Kenneth Romero Linares
Giorgio Wirawan
Alexis Juarez
Sahaf Khan
Chistopher S

Team R

Table of Contents

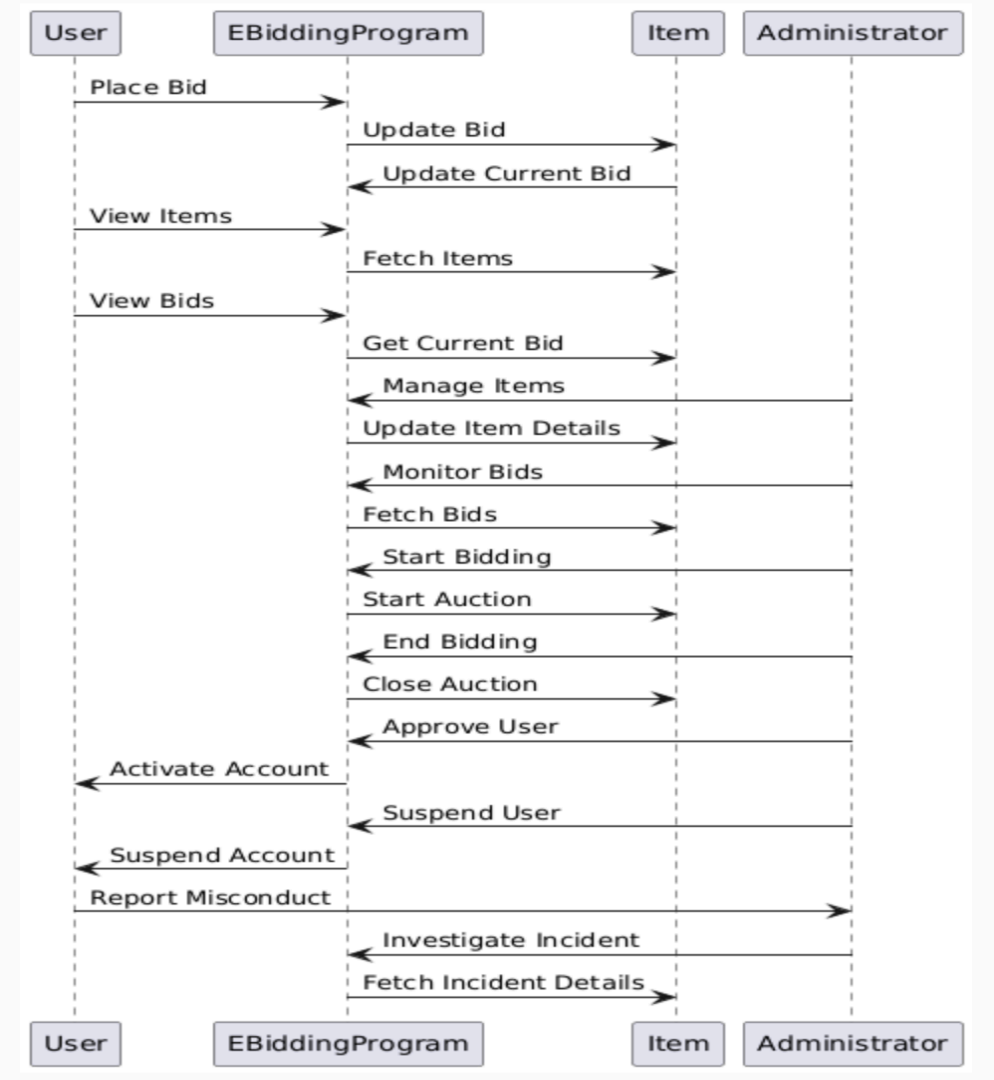
*Pages listed on the top left corner

Preface

This report is a demo of what our final system will contain. In this report, you will find Class diagrams, E/R diagrams, Petri nets, Pseudocode and all the foundations for our final demonstration.

