CSN-254
FINAL PROJECT
ARTHUB

GROUP 16

Members:

- BINDU 20114024
- DEEPANSHU 20114030
- DEEPANSHU MEENA 20114031
- YOUWAN SONI 20114107
- PRIYA SINGH 20114077
- SONALI GUPTA 20114094

Project Proposal

Arthub is an android application designed for all artists and art lovers. Not only artists will have their art displayed to a wide audience but also there will be various competition, which could be free or requires a participation fee, and there will be various incentives for an artist to participate in these competitions. It's not only the winners of the competitions who will get benefit from the competition but also every participant, as their art will be viewed by various people who might be interested in buying their creation, which brings us to the next part of the software where anyone can buy any of the art styles displayed. It will not only provide a platform for the artists but will also help them find mentors as contact details will be mentioned with every painting (which will not be visible to all but only for those who request it) and also bridge the gap between the buyers and sellers of art.

Technology used

- **Payment are done using razorpay.**
- **Upper navigation is used to navigate between different tabs**
- **All the text entries are asked to be filled in edit text boxed which are arranged in linear layout.**
- **There is a firebase based authentication**
- **For Database we have used MySQL and Firebase**

Key Features

The project is to develop an interactive platform where the users can participate in various competitions, can buy arts that they like, and can contact directly to their ideal(mentors). They can then read the reviews, results of the competition, and price of arts or paintings.

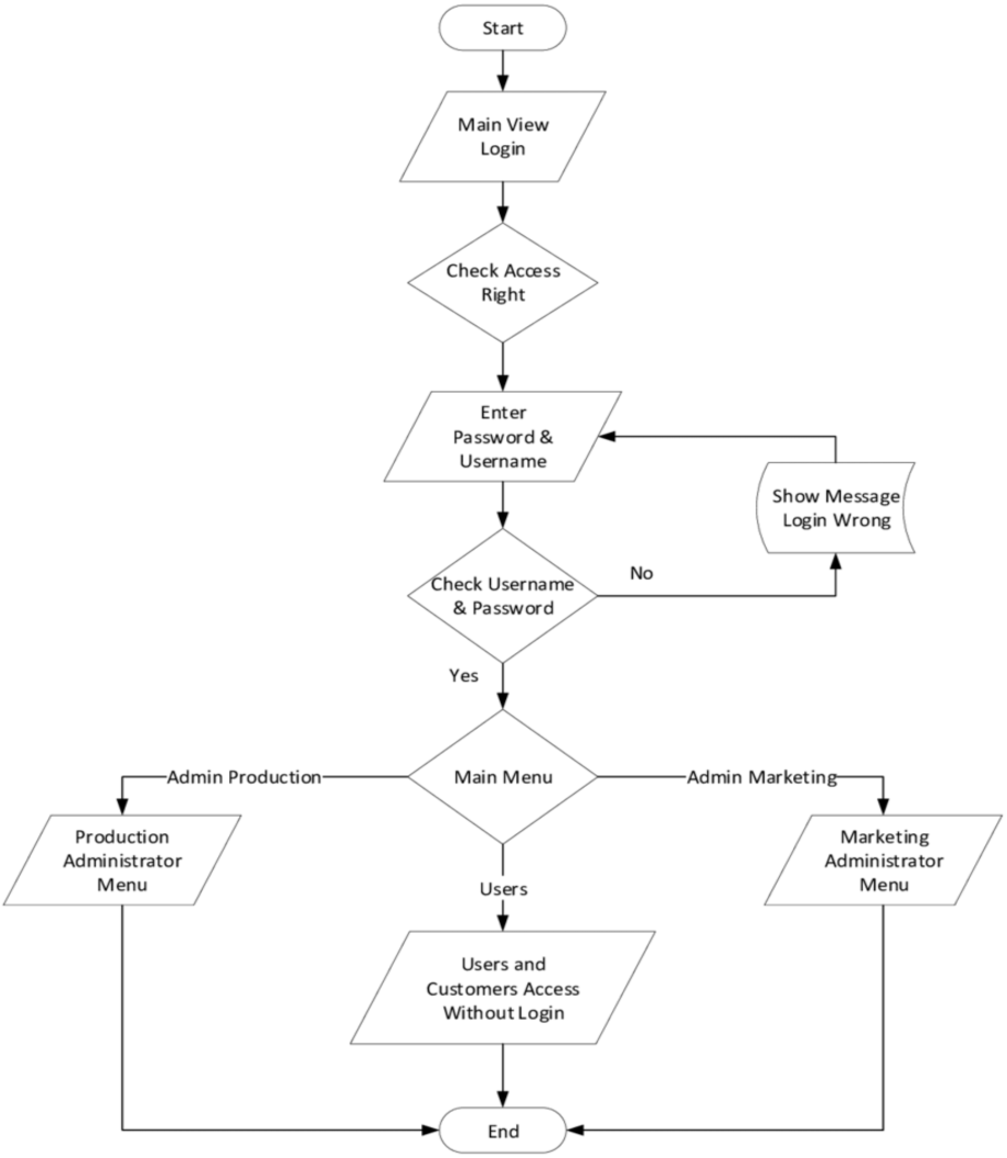
So, the tasks to be implemented are database and Interface. The users can read about the competitions as prize pool, last date to apply, and entry fees. Users can place an order and contact directly to the seller to know more about product. If they want to add their reviews or feedback, they can also add them, determining a database solution to hold the reviews, and feedback. Artist also want a way to find people belonging to their own category of art style and get to know their mindset and art experiences.

