

drawmeet

draw a connection

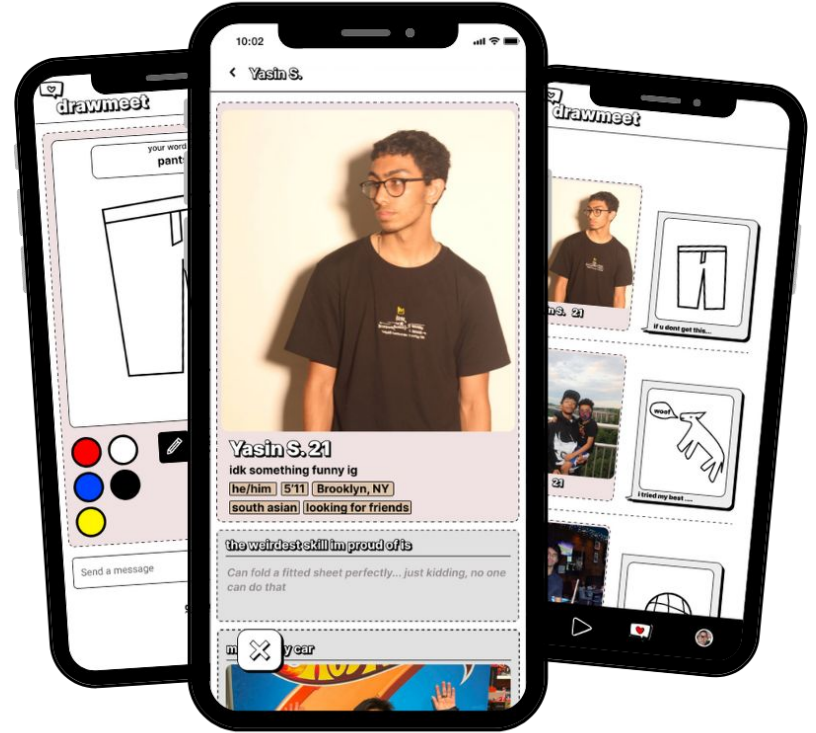
team:

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

David Lema (VC1C) lemadavid1234@gmail.com

supervisor:

Aaryan Shrestha aaryanrajshrestha1@gmail.com



team

David Lema

In charge of both frontend and the backend code through the Flutter framework to ensure the app UI is made like the Figma designs provided and that functions to supabase are being called correctly using the provided API

Bryan Zacarias

Is currently in charge of both backend code through the Flutter framework and database management in Supabase to ensure SQL tables are made to handle our user base and any our cloud storage is able to contain larger amounts of images

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



ClickUp

used as our project management platform



Flutter Framework

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



Github

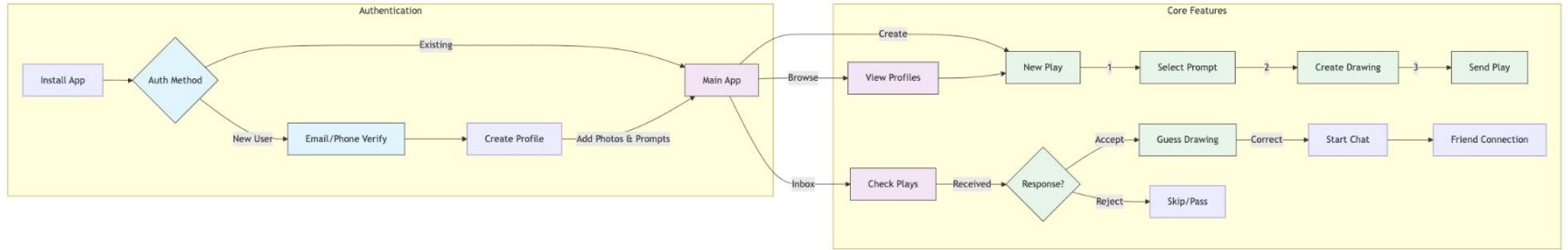
used for having version control as we work on the project



