
ICEBERG

Blue Penguin – E-bidding Application For Bidding Subsystem

Version 2.0

Blue Penguin – E-bidding Application	Version: 2.0
Software Requirement Specification	Date: 11/12/24
Second Draft	

Revision History

Date	Version	Description	Author
09/11/24	2.0	Added Title page, Revision History page, and Table of Contents	Addina Rahaman
10/11/24	2.1	Added Collaboration class diagram (1), git repo (7), and ER diagram (3)	Addina Rahaman & Samiul Saimon
11/11/24	2.2	Added memos (6) and pseudocode (4)	Lily Minchala
12/11/24	2.3	Added use-case diagrams (2) and System screens (5)	Addina Rahaman, Lily Minchala, Samiul Saimon

Blue Penguin – E-bidding Application	Version: 2.0
Software Requirement Specification	Date: 09/11/24
Second Draft	

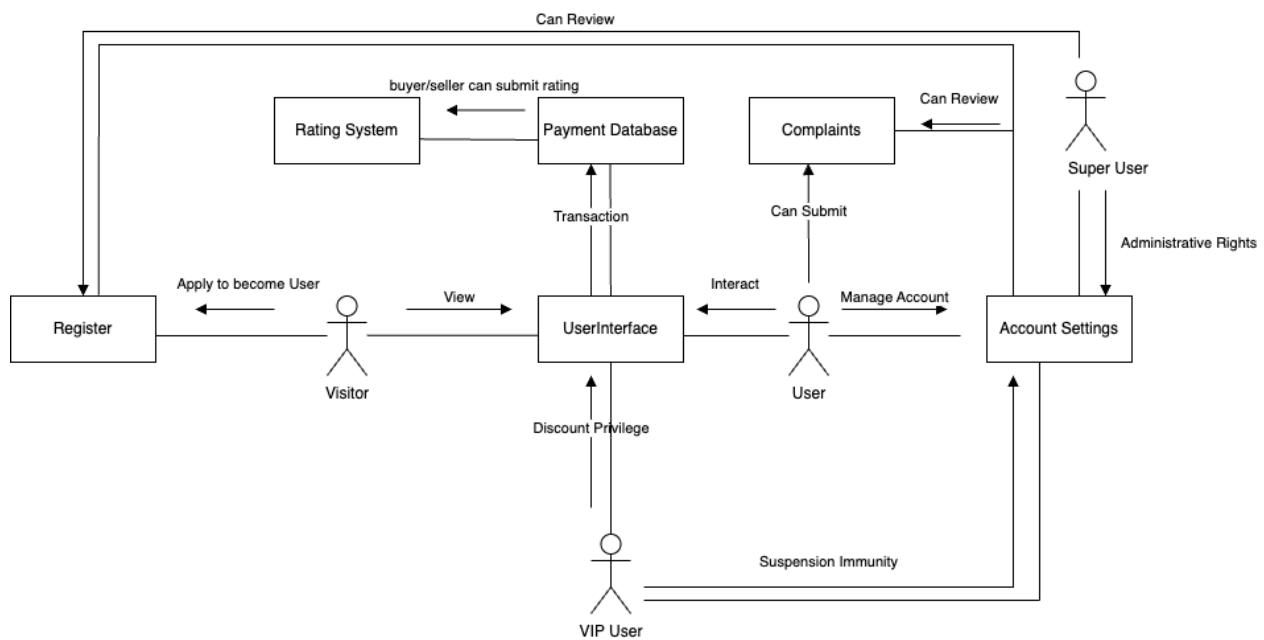
Table of Contents

1. Introduction	4
2. All Use Cases	5
3. ER Diagram for Entire System	17
4. Detailed Design	18
5. System Screens	24
6. Memos	42
7. GitHub Link	44

Blue Penguin – E-bidding Application	Version: 2.1
Software Requirement Specification	Date: 10/11/24
Second Draft	

1. Introduction

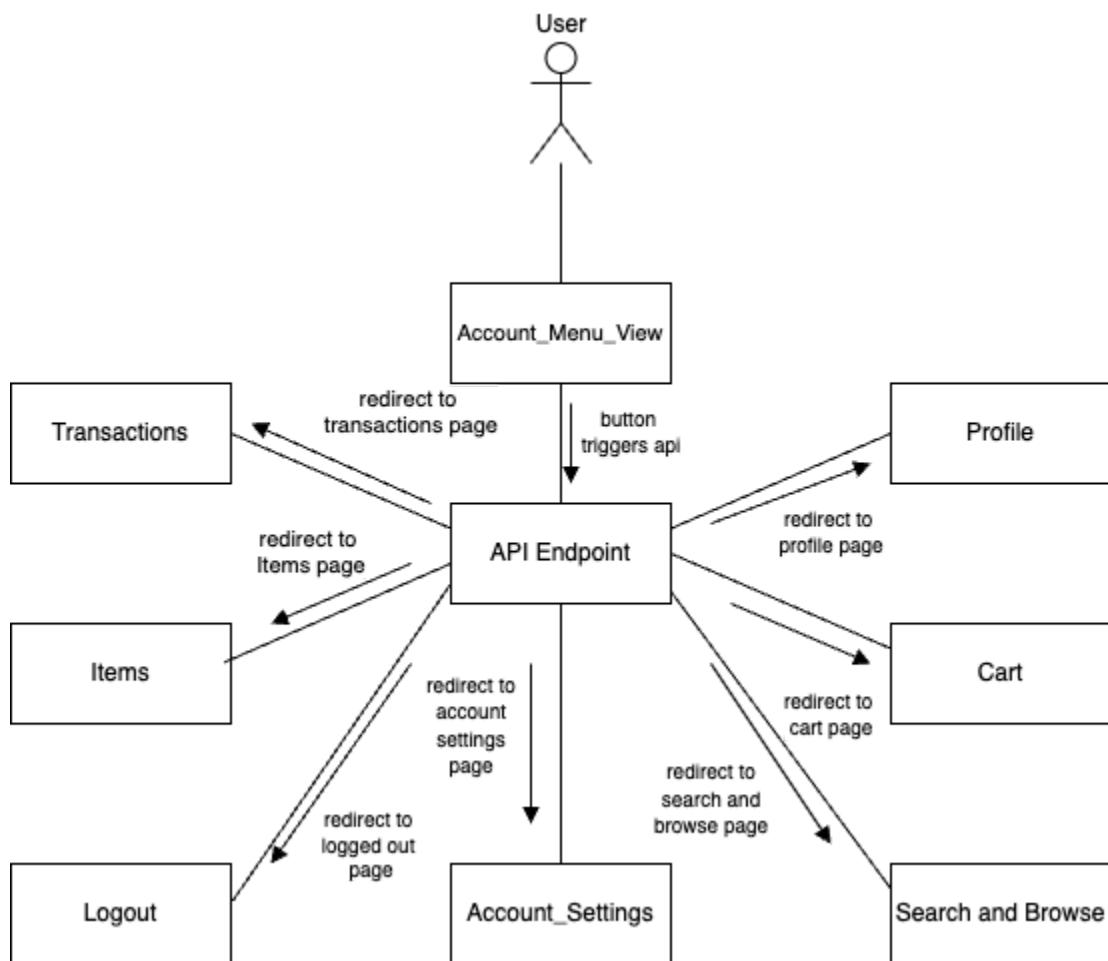
Overall Picture of Class Collaboration Diagram:



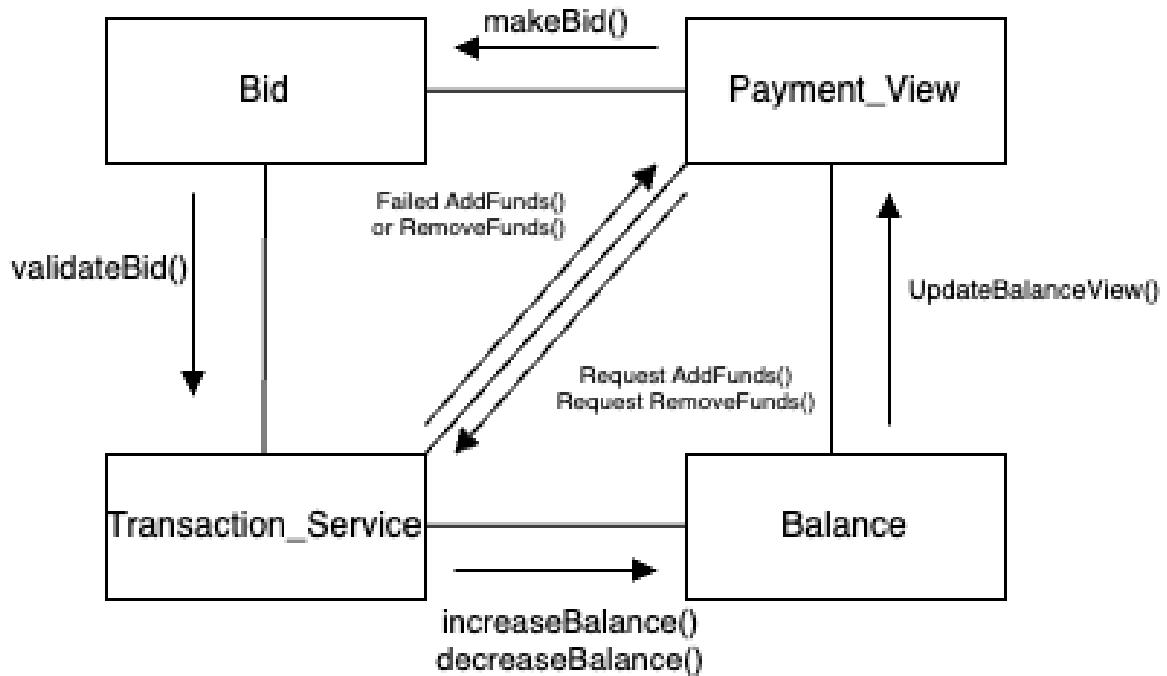
Blue Penguin – E-bidding Application	Version: 2.3
Software Requirement Specification	Date: 12/11/24
Second Draft	

