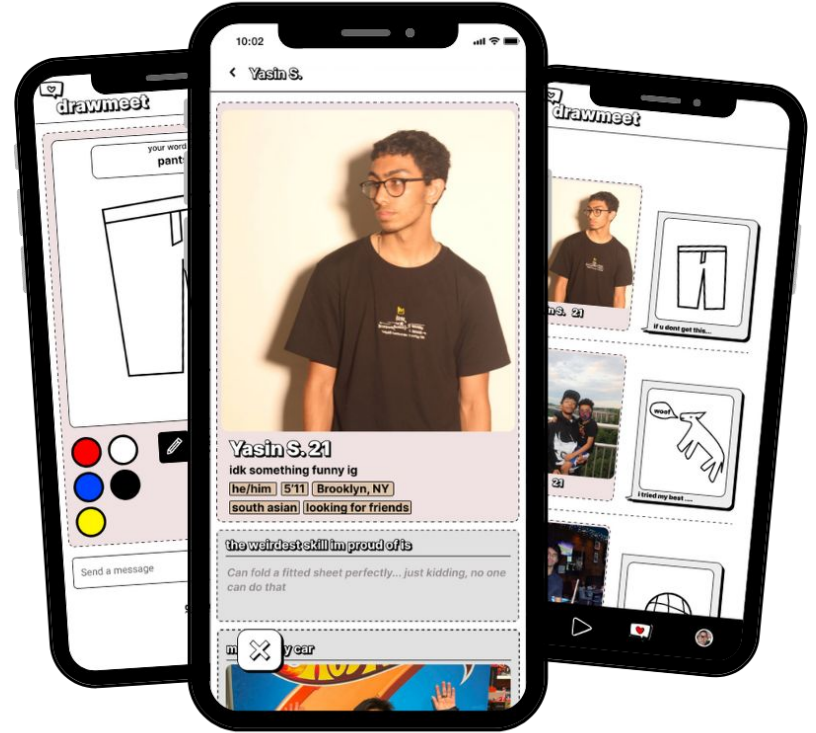# drawmeet

draw a connection

team:

Bryan Zacarias (VC1A) bryan.zacarias7171@gmail.com

David Lema (VC1C) lemadavid1234@gmail.com

supervisor:

Aaryan Shrestha aaryanrajshrestha1@gmail.com

Github repository

https://github.com/Bzacarias03/CISC-4900.git

Clickup board

Reach out for access.
https://app.clickup.com/9013614663/v/li/901307523619

abstract

Apps like hinge and bumble and other traditional matchmaking apps exist for the purpose to solely meet people.

They add in prompts and other hooks for people to chat about. While it serves the purpose of you meeting someone, it doesn't solve the main problem of why people are lonely.
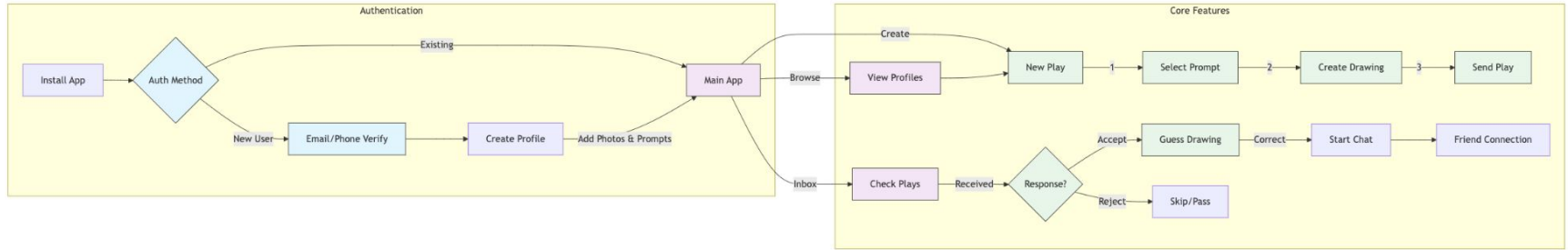
There are plenty of ways to meet people, you'll probably interact with hundreds of people the second you leave your home.

The issue is more about how you can communicate when you are placed in front of these people.

Unless you are incredibly charming or good looking, the odds that you are going to struggle holding a conversation is rather high.

The goal is to implement games to bring out a playful side to someone and ease the tension. People have made friends through gaming ever since online gaming was a thing. We're just expanding on that concept.

user flow

Authentication

Install App → Auth Method

Auth Method —Existing→ Main App

Auth Method —New User→ Email/Phone Verify → Create Profile —Add Photos & Prompts→ Main App

Core Features

Main App —Browse→ View Profiles

Main App —Create→ New Play —1→ Select Prompt —2→ Create Drawing —3→ Send Play

Main App —Inbox→ Check Plays —Received→ Response?

