
Design Document for Cydrop Reminder

Group: 3_mahdi_1

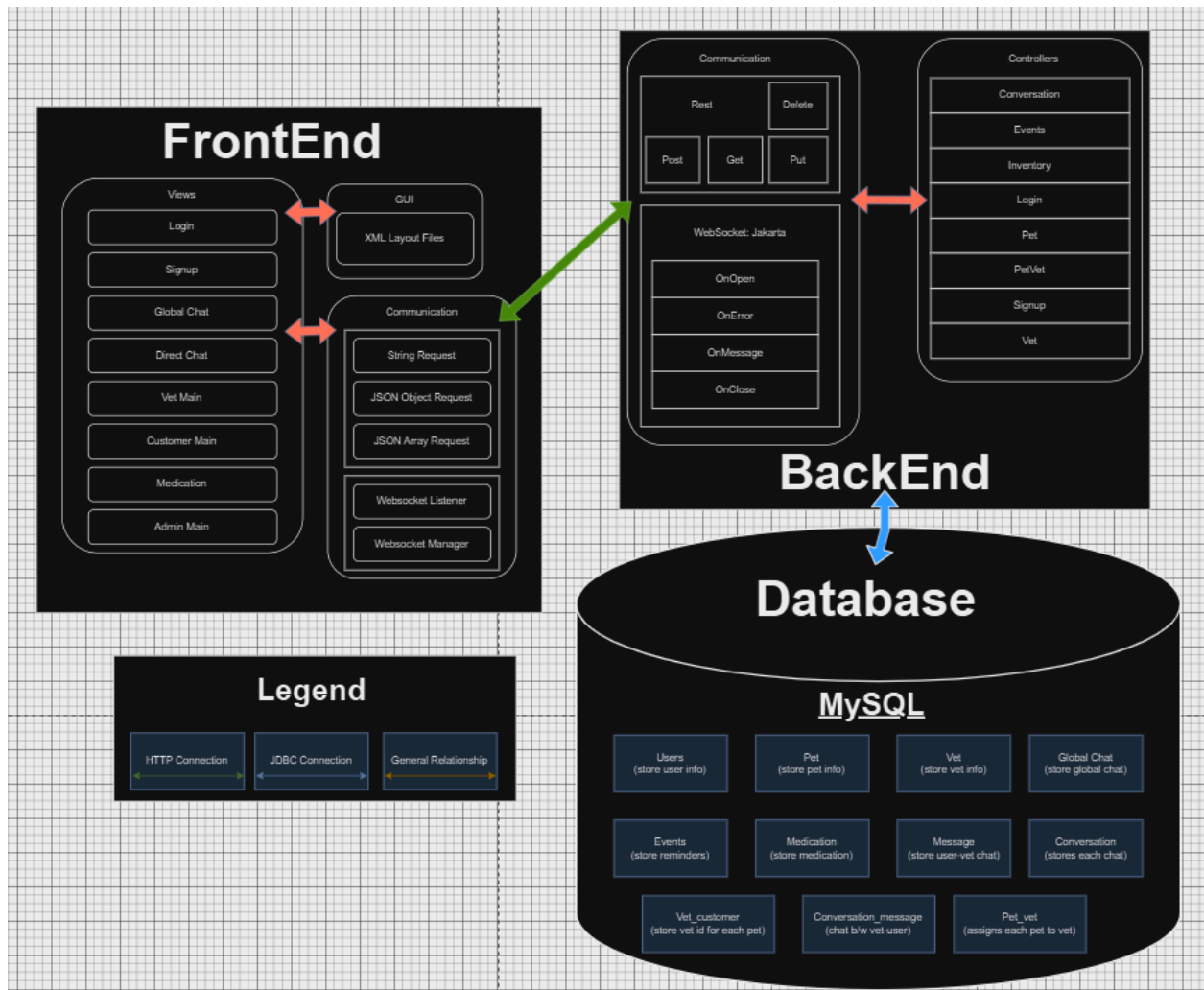
Member1 Name: Madeleine Carydis

Member2 Name: Madison Vosburg

Member3 Name: Fury Poudel

Member4 Name: Niraj Amin

BLOCK DIAGRAM



DESIGN EXPLANATIONS

Frontend

Signup: When clicking the button “Signup” the values of email and password are sent as a POST request to the server.

Login: When clicking the button “Login” the values of email and password are compared to the values in the backend with a POST request. If the values match (ie, the information is valid), the user will be taken to the main menu of their specific user type (customer/vet/admin).

Chat: Global chat between all users, direct chat between vet and user. When clicking the button “Send” the value of the message will be sent through websocket. The message will appear in the scroll view chat.

Main View: Main page of specific user view after successful login. Can navigate to different pages with bottom navbar. Can view and edit their assigned customers or pets, edit details, and manage medications.

Manage Medication: Vets can manage pet medication. Upon loading page, the list of medication information will be displayed in scroll view with a GET request. When the user enters a new medication name and clicks the button “Add medication”, the value of medication is sent as a POST request to the server. When the user enters an existing medication ID and clicks the button “Delete medication”, the value of medication ID is sent as a DELETE request to the server. When clicking the button “Save” on the Edit Medication page, the values of ID, medication name, and stock are sent as a PUT request to the server.

Backend

Communication:

- We use REST API to update the databases
 - Post mapping: to add new thing to the database. Put mapping: to update a thing already in the database. Delete mapping: to remove a thing in the database. Get mapping: to fetch information about a thing in the database
- We use Jakarta Websocket to allow different users to chat in real time
 - OnOpen: all prior messages in database will be loaded. OnMessage: new message will appear and be saved into the database. OnClose: terminates link between the people chatting. OnError: terminates link and notifies of type of error

Controllers:

- Conversation: Contains mappings to read and list conversations. Allows you to fetch all messages in a conversation
- Events: Contains mappings to create events
- Inventory: Contains mappings to create, update, fetch, and delete medications and inventory
- Login: Contains a post mapping to authenticate a user
- Pet: Contains mappings to create, update, fetch, and delete pets. When a pet is created, by default it is linked to an owner
- PetVetAssign: Contains mappings to assign pets to vets and allows fetch and removal of assignments
- Signup: Contains mappings to create, update, fetch, and delete users
- Vet: Contains mappings to create, update, fetch, and delete vets. Contains mappings to create and remove a link between a vet and its customer (a user)

TABLE RELATIONSHIPS DIAGRAM