2. ALL USES CASES

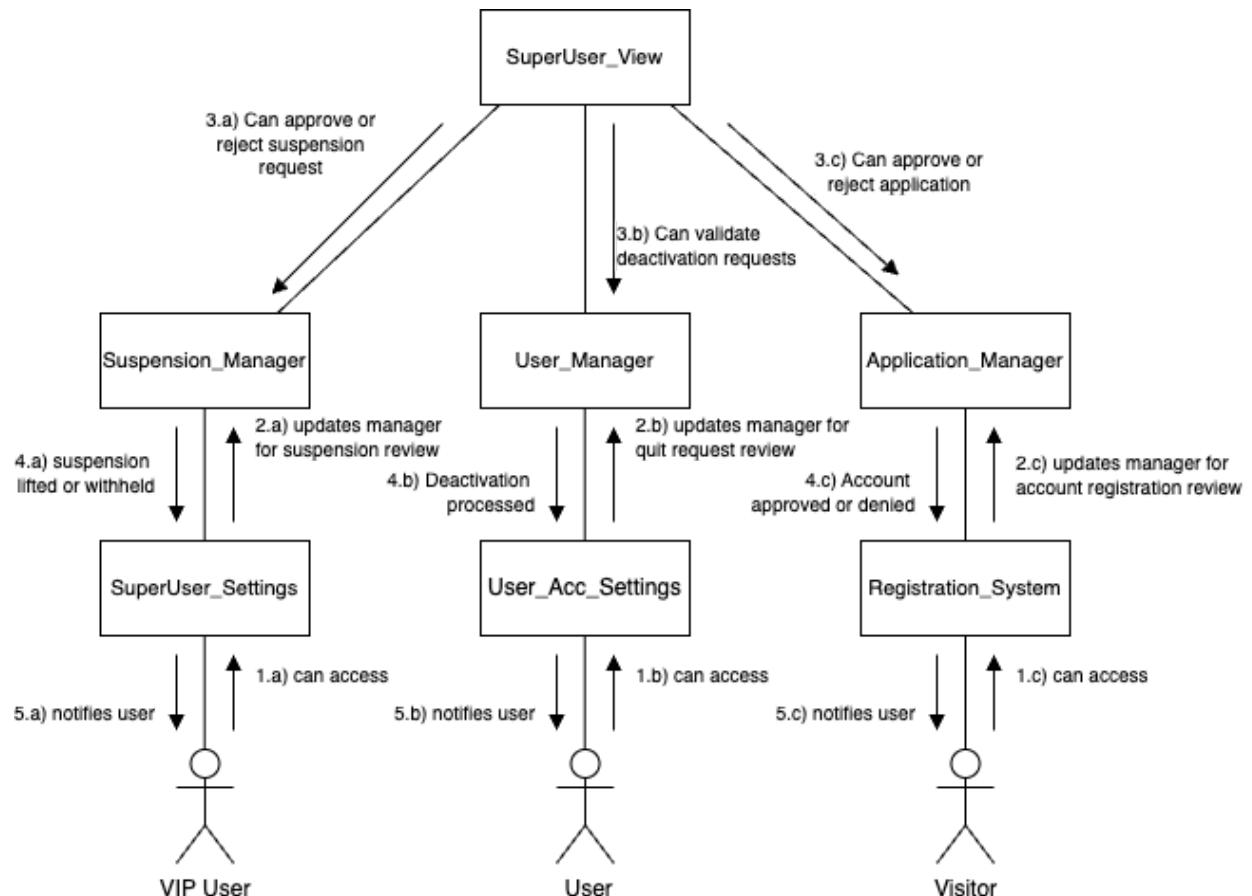
1. Account Menu View:



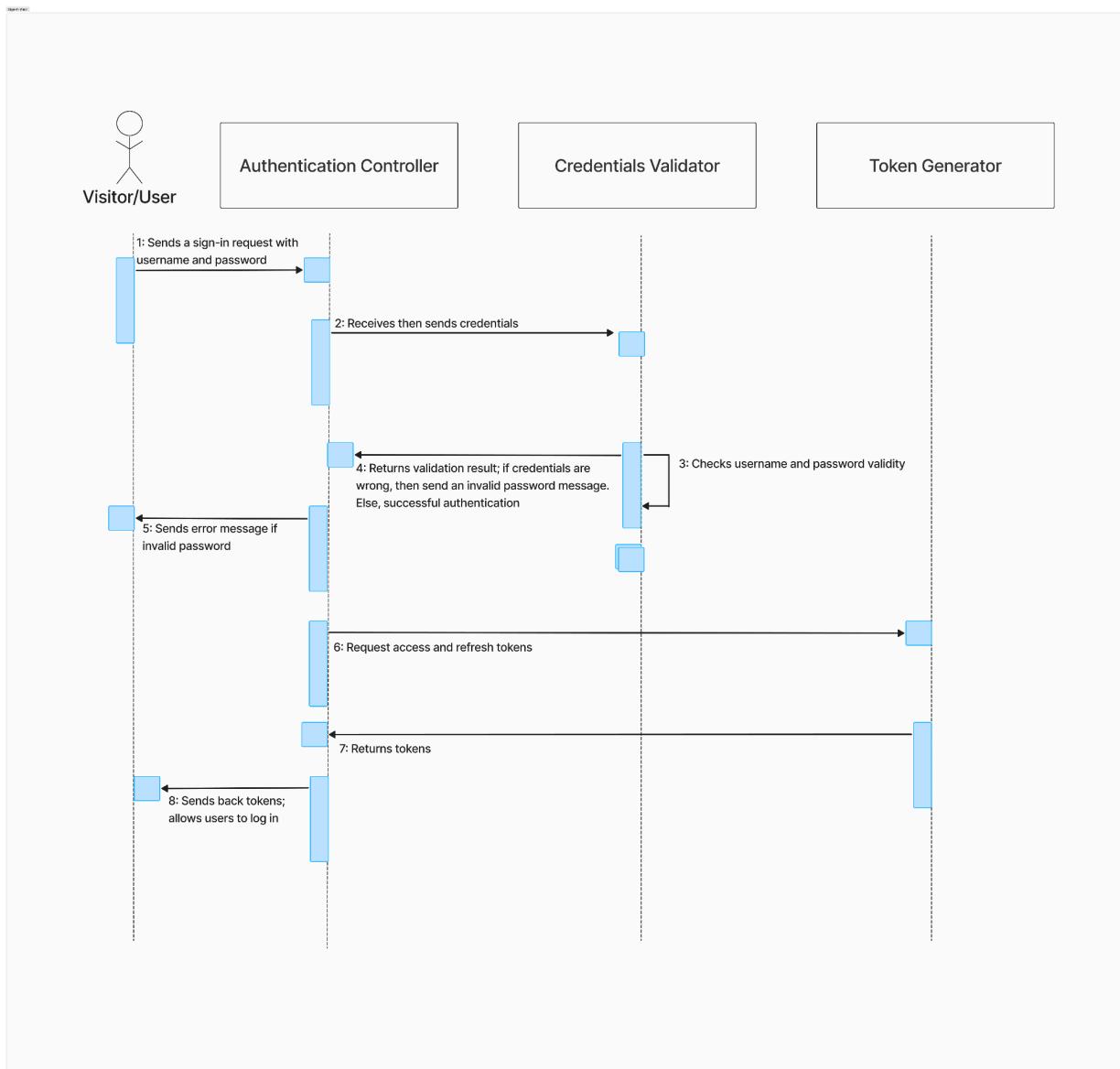
2. Payment View:



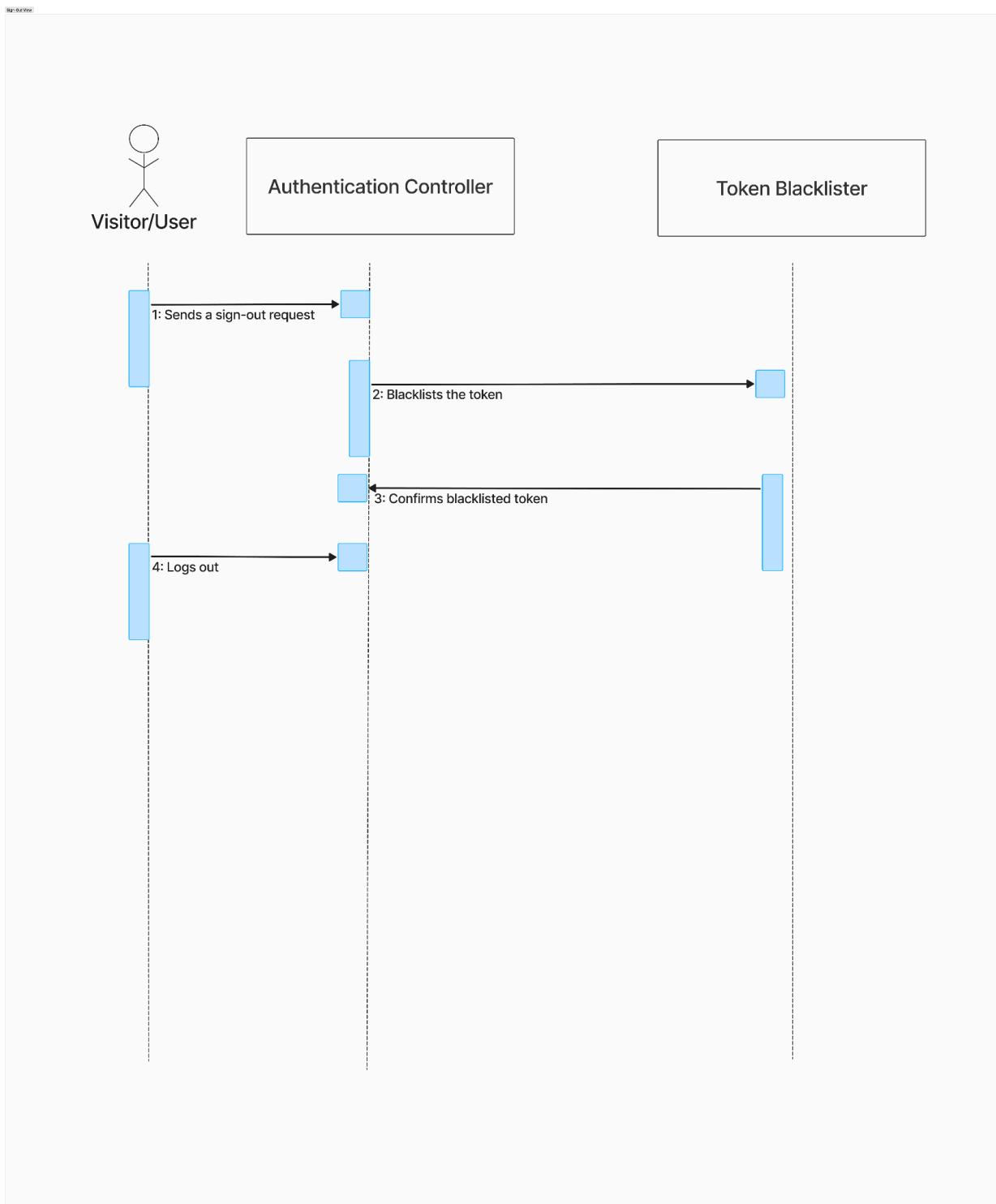
3. SuperUser View:



