

# **Discord SET Bot**

## Midterm Project Report

Alex Algazi  
asa10@hood.edu

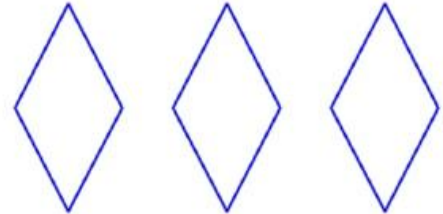
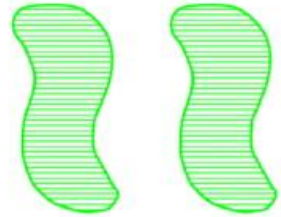
4/4/22

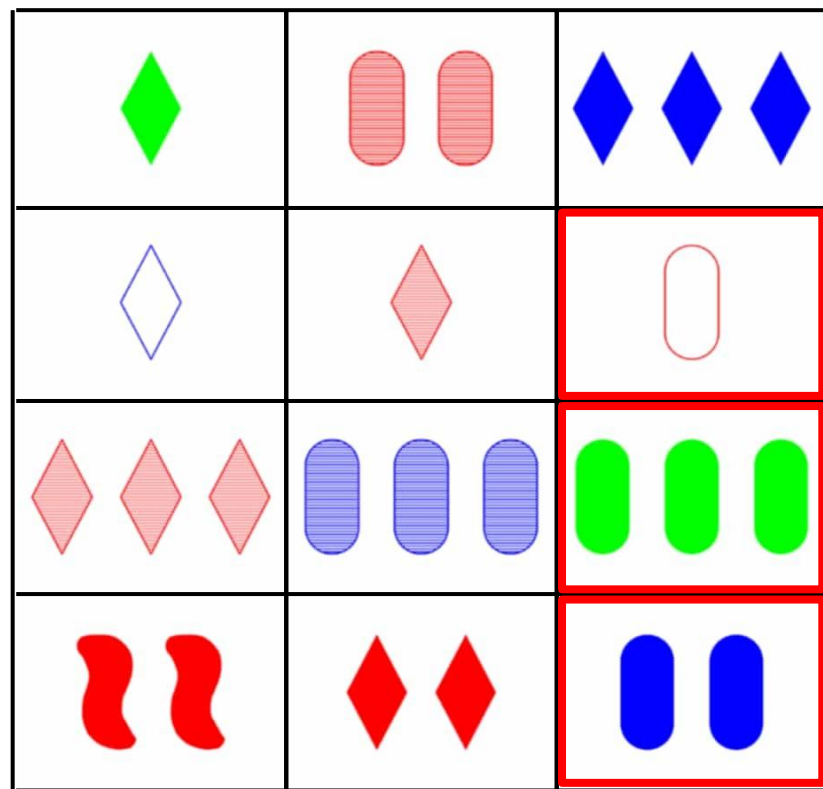
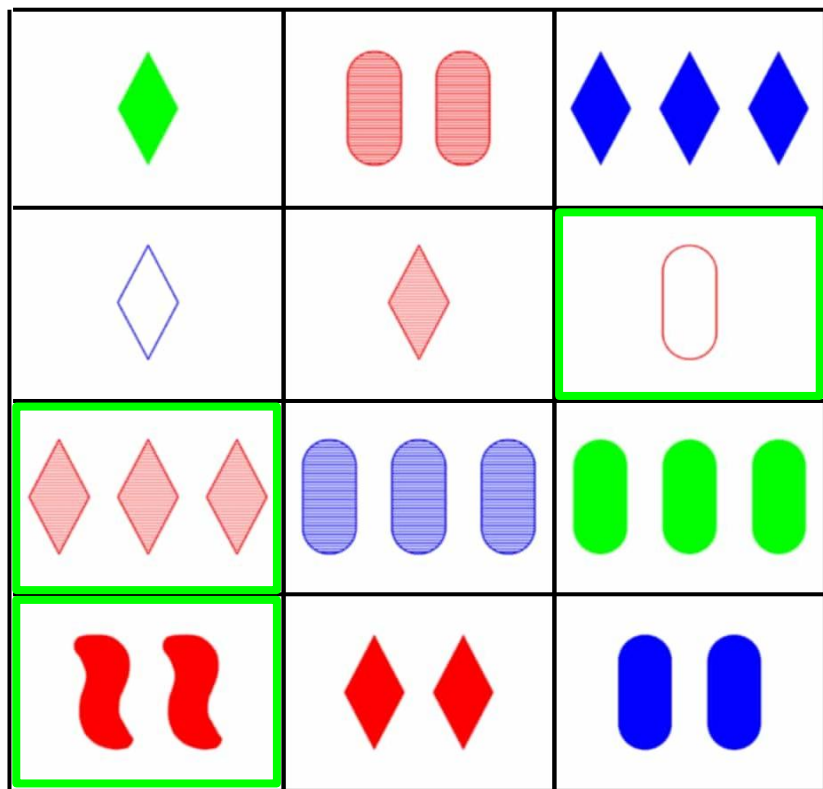
# What is Discord?