Response? —Accept→ Guess Drawing —Correct→ Start Chat → Friend Connection
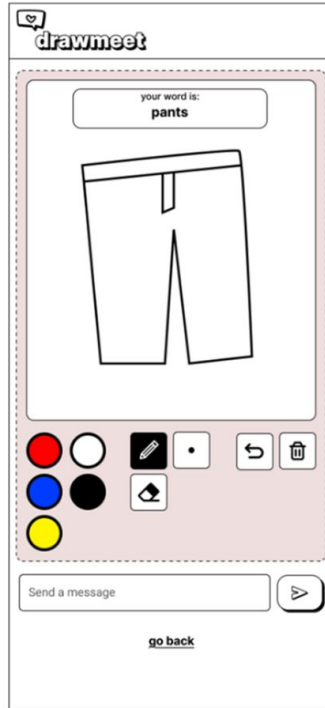
Response? —Reject→ Skip/Pass
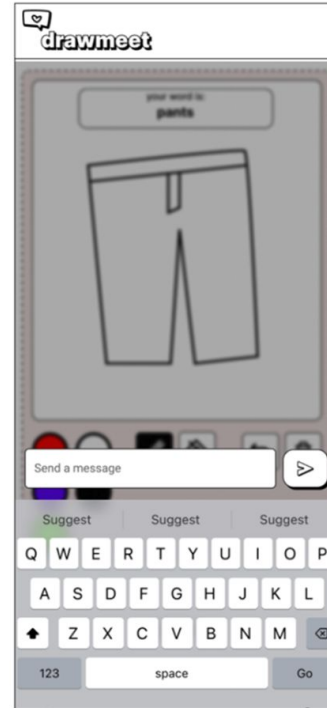
user interface - sending a play
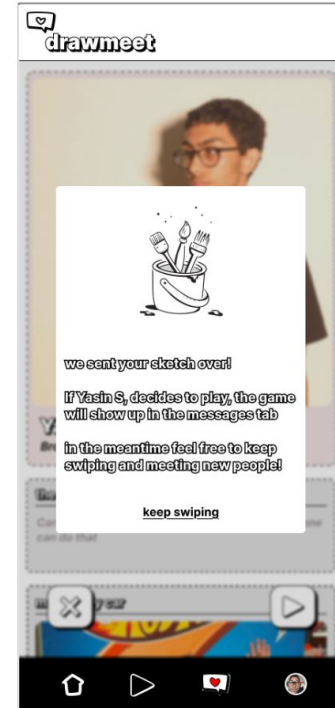

press the X to see new people

press the play button if you would like to send a like
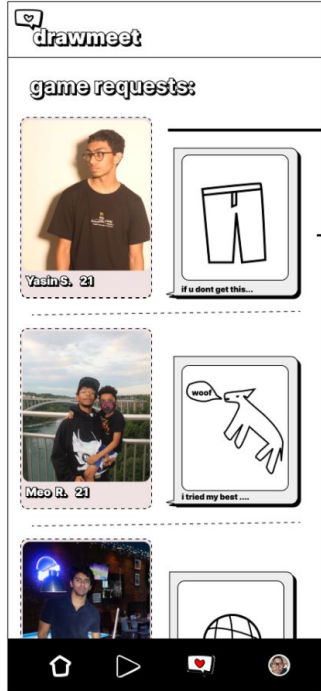
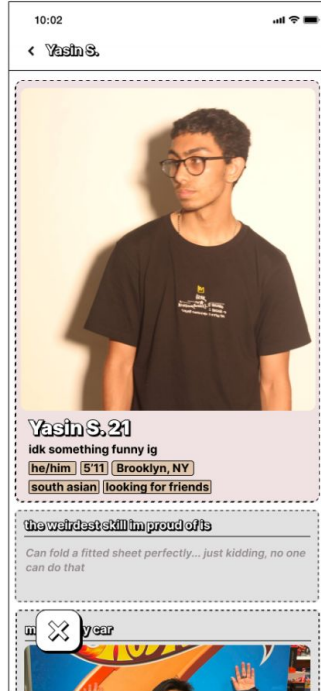send a drawing according to the prompt

send a message (optional)

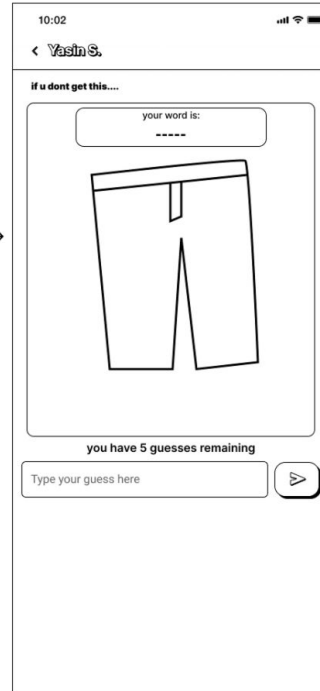you will receive a message that lets you know that the drawing was sent.