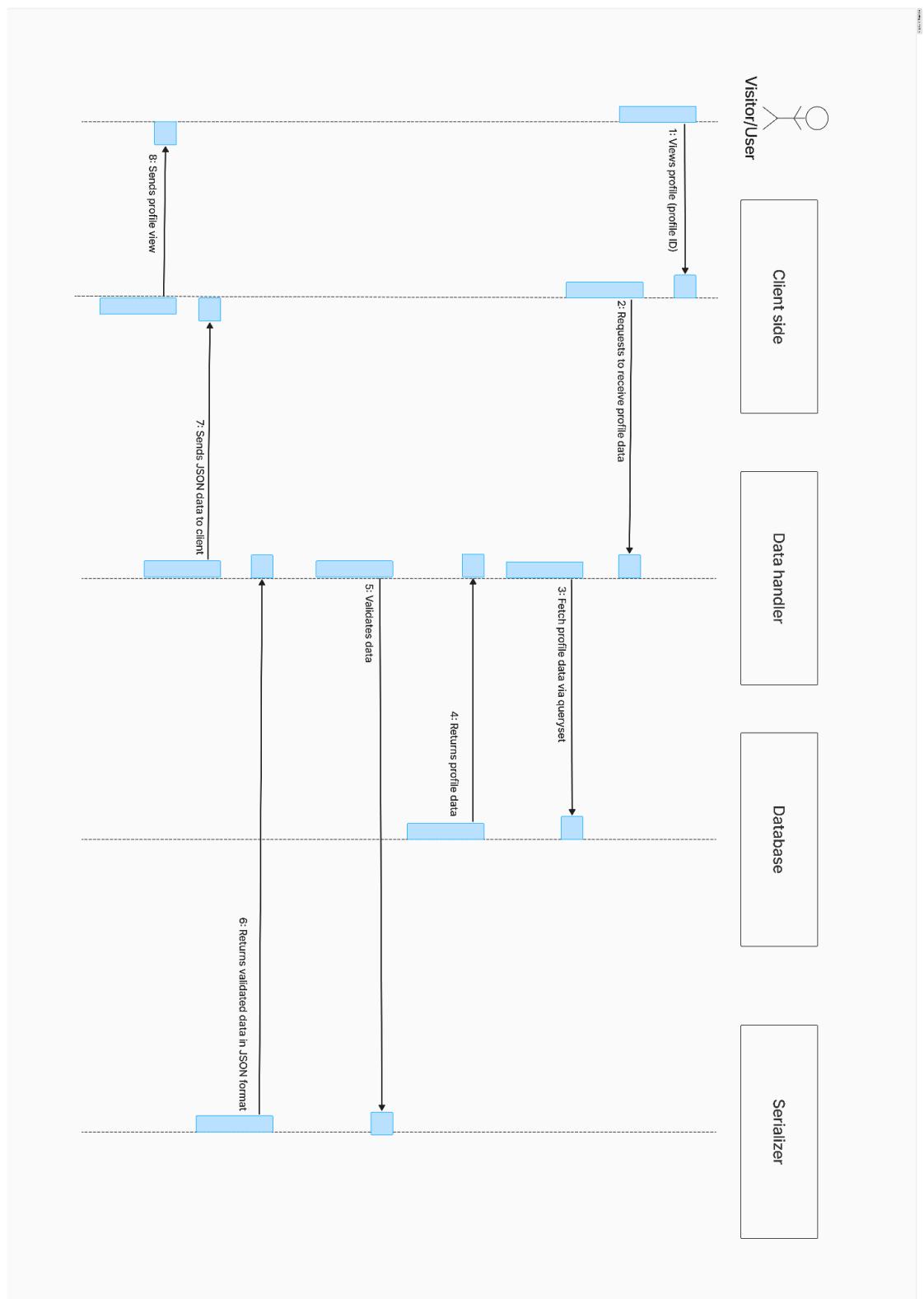
4. Signing In



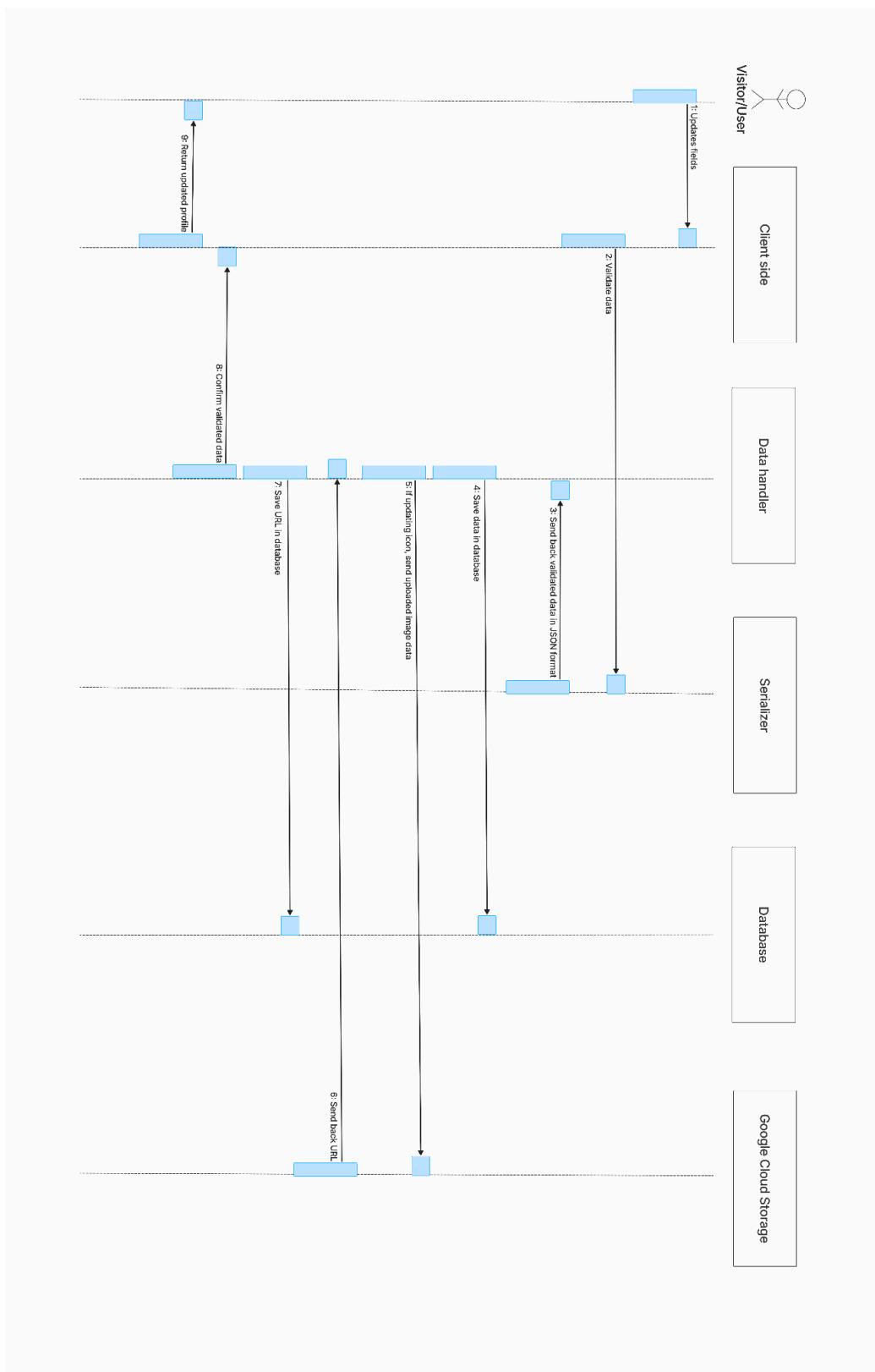
5. Signing Out



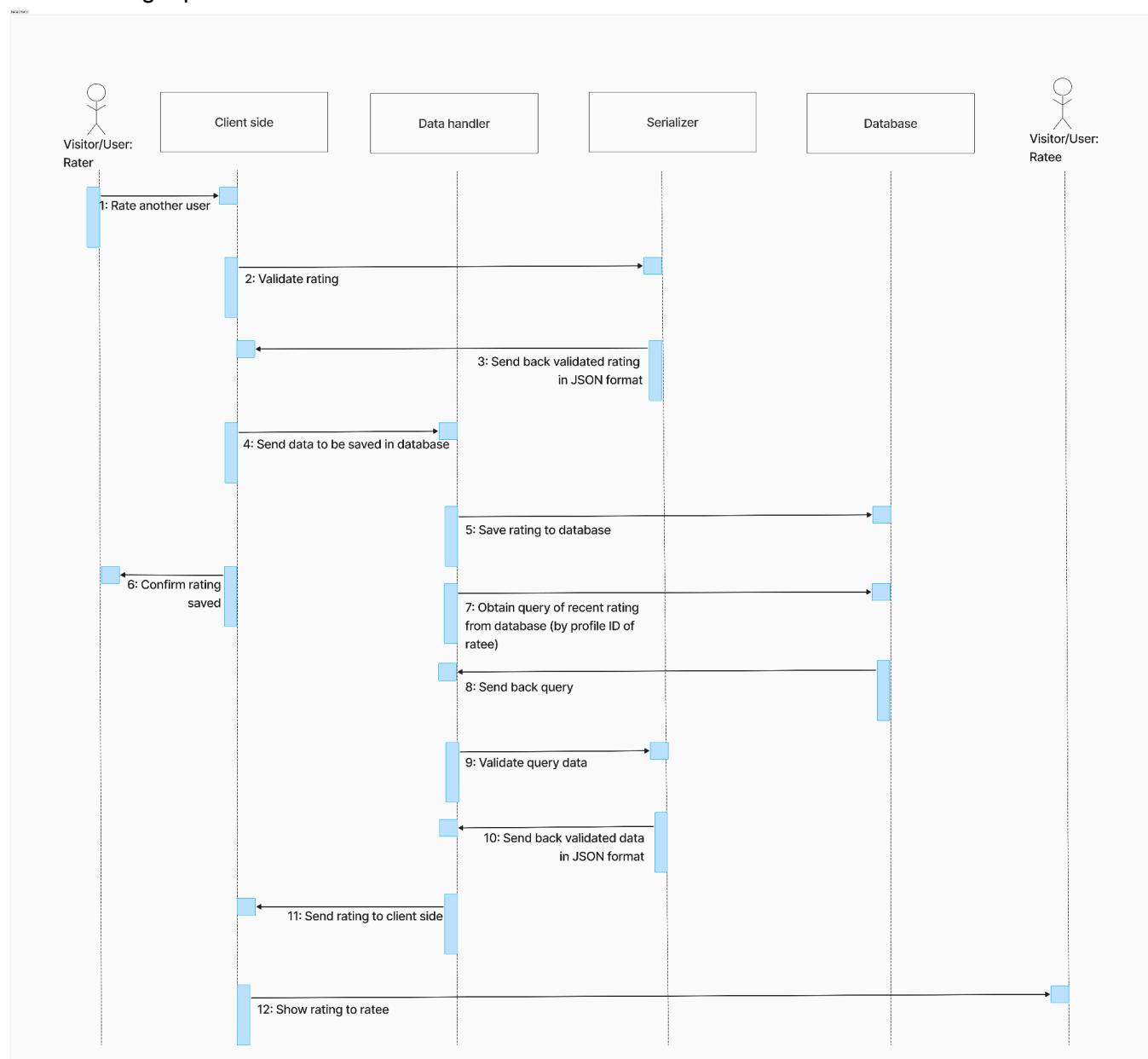
6. Viewing a profile



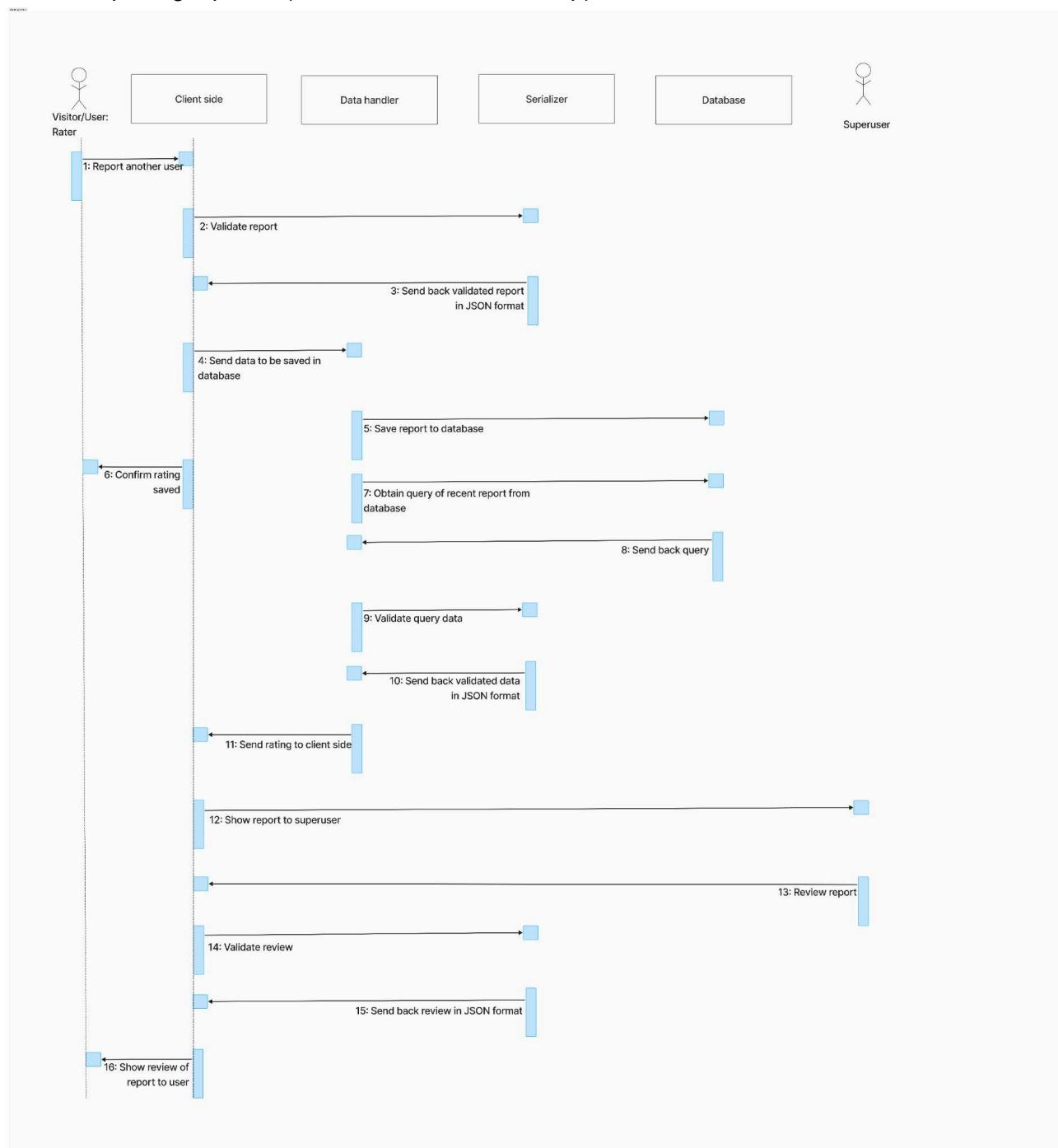
