

Dharmsinh Desai University, Nadiad

Faculty of Technology, Department of Computer Engineering B.Tech. CE Semester – VI

Subject: System Design Practice

Project Title:

# **Online Auction**

Submitted By:

Bhikadiya Yash CE016 17CEUOS107

Kachhia Neel CE052 17CEUOS121

Khunt Deep CE061 17CEUOG096

Guided By:

Prof. Sheetal S. Shah

Assistant Professor, CE Dept.

Dharamsinh Desai University, Nadiad



Dharmsinh Desai University, Nadiad

Faculty of Technology, Department of Computer Engineering

**CERTIFICATE**

This is to certify that System Design Practice project entitled “Online Auction” is the bonafide report of work carried out by

1. **Bhikadiya Yash CE016 17CEUOS107**
2. **Kachhia Neel CE052 17CEUOS121**
3. **Khunt Deep CE061 17CEUOG096**

Of Department of Computer Engineering, Semester VI, academic year 2019-2020, under our supervision and guidance.

Guide HOD

**Prof. Sheetal S. Shah Dr. C. K. Bhensdadia**

Assistant Professor of Head of the Department of

Department of Computer Department of Computer

Engineering, Engineering,

Dharmsinh Desai University, Dharmsinh Desai University,

Nadiad. Nadiad.

**INDEX**

|  |  |
| --- | --- |
| **1.Abstract** | **4** |
| **2.Introduction** | **5** |
| **3.Software Requirement Specifications** | **6** |
| **3.1 End user** | **6** |
| **3.2 Admin** | **7** |
| **4.Design** | **8** |
| **4.1 Activity Diagram** | **8** |
| **4.2 Sequence Diagram** | **10** |
| **4.3 Use-Case Diagram** | **12** |
| **4.4 Class Diagram** | **13** |
| **4.5 State Diagram** | **14** |
| **4.6 ER Diagram** | **15** |
| **4.7 Data Dictionary** | **16** |
| **5.Implemantation Detail** | **18** |
| **5.1 Module** | **18** |
| **6.ScreenShots** | **22** |
| **7.Conclusion** | **29** |
| **8. Limitation and Future Extension** | **29** |
| **9. Reference** | **31** |

# 1. Abstract

Online Auction provides facility of auction to customer. They can take part in auction, view live auction. Customer can also put their products online on our website for the auction, so that other customer can buy it. They can earn profits from it. We are also providing auction for players auction. The teams can take part in it online. Normal user can watch the special auction but cannot take part in it. Authorized special users by admin only can take part in it.

# 2. Introduction

Online auction will provide auction facility to users so that they can buy products online. Users can also view other live auctions. Users can put their products in to our website very easily for the online auction so other user can buy it and they can earn profits. There is one special section for players auction. It is permitted to specific users only by the admin user. Many e-commerce websites provide selling of the products through their website but there is not much website that provide user facility to do auction of their products online. Our aim is to provide auction facility online to all users.

# Tools/Technologies

**Technologies:**

HTML 5

CSS 3

Bootstrap 4

Python

Django

DBSqlLite AWS S3 Buckets

**Tools:**

Visual Studio Code

PyCharm

**Hosted at:** Localhost:8000/

# 3. Software Requirement Specifications

**3.1 Types of User**

1. End users
2. Admin

**3.2 System Functional Requirements**

**R.1: End Users**

R.1.1: Login

IP: Provide username & password

OP: Success or invalid message

Process: validation of credentials

R.1.2: Register

IP: Add required information

OP: Successfully registered

R.1.3: Live Auctions

There are the category vice auctions. User can see live, upcoming and past auctions.

IP: Enter to home page of auctions.

OP: View auctions.

R.1.4: Take Part

User can take part in auctions by raising the price in live auctions.

If that user is last in incrementing the price then he/she will be

the owner of that item

IP: raise price.

OP: user name store to data base.

R.1.5: Add the product in the Auctions.

User can add the product in the auctions, by adding the auction detail.

IP: provide the all product detail.

OP: Item added in the auction .

**R.2: Admin**

R.2.1: Login

IP: Provide username & password

OP: Success or invalid message

Process: validation of credentials

R.2.2: Show users

IP: show request of users

OP: List of users

R.2.3: Organize the Auctions.

Admin can organize the auctions, by adding the auction detail.

IP: provide the all auction detail.

OP: auction will be added.

R.2.4: manage the user

Give authority to particular users, only that can raise the

price

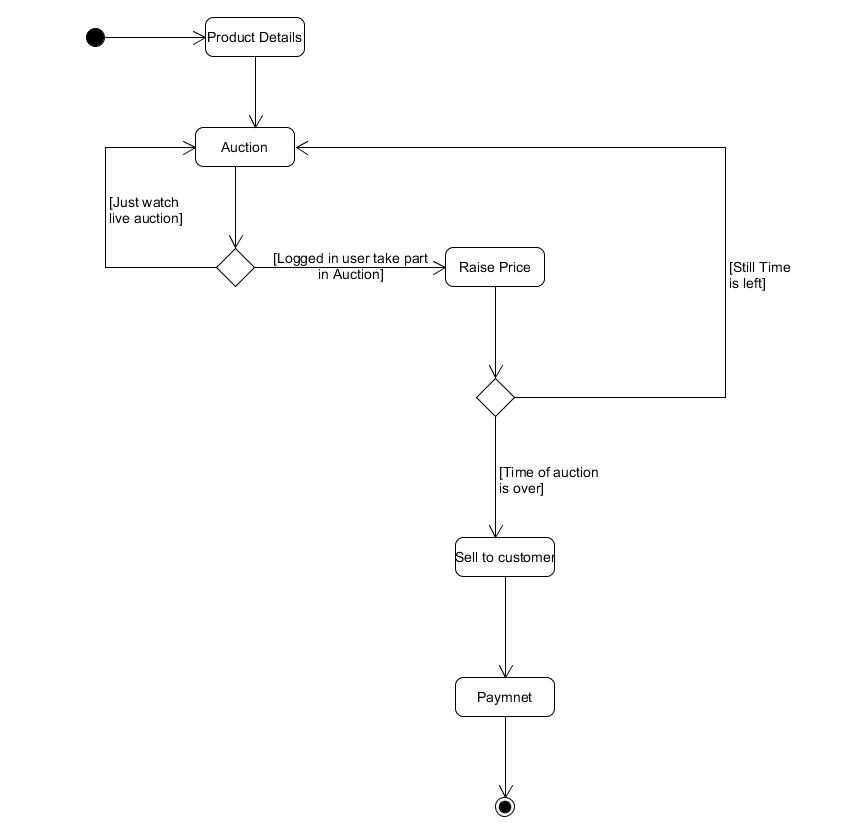
IP: make that user to authorize.

OP: mark that user authorize in data base.

