: Go 0 of 4 selected

☐ COUNTRY

☐ india

☐ China

☐ USA

☐ India

4 countrys

- **Album:**

Django administration WELCOME ROOT. VIEW SITE / CHANGE PASSWORD / LOG OUT

Home » Stocking » Albums

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

Groups [+ Add](#)

Users [+ Add](#)

STOCKING

Albumimages [+ Add](#)

Albums [+ Add](#)

Catagorys [+ Add](#)

Comments [+ Add](#)

Countryys [+ Add](#)

Imgs [+ Add](#)

Likes [+ Add](#)

Orders [+ Add](#)

Saves [+ Add](#)

States [+ Add](#)

Tagss [+ Add](#)

Ucltys [+ Add](#)

Users [+ Add](#)

Select album to change [ADD ALBUM +](#)

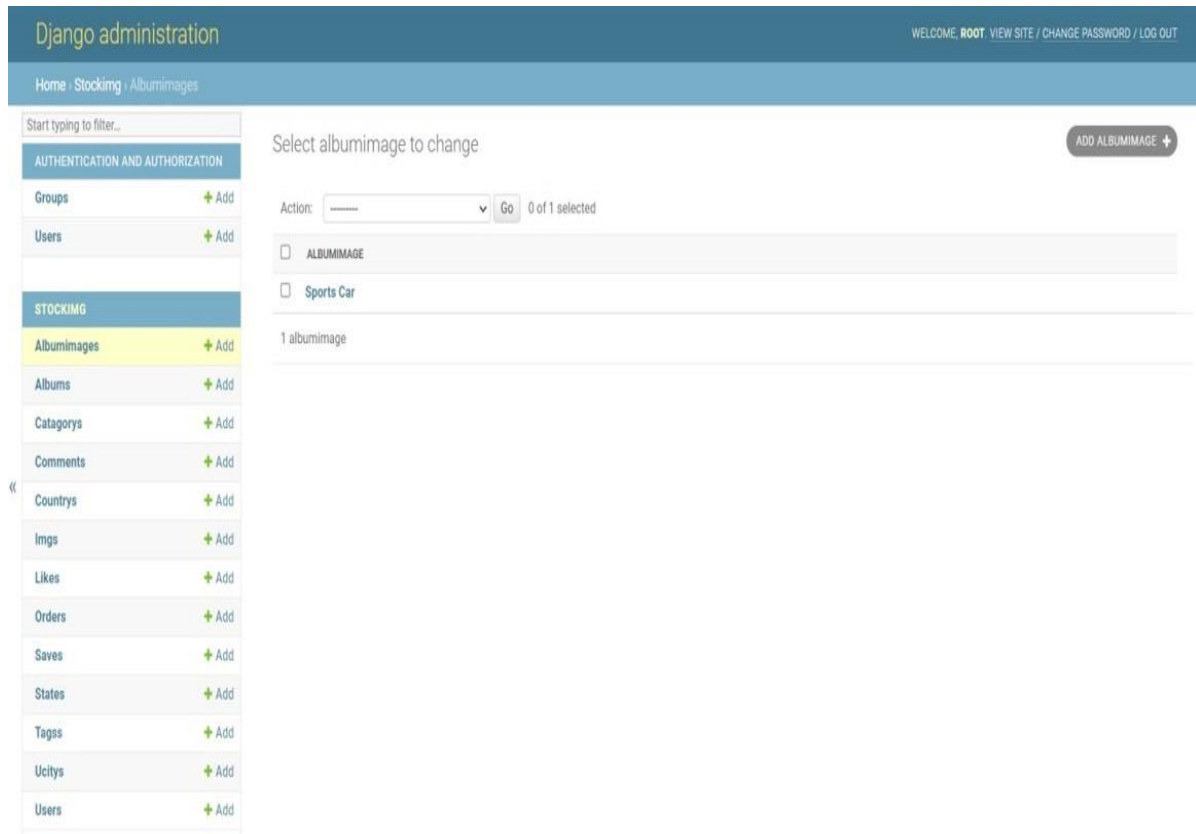
Action: 0 of 1 selected

☐ ALBUM

☐ Sports Car

1 album

- **Album Image:**



The screenshot shows the Django administration interface for the 'Album Images' model. The interface is divided into a left sidebar for navigation and a main content area for object management.

