Sprint 1

All Pay Auction
CSCE 606 - Spring 2025

January 31, 2025

1 Team Roles

- Product owner: Jason Le
- Scrum master: Praewa Pitiphat
- Developers:
  - Arkan Abuyazid
  - Ruvail Shahzad
  - Steve Wang

---

2 Customer Meetings

2.1 Date/Time/Place

The meeting will be held through zoom at 2PM on 30/1/2025.

2.2 Summary

The main customer need for the All_Pay_Auction application is to implement an all-pay auction system where each bidder pays their bid regardless of winning. This system is designed to ensure fair competition and provide sellers with a reliable method to auction items. The application will feature functionalities such as real-time auction tracking, automated payment processing, and user authentication for both sellers and buyers. By utilizing technologies like Rails 7.2, Ruby 3.3, Hotwire, Stimulus, and PSQL, the application aims to deliver a seamless and intuitive experience for users, while ensuring data integrity and security.

The primary stakeholders for this application include sellers who want to auction their items, buyers who participate in the auctions, and the development team responsible for creating and maintaining the system. Sellers will benefit from the ability to create and manage auctions, track bids, and receive payments automatically. Buyers will have a user-friendly interface to place bids and monitor their status in the auction. Additionally, the development team will ensure that the application is hosted either on Heroku or locally, depending on the budget, and will use tools like Git and Docker for version control and deployment. Overall, the application aims to meet the needs of all stakeholders by providing a robust and efficient all-pay auction platform.

---

3 Sprint backlog

3.1 What is the sprint goal?

The sprint goal is to ensure that buyers and sellers are able to do the following:

- both buyers and sellers arrive at the same landing page
- both buyers and sellers are able to create their own accounts

3.2 Which stories were pulled into the Sprint?

The stories that were pulled into this sprint are the following:

1. Feature: Landing Page
   As a user,
   I want to create an account as either a buyer or seller
   so that I can interact with the auctions
2. Feature: Buyer's Login Page
   As a buyer,
   I want to log in,
   so that I can participate in auctions.
3. Feature: Seller's Login Page
   As a seller,
   I want to log in,
   so that I can set the auctions.
4. Feature: Buyer's Account Creation Page
   As a new buyer,
   I want to create a buyers account,
   so that I can login and view auction items to purchase.
5. Feature: Seller's Account Creation Page
   As a new seller,
   I want to create an seller's account,
   so that I can sell items.
6. Feature: Database Creation
   As a platform administrator,
   I want to create and maintain a reliable database that securely stores buyer and seller information
   so that users can safely register, log in, and manage their accounts.

3.3 How many points?

1. 0.5 points
2. 1.0 points
3. 1.0 points
4. 1.0 points
5. 1.0 points
6. 1.5 points

3.4 To whom is each story assigned?

1. Jason Le
2. Steve Wang
3. Steve Wang
4. Ruvail Shahzad
5. Ruvail Shahzad
6. Arkan Abuyazid

3.5 What are the tasks and their time estimates?

1. Tasks:
    - Create basic UI (approx. 30 minuites)
    - Implement page navigation (approx. 30 minuites)
    - Buttons for creating account and logging in (approx. 30 minuites)
2. Tasks:
    - Create basic UI (approx. 30 minuites)
    - Implement page navigation (approx. 30 minuites)
    - Confirm user/password exists (approx. 30 minuites)
3. Tasks:
    - Create basic UI (approx. 30 minuites)
    - Implement page navigation (approx. 30 minuites)
    - Confirm user/password exists (approx. 30 minuites)
4. Tasks:
    - Create basic UI (approx. 30 minuites)
    - Implement page navigation (approx. 30 minuites)
    - Update database with account information (approx. 30 minuites)
5. Tasks:
    - Create basic UI (approx. 30 minuites)
    - Implement page navigation (approx. 30 minuites)
    - Update database with account information (approx. 30 minuites)
6. Tasks:
    - Draw an E-R Diagram (approx. 1 hour)
    - Write schemas to set up the database (approx. 3 hour)
    - Write the seed script to population the database (approx. 3 hour)
    - Create unit tests (approx. 3 hours)

---

4 User Stories

1. Feature: Seller Creates an Auction
   As a seller,
   I want to create an auction with a closing date,
   so that I can start accepting bids for my item.

2. Feature: User Places a Bid
   As a user,
   I want to place a bid in the auction,
   so that I have a chance to win the item.