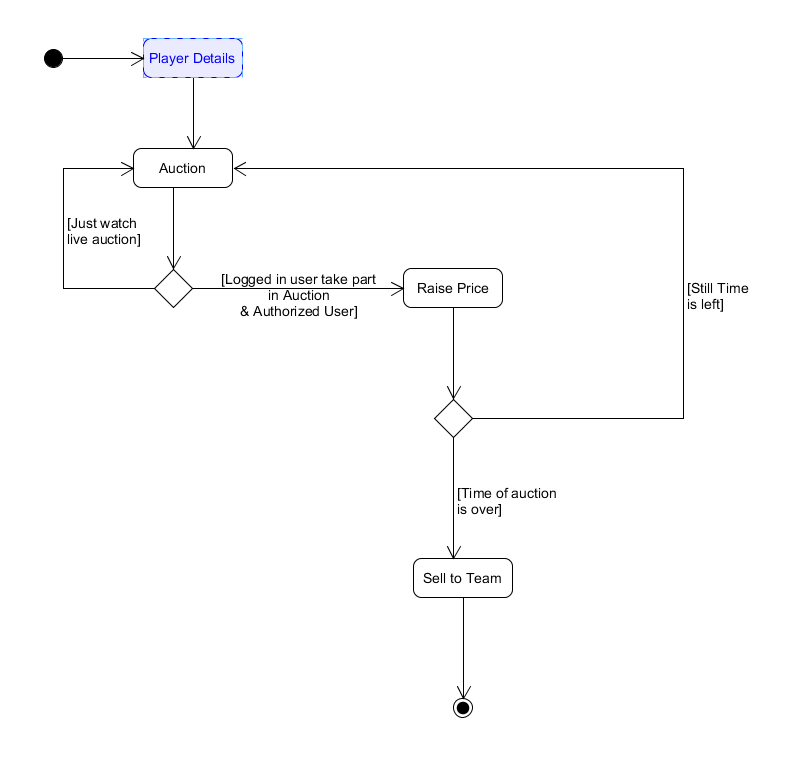
# 4. Design

## Activity Diagrams

1. Product Auction

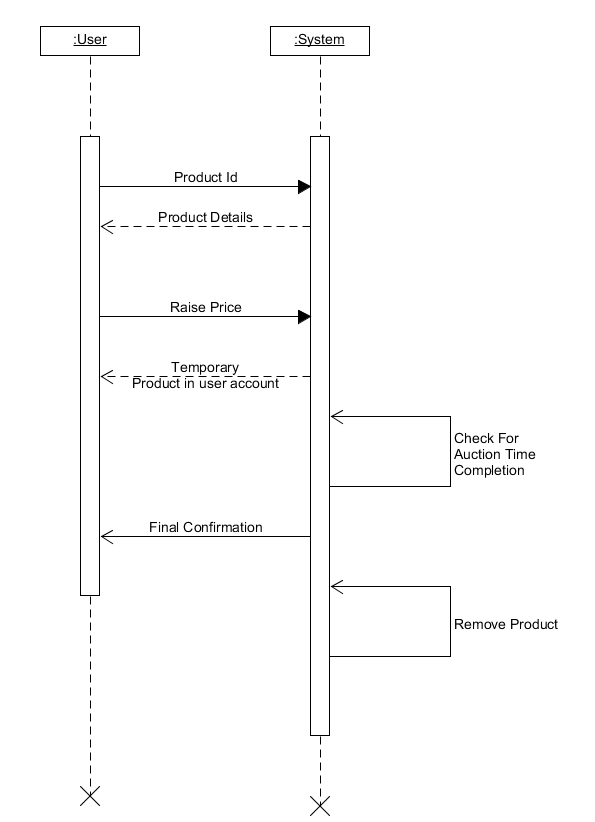


1. Player Auction

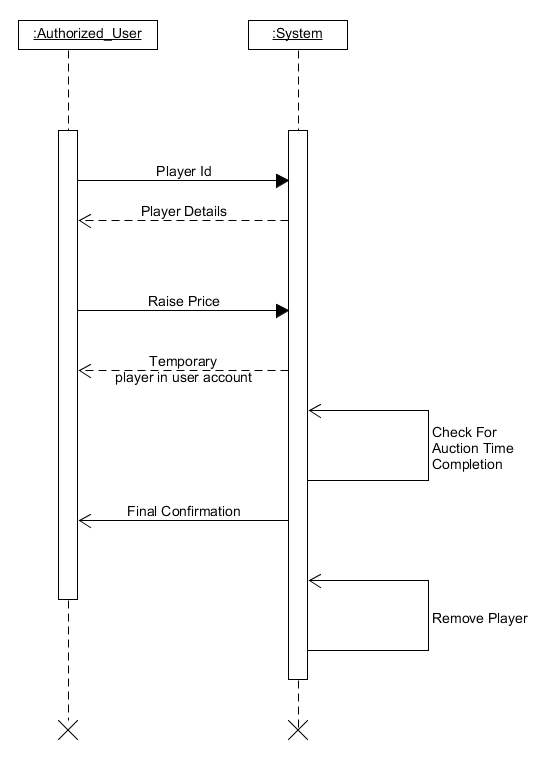


## Sequence Diagrams

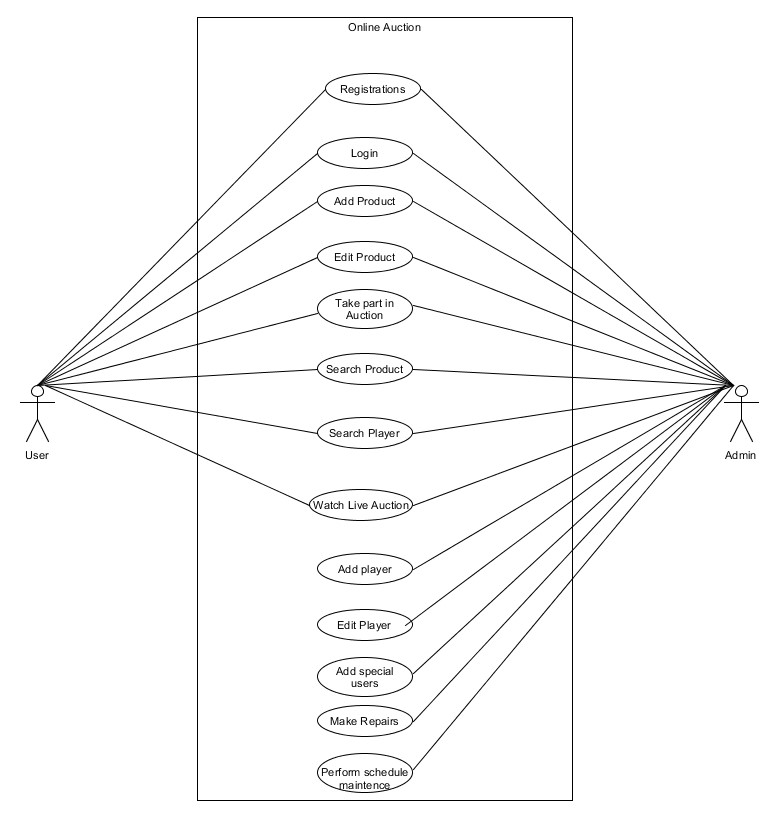
1)Product Auction



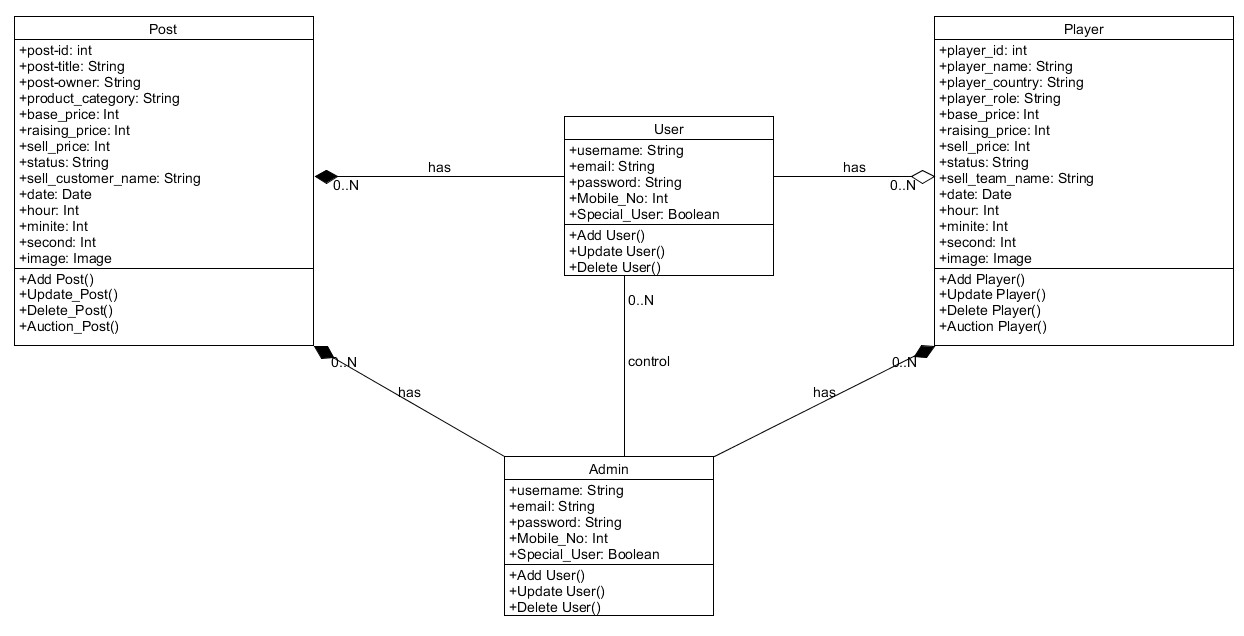
2)Player Auction



# **Use-Case Diagram**

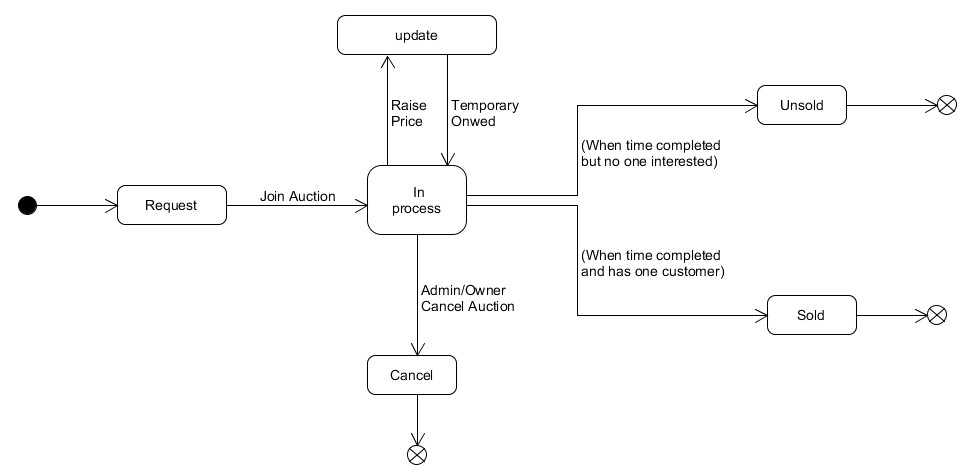


## Class Diagram

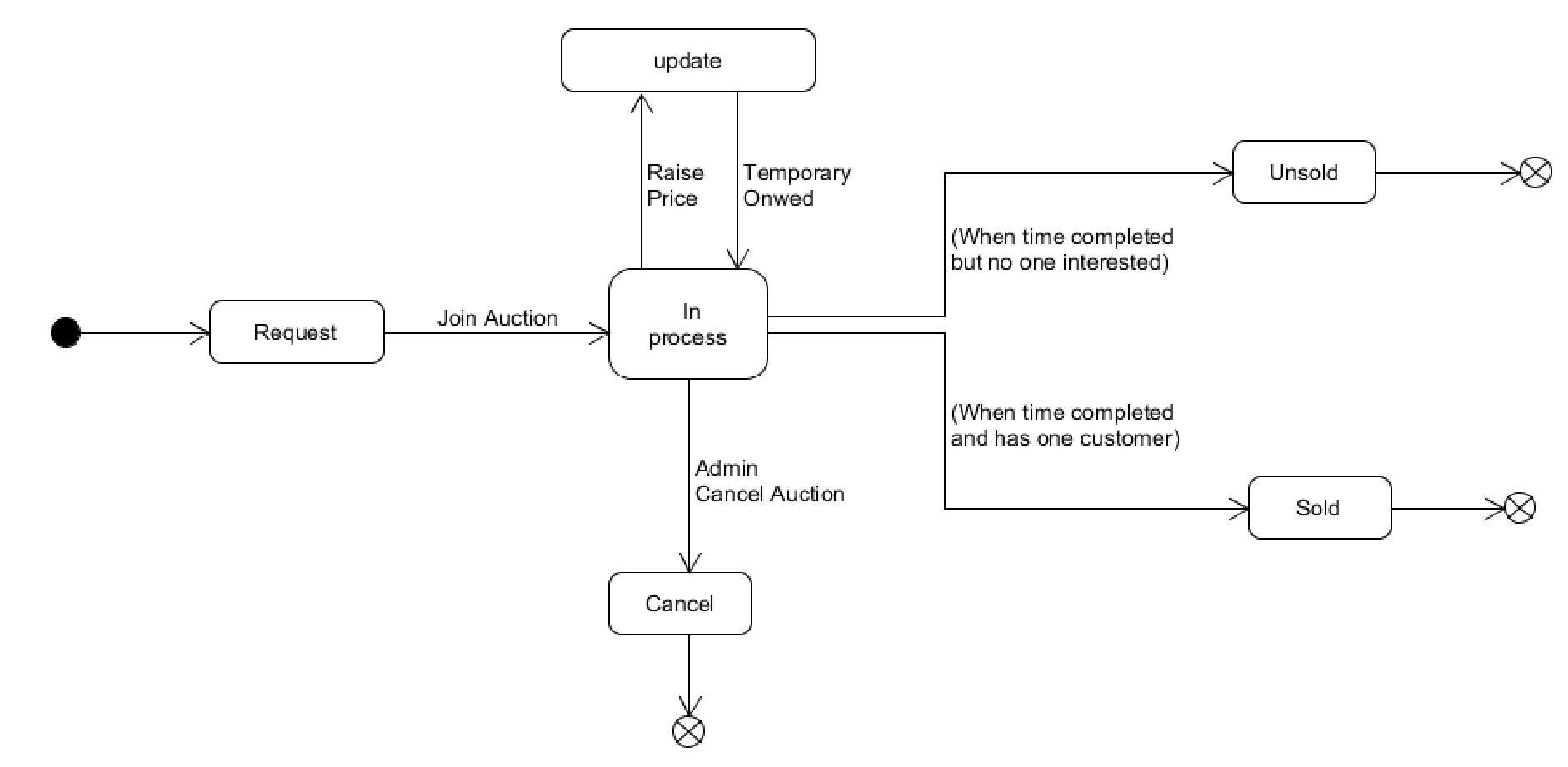


**State Diagram**

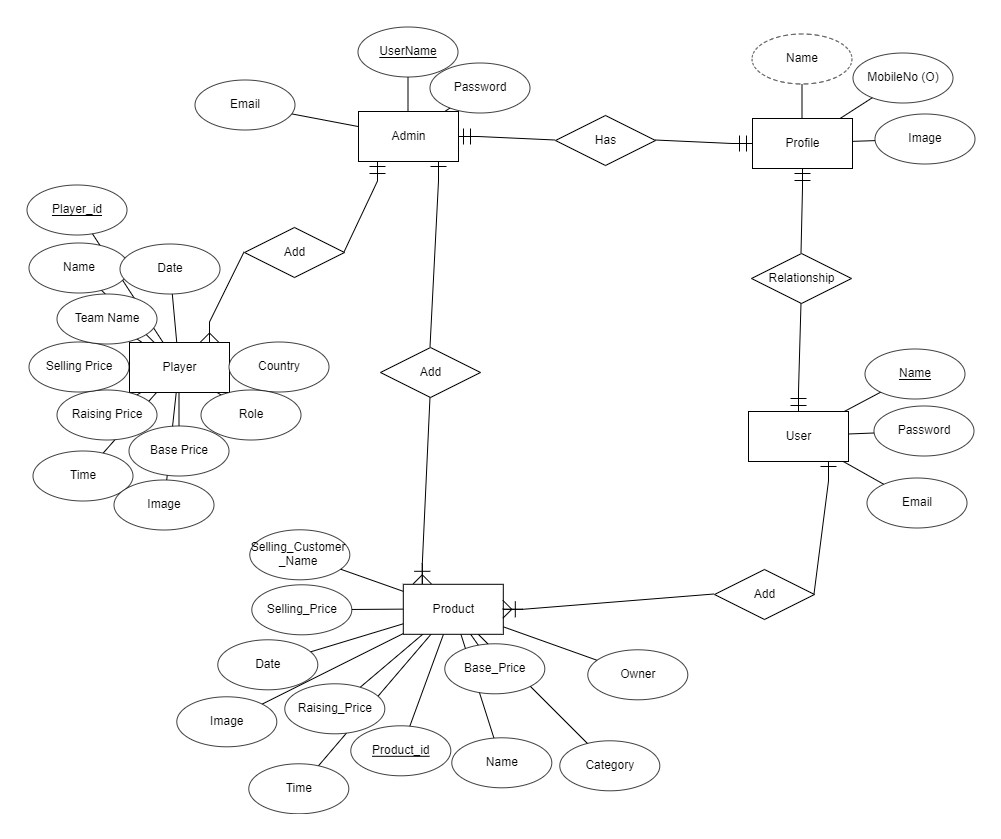
1. Products State Diagram



1. Players State Diagram



## ER Diagram



## 

## Data dictionary

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Admin Table | | | | | | | | |
| Sr. No. | Field Name | Data Type | Width | Required | Unique | PK/FK | Referenced Table | Description |
| 1 | Name | varchar2 | 10 | Yes | Yes | PK |  |  |
| 2 | Email | varchar2 | 20 | Yes | No | No |  |  |
| 3 | Password | varchar2 | 12 | Yes | No | No |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| Profile Table | | | | | | | | |
| Sr. No. | Field Name | Data Type | Width | Required | Unique | PK/FK | Referenced Table | Description |
| 1 | Name | varchar2 | 10 | Yes | Yes | FK | Admin Table |  |
| 2 | Mobile No | int | 10 | No | No | No |  |  |
| 3 | Image | image | 12 | Yes | No | No |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| Player Details | | | | | | | | |
| Sr. No. | Field Name | Data Type | Width | Required | Unique | PK/FK | Referenced Table | Description |
| 1 | Player\_id | int | 25 | Yes | Yes | PK |  |  |
| 2 | Name | Varchar2 | 10 | Yes | No | No |  |  |
| 3 | player\_country | Varchar2 | 10 | Yes | No | No |  |  |
| 4 | player\_role | Varchar2 | 11 | Yes | No | No |  |  |
| 5 | base\_price | int | 12 | Yes | No | No |  |  |
| 6 | raising\_price | int | 12 | Yes | No | No |  |  |
| 7 | sell\_price | int | 12 | Yes | No | No |  |  |
| 8 | sell\_teamName | varchar2 | 25 | Yes | No | FK | Profile Table |  |
| 9 | status | boolean | 5 | Yes | No | No |  |  |
| 10 | date | date | 8 | Yes | No | No |  |  |