7. Updating your profile



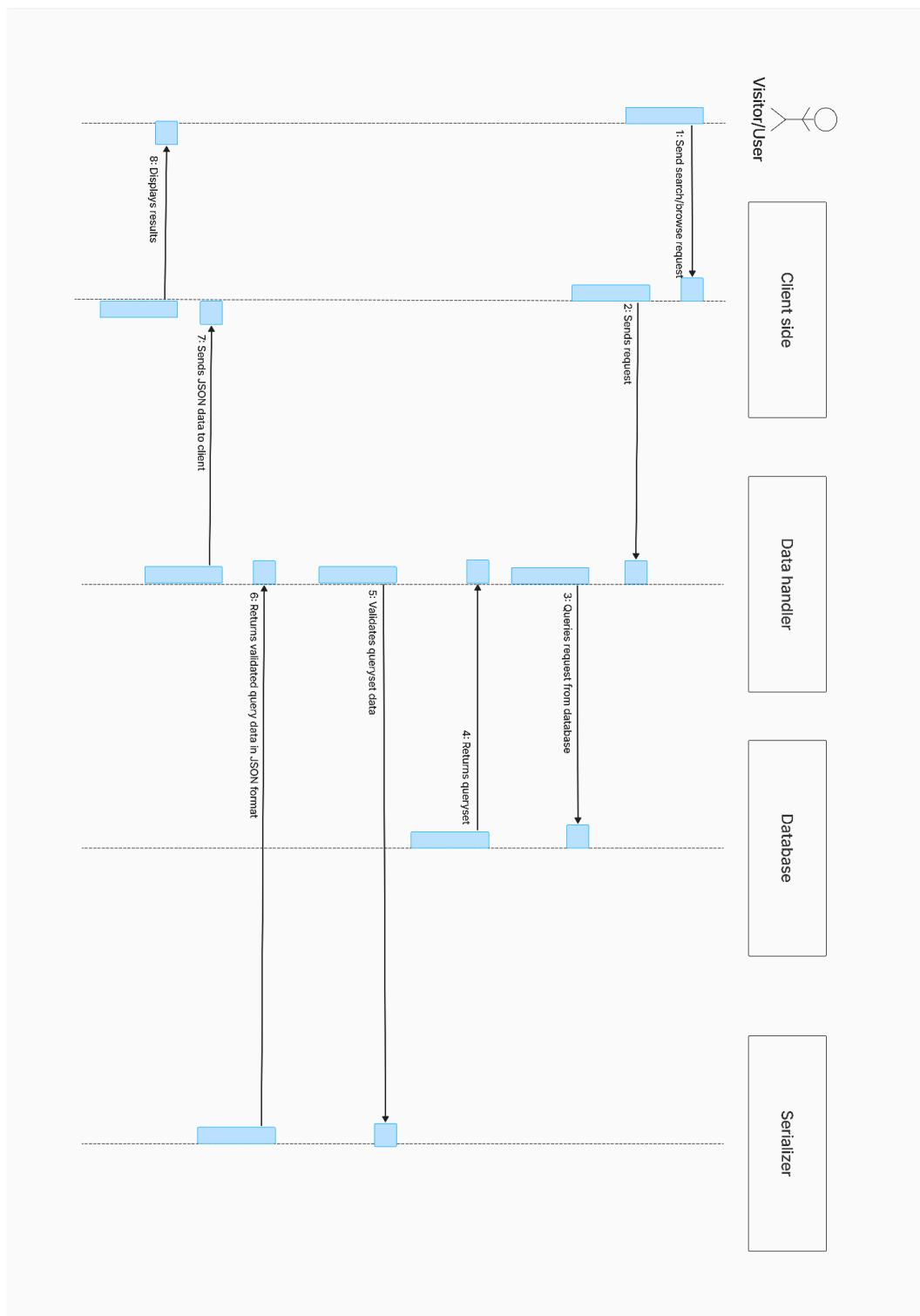
8. Rating a profile



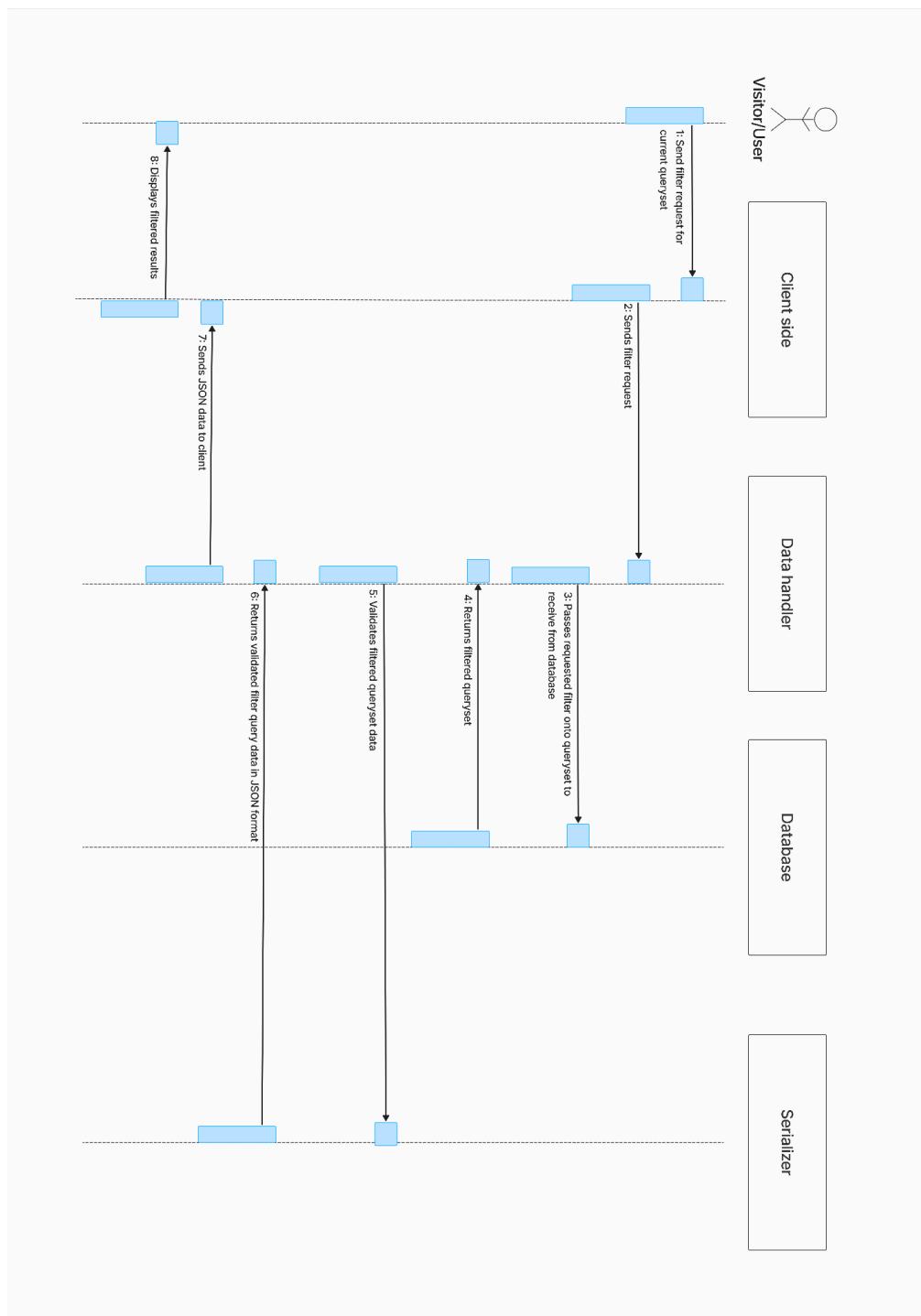
9. Reporting a profile (the lines could not show up)