Navigation Sidebar:

- Start typing to filter...**
- AUTHENTICATION AND AUTHORIZATION**
 - Groups [+ Add](#)
 - Users [+ Add](#)
- STOCKING**
 - Albumimages** [+ Add](#)
 - Albums [+ Add](#)
 - Catagorys [+ Add](#)
 - Comments [+ Add](#)
 - Countrys [+ Add](#)
 - Imgs [+ Add](#)
 - Likes [+ Add](#)
 - Orders [+ Add](#)
 - Saves [+ Add](#)
 - States [+ Add](#)
 - Tagss [+ Add](#)
 - Ucitys [+ Add](#)
 - Users [+ Add](#)

Main Content Area:

Select albumimage to change [ADD ALBUMIMAGE +](#)

Action: 0 of 1 selected

<input type="checkbox"/>	ALBUMIMAGE
<input type="checkbox"/>	Sports Car

1 albumimage

6. Software Testing

6.1 Introduction

- Testing is the process of detecting errors. Testing performs a very critical role for quality assurance and for ensuring the reliability of software. The results of testing are used later during maintenance also.

6.2 Objectives

- Testing is a process of executing a program with the intent of finding an error.
- A good test case is one that has a high probability of finding yet undiscovered error.
- A successful test is one that uncovers yet undiscovered error.

6.3 Levels of Testing

- **Unit Testing**

Unit testing focuses verification effort on the smallest unit of software design (i.e.), the module. Unit Testing exercise specific paths in a module's control structure to ensure complete coverage and maximum error detection.

- **Integration Testing**

Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing.

- **Validation Testing**

At the end of Integration Testing, software is completely assembled as a package, interfacing errors have been uncovered and correction testing begins. Software Testing and Validation is achieved through serried of Black Box tests that demonstrate conformity with the requirements.

➤ **System Testing**

System testing is series of different tests whose primary purpose is to fully exercise the computer based system. Types of system test includes : recovery testing, security testing, stress testing, performance testing. Although each test has a different purpose, all the work should verify that all system elements have been properly integrated and perform allocated functions.

- **Performance Testing**

Performance testing is designed to test run-time performance of application within the context of an integrated system . Proper response time for user actions is critical to maintaining and enhancing user base.

- **Load Testing**

Load testing demonstrates how the application performs under concurrent user sessions for typical user scenarios. Setting up common scenarios that execute for a short period of time allows seeing how the application operates under a multiple-user load.

- **Stress Testing**

Stress test allows examining how the application behaves under a maximum user load. To stress test application, remove the think time for load scripts and execute the scripts against the server to overload use of the application. If there are unhandled exceptions in a stress test, the application may not be robust enough to handle a sudden unexpected increase in user activity. Stress tests generally execute for a longer period of time, and can be used to catch difficult-to-diagnose problems like subtle memory leaks in the application

6.4 Test Cases

- **Testcase-1: User Registration**

Field No.	Field Name	Invalid Values	Valid Values	Messages
1.	Username	@maan123	Amaan	Please enter a Valid name
2.	Password	Null	145swwfd@7	Please Entera Valid Password
3.	E-mail	@maan@hi	amaan@gmail.com	Please enter a valid EmailID
4.	Phone Number	123456482555488	9876543210	Please Entera Valid Nunber
5.	City name	21dasa	Surat	Please Entera Valid Cityname
6.	Bio	ertyuidfghjkbnm	Abcxyz	Please Entera Valid Bio
7.	Date and Time	dd-mm-yy	12/01/2012	Please Entera ValidDATE
8.	Gender	NULL	Male	Please Entera Valid Gender
9.	DOB	dd-mm-yyyy	10-06-2021	Please Entera Valid Date

