

Product backlog - General Workflow

1. Authentication and Login

- Login Options: Users log in either through Google OAuth or email-based authentication.
- 2-Factor Authentication (2FA): Set up two-factor authentication for added security, triggered after initial password entry or OAuth.

2. User Information Data Storage

- Data Storage: Store user data using SQL (e.g., MySQL or PostgreSQL) for structured data management.
- Location Integration: Capture geolocation data (latitude and longitude) using the Google Maps API.

3. Matching System

- Red Flag Filtering: Filter out users based on "red flags" (preferences or dealbreakers set by each user).
- GPT API Integration: Use the OpenAI GPT API to generate a recommendation order based on user preferences and profile analysis.
- Sorting and Scoring: Calculate and sort matches by compatibility scores using pre-trained or in-house models.

4. Messaging System

- Real-Time Messaging: Use a real-time, scalable messaging system
- Storage and History: Store message history securely in SQL for retrieval.

Sprint #1 backlog

1. UI mockup (together)
2. Database and workflow discuss (together)

Product owner

Miaoyang Ye

Data model

1. Users Table

Stores basic user information and supports both OAuth and email login methods.

Field	Type	Description
-------	------	-------------

username	VARCHAR	Username (Primary Key)
oauth_id	VARCHAR	User ID returned by the OAuth provider
email	VARCHAR	User's email address
password	VARCHAR	Encrypted password (only for email-based users)
provider	VARCHAR	OAuth provider name or "email"
created_at	TIMESTAMP	Account creation date
updated_at	TIMESTAMP	Last update date

2. User_Pets Table

Stores the relationship between users and their pets; a user can have multiple pets.

Field	Type	Description
username	VARCHAR	User's username (Foreign Key to Users)
pet_name	VARCHAR	Pet's name (Foreign Key to Pet Information)

3. Pet Information Table

Stores detailed information about each pet.

Field	Type	Description
pet_name	VARCHAR	Pet's name (Primary Key)
gender	VARCHAR	Gender of the pet
region	VARCHAR	Location/Region of the pet
timeslot	VARCHAR	Available time slots
breed	VARCHAR	Pet's breed
age	INT	Age of the pet
weight	DECIMAL	Weight of the pet
health	VARCHAR	Health details (e.g., vaccinations, allergies)
photo	VARCHAR	Link to pet's photo

4. Pet Friends Table

Stores friendships between pets.

Field	Type	Description
<code>pet_name_1</code>	VARCHAR	Name of Pet 1 (Foreign Key to Pet Information)
<code>pet_name_2</code>	VARCHAR	Name of Pet 2 (Foreign Key to Pet Information)

5. Password_Resets Table

Stores verification details for password recovery requests.

Field	Type	Description
<code>email</code>	VARCHAR	User's email to identify the account
<code>token</code>	VARCHAR	Unique Token for verification
<code>expires_at</code>	TIMESTAMP	Expiration time for the Token
<code>used</code>	BOOLEAN	Status indicating if the Token has been used

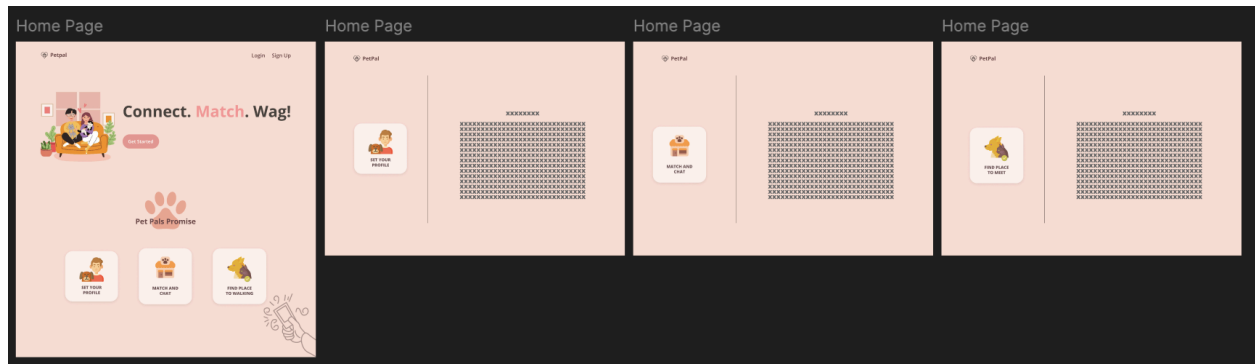
6. Location_Marks Table

Stores locations marked by users on the map.