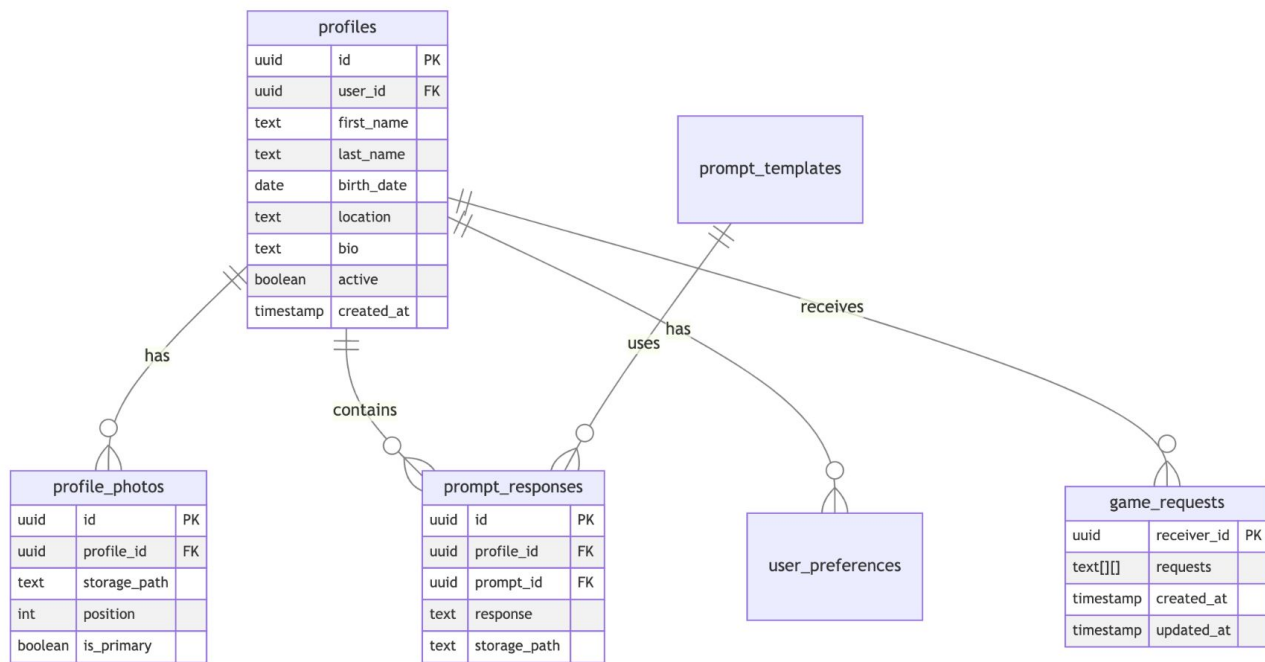
Firebase

used for sending users push notifications

user flow



database schema

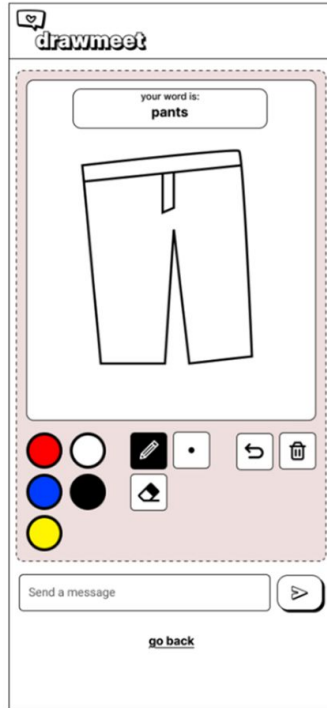


user interface - sending a play

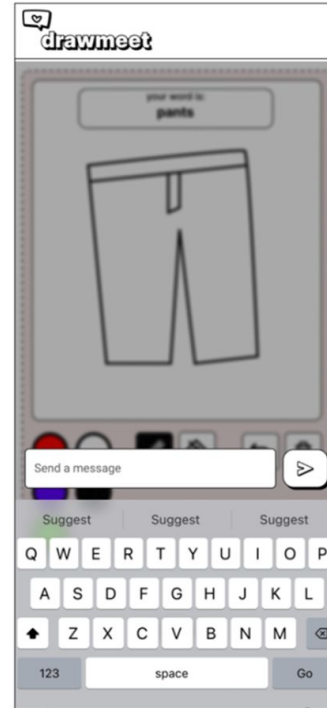


press the X to see new people

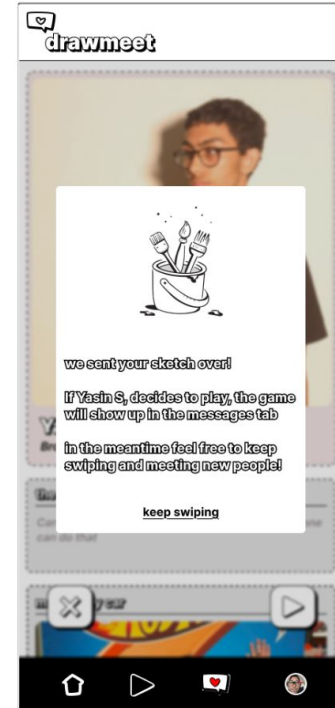