- **Latest Updates:** So the latest local updates in an area is that users can buy creation after the competition
- **Promoting beginners**
- **Ease of Access:** Within a single website the user can find the best art at best price and artist can participate in big competition without going anywhere and can show their talent in large number of audience.
- **Updated Feedback and Review:** Survey of the local people with the daily add-on of the feedback by the users shapes up the whole rating infrastructure of the android app.

Design Diagram

- **Data flow diagram**
- **Use Case Diagram**
- **Class Diagram**

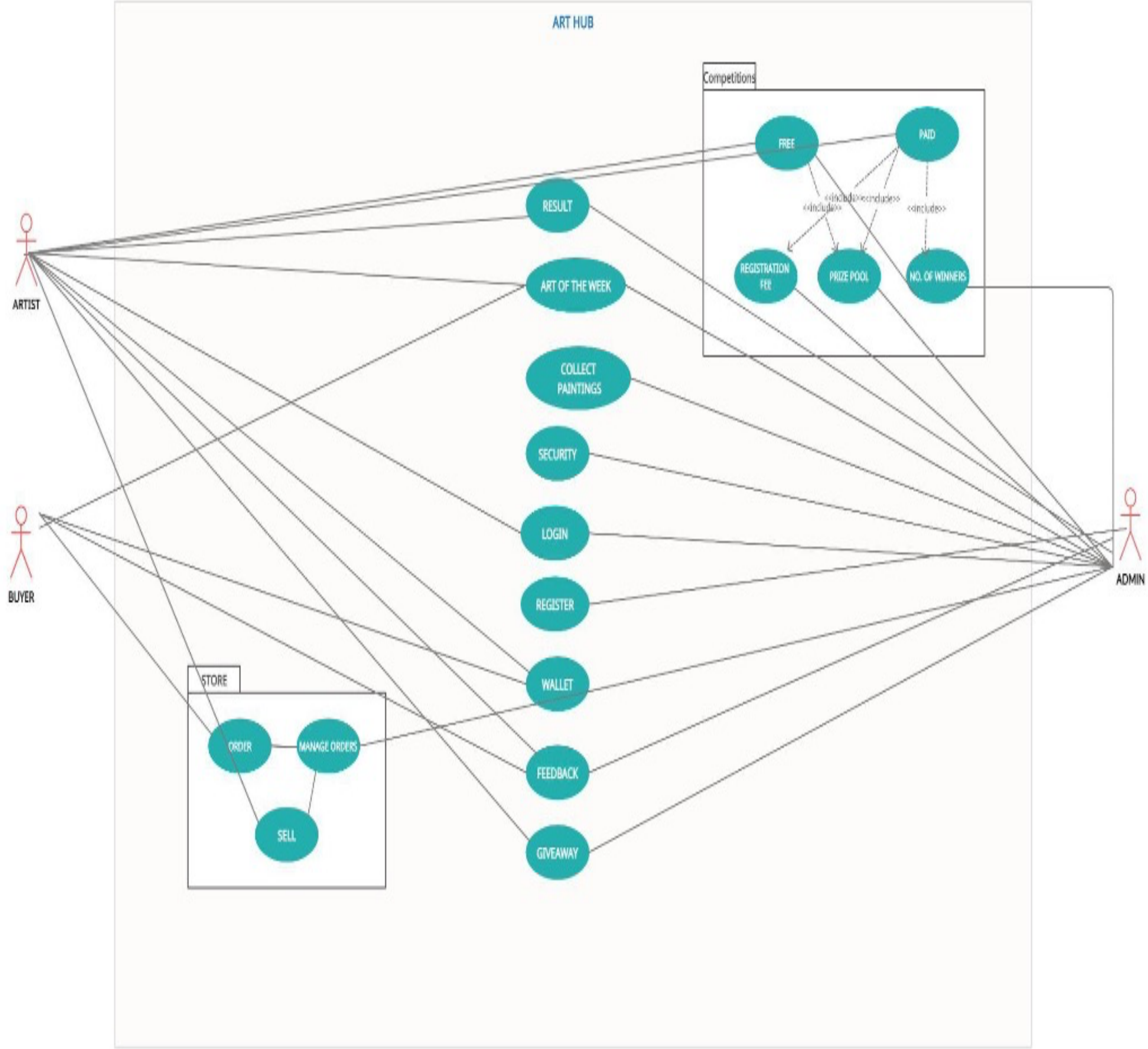
1. Data Flow Diagram


Start
Main View Login
Check Access Right
Enter Password & Username
Show Message Login Wrong
Check Username & Password
No
Yes
Main Menu
Admin Production
Admin Marketing
Production Administrator Menu
Marketing Administrator Menu
Users
Users and Customers Access Without Login
End

- A Data Flow Diagram basically tells us the flow of information in a process or a system.

- It shows us the incoming data, outgoing data and the stored data and the various subprocesses through which the data flows.