1. Introduction

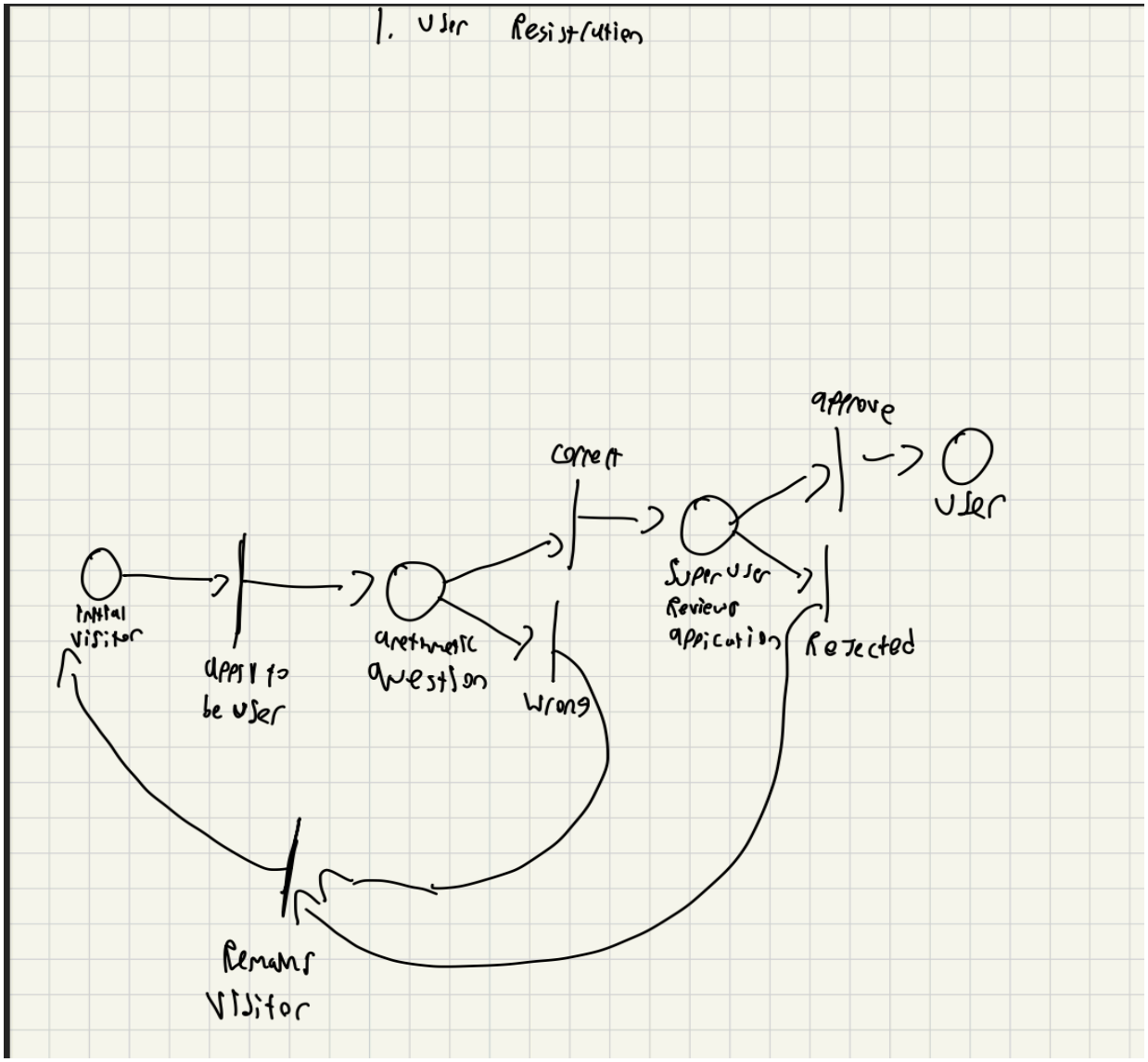
Class diagram of system



2. All use cases in the system

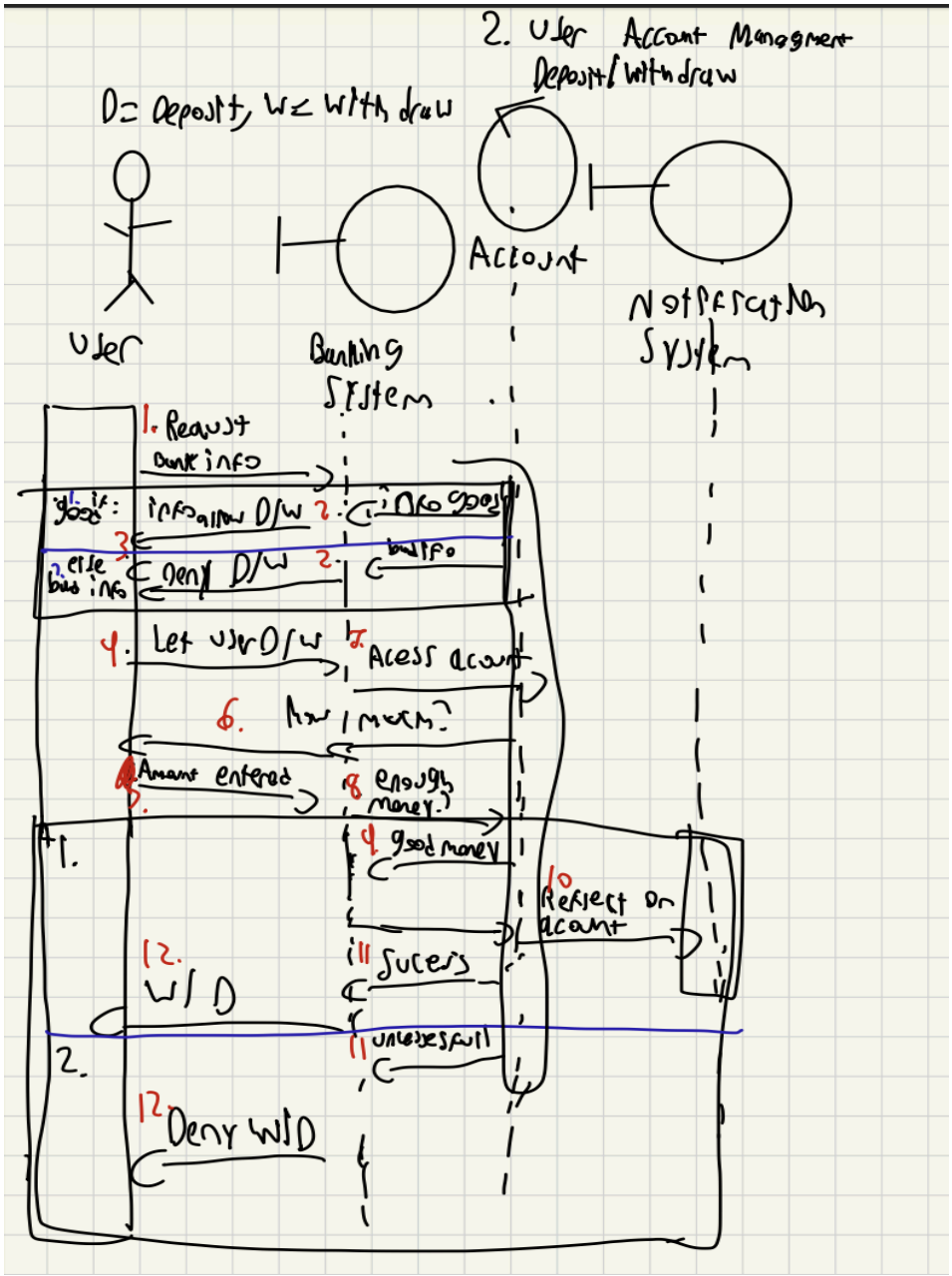
2a) User Registration

The 1st Use case is user registration. The Petri Net below demonstrates a visitor who wants to become a User.