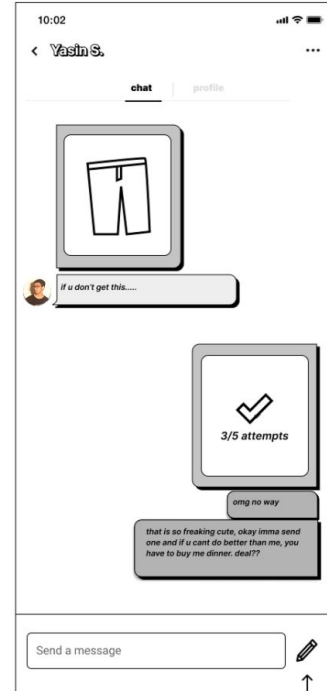# user interface - starting a conversation

view all of your game requests

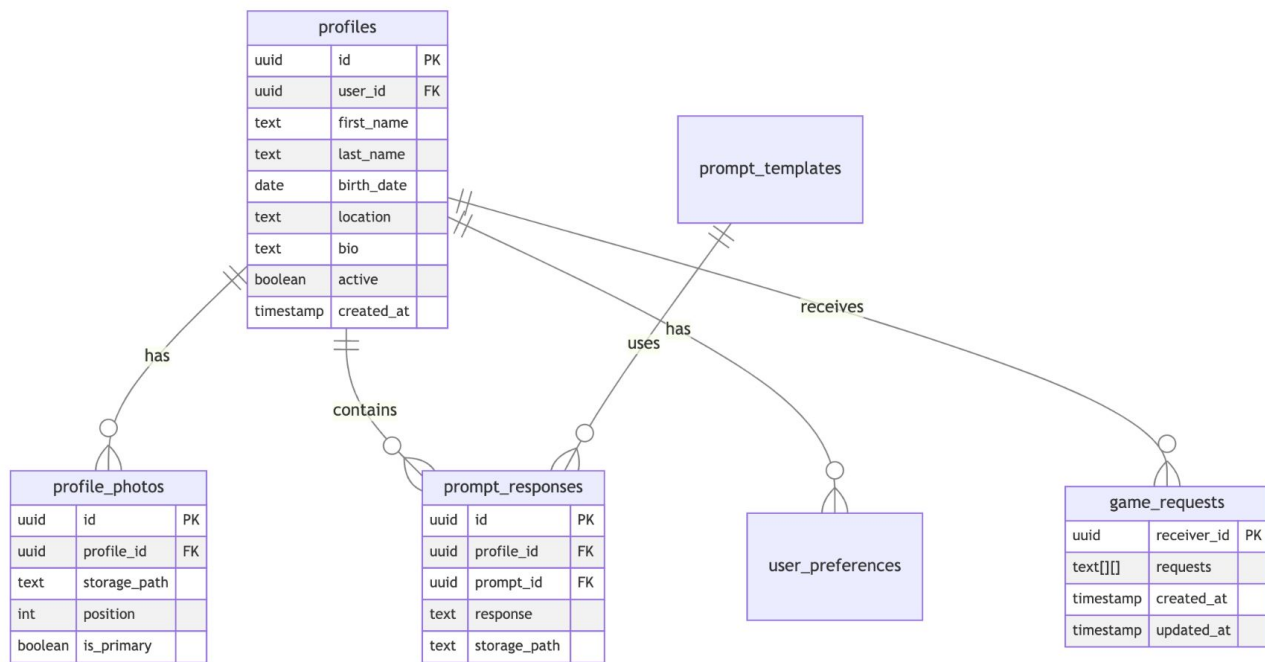if you click their picture you will be able to view their profile. u can click the x to reject them

guess what someone drew, 5 get up to 5 guesses. fewest amount of guesses wins

chat screen. press the pencil and you will be able to send a drawing for them to guess

database schema

profiles

uuid	id	PK
uuid	user_id	FK
text	first_name	
text	last_name	
date	birth_date	
text	location	
text	bio	
boolean	active	
timestamp	created_at	

prompt_templates

profile_photos

uuid	id	PK
uuid	profile_id	FK
text	storage_path	
int	position	
boolean	is_primary	

prompt_responses

uuid	id	PK
uuid	profile_id	FK
uuid	prompt_id	FK
text	response	
text	storage_path	

user_preferences

game_requests

uuid	receiver_id	PK
text[][]	requests	
timestamp	created_at	
timestamp	updated_at	