press the play button if you would like to send a like



send a drawing according to the prompt

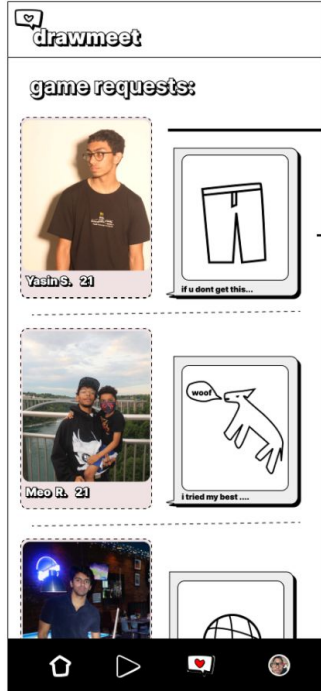


send a message (optional)

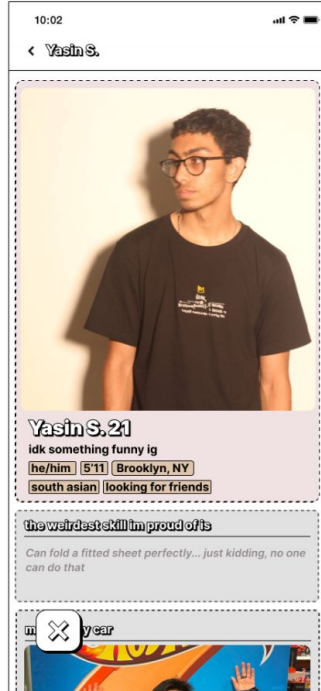


you will receive a message that lets you know that the drawing was sent.

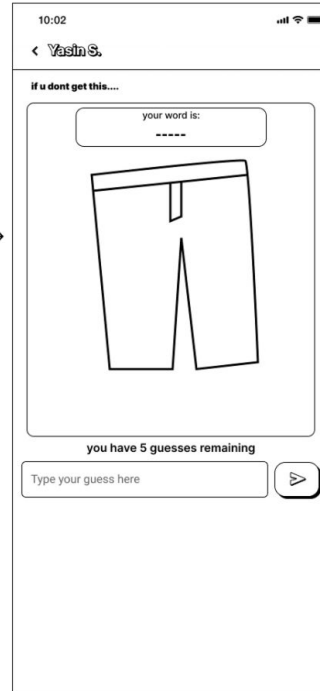
user interface - starting a conversation