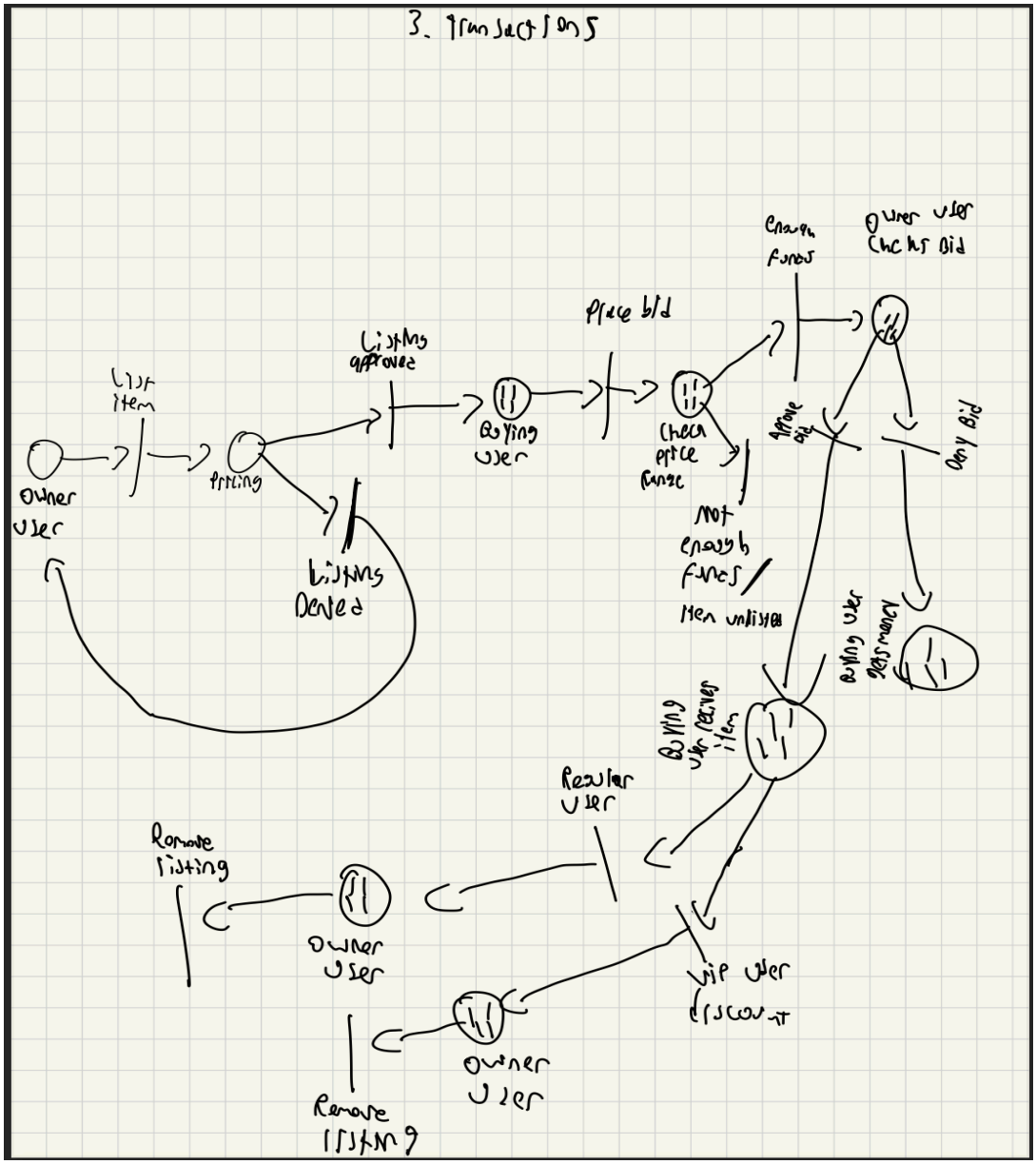
2b) User Account Management

The 2nd use case is User Account management. The sequence class diagram below demonstrates the user wanting to make a deposit/withdrawal and all the transactions that happen in between.



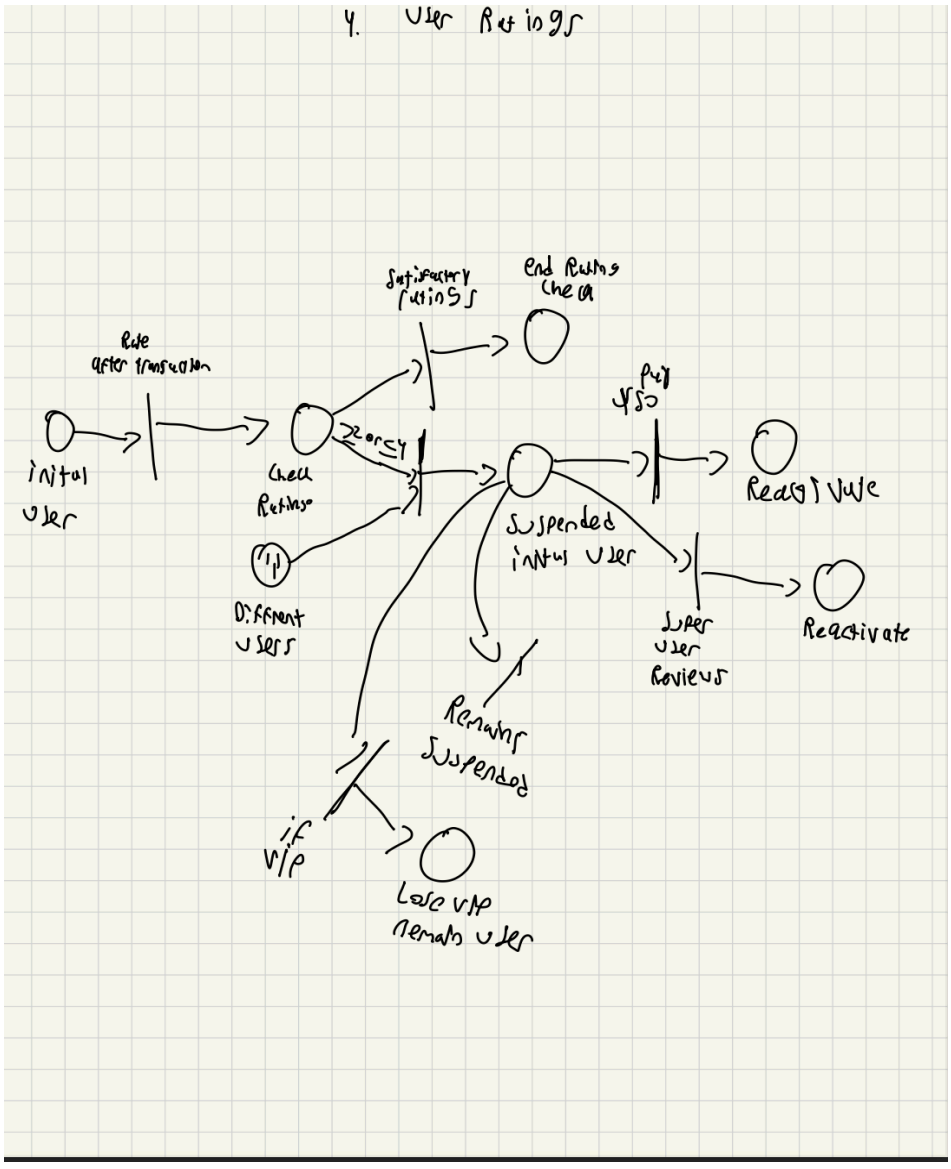
2c) Transactions

The 3rd use case is Transactions. The Petri Net Bellow describes the transactions between an Owner User who wants to list an item and the Buying user who wants to buy the item the Owner User Listed.