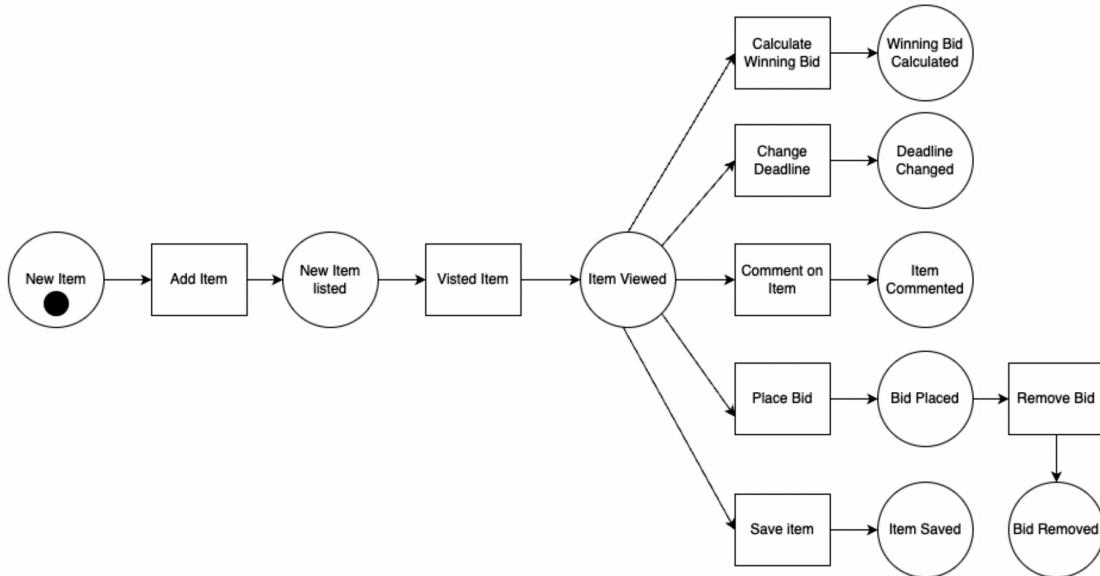
10. Displaying search/browse results



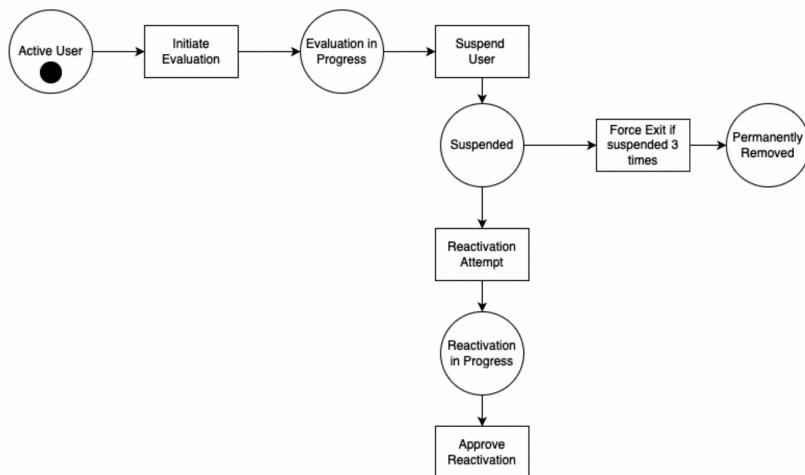
11. Filtering results



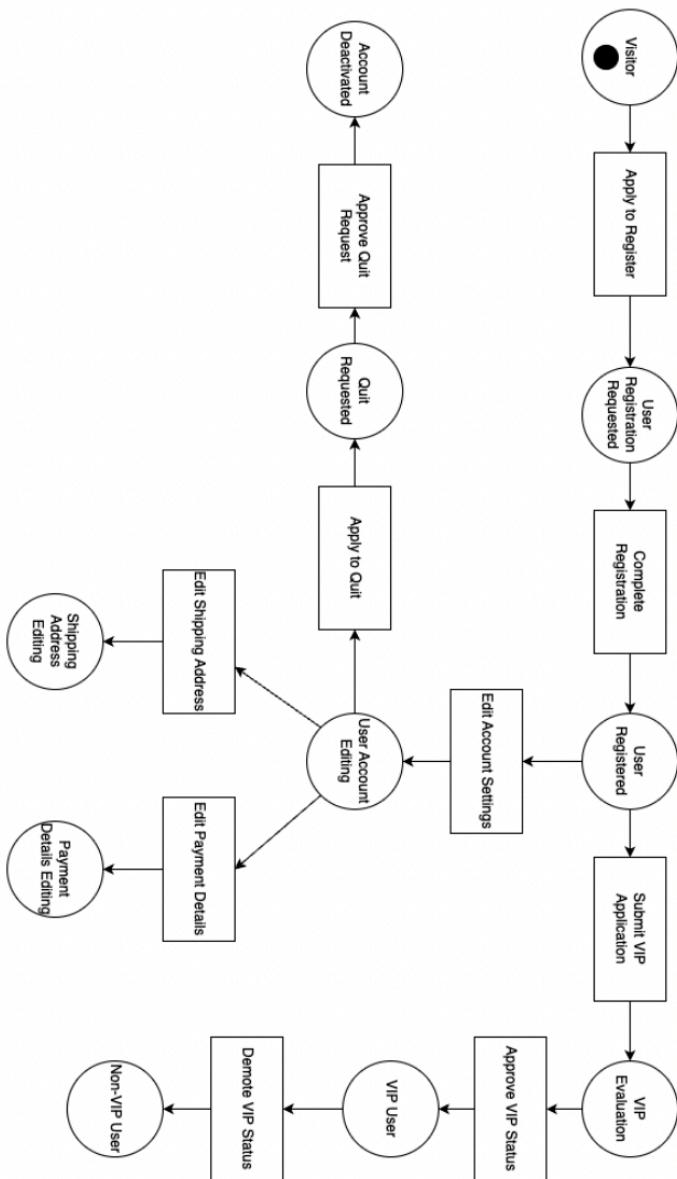
12. Item view class



13. Suspension view class

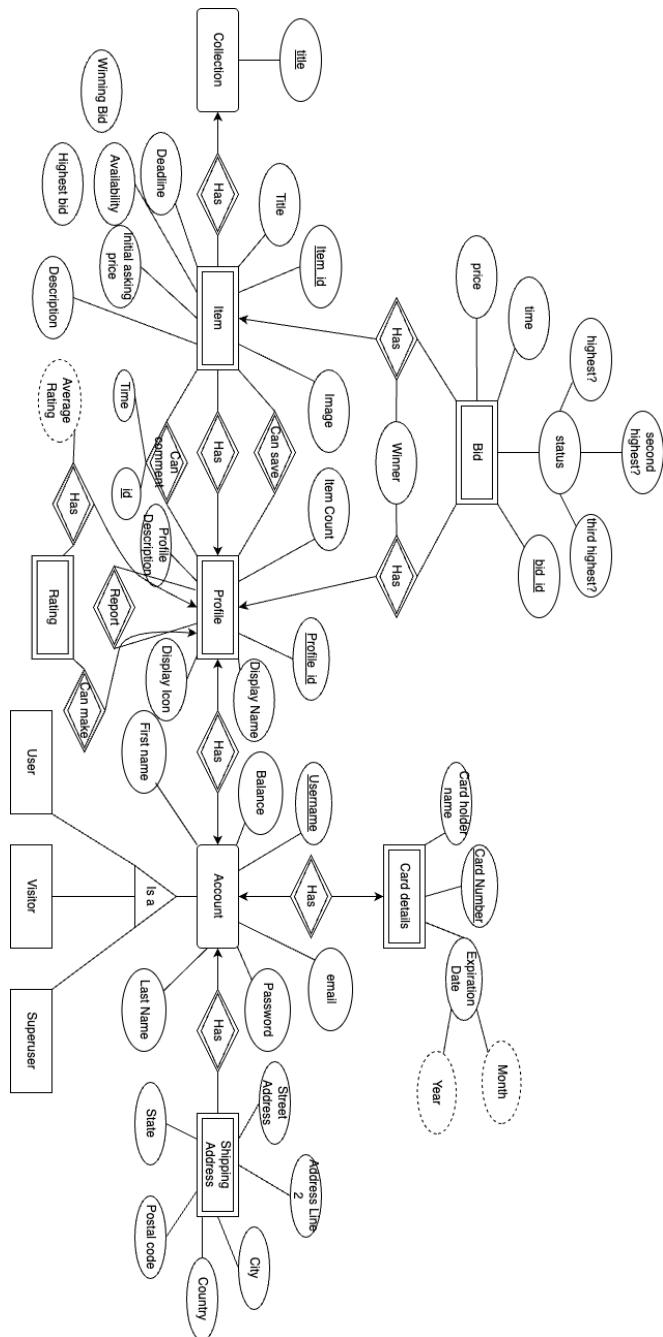


14. Account view class



Blue Penguin – E-bidding Application	Version: 2.1
Software Requirement Specification	Date: 10/11/24
Second Draft	

3. E-R diagram for the Entire system



Blue Penguin – E-bidding Application	Version: 2.2
Software Requirement Specification	Date: 11/11/24
Second Draft	

4. Detail Design:

1. **Account view class:** Handles user registration, account updates, VIP management, and quit requests

Pseudo-Code:

```

class AccountView:
    # User registration (Visitor applies to become User, with bot prevention)
    def register_user(request):
        # Input: { "username": str, "password": str, "email": str, "answer": int }
        # Output: { "status": "success/failure", "message": str }
        pass

    # Edit account settings (email, username, etc.)
    def edit_account_settings(request):
        # Input: { "user_id": int, "settings": { "email": str, "username": str, etc. } }
        # Output: { "status": "success/failure", "message": str }
        pass

    # Edit shipping address
    def edit_shipping_address(request):
        # Input: { "user_id": int, "address": { "street": str, "city": str, etc. } }
        # Output: { "status": "success/failure", "message": str }
        pass

    # Edit payment details
    def edit_payment_details(request):
        # Input: { "user_id": int, "payment_info": { "card_number": str, "expiry_date": str, etc. } }
        # Output: { "status": "success/failure", "message": str }
        pass

    # Apply to quit
    def apply_to_quit(request):
        # Input: { "user_id": int }
        # Output: { "status": "success/failure", "message": str }
        pass
}

```

```

# Check VIP eligibility and update status
def update_vip_status(request):
    # Input: { "user_id": int }
    # Output: { "status": "success/failure", "is_vip": bool, "message": str }
    pass

```

2. **Sign-in view class:** Handles sign-in, log-out, and password recovery.

Pseudo-Code:

```

class SignInView:
    # Sign in
    def sign_in(request):
        # Input: { "username": str, "password": str }
        # Output: { "status": "success/failure", "token": str, "message": str }
        pass

    # Log out
    def log_out(request):
        # Input: { "token": str }
        # Output: { "status": "success/failure", "message": str }
        pass

    # Password recovery
    def password_recovery(request):
        # Input: { "email": str }
        # Output: { "status": "success/failure", "message": str }
        pass

```

3. **Account Menu view class:** Serves as the navigation endpoint; each link is a separate function.

Pseudo-Code:

```

class AccountMenuView:
    # Example function for viewing user details
    def view_user_details(request):
        # Input: { "user_id": int }
        # Output: { "user_details": { "username": str, "email": str, etc. } }
        pass

```

4. **Item view class:** Handles item management, comments, bidding, etc.

Pseudo-Code:

```

class ItemView:
    # Add a new item
    def add_item(request):
        # Input: { "user_id": int, "item_details": { "name": str, "description": str, etc. } }

```

```

# Output: { "status": "success/failure", "item_id": int, "message": str }
pass

# View item details
def view_item(request):
    # Input: { "item_id": int }
    # Output: { "item_details": { "name": str, "description": str, etc. } }
    pass

# Comment on an item
def comment_item(request):
    # Input: { "user_id": int, "item_id": int, "comment": str }
    # Output: { "status": "success/failure", "message": str }
    pass

# Bid on an item
def bid_item(request):
    # Input: { "user_id": int, "item_id": int, "bid_amount": float }
    # Output: { "status": "success/failure", "message": str }
    pass

# Save an item
def save_item(request):
    # Input: { "user_id": int, "item_id": int }
    # Output: { "status": "success/failure", "message": str }
    pass

# Remove a bid
def remove_bid(request):
    # Input: { "user_id": int, "item_id": int }
    # Output: { "status": "success/failure", "message": str }
    pass

# Calculate the winning bid and notify user
def calculate_winning_bid(request):
    # Input: { "item_id": int }
    # Output: { "status": "success/failure", "winning_user": int, "message": str }
    pass

```

5. **Profile view class:** Handles viewing, editing, rating, and complaints on user profiles.