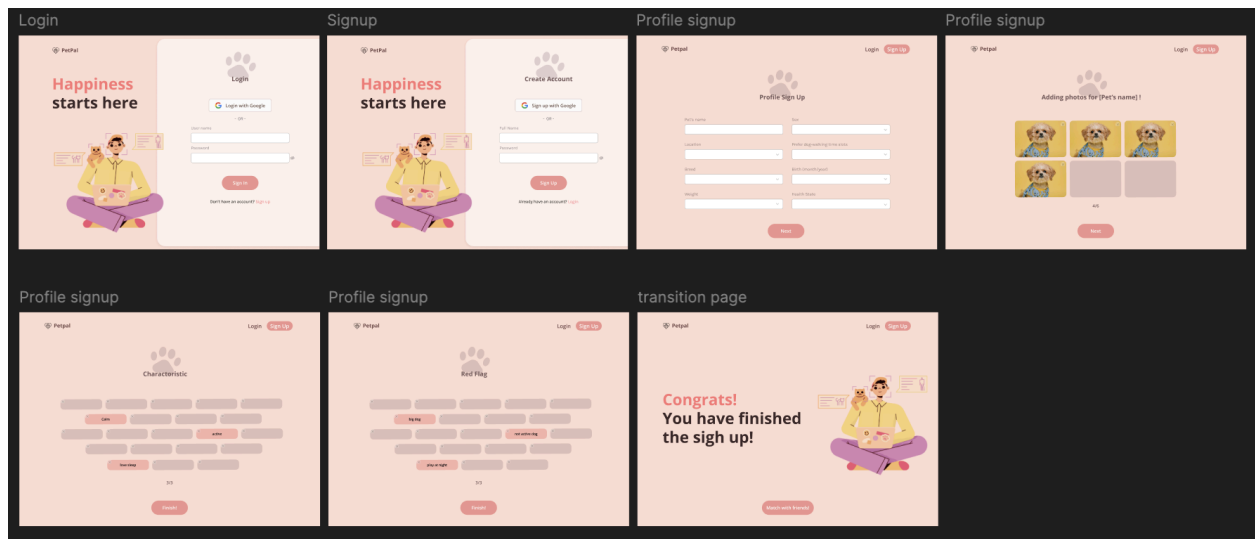
Field	Type	Description
<code>mark_id</code>	INT	Unique ID for each location mark (Primary Key)
<code>username</code>	VARCHAR	Username of the user who marked the location
<code>location_name</code>	VARCHAR	Name of the location
<code>latitude</code>	DECIMAL	Latitude of the location
<code>longitude</code>	DECIMAL	Longitude of the location
<code>description</code>	TEXT	Description or comment on the location
<code>created_at</code>	TIMESTAMP	Date when the mark was created
<code>photo_url</code>	VARCHAR	Optional link to a photo uploaded by the user

UI mockup

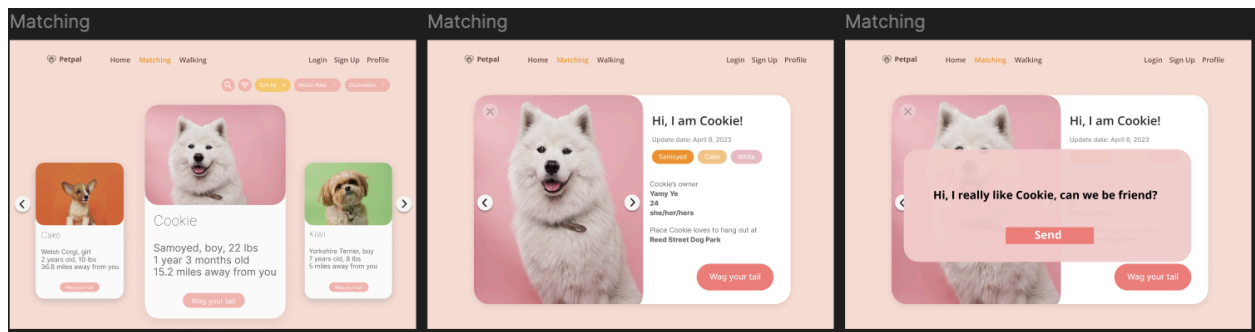
0.Home Page



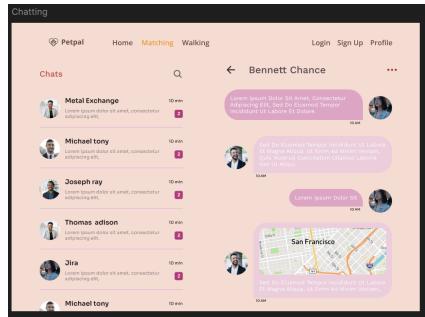
1.Login & Register



2.Matching



3.Chatting



4.Mapping