3. Feature: Seller Settles the Auction
   As a seller,
   I want to settle the auction after the closing date,
   so that I can determine the winner and collect payments from all bidders.

4. Feature: User Views Auction Status
   As a user,
   I want to view the current status of the auction,
   so that I can decide whether to place a bid or increase my bid.

5. Feature: Seller Views Auction Results
   As a seller,
   I want to view the results of the auction after it closes,
   so that I can see who won and how much was collected.

6. Feature: User's Login page
   As a user,
   I want to log in,
   so that I can participate in auctions.

7. Feature: Seller's Login page
   As a seller,
   I want to log in,
   so that I can set the auctions.

8. Feature: Seller interface
   As a seller,
   I want to create an auction for an item (with description) that has minimum bid, minimum bid increment, and the time range (1 to 30),
   so that the bid is active.

---

5 User interface (Lofi-sketch) and storyboard

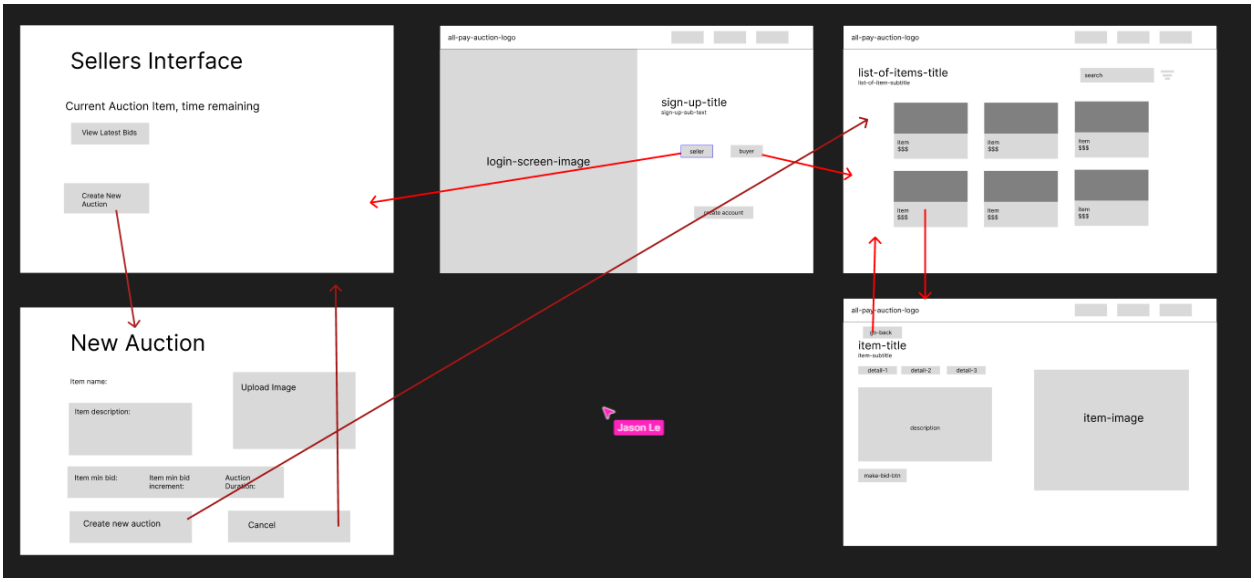

Figure 1: Story board of the auction system

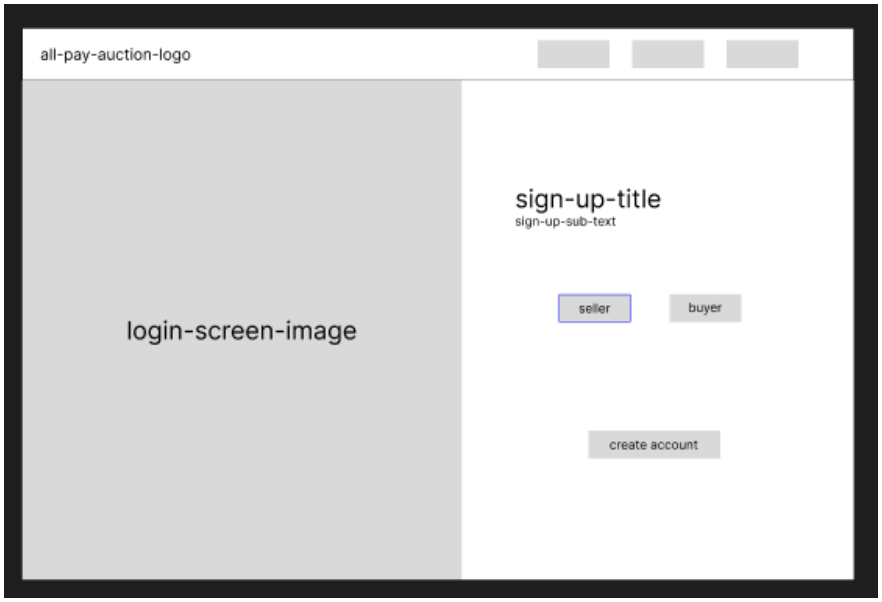

Figure 2: Login page

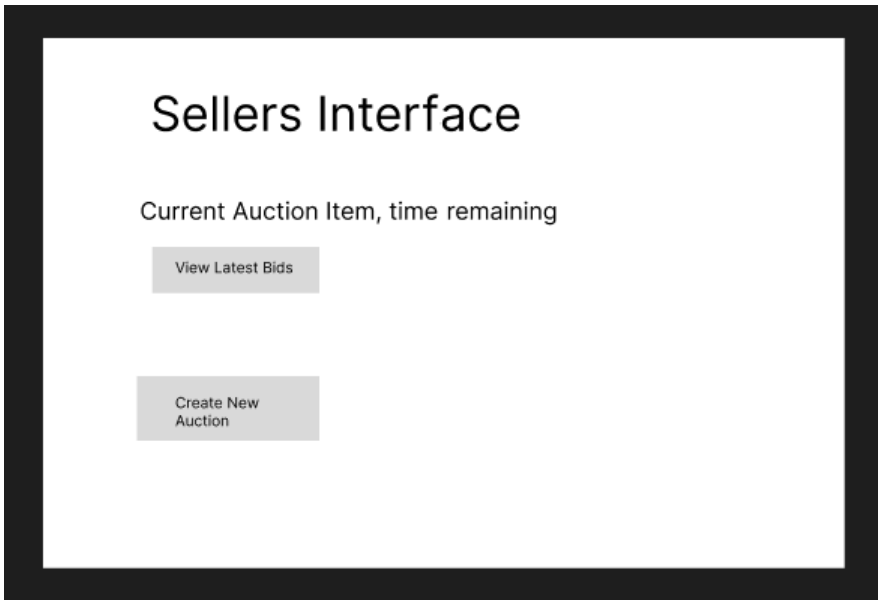

Figure 3: Seller interface

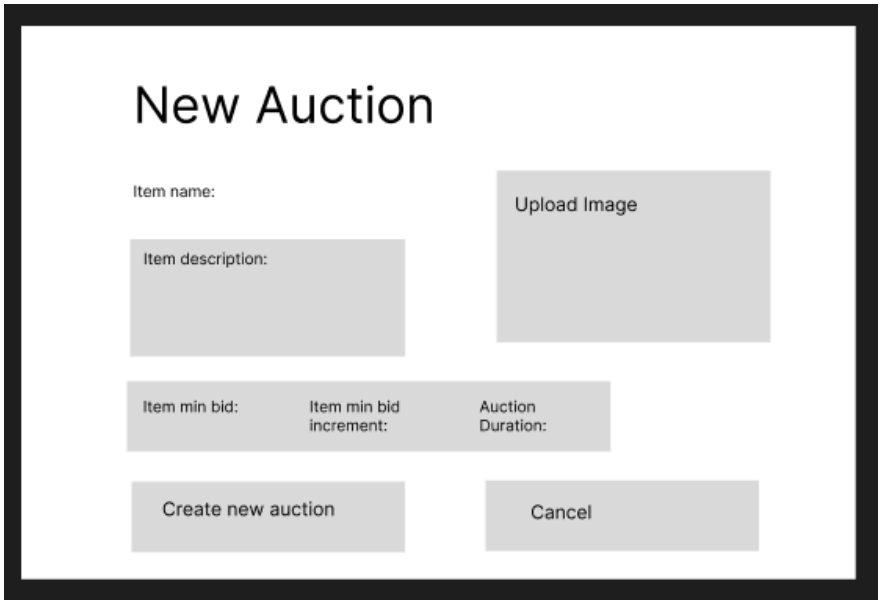

Figure 4: Create new auction

Figure 5: Auction item page

Figure 6: Specific auction item page

---

6 Links

- GitHub Repo
- Discord

Note:

- The team has permission from Professor Ritchey to use Discord instead of Slack.

- Since pivotal tracker is no longer available, our team use GitHub Projects instead.