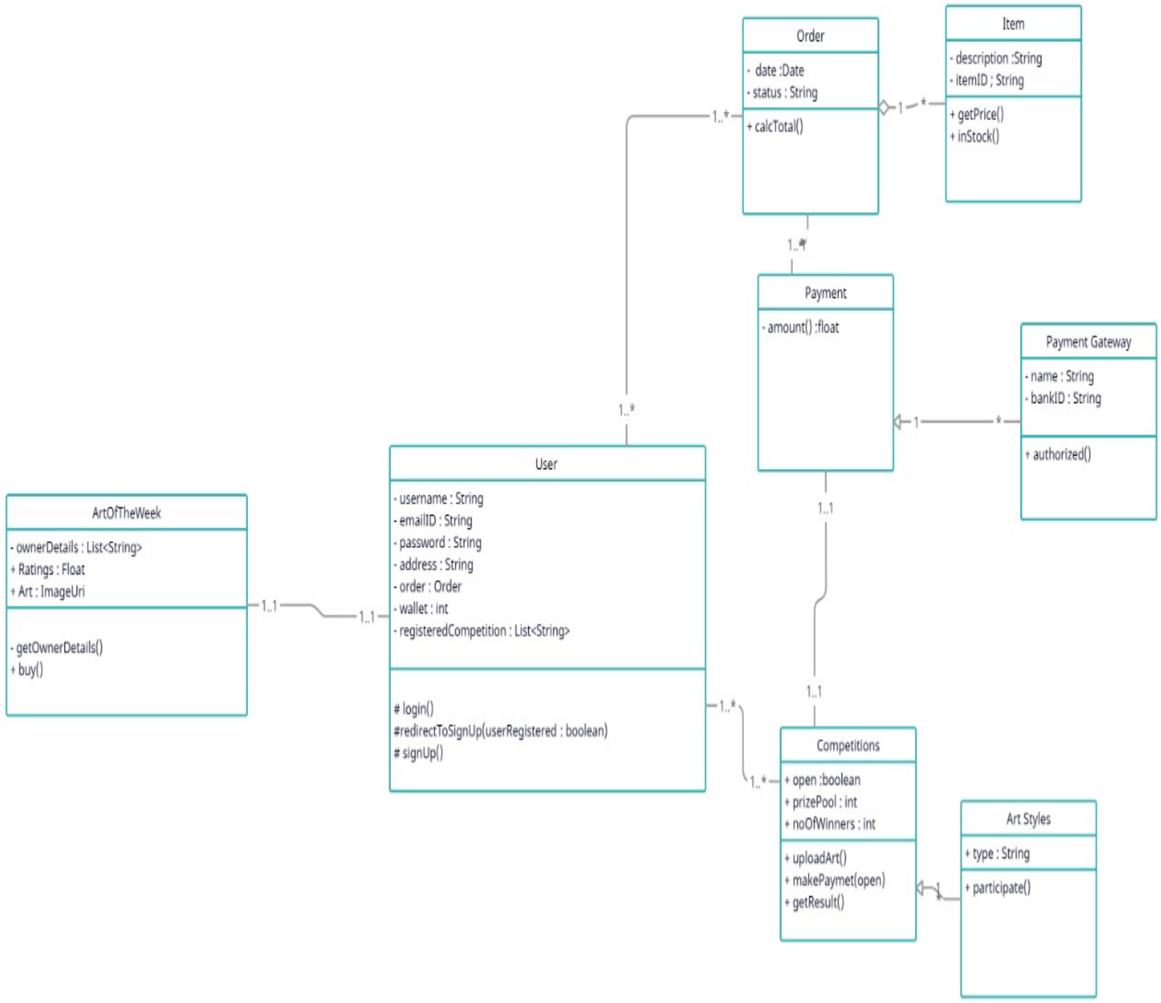
2. Use Case Diagram:



- Basically, a use case diagram shows us the details of how the system's users (Actors) will interact with the system.

- The use cases are the tasks being performed by the system and the actors are those who compel the system to perform these tasks.

3. Class Diagram



- As it is known to us that the class diagram is a static diagram basically it shows us that how the system is implemented with the help of different classes and the specific relations between different classes.

- It also tells us the access specifiers of different variables used in a class and the return type of different methods included in a class

Coding and implementation phase

- **Frontend**

  Frontend of the application implements various features of the android library like the recycler view. Different art styles have their own recycles view tabs each of them has its own fragments. There is a bottom navigation bar that helps us to navigate through the app. The theme of the app is chosen to be white-blue. There are different buttons to help the user access various features of the app like "Art of the week", "My Competitions", etc.