| 11 | hour | int | 2 | Yes | No | No |  |  |
| 12 | minite | int | 2 | Yes | No | No |  |  |
| 13 | seconds | int | 2 | Yes | No | No |  |  |
| 14 | image | image | 500 | Yes | No | No |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| Post Details | | | | | | | | |
| Sr. No. | Field Name | Data Type | Width | Required | Unique | PK/FK | Referenced Table | Description |
| 1 | Post\_id | int | 25 | Yes | Yes | PK |  |  |
| 2 | Post\_title | Varchar2 | 10 | Yes | No | No |  |  |
| 3 | Post\_owner | Varchar2 | 10 | Yes | No | No |  |  |
| 4 | product\_category | Varchar2 | 11 | Yes | No | No |  |  |
| 5 | base\_price | int | 12 | Yes | No | No |  |  |
| 6 | raising\_price | int | 12 | Yes | No | No |  |  |
| 7 | sell\_price | int | 12 | Yes | No | No |  |  |
| 8 | sell\_customerName | varchar2 | 25 | Yes | No | FK | Profile Table |  |
| 9 | status | boolean | 5 | Yes | No | No |  |  |
| 10 | date | date | 8 | Yes | No | No |  |  |
| 11 | hour | int | 2 | Yes | No | No |  |  |
| 12 | minite | int | 2 | Yes | No | No |  |  |
| 13 | seconds | int | 2 | Yes | No | No |  |  |
| 14 | image | image | 500 | Yes | No | No |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
| User Table | | | | | | | | |
| Sr. No. | Field Name | Data Type | Width | Required | Unique | PK/FK | Referenced Table | Description |
| 1 | Name | varchar2 | 10 | Yes | Yes | PK |  |  |
| 2 | Email | varchar2 | 20 | Yes | No | No |  |  |
| 3 | Password | varchar2 | 12 | Yes | No | No |  |  |

## 5. Implementation Details

**5.1 Modules created and brief description of each modules**

**Admin Module:**

Admin can add products and players to the auction. Admin will provide date and time of the end of the auction. Admin will also select categories of the products. Admin will have access to all the products that are added by any user. Admin will have access to permit the auction that is added by user. Admin will also able to specialize user for allowing user to player auction. Admin can also see the AWS S3 Bucket that stores the all the images of the system.