- VoIP/instant messaging platform (2015)
- Communities: guilds, or “servers”
- Extensive API for bots
  - Bots range from call-and-response text messaging to music streaming
- Initially popular among video game enthusiasts, now general purpose
- Bots introduced with official API (2016)
- Notable examples of game bots include:
  - Pokétwo - emulates the experience of playing a Pokémon game
  - EPIC RPG - a bot which has dungeons and an in-game economy, as well as PVP
  - Gamebot - collection of games, including Chess, Connect 4, and Cards Against Humanity
- However, no one has ever attempted to make a bot for the SET card game...

# What is SET?

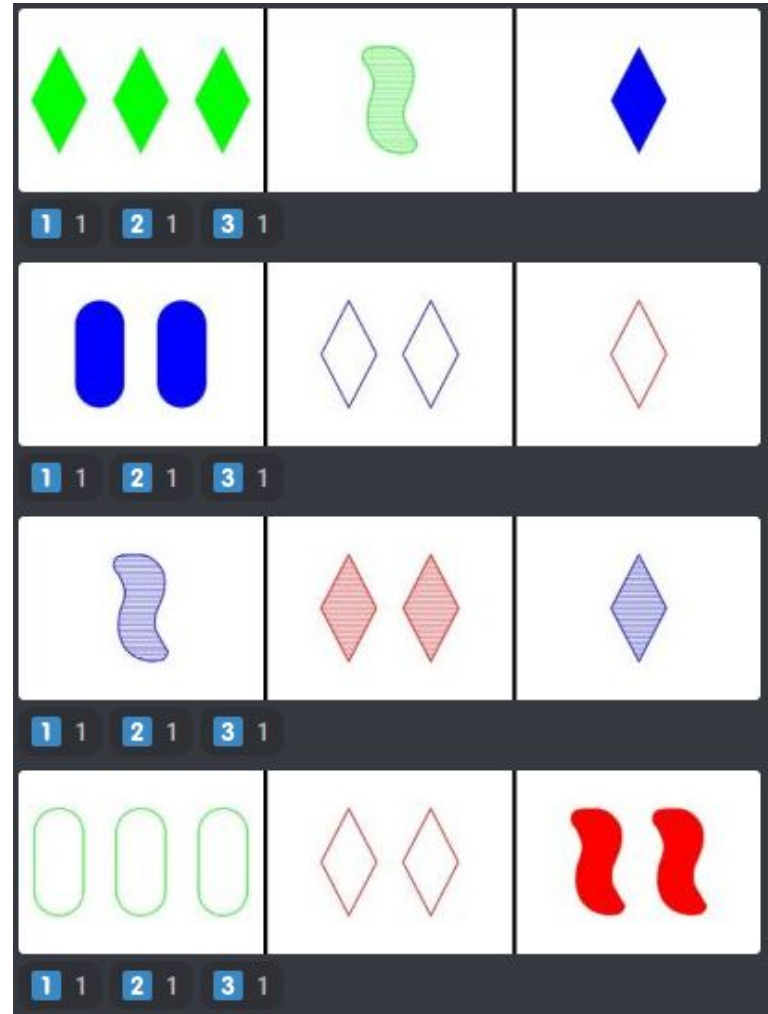
- Pattern-recognition card game, invented in 1974
- 4 attributes: number, shading, color, and shape
- Three possibilities for each attribute:
  - Number: One, Two, Three
  - Shading: Solid, Striped, Open
  - Color: Red, Green, Blue
  - Shape: Oval, Squiggle, Diamond
- 81 cards in the deck, one for each combination ( $3^4$ )
- A set is a group of three cards where each attribute is **either** the same for all three cards **OR** different for all three cards
- Players compete to find sets on the board





# Objectives

1. Generate board visible to all
2. Recognize sets using arithmetic
3. Check boards for sets, add 3 if no set
4. User inputs using Discord emojis
5. Regenerate the board after set is found
6. Show user scores in a leaderboard
7. Track relevant statistics in each guild