Pseudo-Code:

```

class ProfileView:
    # View profile
    def view_profile(request):
        # Input: { "profile_id": int }

```

```

# Output: { "profile_details": { "username": str, "rating": float, etc. } }
pass

# Edit profile
def edit_profile(request):
    # Input: { "user_id": int, "profile_updates": { "bio": str, "image": str, etc. } }
    # Output: { "status": "success/failure", "message": str }
    pass

# Rate a profile
def rate_profile(request):
    # Input: { "rater_id": int, "profile_id": int, "rating": float }
    # Output: { "status": "success/failure", "message": str }
    pass

# Complain about a profile
def complain_profile(request):
    # Input: { "complainant_id": int, "profile_id": int, "complaint": str }
    # Output: { "status": "success/failure", "message": str }
    pass

```

6. Payment view class: Manages funds for user's bids.

Pseudo-Code:

```

class PaymentView:
    # Add funds to balance
    def add_funds(request):
        # Input: { "user_id": int, "amount": float }
        # Output: { "status": "success/failure", "balance": float, "message": str }
        pass

    # Remove funds when a bid is drawn
    def remove_funds(request):
        # Input: { "user_id": int, "amount": float }
        # Output: { "status": "success/failure", "balance": float, "message": str }
        pass

```

7. Listing view class: Supports browsing, searching, and filtering for items.

Pseudo-Code:

```

class ListingsView:
    # Browse collections
    def browse_collections(request):
        # Input: { "category": str }
        # Output: { "items": [ { "item_id": int, "name": str, etc. } ] }
        pass

```

```

# Browse explore page
def explore(request):
    # Input: { "filter": str }
    # Output: { "items": [ { "item_id": int, "name": str, etc. } ] }
    pass

# Search items
def search_items(request):
    # Input: { "query": str }
    # Output: { "results": [ { "item_id": int, "name": str, etc. } ] }
    pass

```

8. **Suspension view class:** Handles user suspensions based on ratings.

Pseudo-Code:

```

class SuspensionView:
    # Evaluate and suspend user based on rating
    def evaluate_user(request):
        # Input: { "user_id": int }
        # Output: { "status": "suspended/active", "message": str }
        pass

    # Force user out after 3 suspensions
    def force_out_user(request):
        # Input: { "user_id": int }
        # Output: { "status": "removed", "message": str }
        pass

```

9. **Superuser view class:** Manages applications, suspensions, and quit requests.

Pseudo-Code:

```

class SuperuserView:
    # Approve or deny user quit requests
    def manage_quit_request(request):
        # Input: { "user_id": int, "decision": "approve/deny" }
        # Output: { "status": "success/failure", "message": str }
        pass

    # Manage suspensions
    def manage_susensions(request):
        # Input: { "user_id": int, "action": "suspend/reactivate" }
        # Output: { "status": "success/failure", "message": str }
        pass

    # Manage Visitor applications

```

```
def manage_visitor_application(request):
    # Input: { "visitor_id": int, "decision": "approve/deny" }
    # Output: { "status": "success/failure", "message": str }
    pass
```

Blue Penguin – E-bidding Application	Version: 2.3
Software Requirement Specification	Date: 12/11/24
Second Draft	

5. System Screens

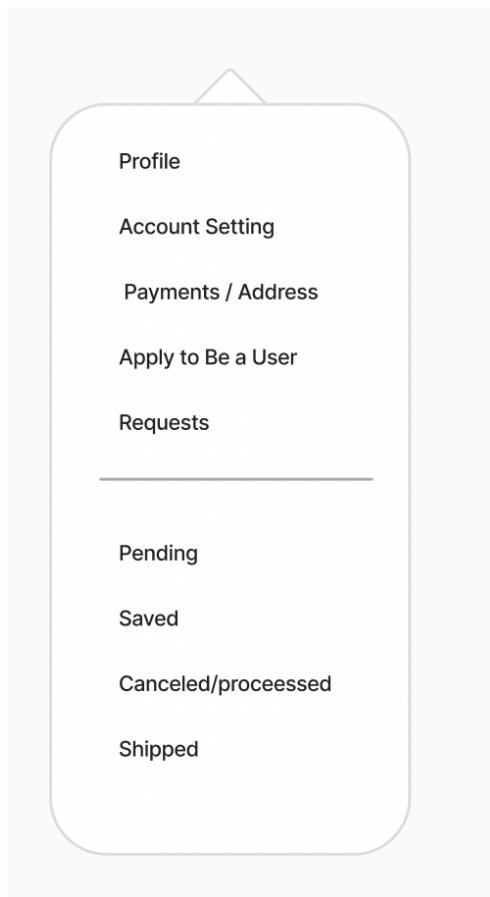
A. Homepage Design

The screenshot shows the homepage of the Blue Penguin E-bidding Application. At the top, there is a navigation bar with the logo '{BLUE PENGUIN}', a 'Categories' menu, a search bar containing 'What would you like to find?', a magnifying glass icon, a 'Sign in' link, and a shopping cart icon. Below the navigation bar, there is a section titled 'Recently Added' which displays five items, each represented by a small thumbnail image showing a penguin holding a sign that says 'complex' and 'fix'. To the right of this section is a '>>' button. Below this, there is a larger section titled 'Shop By Price' which also displays three items with similar thumbnails. At the bottom of the page, there is a blue footer bar with links for 'About us' and 'Credit', and the '{BLUE PENGUIN}' logo.

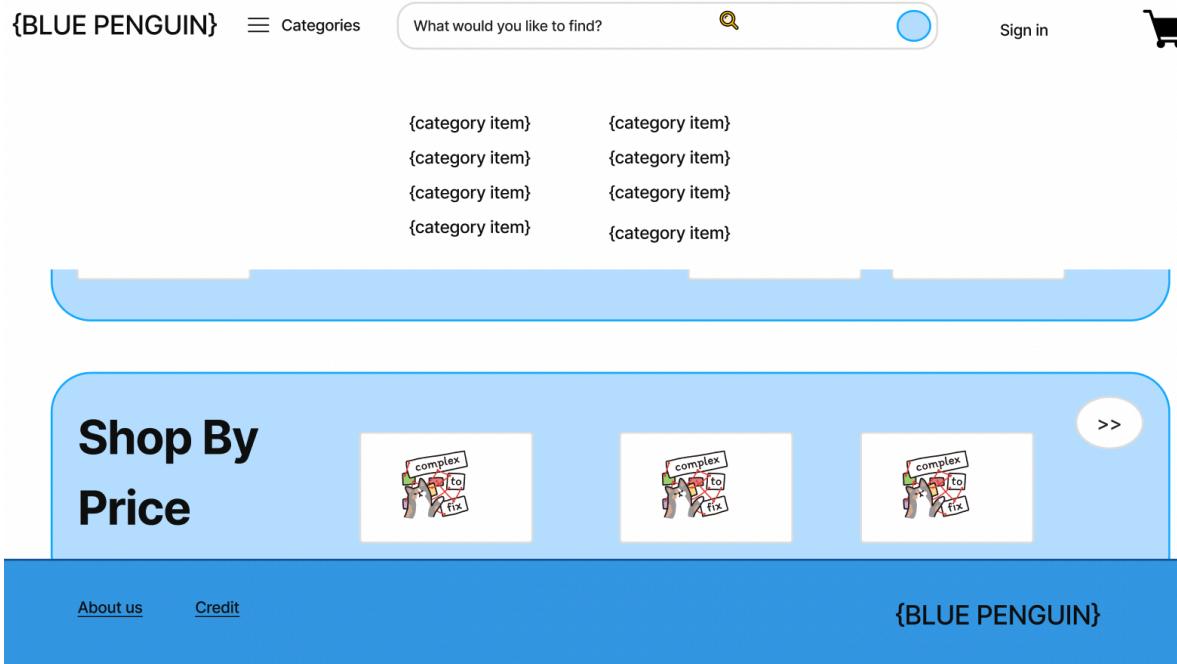
B. Drop-down design for Sign-In and Registration

The wireframe illustrates a mobile application's sign-in and registration screens. On the left, the 'Sign in !' screen features fields for 'Username *' and 'Password *', with links for 'Forgot Password?' and 'Not a member? Register'. It includes social login options for 'Continue with Google' and 'Continue with FaceBook'. A large blue 'Sign in' button is at the bottom. An 'OR' separator line follows. On the right, the 'Register Today!' screen has fields for 'Username *', 'Enter password *', 'Re-enter password *', and 'Email *'. There is a checkbox for accepting terms of service and a 'Register' button. Another 'OR' separator line follows, followed by social login options for 'Continue with Google' and 'Continue with FaceBook'.

C. Account menu drop-down design



D. Category drop-down menu design



E. User's view on the item's bidding page

Item Name **Time Ends**

\$00.00 {username of Highest Bidder}

Description
140 characters max**

Seller : Username

Comments (#)

Start Commenting on this item

MM/DD/YYYY Report

Reply

MM/DD/YYYY Report

Reply

MM/DD/YYYY Report

Reply

Load More

Save **Start Bidding**

Bid Amount

By Clicking I understand I am participating in this bid **BID**

About us Credit **{BLUE PENGUIN}**

F. Search Results Page

{BLUE PENGUIN} ≡ Categories What would you like to find? 🔍 Sign in 🛒

Filters

Categories

- {title}
- {title}
- {title}
- {title}

>> More

Others

- Low to High (\$)
- High to Low (\$)
- Recently Added
- Colors

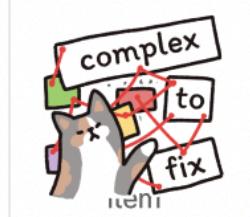
>> More

{result}

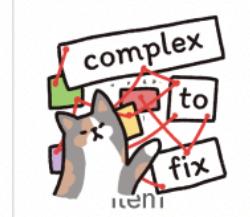
1 - 25 of (n_results)



Item Name
\${Price}
Closes : MM/DD/YYYY



Item Name
\${Price}
Closes : MM/DD/YYYY



Item Name
\${Price}
Closes : MM/DD/YYYY



Item Name
\${Price}
Closes : MM/DD/YYYY



Item Name
\${Price}
Closes : MM/DD/YYYY



Item Name
\${Price}
Closes : MM/DD/YYYY

1 2 ... N ▶

G. Cart design



≡ Filters

Saved

Closed

Pending

Ongoing

...

Pops up when bid is won (either from user or of a listing)

Pending bids (#)

Date Added

Product

Status



Product Name

Authors description of the item.
Purpose
Dimension
Color

Closes: MM/DD/YY

Bid in lead

Outbidded

Accept Item

| Remove

Next Actions (#)

Product



Product Name

Authors description of the item.
Purpose
Dimension
Color

Closes: MM/DD/YY

{username} won item!