- **Testcase-2: User Login**

Field No.	Field Name	Invalid Values	Valid Values	Messages
1.	Username	@maan123	Amaan	Please enter a Valid name
2.	Password	Null	Fghj[;']	Please enter a Valid Password

- **Testcase-3: Add New Image**

Field No.	Field Name	Invalid Values	Valid Values	Messages
1.	Title	Null	Sports Car	Please enter a Valid Title
2.	URL	rtyu@.com	Google.com	Please enter a Valid url
3.	Description	Null	Abcxyz	Please enter a Valid Description
4.	Price	Null	200	Please enter a Valid Price
5.	Catagory	Null	Vehical	Please enter a Valid Catagory

7. Limitation and System Enhancement

7.1 Limitation

- **Lack of originality:** Since many people have access to the same stock images, there is a risk that the images will be overused, which can reduce their impact and make them less effective in conveying a unique message.
- **Limited customization options:** Stock images are often pre-designed, so customers may not have the flexibility to make changes or adjustments to fit their specific needs.
- **Quality concerns:** While many stock image websites offer high-quality images, there is always a risk that customers may not be completely satisfied with the quality of the images they purchase.
- **Legal issues:** Some stock images may infringe on copyright or other legal restrictions, which can lead to legal issues for the customer or the website.
- **Cost:** While stock images can be an affordable option for some customers, the cost can add up quickly if multiple images are needed, and there may be additional fees for extended usage rights or exclusive use.
- **Limited variety:** Despite the large number of images available on stock image websites, there may not be enough variety to meet all a customer's needs, especially if they require highly specialized images.

7.2 System Enhancement

- **Improved Search Functionality:** One of the most important features of a stock image website is its search functionality. Enhancing the search algorithm to provide more accurate results based on keywords, image orientation, colour scheme, image style, or other criteria could help customers find the perfect image more quickly and easily.
- **Customizable Images:** Offering more customization options could make stock images more appealing to customers. For example, allowing customers to add or remove objects from an image or change its colour scheme to fit their needs could provide more value to them.
- **Advanced Editing Tools:** Providing advanced editing tools such as cropping, resizing, and background removal within the platform could save customers time and money by reducing the need to use separate editing software.
- **Exclusive Image Rights:** Offering customers exclusive usage rights to a particular image or a set of images could be a valuable selling point for the website.
- **User Feedback:** Collecting feedback from users about their experience using the website and the quality of the images could help the website administrators to identify areas for improvement.
- **Mobile App:** Developing a mobile app for the website could increase accessibility for customers and make it easier for them to browse and purchase images on the go.

8. Conclusion

- stock image selling websites provide a valuable service to businesses and individuals looking for high-quality, professional images for their projects. These websites offer a wide variety of images that can be easily downloaded and used for commercial and personal purposes.
- However, the competition in this industry is high, and success on these platforms requires not only quality images but also a strong marketing strategy and a willingness to continually update and improve one's portfolio. It is important to carefully review and understand the terms and conditions of each platform before uploading and selling images, as they can vary significantly from site to site.

9. Bibliography

1. [http://preview.themeforest.net/item/cassie-responsive-dashboard-and-admin-html5-template/full-screen-preview/24369212? ga=2.10033587.787056111.1677068572-1608235303.1675426332](http://preview.themeforest.net/item/cassie-responsive-dashboard-and-admin-html5-template/full-screen-preview/24369212?ga=2.10033587.787056111.1677068572-1608235303.1675426332)
2. <http://themepixels.me/demo/cassie/pages/dashboard-three.html>
3. <https://desygner.com/blog/benefits-stock-photos/>
4. <https://app.diagrams.net/>
5. <https://stackoverflow.com>
6. <https://www.w3schools.com>
7. <https://www.geeksforgeeks.org>
8. <https://www.tutorialspoint.com>
9. <https://www.youtube.com/@Django>
10. <https://www.youtube.com/@CodeWithHarry>