# Project Plan

- Create a game loop
- Encode the cards mathematically
- Stitch images of the cards together into rows
- Use reaction collectors to record user inputs
- Check the board for sets efficiently
- Store game data in a JSON file
- Send game data to a database
- Host the bot on Hood's "pluto" server

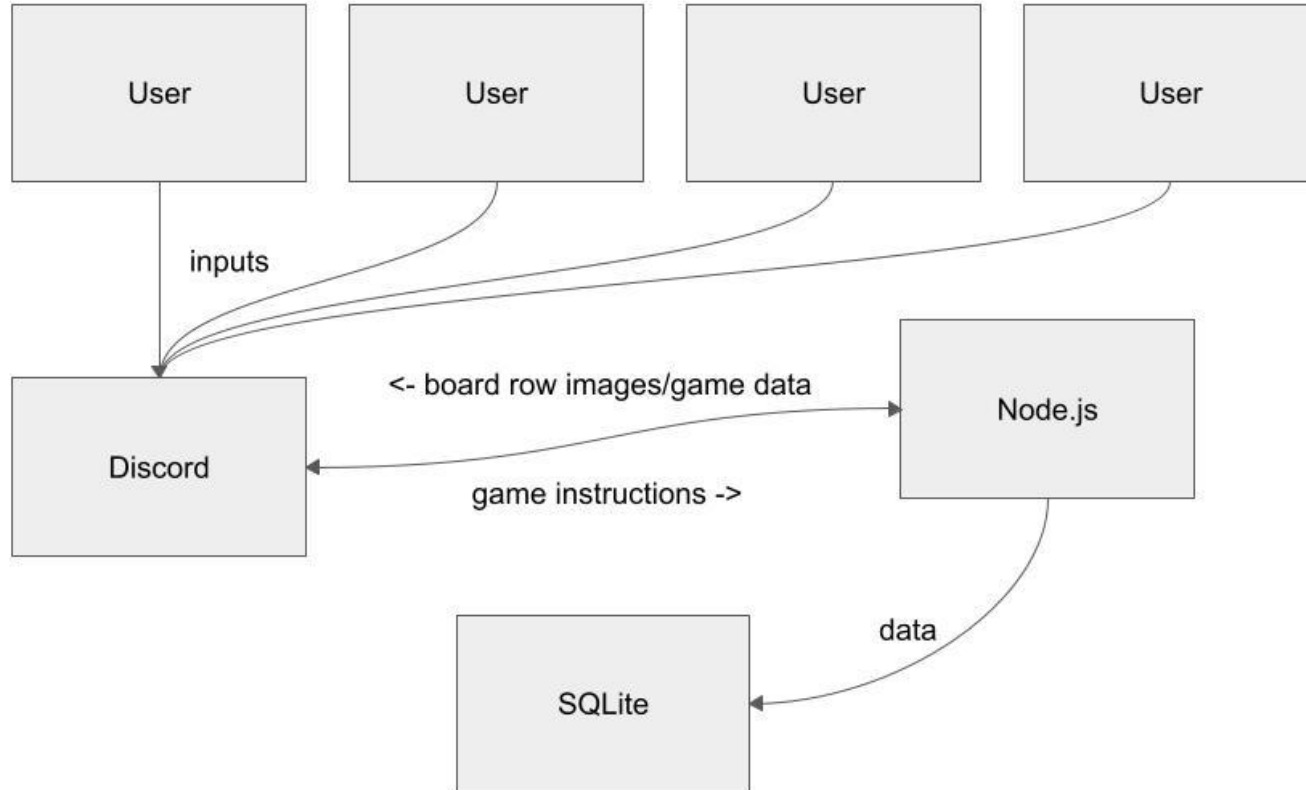
# Tech Stack

- JavaScript, for the game logic
- Node.js, a server-side runtime environment
- Discord.js, a Node framework for the Discord API
- SQLite, to store game data in a discrete file
- pluto@hood.edu, as a host for the bot and all its data