- **Backend**

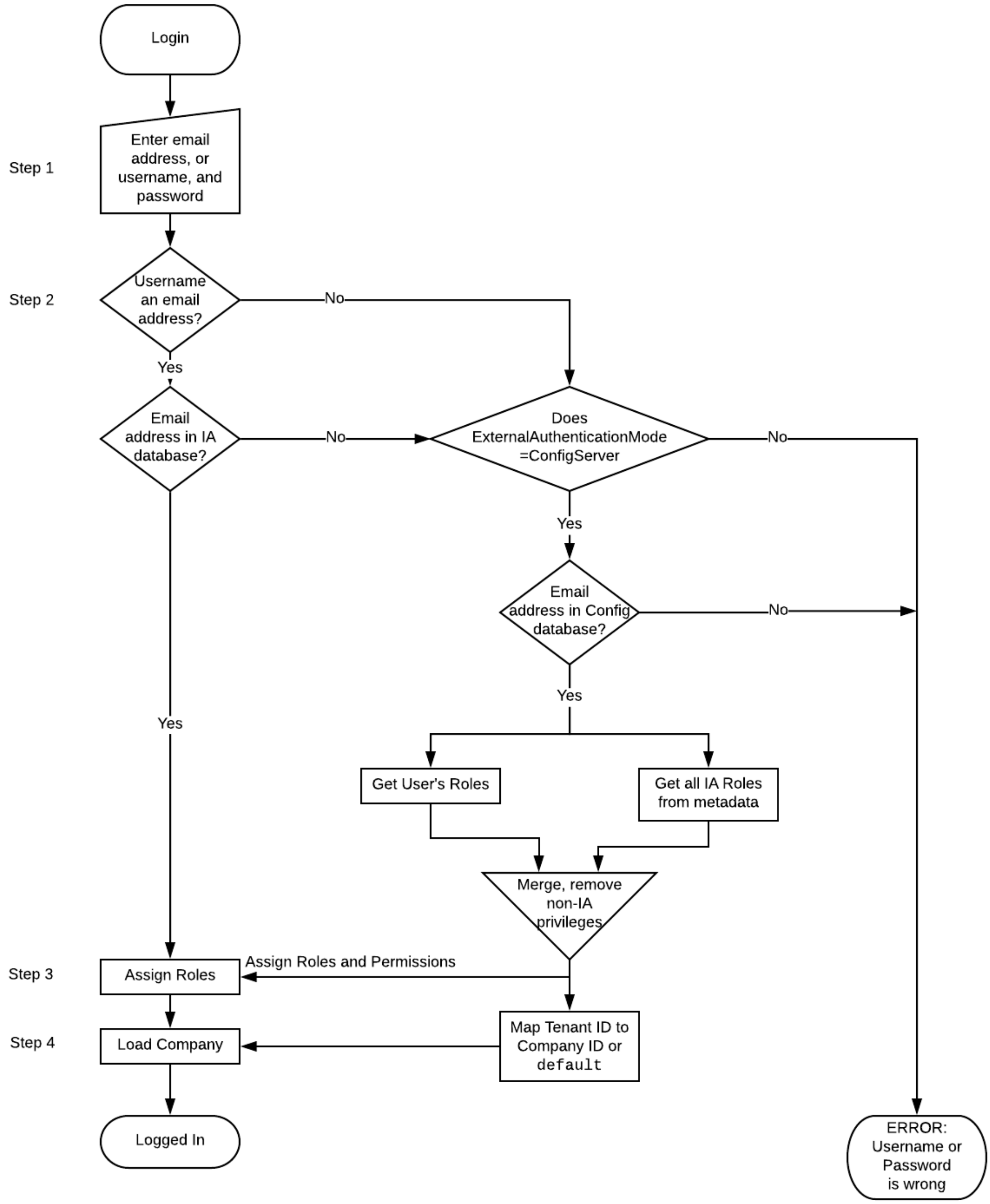
  1. **Authentication:**

Our app uses the authentication provided by firebase authentication. Our app needs to know the identity of a user. Knowing a user's identity allows our app to securely save user data in the cloud and provide the same personalized experience across all the user's devices.

Firebase Authentication provides backend services, easy-to-use SDKs, and ready-made UI libraries to authenticate users to our app. It supports authentication like passwords, phone numbers, and popular federated identity providers like Google, Facebook, Github, and many more.

Firebase Authentication integrates tightly with other Firebase services, and it leverages industry standards like OAuth 2.0 and OpenID Connect, so it can be easily integrated with your custom backend.

We have used the email and password-based authentication provided by firebase. Authenticate users with their email addresses and passwords. The Firebase Authentication SDK provides methods to create and manage users that use their email addresses and passwords to sign in. Firebase Authentication also handles sending password reset emails.


Login
Step 1
Enter email address, or username, and password
Step 2
Username an email address?
No
Yes
Email address in IA database?
No
Does ExternalAuthenticationMode =ConfigServer
No
Yes
Email address in Config database?
No
Yes
Yes
Get User's Roles
Get all IA Roles from metadata
Merge, remove non-IA privileges
Step 3
Assign Roles
Assign Roles and Permissions
Step 4
Load Company
Map Tenant ID to Company ID or default
Logged In
ERROR: Username or Password is wrong

2. Participation:

One of the most important features of our application is participation in various art competitions. Using various classes of each art style, we implemented this feature. We have various methods like PARTICIPATE, MAKEPAYMENT, CANCELPAYMENT, etc.