**User Module:**

Users will able to login and register. For all the new registration they will receive email from the system automatically. User can also reset the password by the link that system provide in their email address.

User can also add products in auction, edit it & also delete it.

**Auction Module:**

Without login user can only view live auction and players auction. If user wants to start bidding in auction they must need to login to the system. At the end of the timer which ever user have highest bidding that product or player will be sell to him. To bid in player auction user will need special permission from the admin side.

**5.2 Function Prototypes**

1. Registration

It will take username, email and password to register new user. Username must be unique to all the users. Email will be sent to email address after the successful registration of the user.

1. Login

It will take username and password to login of the user.

1. Forgotten Password

It will take register email id of the user. On that email it will send one password change link with encryption. User must need to click on that link to change their password.

1. Profile

In this section user can upload image to their profile. User can update mobile no, email and all other user details.

1. Details

This function will provide all the details of the product or player to the user.

6)Auction

This function is the most important function of the system. It will take Id of the product. If user want to bid on the product it will add its name and price to the auction. After completion of the auction it will redirected to sold function.

7)Sold

It will check if the product has no bidder than it will be unsold. But if it has bidder than it will display the new owner name, selling price of that product.

8)History

It will display all the details of the product and player whose auction is completed. At what price and whom it is sold will be displayed here.

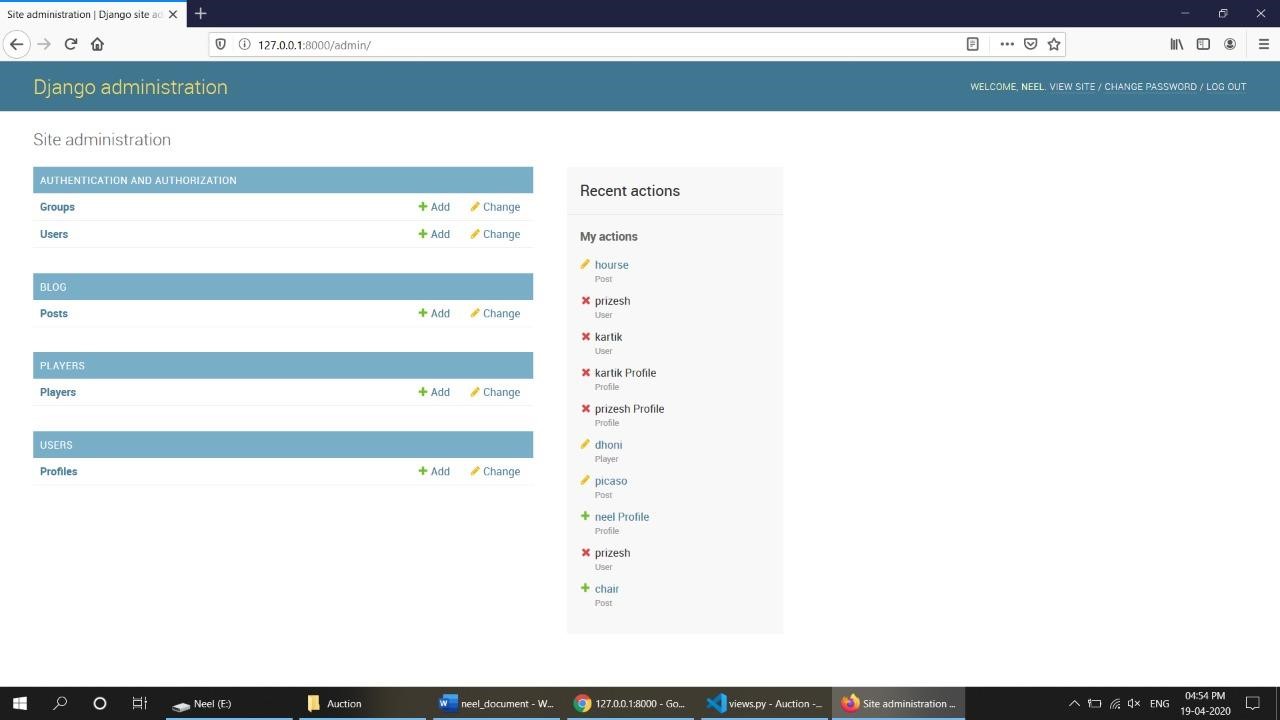
9)Your Players

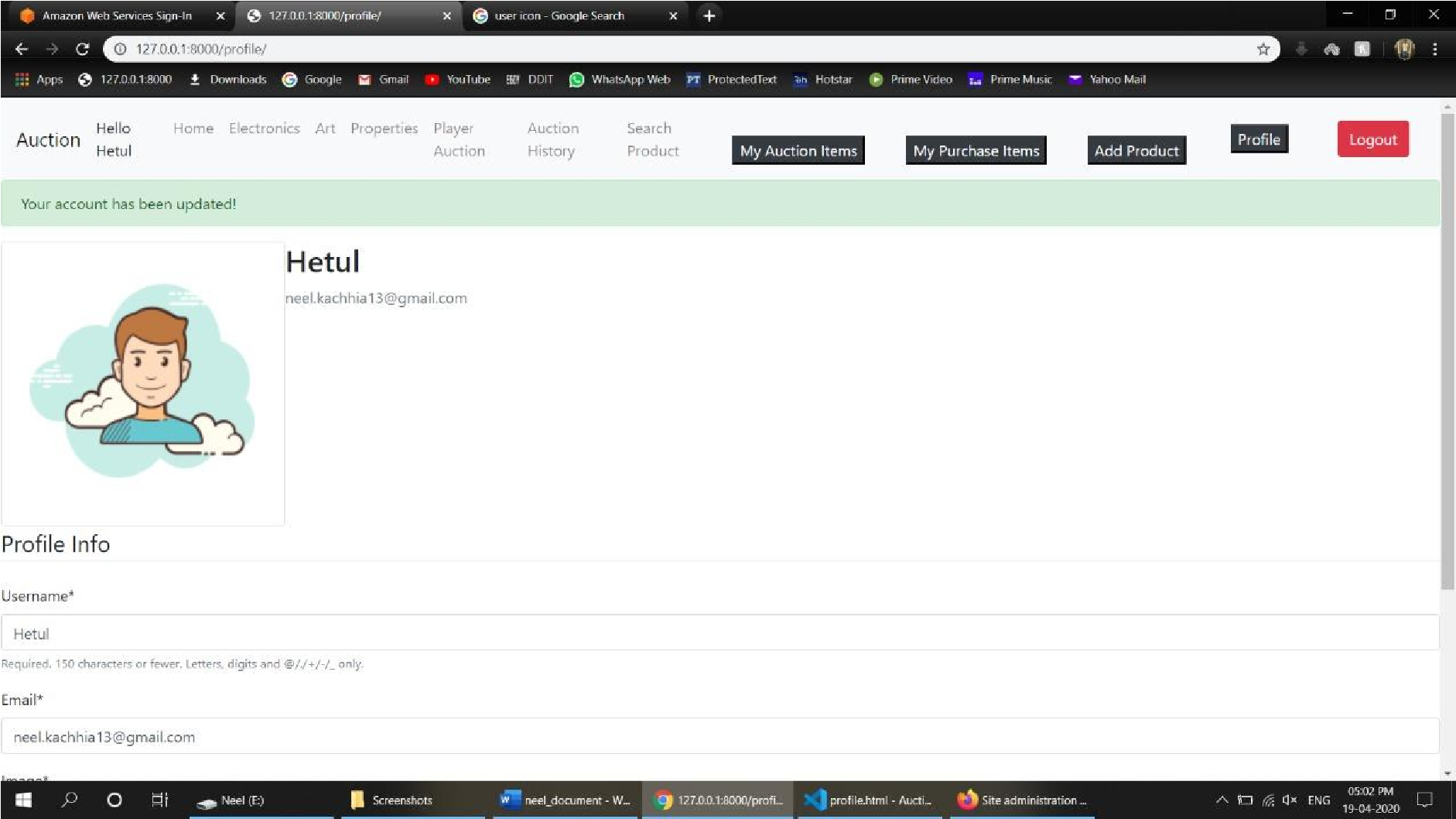
It will display the batsman, bowlers, allrounders of the user that they have bought during the auction.