Ship Item

OR

Return Money

Awaiting Arrival(#)

Product



Product Name

Authors description of the item.
Purpose
Dimension
Color

Closes: MM/DD/YY

Status

On route

Arrived

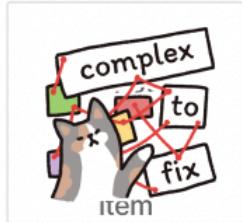
Rate the Product | Problems?

Saved Items (#)

Item Name

\$(Price)

Closes : MM/DD/YYYY



Item Name

\$(Price)

Closes : MM/DD/YYYY



Item Name

\$(Price)

Closes : MM/DD/YYYY

H. Profile design

{BLUE PENGUIN} Categories

What would you like to find?

My Account

Cart

Username

Legal Name | Status | Joined MM/DD/YYYY | Rating : *****

Description of the {username}

Product Listings (#)

Reviews (#)

MM/DD/YYYY

Report

Item: Descriptions Descriptions Descriptions Descriptions Descriptions

Leave a review

I. Design for an item whose auction has been closed for less than 24 hours

{BLUE PENGUIN} Categories What would you like to find? My Account

Item Name Closed

\$0.00 {username of Highest Bidder}

Description
140 characters max**

Comments (#)

Start Commenting on this item

MM/DD/YYYY Report

MM/DD/YYYY Report

MM/DD/YYYY Report

Related Items

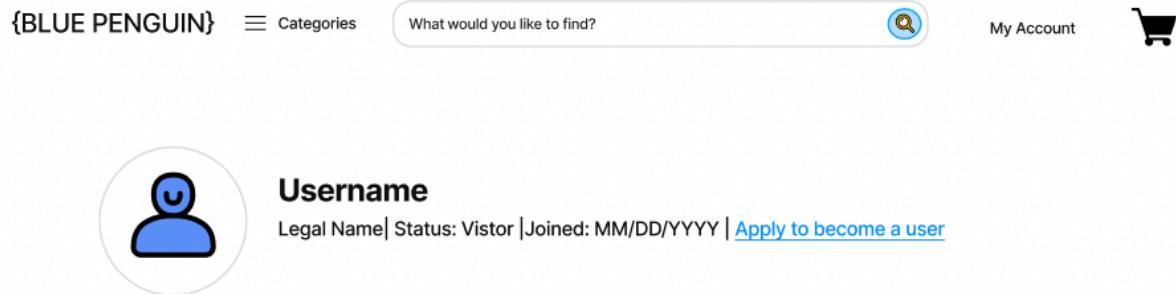
Item Name \$Price Closes : MM/DD/YYYY

Item Name \$Price Closes : MM/DD/YYYY

Item Name \$Price Closes : MM/DD/YYYY

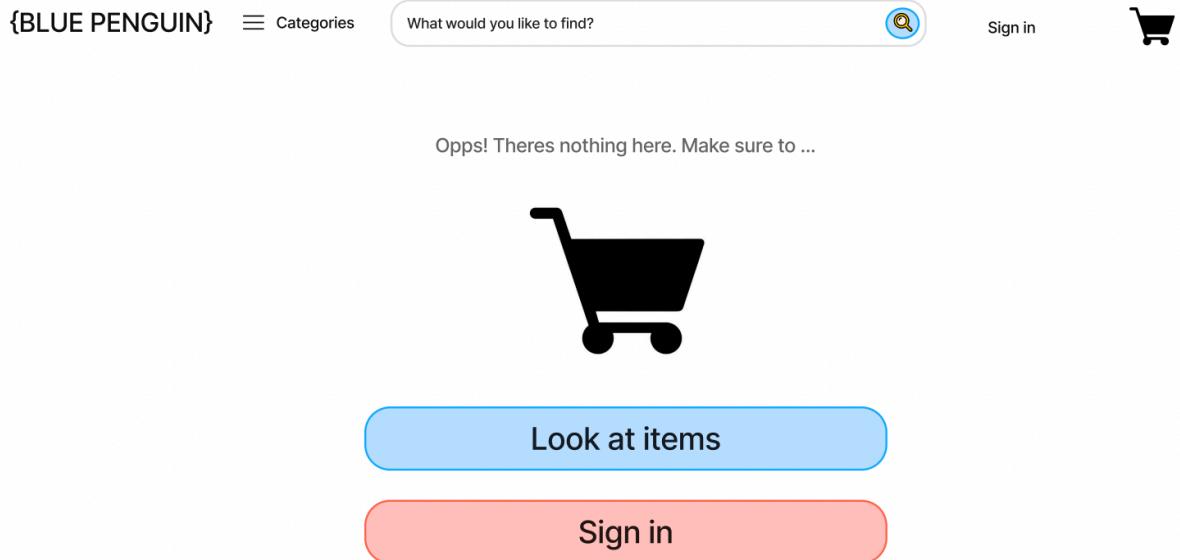
About us Credit {BLUE PENGUIN}

J. Profile design for a visitor profile



The screenshot shows a visitor profile page. At the top, there is a header bar with the text '{BLUE PENGUIN}' and a categories menu icon. To the right of the header is a search bar containing the placeholder 'What would you like to find?' with a magnifying glass icon. Further to the right are 'My Account' and a shopping cart icon. Below the header, on the left, is a circular user icon containing a blue penguin silhouette. To the right of the icon, the word 'Username' is displayed in bold. Below 'Username' is a line of text showing 'Legal Name| Status: Vistor |Joined: MM/DD/YYYY | [Apply to become a user](#)'. The rest of the page is blank.

K. Design for a visitor's cart page



The screenshot shows a visitor's cart page. At the top, there is a header bar with the text '{BLUE PENGUIN}' and a categories menu icon. To the right of the header is a search bar containing the placeholder 'What would you like to find?' with a magnifying glass icon. Further to the right are 'Sign in' and a shopping cart icon. In the center of the page, the text 'Opps! Theres nothing here. Make sure to ...' is displayed above a large black shopping cart icon. Below the icon are two buttons: a blue button labeled 'Look at items' and a red button labeled 'Sign in'.

L. Sign of a listed item's page (Owner's point of view)

{BLUE PENGUIN} Categories What would you like to find? My Account

Item Name Time Ends

\$00.00 {username of Highest Bidder}

Description
140 characters max**

Comments (#)

Start Commenting on this item

MM/DD/YYYY Report

MM/DD/YYYY Report

MM/DD/YYYY Report

Load More

Related Items

Item Name \$Price Closes: MM/DD/YYYY

Item Name \$Price Closes: MM/DD/YYYY

Item Name \$Price Closes: MM/DD/YYYY

About us Credit {BLUE PENGUIN}

M. Transaction/Address Page of a user

The screenshot shows a mobile application interface for a financial account. At the top, there is a header with the text '{BLUE PENGUIN}' and a 'Categories' button. A search bar with the placeholder 'What would you like to find?' is positioned next to a magnifying glass icon. To the right are 'Sign in' and a shopping cart icon.

Available Balance

\$ 00.00

Add Money | Zelle | PayPal

Shipping Address

Street Address,
City, State (abbreviation), Zip code

Change Address

Transactions

#Account it came from
MM/DD/YYYY \$ 00.00

#Account it came from
MM/DD/YYYY \$ 00.00

#Account it came from
MM/DD/YYYY \$ 00.00

#Account it came from

N. Make Request - File Complaint Against Another User

The screenshot shows a 'Requests' section within a mobile application. At the top, there is a header with the text '{BLUE PENGUIN}', a 'Categories' button, a search bar with the placeholder 'What would you like to find?', a magnifying glass icon, 'My Account' link, and a shopping cart icon.

Requests

File Compliant against User

Quit as User

Pay Suspension fine

Your Username
[Redacted]

Username of offender
[Redacted]

Description
[Large text input field]
140 char max*

Submit Compliant

O. Make Request - Quit as User

The screenshot shows the Blue Penguin platform's interface. At the top, there is a header with the logo '{BLUE PENGUIN}', a 'Categories' menu, a search bar containing 'What would you like to find?', a magnifying glass icon, 'My Account', and a shopping cart icon. Below the header, the page title is 'Requests'. On the left side, there is a sidebar with three options: 'File Compliant against User', 'Quit as User', and 'Pay Suspension fine'. The 'Quit as User' option is selected. The main content area contains fields for 'Your Username' and 'Password', a 'Reason' section with a list of checkboxes for various reasons, and a red 'Submit' button.

Your Username
Password

Reason

- "I no longer wish to use this service"
- "I received too many email notifications"
- "I found a better alternative."
- "This platform is difficult to use."
- "I created a duplicate account."
- Other

Submit

P. Make Request - Pay Suspension Fine

The screenshot shows the Blue Penguin platform's interface. At the top, there is a header with the logo '{BLUE PENGUIN}', a 'Categories' menu, a search bar containing 'What would you like to find?', a magnifying glass icon, 'My Account', and a shopping cart icon. Below the header, the page title is 'Requests'. On the left side, there is a sidebar with three options: 'File Compliant against User', 'Quit as User', and 'Pay Suspension fine'. The 'Pay Suspension fine' option is selected. A message in the center states: 'Your account is currently suspended. A fee of \$50 and admins approval is required to reinstate your account'. The main content area contains fields for 'Username*' and 'Password *', a '\$50.00' button, and a note about agreeing to a withdrawal of \$50. There is also a red 'Submit' button.

Your account is currently suspended. A fee of \$50 and admins approval is required to reinstate your account

Username*
Password *

\$50.00

By clicking this amount, I'm agreeing to a withdrawal of \$50 to the Blue Penguin Platform to pay the suspension fee.

Submit

Q. Account Setting - Edit Account Infomation

The screenshot shows the 'Account Setting' section of a website. At the top, there is a header with the logo '{BLUE PENGUIN}', a 'Categories' menu, a search bar containing 'What would you like to find?' with a magnifying glass icon, 'My Account', and a shopping cart icon. Below the header, the 'Account Setting' title is displayed. The form contains fields for First name, Last Name, Email, Username, Password, Card Details, and Address. Each field has a placeholder value: {Fname}, {Lname}, {Email}, {Username}, {Password}, {Card Details}, and {Address}. To the right of the address field are two buttons: 'Cancel' and 'Save Changes'. At the bottom of the page, there is a blue footer bar with links for 'About us' and 'Credit', and the '{BLUE PENGUIN}' logo.