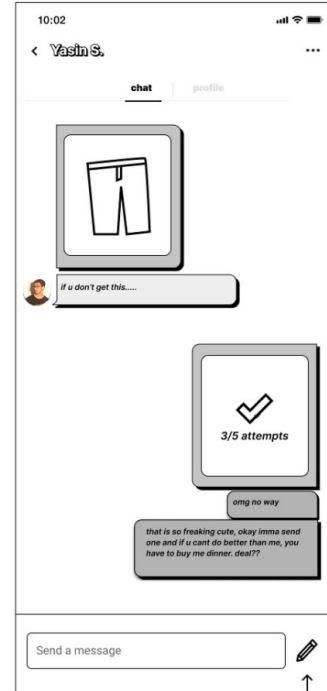
view all of your game requests



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

before

Initially the app was supposed to be an app where you met people and played a drawing game with them via video call.

The idea was you would be able to foster better connections on the phone rather than text. You were still able to text someone after you match but the main games would be played on the phone.

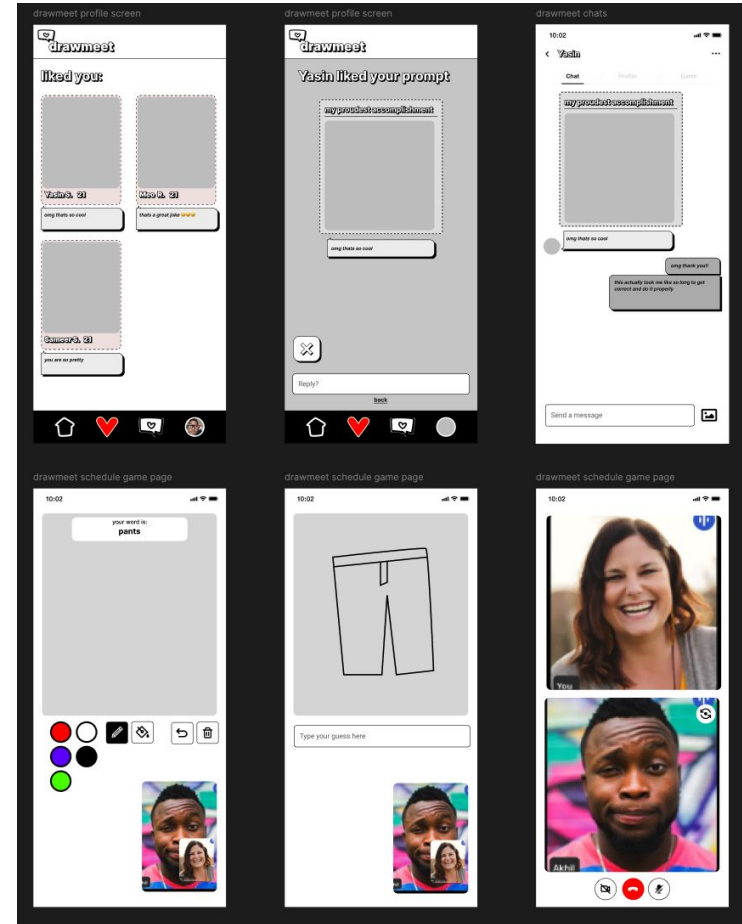
This lead to some issues, the obvious one being people don't feel comfortable hopping on calls with strangers. Also issues around moderation came about. We didn't want any inappropriate things to happen when users were on a video call

Which led us to change a few things around.

our big pivot

So originally we had:

- Likes similar to Hinge where you were able to respond to messages received based on your prompts
- Once you were chatting with someone then you can hit the game tab on the top of the messages and hop on a video call and play the pictionary game



after

After we decided on the pivot, we had to fix a few things. We decided to let users send a drawing game first before hopping on chat and basing it on something like imessage games.

A “like” would mean more in our app because someone took time into drawing something and the other person took time into guessing what the drawing was.