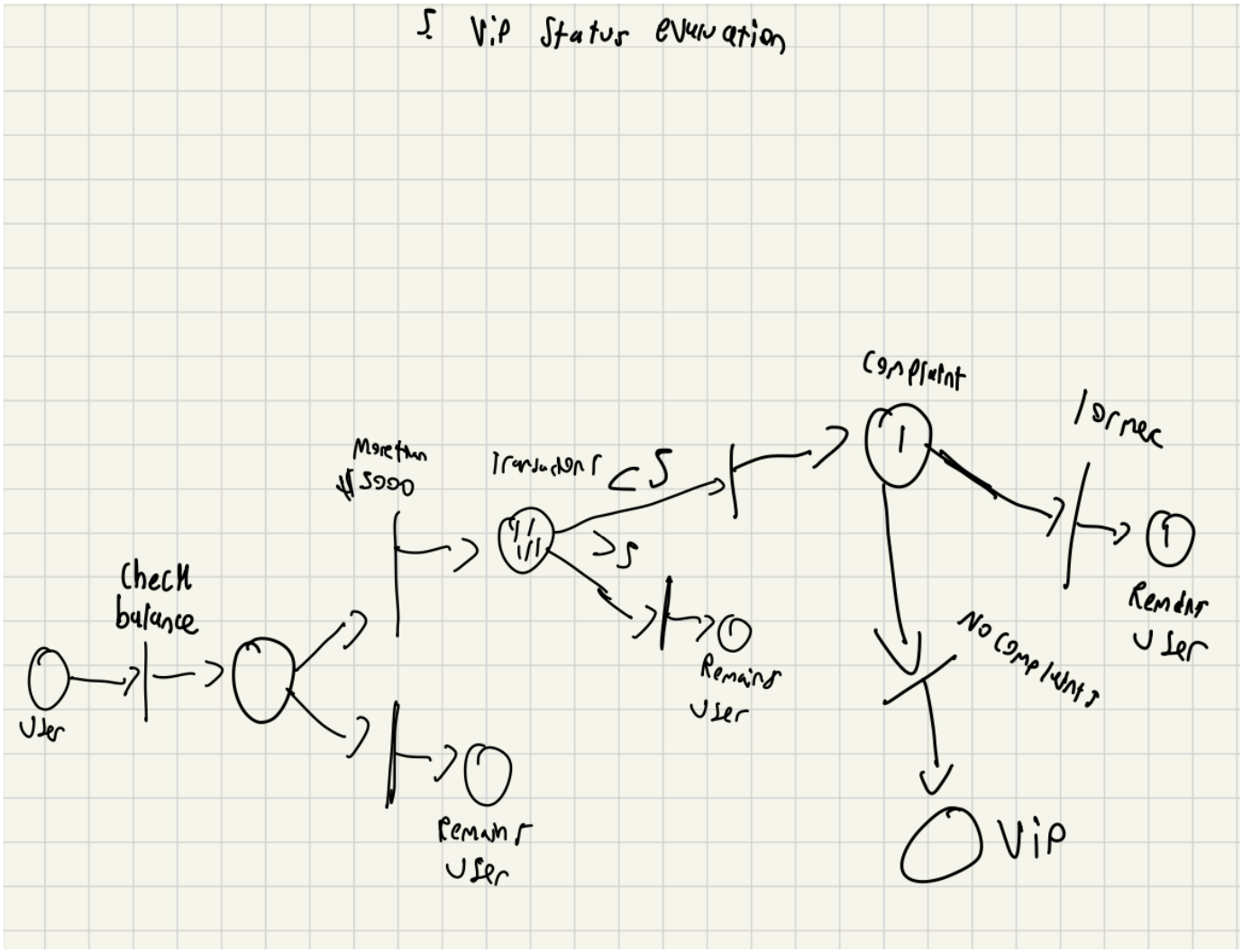


2d) User Ratings

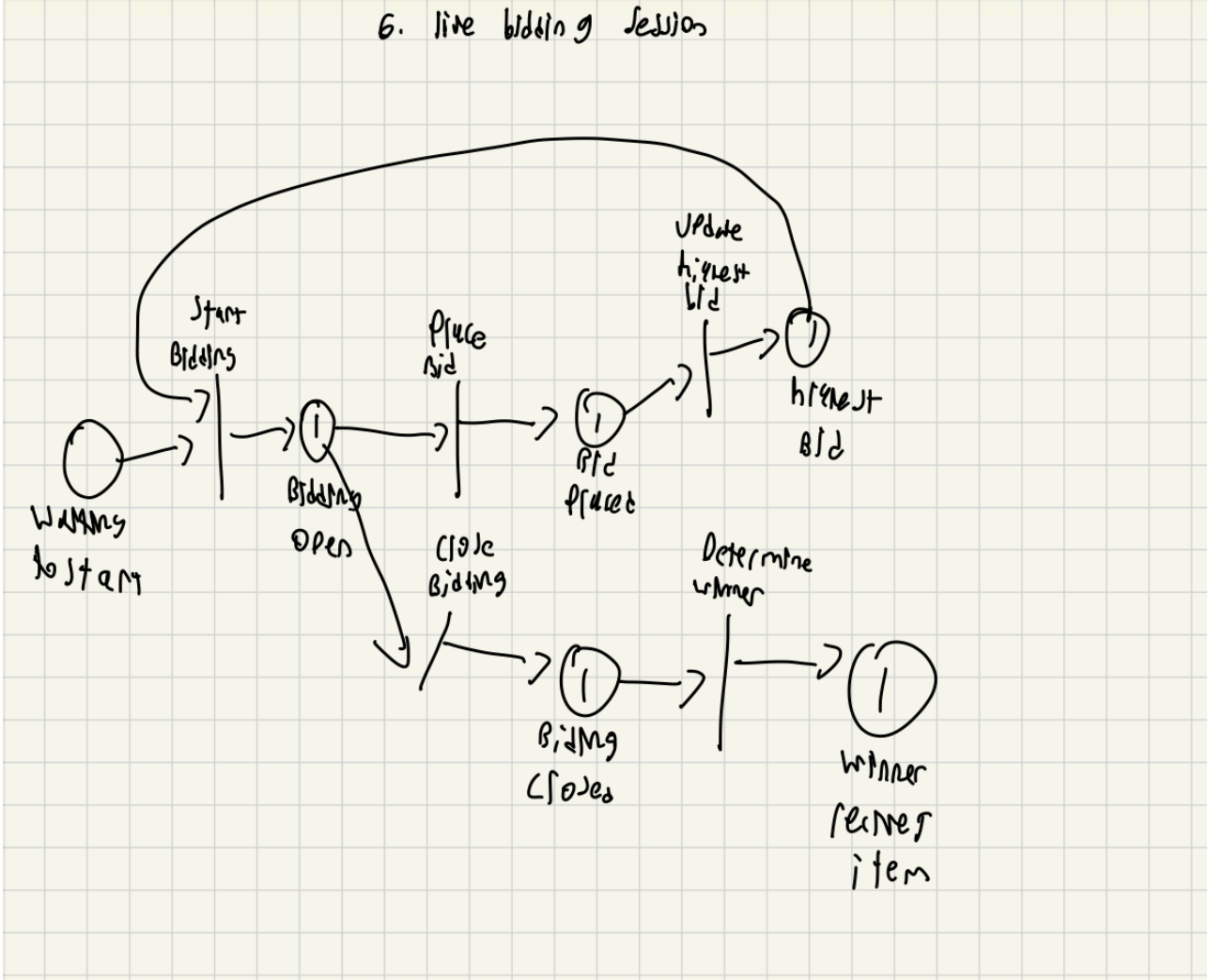
The 4th Use Case is user ratings. The Petri Net Bellow describes the system of when a user rates another user and the possible outcomes that can happen given the users ratings.