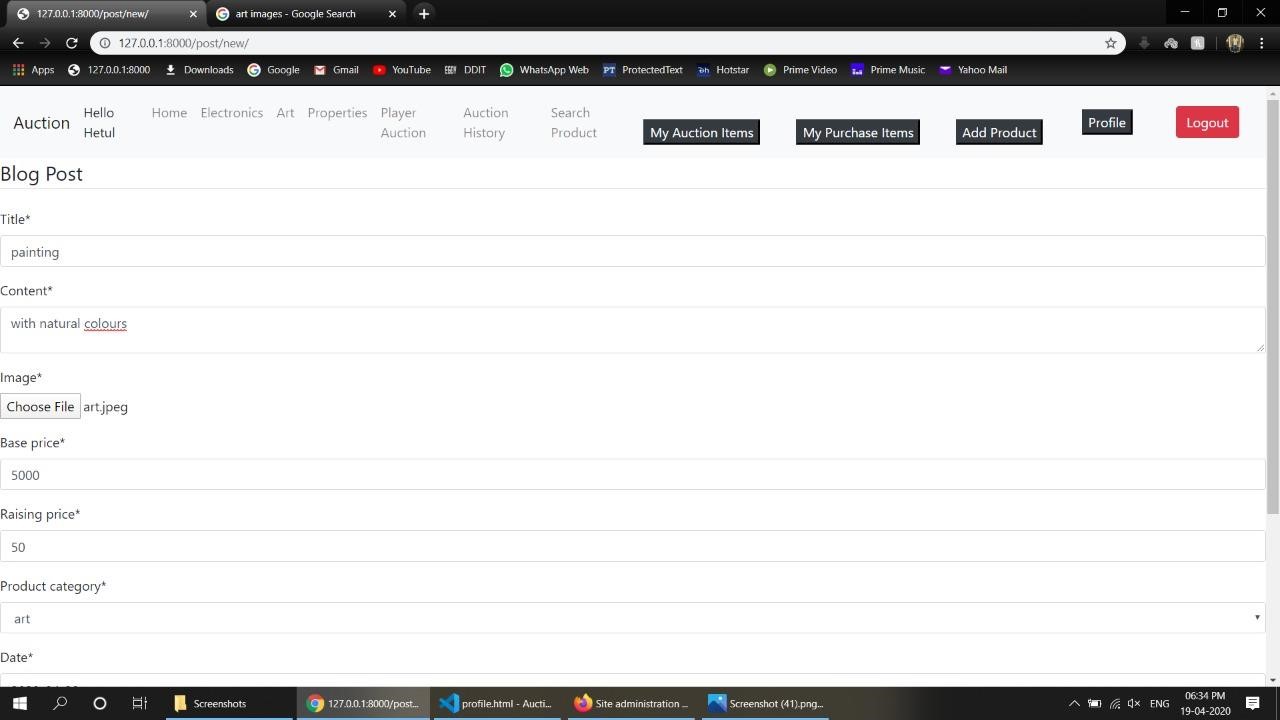
10) Add Product

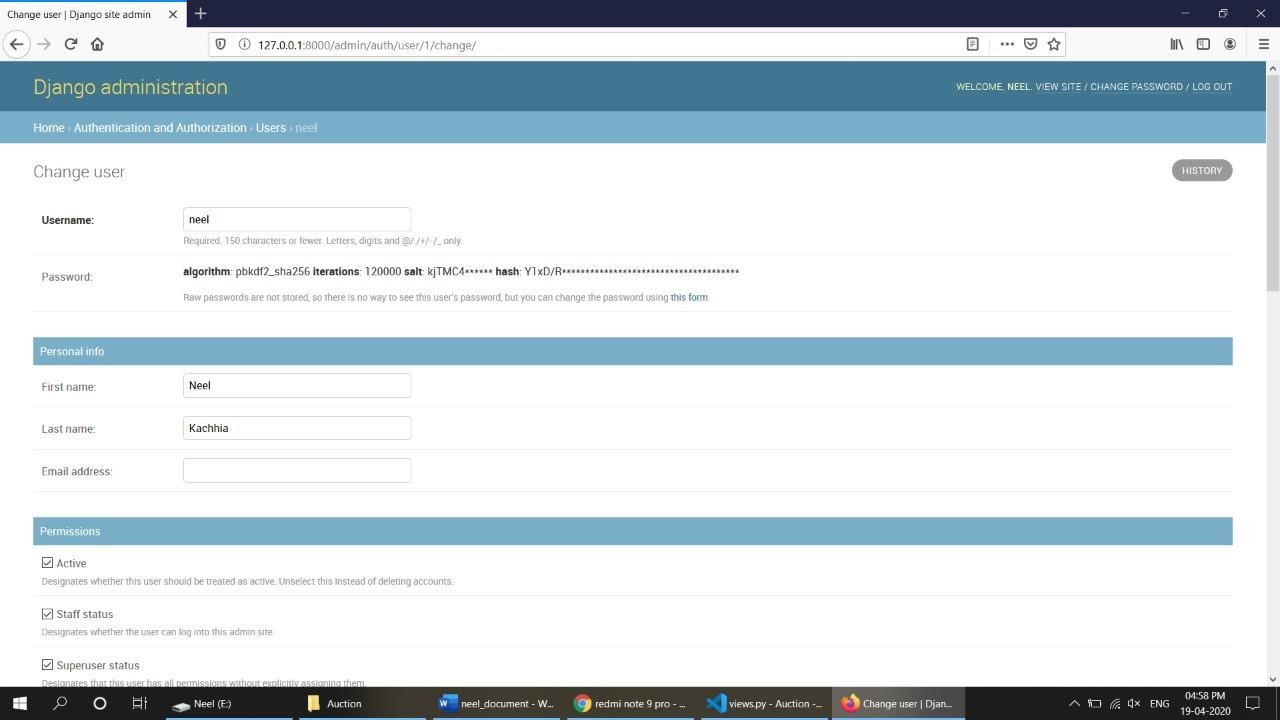
This function will provide functionality to user to add products in the online auction. User need to provide all the details of the product.