We had to decide what a win would look like in the system so we added a guess limit. You can also send additional games

We also realized that if someone would wanna hop on a call, more often than not they are exchanging phone numbers so implementing a call feature would not be the best use of our resources.

ux/ui

we chose a **monochrome color palette** in order to ensure a sleek, modern, and minimalistic look. this ensures that the user's attention is **focused** on people's profile picture, prompt photos, and **more importantly**, their **drawing request**.

the use of **dashed borders** gives the interface a creative, **sketch like aesthetic**, which mimics the **feel of a drawing**, making app feel more dynamic and **engaging**.

adding an **offset shadow** to buttons creates a slight 3D effect, making them more noticeable and **inviting to tap**. this guides the user towards important actions such as continuing with onboarding, accepting a game, or customizing their profile.

start searching

buy more plays

get drawmeet+

we added a **colored background** to the main profile picture to make user's key **information stand out** at a glance. it enhances the apps **usability** while **maintaining** a clean, modern aesthetic.

we added various **hand drawn characters** to enhance the playful and artistic theme of Drawmeet. they also serve to **add personality** to the app, making it feel more inviting. they also are purposefully **monochrome**.



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

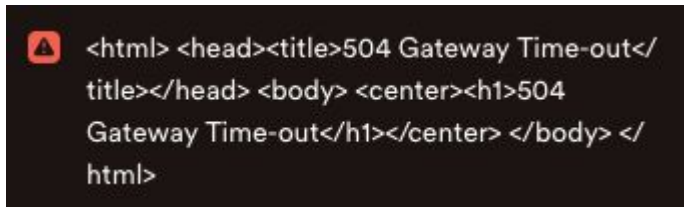
- **Input:** User swipes right or left on other profiles.
- **Process:** Flutter UI updates the swipe action, and Supabase tracks the swipe data.
- **Output:** If two users swipe right on each other, a "match" notification is sent.

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.

setbacks

February presented us with a series of errors that or both David and I at the time were out of the blue. Code that had worked the day before suddenly didn't with no clear explanation, even after consulting multiple forms which didn't seem to provide and solutions either. We reached out to the support staff at Supabase to provide us with some answers and possible solutions for these new errors.



```
[ERROR:flutter/runtime/dart_vm_initializer.cc(40)] Unhandled Exception: Exception: Failed to upload game image: StorageException(message: <html>
<head><title>504 Gateway Time-out</title></head>
<body>
<center><h1>504 Gateway Time-out</h1></center>
</body>
</html>
, statusCode: 504, error: null)
#0      uploadGameImage (package:drawmeet/core/image_processor.dart:240:5)
<asynchronous suspension>
#1      _DrawingScreenState._saveDrawing (package:drawmeet/screens/chat/drawing_screen.dart:545:5)
<asynchronous suspension>
```

setbacks

That was when we were hit with the unexpected news that the east coast servers at Supabase had encountered issues that the root of our issues. This caused a huge pause in the coding process for David and I that was met us questioning the sustainability of the app if we were to encounter errors like this more often in the future.



SUBSCRIBE TO UPDATES

Unhealthy Host in Connection Pooler Load Balancer Affecting Projects in us-east-1

Investigating - - We are still seeing elevated connection error rates for some customers

Mar 08, 2025 - 00:01 UTC

Identified - Some users are still seeing intermittent connectivity issues with Supvisor in us-east-1. The team is continuing to work on a fix.

Mar 08, 2025 - 00:00 UTC

Update - - We are still seeing some connection timeouts for a few customers

Mar 07, 2025 - 23:00 UTC

Monitoring - - Our AWS health check failed and rebooted a Supervisor node

- This reboot caused some timeouts for new connections for some customer connection pools

- We have added resources to the cluster to prevent further node reboots

- Timeouts have subsided after ~45 minutes

Mar 07, 2025 - 22:10 UTC

Investigating - We are performing some emergency maintenance on our connection pooler cluster in us-east-1.

New connections may timeout during this period.

Expected maintenance period is roughly 30 minutes.

Mar 07, 2025 - 21:40 UTC

github repository

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

clickup board

Reach out for access.

<https://app.clickup.com/9013614663/v/li/901307523619>