The android studios environment helps integrate frontend UI with the backend coded in java. All the errors are handled by the try-catch blocks provided by java.

3. Make/Cancel order:

Another important feature, of buying paintings that were part of the competitions. Orders will be shipped to the address of the user and the cost of framing the painting and type of canvas will also be added. An API for making payments is also implemented. Razorpay helps to manage payments for the app.

4. Database:

We have used the Firebase database for storing all the user information and the paintings received in the competitions.

There are two crucial things needed to be handled in the database:

1. Creating models:

Creating models means creating a different types of classes in database, for storing all the kinds of data. We have basically created 8 models:

- User
- cancel order model
- different art styles model(digital art model, color pencils model, etc.)
- my order model
- feedback model
- competition list model
- prize winners model

The code for the same is present in the models folder. This part is explained in complete detail in the class diagram of the design Document.

2. Integrating database to the software:

For this purpose, we have used the firebase database for storing the data
and integrated it into our project,written in java, using Firebase library provided by android studios. We have used the same firebase database for storing paintings and we had basically used

a decentralizing algorithm, whose implementation details are well explained in the design document, for making the storing, render and update the process faster.

5. Deployment:

To deploy our application, the user has to follow the below-provided steps.

- Download the code provided.
- Open the app folder in android studios.
- In the Android menu, go to **Build > Build Bundle(s) / APK (s) > Build APK(s)**.
- Android Studio will start building the APK for you. Once done, a pop-up on the bottom right will notify you of its completion. Click the '**locate**' button in this dialog.
- The 'locate' button should open File Explorer with the debug folder open that contains a file called "app-debug.apk".
- Run the "app-debug.apk" file on ur device and the app is ready to be used.

There is a change of a possible error in the Gradle build of the app src due to the mismatch between the SDK used in the user machine and the SDK of the host machine. Hence an easier way to install is using the below link.

https://drive.google.com/file/d/1IWlxB46GeUuRj65iHp7Gf5hQPBg1023O/view?usp=drivesdk

This link contains the already installed apk file and hence there won't be an issue with the mismatch of SDK.

Integrating frontend, backend, database, and API.

- Our frontend is written using XML.
- The database used is firebase.
- Also, we have made use of an API for payment management namely Razorpay.

For integrating all this, android studios IDE provides a complete environment. The XML files which build up the frontend can be fetched in the backend using the "findViewById" method. This allows us to access each and every element of the frontend screen.

The database is integrated with the use of the firebase library which is imported into the project.

API is also fetched and rendered from amazon AWS's Razorpay mobile API.

Innovation

- Promoting new and small art businesses
- Ease of buying and viewing the art for the art lovers sitting around the globe and also ease of selling an art at its right price while sitting at home.
- A platform to showcase art skills and win exciting prizes by taking part in variety of competitions.
- A very platform for new artists to interact with artists of same art style and get more exposure to grow.
- Ending monopoly of big marketing brands by providing artists with an platform to showcase their skills and to sell it without almost anything deducted from the original money.

Now various artists can show their art to the big online world just by a click and can get the deserved appreciation and reward for it. Similarly, just by a click, you can get a beautiful art form to decorate your house.

Future Scope

In the initial release of our project, we have different art styles, different competitions paid and unpaid, exhibitions, rewards and offers to attract the attention of users. Also our present model replicates most of the features defined earlier. But some of the features of our project can only be implemented once we have a sufficient knowledge. The feature we are trying to implement are:

i) Let's say a user want to buy some painting than for that we are trying to include biding system feature in our application.

ii) With the newer version we will try to add more art styles, find ways to improve the connection between the artists around the globe, more attractive competitions, better rewards etc. we will also try to add a feature that would recommend paintings to different users according to their previous purchases and previous liked paintings.

iii) We are also planning to add feature by which artists can carry out live exhibitions.