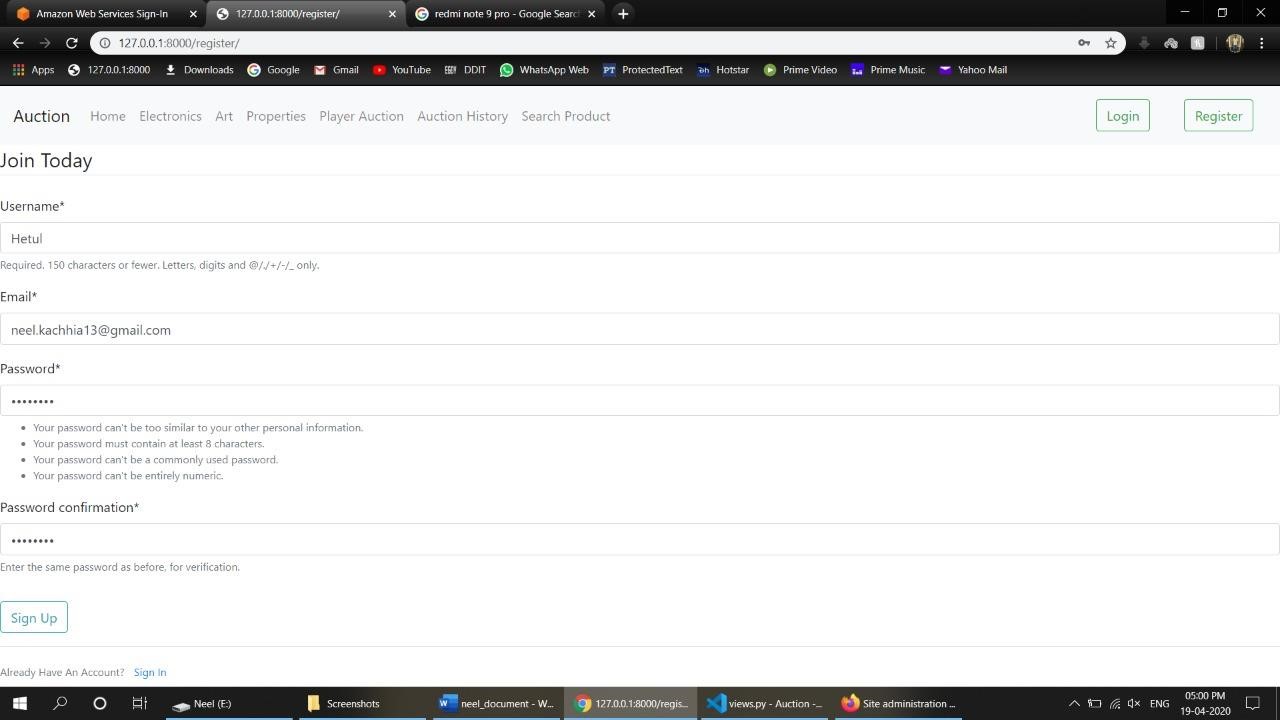
## 6. Screenshots

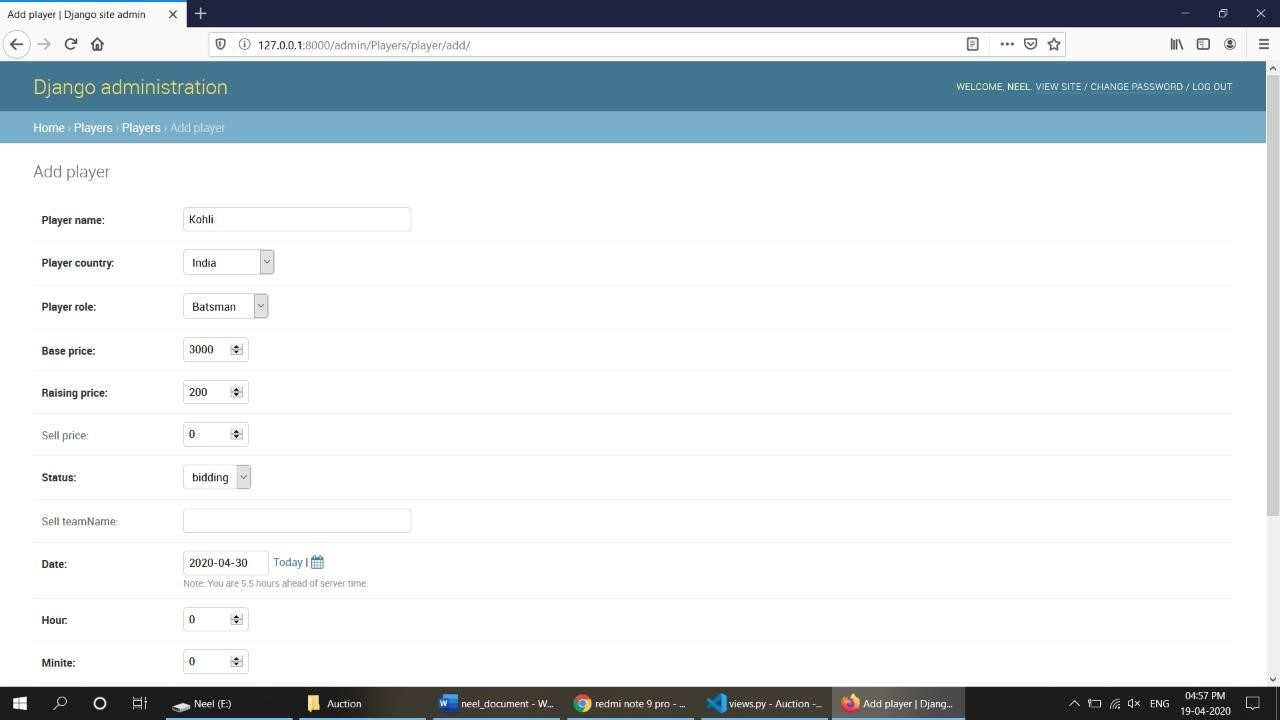


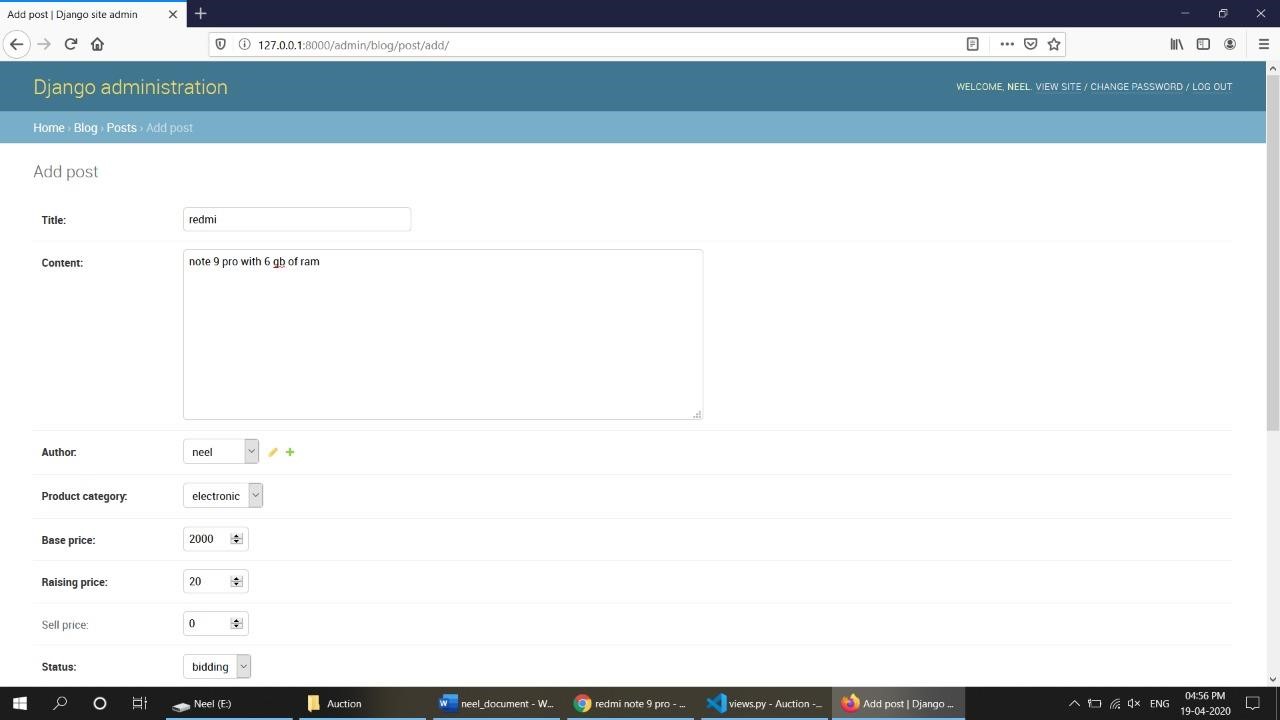


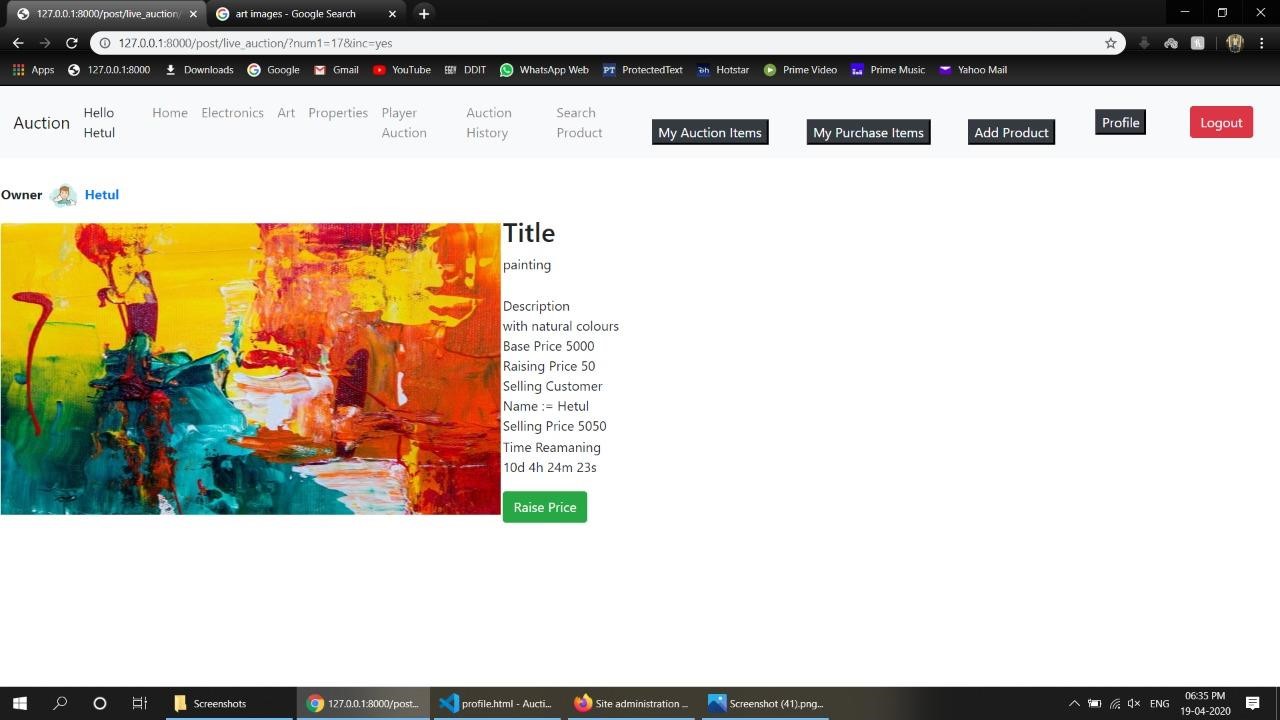


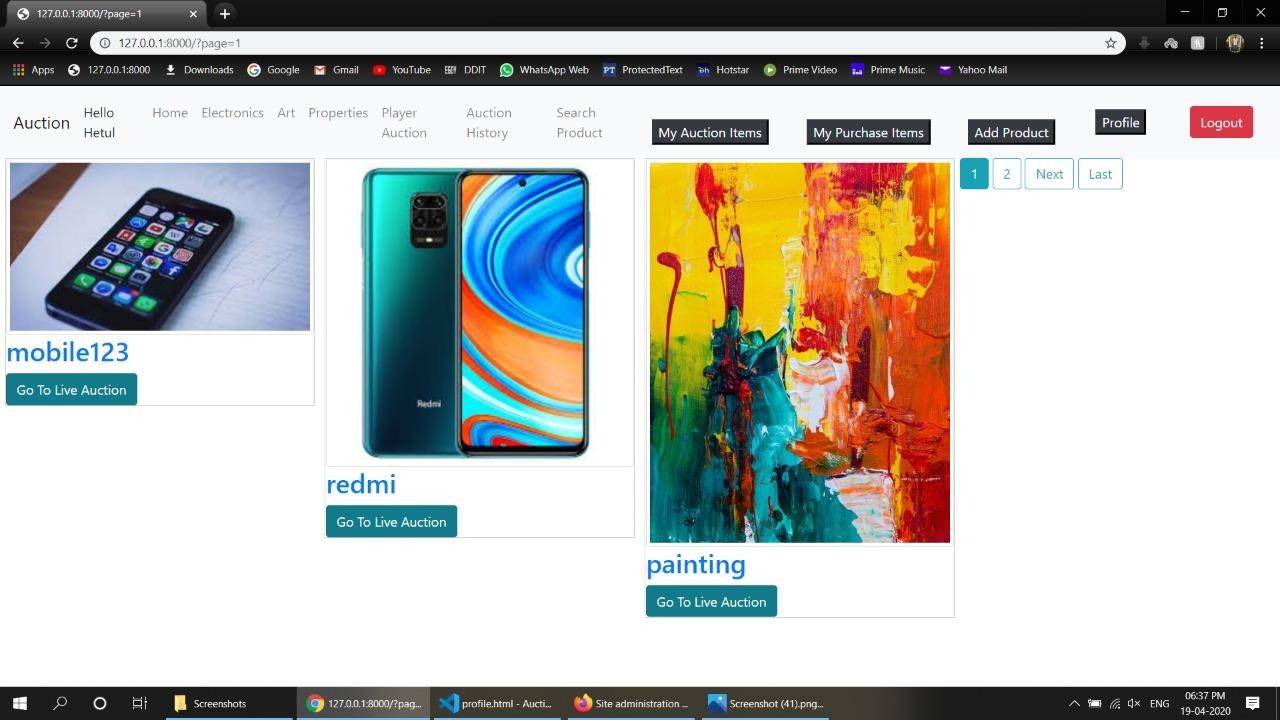


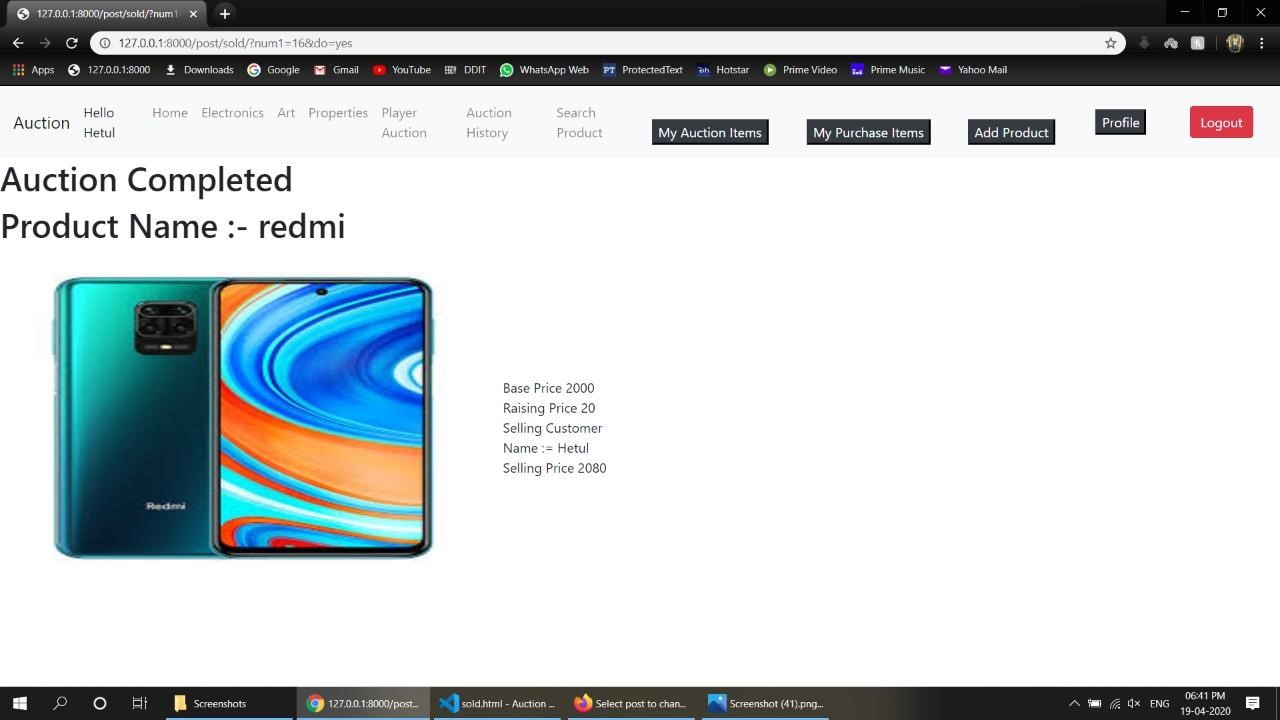


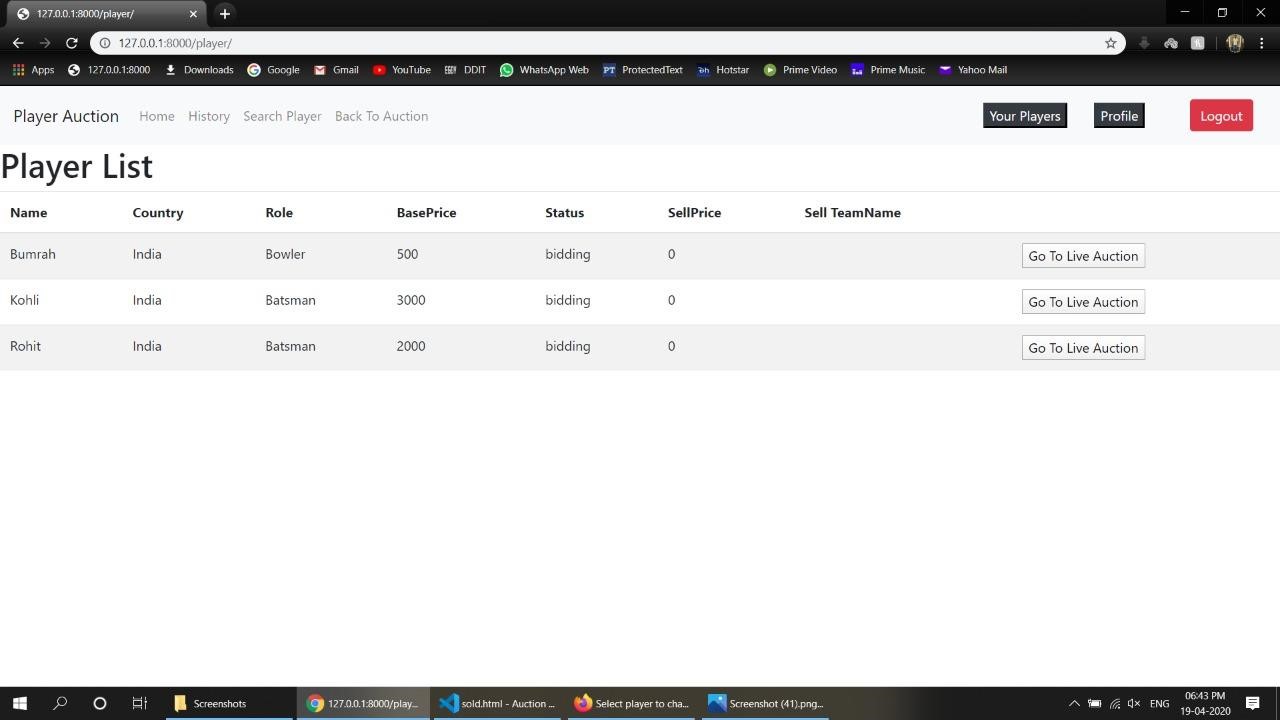


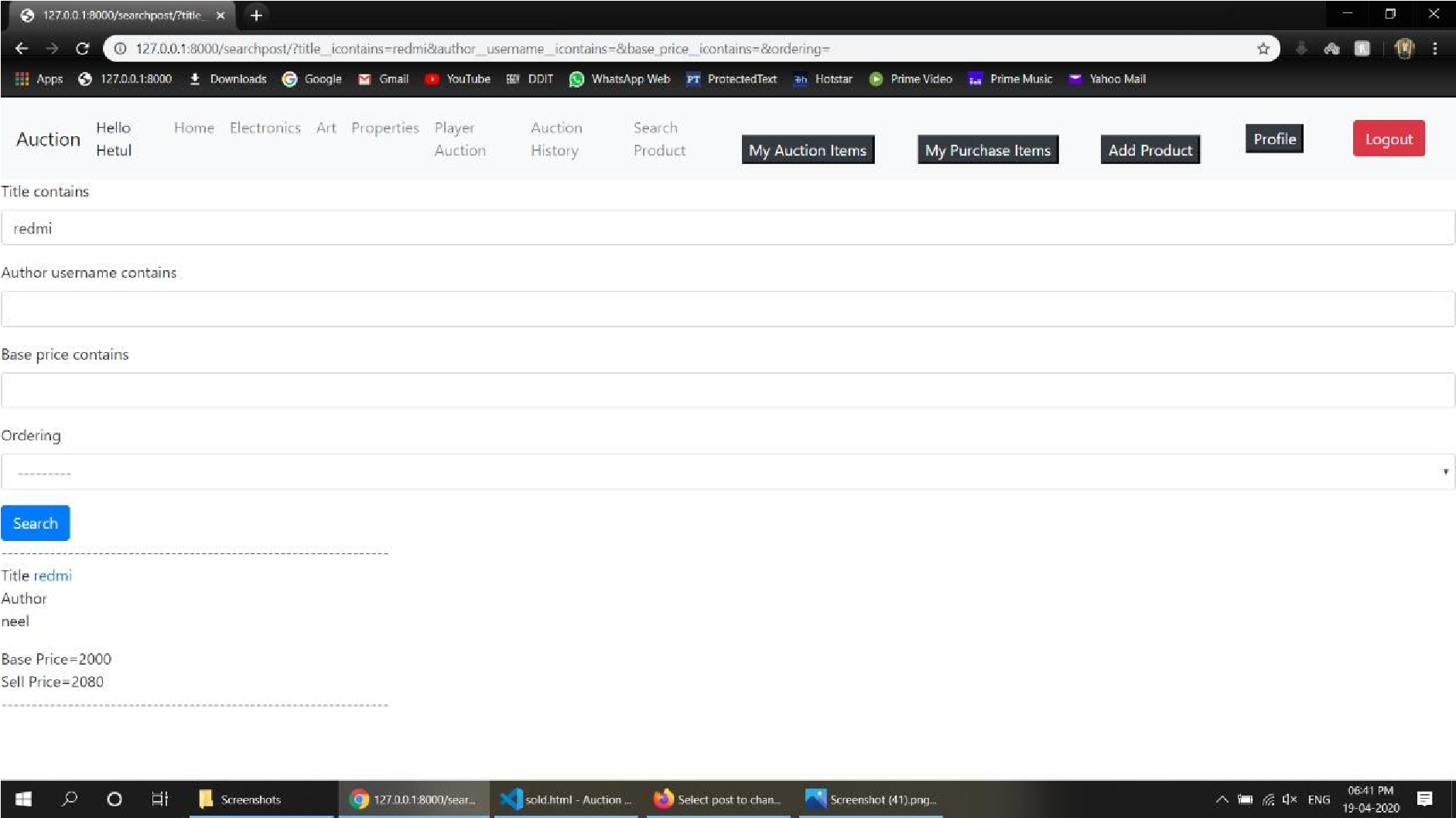


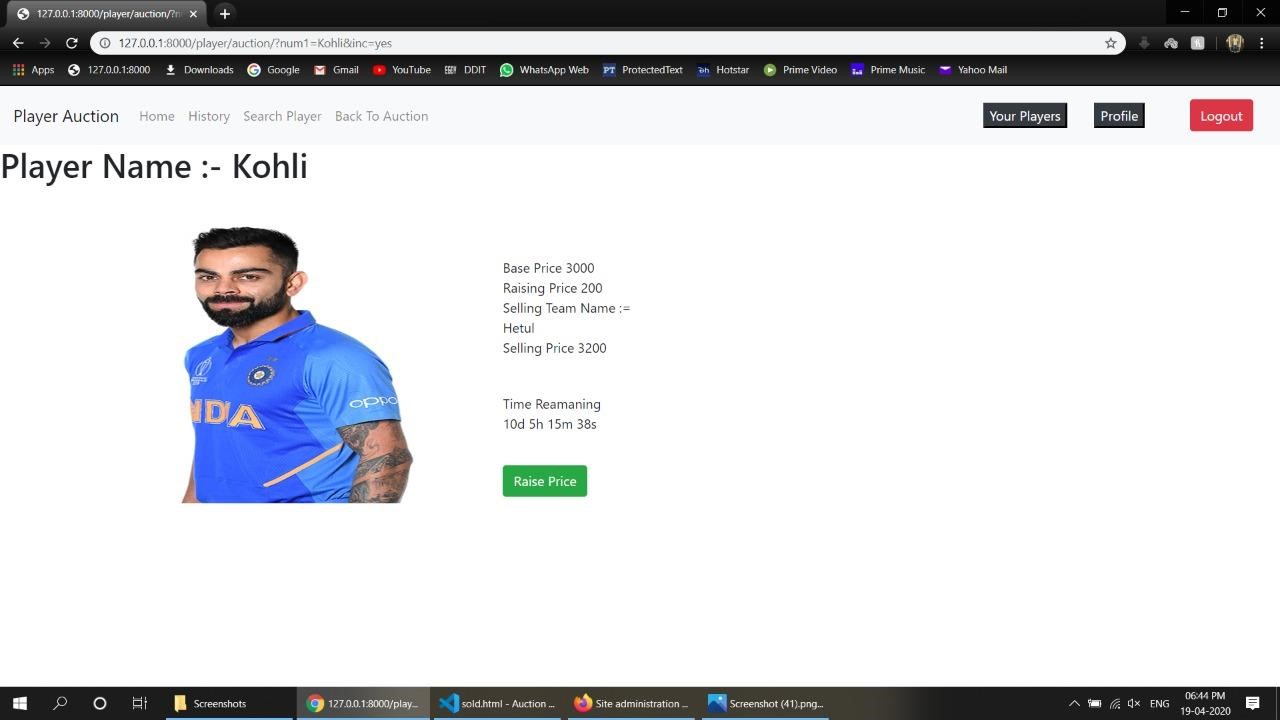


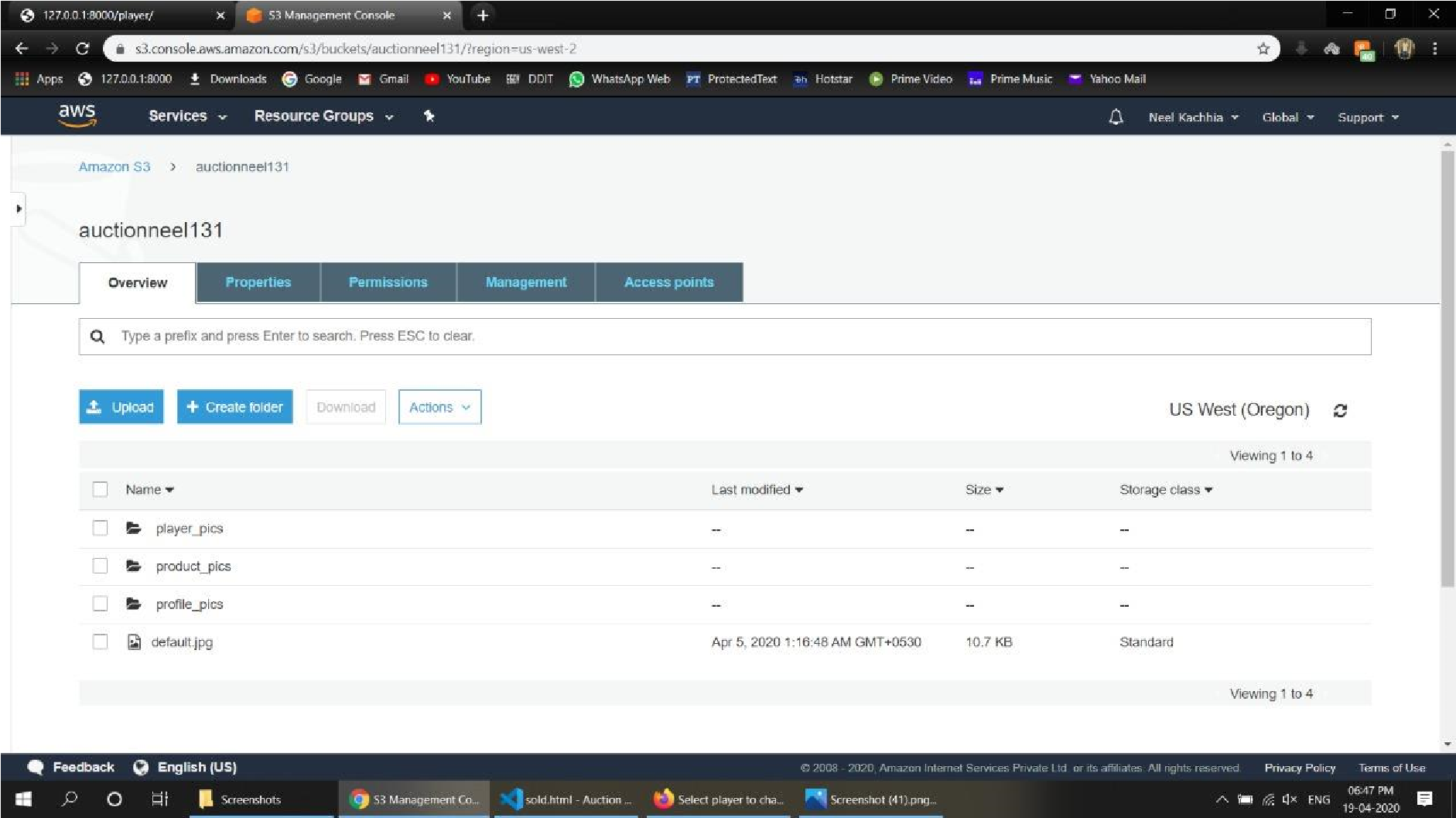












## 7. Conclusion

We have successfully implemented login, signup, change forgotten password, add new products, auction of the products and players, update products, update profile. We are able to store image on AWS S3 Buckets online. Admin can have access to all users, all the auction, can give special authority to users.