Account Setting

First name :
{Fname}

Last Name :
{Lname}

Email :
{Email}

Username :
{Username}

Password :
{Password}

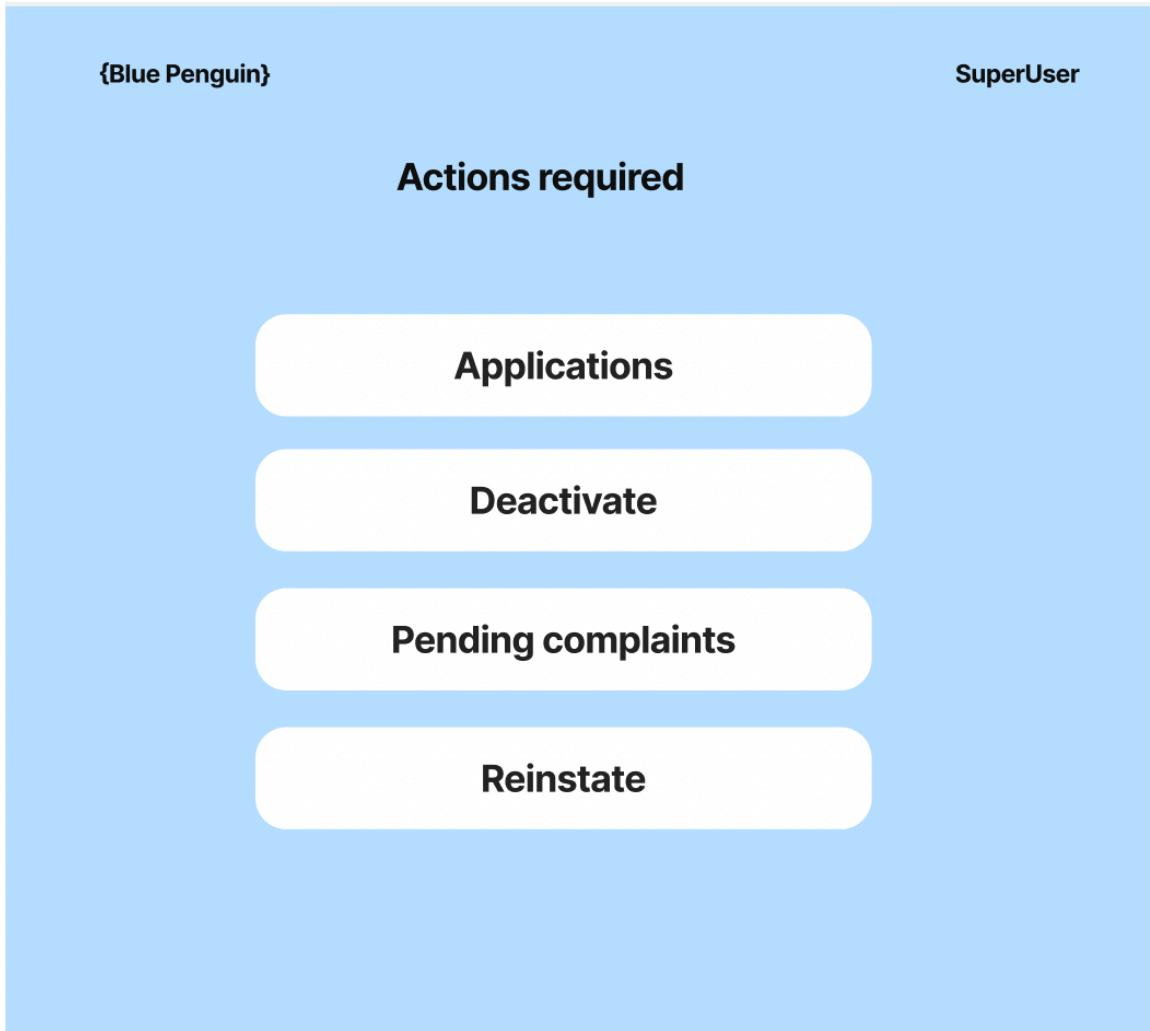
Card Details :
{Card Details}

Address :
{Address}

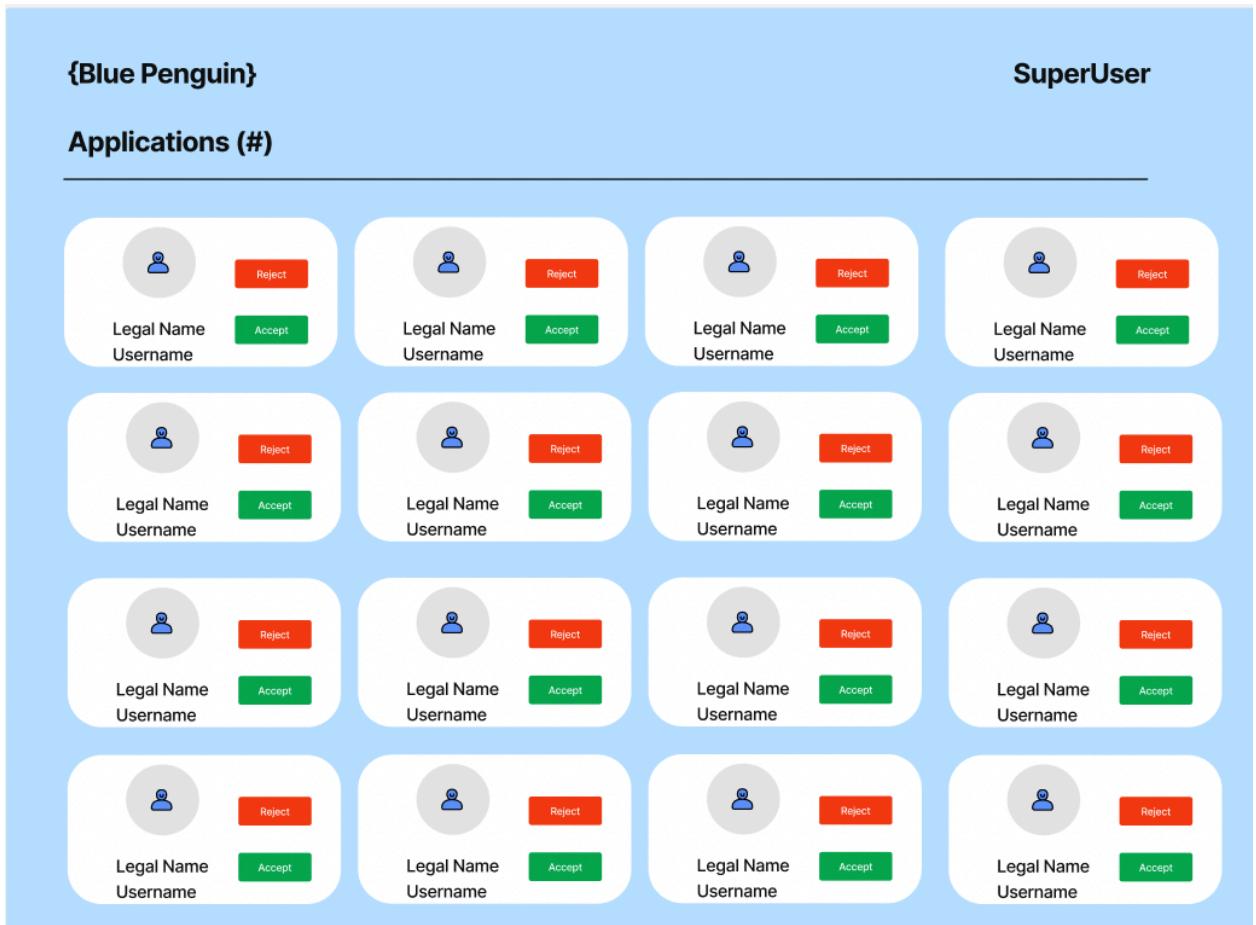
Cancel Save Changes

[About us](#) [Credit](#) **{BLUE PENGUIN}**

R. Super User Homepage



S. Super User - Applications



T. Super User - Deactivate

{Blue Penguin}

SuperUser

Deactivate (#)

The interface displays a 4x4 grid of user profiles. Each profile consists of a circular icon with a person symbol, the text "Legal Name" and "Username" below it, and a green "Accept" button to its right. The profiles are arranged in four rows and four columns.

Column 1	Column 2	Column 3	Column 4
Legal Name Username	Legal Name Username	Legal Name Username	Legal Name Username
Legal Name Username	Legal Name Username	Legal Name Username	Legal Name Username
Legal Name Username	Legal Name Username	Legal Name Username	Legal Name Username
Legal Name Username	Legal Name Username	Legal Name Username	Legal Name Username

U. Super User - Reinstates

{Blue Penguin}

SuperUser

Reinstate (#)

 Legal Name Username	Reject	Accept
 Legal Name Username	Reject	Accept
 Legal Name Username	Reject	Accept
 Legal Name Username	Reject	Accept
 Legal Name Username	Reject	Accept
 Legal Name Username	Reject	Accept
 Legal Name Username	Reject	Accept
 Legal Name Username	Reject	Accept
 Legal Name Username	Reject	Accept
 Legal Name Username	Reject	Accept
 Legal Name Username	Reject	Accept
 Legal Name Username	Reject	Accept
 Legal Name Username	Reject	Accept
 Legal Name Username	Reject	Accept
 Legal Name Username	Reject	Accept
 Legal Name Username	Reject	Accept

V. Super User - Complaints

The interface is titled "SuperUser" at the top right. At the top left, it says "{Blue Penguin}" with a "Complaints (#)" button below it. The main area contains a grid of 9 complaint entries, arranged in a 3x3 grid. Each entry has a yellow header with "Offending Username" and "complaint description". Below the header is a dashed-line text area. To the right of the text area are three buttons: "Review" (light blue), "Reject" (red), and "Agree" (green). The "Agree" button is only visible for the first row of complaints.

Offending Username	complaint description	Review	Reject	Agree
Offending Username	complaint description	Review	Reject	
Offending Username	complaint description	Review	Reject	Agree
Offending Username	complaint description	Review	Reject	
Offending Username	complaint description	Review	Reject	Agree
Offending Username	complaint description	Review	Reject	
Offending Username	complaint description	Review	Reject	
Offending Username	complaint description	Review	Reject	Agree
Offending Username	complaint description	Review	Reject	

Blue Penguin – E-bidding Application	Version: 2.2
Software Requirement Specification	Date: 11/11/24
Second Draft	

6. Memos of group meetings and possible concerns

A. Meeting 1

Team Meeting Minutes

06 October 2024 / 12:30 PM - 3:15 PM / Video Conference

Attendees

- Addina R.
- Samiul S.
- Lily M.

Agenda

Follow-up from Last Meeting

- Finding a reliable method to connect the front end to the back end
- Begin drafting the initial design concept.
- E-bidding website is officially named **Blue Penguin**

New Business

- Review and provide feedback on design drafts for the Blue Penguin platform.
- Clarify overall required actions and user functionalities
- Assign and understand criteria for each section of the first report.

Notes

- The front-end and back-end teams will work independently on their respective sections before integrating them.
- Several initial design concepts were approved for further development on the Figma Jamboard (with a star stamp). Approvements and changes were noted on grey sticky notes listed on the Jamboard.

Action Items

- All team members must familiarize themselves with the programming languages and tools required for the project. (**insert languages**)
- Begin coding respective sections of the project.
- Draft and add new elements (beyond layouts) for the Blue Penguin website.
- Start and share a new document for **Report 1 (Due: 10/17/2024)**

B. Meeting 2

Team Meeting Minutes

06 November 2024 / 5:00 PM - 7:00 PM / Video Conference

Attendees

- Addina R.
- Samiul S.
- Lily M.

Agenda

Follow-up from Last Meeting

- **Report 1** was completed and submitted on time
- Integration of back-end and front-end

New Business

- Assign and understand criteria for sections required of the second report

Notes

- Integration of the initial sections of the back-end and front-end has been successful.
- Addition items needed:
 - A separate account settings page
 - Delete 'edit username' from the user's profile
- Division of **Report 2** has been noted and reflected in the official **Report 2** Google doc

Action Items

- Each member should start working on their section(s) for **Report 2 (Due: 10/12/2024)**.
- Questions should be posted in the general chat created at the start of the collaboration. (will not be noted here for privacy reasons)

Blue Penguin – E-bidding Application	Version: 1.0
Software Requirement Specification	Date: 10/11/24
Second Draft	

7. Address of the git repo:

Repo Link: <https://github.com/iDaManall/BluePenguin>