has

contains

uses

has

receives

tools

Supabase

used for user database SQL tables and storage for user uploaded images

Twilio Verify

used for OTP phone authentication for account creation and login

Google Cloud Platform

used for implementing Google authentication for account login

Flutter Framework

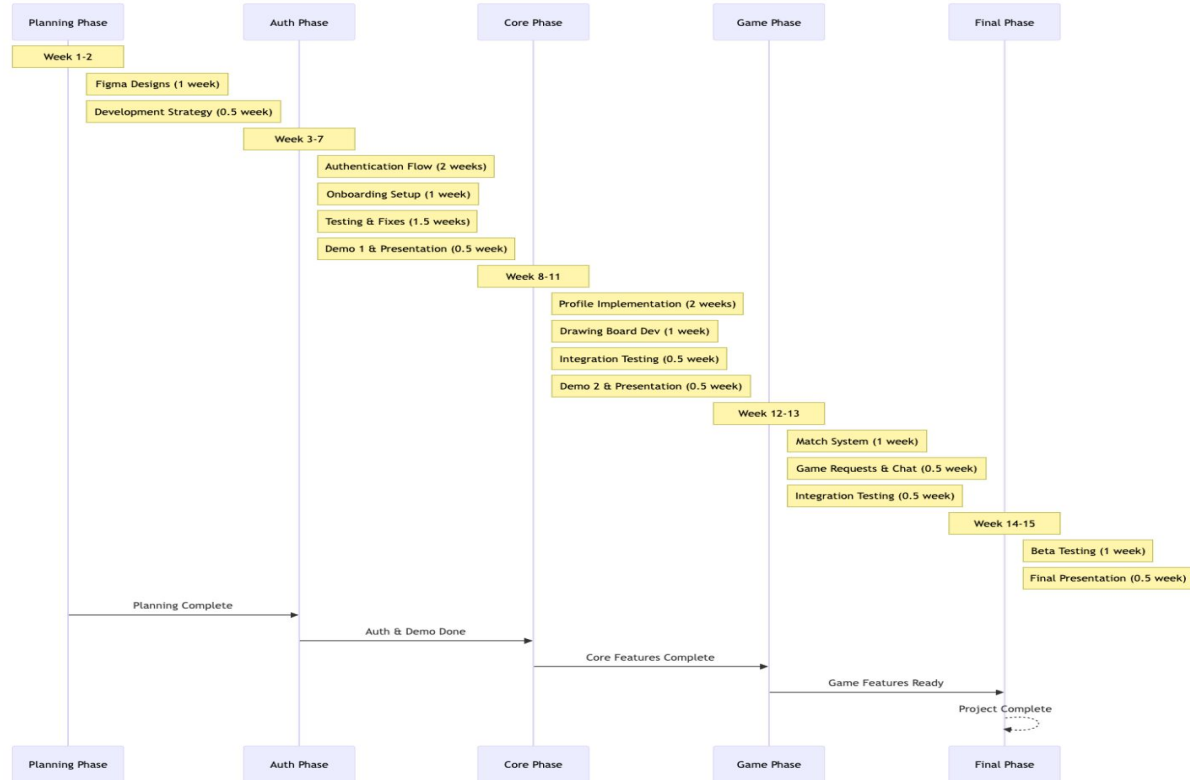**used for developing multi-platform front-end on Android and iOS**

Github

used for having version control as we work on the project

Firebase

used for sending users push notifications

tentative schedule:

data sources:

Our primary data sources come from user-provided information and interaction data. During the registration process, users will provide essential details such as their name, age, gender, bio, and dating preferences.

This information is collected through the Flutter-based interface and securely stored in Supabase.

Users will express interest by creating and sending a Pictionary-style drawing to the person they are interested in.

The recipient must then guess what the drawing represents. Once the recipient accepts the match request—regardless of whether their guess is correct or not—both users are notified and can begin chatting.

All drawing data, guesses, and timestamps are stored in Supabase, ensuring real-time updates and seamless communication. This interaction data is essential for maintaining the core functionality of the app.

use cases:

User Registration and Profile Creation

- **Input:** New user signs up with email and password.
- **Process:** Supabase authenticates the user and stores profile details like name, age, bio, and preferences.
- **Output:** A personalized profile is created and visible on the app.

Matching and Swiping

- **Input:** User swipes right or left on other profiles.
- **Process:** Flutter UI updates the swipe action, and Supabase tracks the swipe data.
- **Output:** If two users swipe right on each other, a "match" notification is sent.

In-App Messaging

- **Input:** User sends a message to a match.
- **Process:** Supabase handles real-time message syncing.
- **Output:** Message appears instantly on both devices.