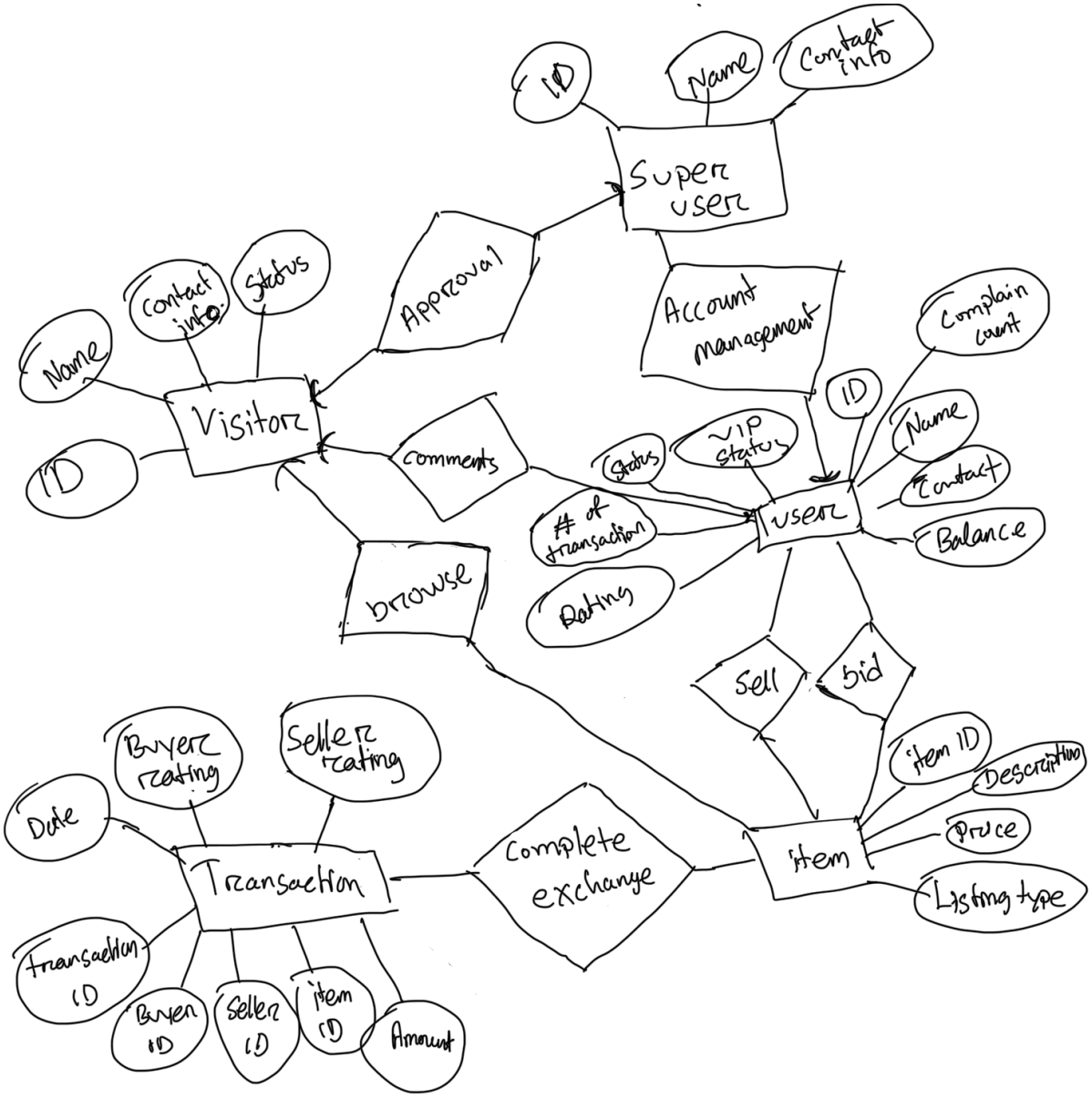
2e)The 5th Use case is Vip status evaluation. The Petri Net Bellow describes the user and their process to become a Vip.



2f) The 6th use case is the live bidding session. The Petri Net below describes the live bidding session and how the Vip users interact in the session.