## 8. Limitations and Future Extension

### 8.1 Limitation

If more than one user hits the raise price of the same product at the same time then our system fails. We have don’t use any test cases that are mentation in Software Engineering. If the large amount of data that need to processed then also our system slows down. Our system doesn’t have real world payment system for payments of real money through banks or online wallets.

### 8.2 Future Extension

* We want to host our website online on Heroku so that any one with internet access can use it.
* We want to implement payment gateway through banking so that our system work with real money.
* We will also want to provide better database system so that loading of the website will be much faster.
* We will provide some online cart type of mechanism.

## 9. Reference / Bibliography

Following links and websites were referred during the development of this project.

* <https://docs.djangoproject.com/en/3.0/>
* <https://www.w3schools.com/w3css/>
* <https://www.w3schools.com/bootstrap4/default.asp>

[➢](file:///E:\Sem%206\➢)<https://getbootstrap.com/docs/4.0/components/navbar/>

* [https://www.youtube.com/watch?v=UmljXZIypDc&list=PLosiE80TeTtoQCKZ03TU5fNfx2UY6U4p](https://www.youtube.com/watch?v=UmljXZIypDc&list=PL-osiE80TeTtoQCKZ03TU5fNfx2UY6U4p)
* [https://www.youtube.com/watch?v=SIyxjRJ8VNY&list=PLsyeobzWxl7r2ukVgTqIQcl1T0C2mzau](https://www.youtube.com/watch?v=SIyxjRJ8VNY&list=PLsyeobzWxl7r2ukVgTqIQcl-1T0C2mzau)
* <https://github.com/CoreyMSchafer/code_snippets/tree/master/Django_Blog>
* <https://stackoverflow.com/questions/tagged/django>