3. E-R Diagram for system



4. Detailed Design (Pseudo Code)

Pseudo code for functionalities

//Basic Pseudo Code on how the functions inside the program will work

```
Function createAccount(username, password, email):
        If username, password and email are not valid
                return string “Invalid input”
        If username already exists in database
                return string “Username is already taken”
        Else
                Create a new user with arguments username, password and email
                Save the user to the database
                // May add a feature to send for confirmation email
return string “Account was successfully created!”

Function logTransaction(user, transactionType, amount, status):
   Log transaction details in transaction history table for auditing
   Return None

Function lockItem(itemId):
   Lock the item to prevent concurrent actions on it
   Return True if lock is successful, False otherwise

Function unlockItem(itemId):
   Unlock the item after action is complete
   Return True if unlock is successful, False otherwise


Function retrivalBalance(user):
        Check the data base to get user balance
        Return balance or error depending on if the data base is successful or not

Function visitorToUser(username):
//User would click on a button to apply for User Status

Once button is clicked:
        Arethmetic question is showed
        If correct:
                Request gets sent to Super user
                Return “Request sent to super user for approval”
        Else:
                print ( Wrong, your account remains visitor)
```

```
Function item description(itemId):
        If itemId is not valid
                return string “Invalid item”
        Else
                retrieve the item data from database and store it some variable
                Display item data on the webpage
return string “Item displayed successfully!”


Function Deposit(user, amount):
        If user is invalid
                Return “Invalid user”
        If amount<=0
                Return “Invalid deposit amount”
        balance=retrieveBalance(user)
        balance=balanace+amount
        Update the data base with the new balance value
return “Deposit successful, current balance is “ + balance

Function Withdraw(user, amount):
        If user is invalid
                Return “Invalid user”
        If amount<=0
                Return  “Invalid withdrawl amount”
        balance=retrieveBalance(user)
                If balance<amount:
                        Return an error msg
        balance=balance-amount
        Update the new balance in data base
return “Withdraw successful, current balance is “ +balance

Function viewBalance(user):
        balance=retrieveBalance(user)
        Return balance

Function makeBid(amount, itemId,balance,priceRange)
        //User would click a button to bid on the item
If lockItem(itemId) is false:
Return error, item is currently locked
        Balance=retrieveBalance(user)
        priceRange=getPricerange(itemID)
        If Balance>=PriceRange:
                Place the bid
```

```
            notifyItemOwner(user)
            unlockItem(itemId)
            Return: Bid was placed successfully
        Else:
            unlockItem(itemId)
            Return Not enough funds in your account to buy this item.
```

```
Function Rate User(username,rating,Vip)
    //After a transaction the user can give the other user a rating
    //First check the ratings the user gives
    If usernameRating>2 and <4:
        print("Ratings Satisfactory")
    Else:
        If username a vip:
            Remove vip
            Return username and eliminate Vip status, reamians user
        Else:
            While user has not paid $50 or super User reactivates account
                User suspended
```

```
Function applyForVip(user):
    //This function will allow user to become a VIP
    //USer will click on button "Apply for Vip"
    balance=retrieveBalance(user)
    transactionCount=get the transaction count from user
    rating=get the average rating from user
    Complaint count=get the complaint count from user
    //Once button is clicked to apply for vip
        If balance<5000:
            Return Not able for vip at this time: Reason: Balance
        If transactionCount<5
            Return Not able for vip at this time: Reason :not enough transactions
        If rating>4 and rating<2 and username complainCount>0:
            Return Not able for vip at this time, Reason:acount not in good standing
guidelines, file complaint here( user can complain to super user to appeal)

            //Once all the if statements are passed the user will be able to be VIP
            Return: Congrats you are now a VIP!!
```

```
Function liveBid(username,balance,Vip)
//This is for when a user bids on item
highestBid=0
While Time>0 or Bid not accepted:
        If bid Button is clicked:
                If balance>currentBid:
                        currBid=UsersBid
                        highestBid=currBid
                        Balance=Balance-highestBid
                Else:
                        Alert the user saying not enough Balance
        Elif bid button is clicked by someone else when you bid for it:
                balance=balance+highestBid
                highestBid=the other users current Bid
        Else;
                highestBid=Other users current bid
unlockItem(itemId)
Complete the transaction with the person who bid the most
Return The bid has ended and (put the user of person who won) won the bid.

//This is fundamental,there is going to be more complex layers added to this part but this is just the basics.
```

Note: Functions are subject to change.

5. System screens Demo: The main page took some inspiration from eBay. We will also have unique features such as account balance, deposit, and withdrawal.

Hello! Sign in or Register Deposit Withdraw Balance: $100.00 Sell Watchlist

E-Bid

Search for anything | Filter | Search

Categories

- Trading cards
- Cars
- Car Parts
- Computers
- Comics
- Collectibles

LIVE NOW

PSA 10 Charizard-Holo #4 (1st Edition)

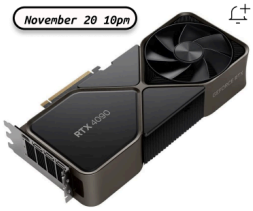
Honda Civic Typer R 2020

iPhone 16 Pro 128Gb Unlocked

Lego Death Star Item: #75159

Next Up

November 20 10pm



NVIDIA GeForce RTX 4090 Founders Edition

November 21 9pm

Samsung Galaxy S24 Ultra 12GB/1TB Black

November 21 10pm

PSA 10 Blastoise-Holo #2(1st Edition)

November 22 1pm

Jordan 11 Space Jam Size 11

E-Bid

Login in to E-Bid

Username or Email Address:

Password:

I'm not a robot reCAPTCHA Privacy - Terms

Log In

Forgot your password?

OR

Create an Account!

6. Memo:

The purpose of this memo is to inform Professor Jie Wei about the final project and how our group has collaborated and worked on it so far. We started off our meeting by deciding what would be the best way to approach this project. We decided to make a website since most of us feel comfortable making one and have previous experience. We then decided on the programming language. This took some time since we are restricted in this category, and some of us haven't learned Javascript or CSS. However, to overcome this issue we will use HTML and CSS for the front end and Javascript for the backend that way everyone is contributing. After this, we discussed ways to prevent others from not contributing and splitting the work equally. In order to provide a fair environment for all we will make sure to remind everyone of their task and roles in this group and the importance of doing their duties as part of this group. As a penalty for not obliging to their duties although we can not remove them from our team, it will tarnish the student's reputation and for future courses, many will not want to team up with them. After discussing this we split the workload of the second report and decided to get most of the report done before November 12. This way we can meet after class and do some finishing touches and talk about any concerns we have. We ended the meeting by deciding to start on making the website on November 15th to get ahead of Finals month and allow everyone to focus on this final project.

7. Github/Git Repo link: https://github.com/alexisj890/FinalProject_323