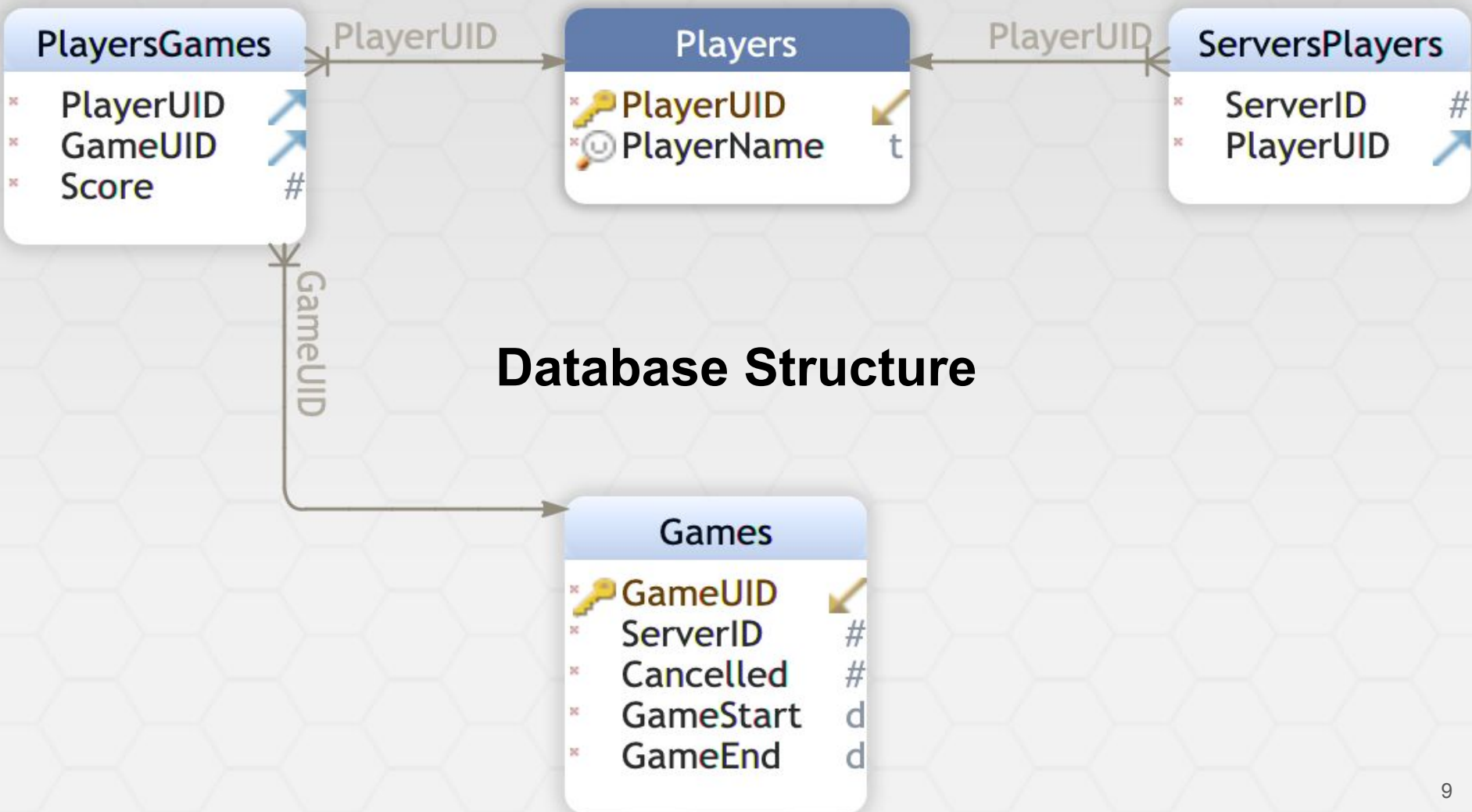
## Required Functionality/Features

- “/newgame” command starts game, generating a board
- Players select cards; when set is found, replace used cards
- Game continues until deck is empty or game is cancelled
- Scores displayed, and saved to DB

# Architecture







# Database Structure

# Accomplished Work

- Besides the database, all functionalities are implemented
  - Game loop achieved through recursion
  - Cards encoded using four digits
  - Images composited using images.js
  - Six reaction collectors, one for each row
  - Boards checked using optimum pair check
- Bot not yet hosted, but we have a Unix service file
- Main file has 1000+ lines of code
- The project is living up to our initial expectations



kami used /newgame

setbot BOT 03/22/2022

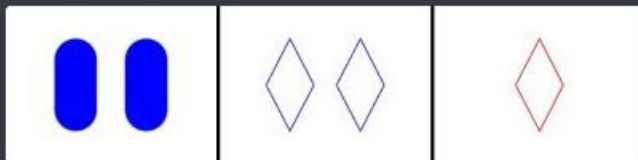
New game! For help, use the "/howtoplay" command.

69 cards remaining. To cancel current game, press the stop sign. (Only the user who started the game can cancel it)

1



1 1 2 1 3 1



1 1 2 1 3 1



1 1 2 1 3 1



1 1 2 1 3 1

Game over!

Scores:

User	Score
-----+-----	
kami#4932	18
Lil Smudge#4494	7

# Effort

Team Member Name	Responsibilities/ Accomplishments	Hours Spent
Alex Algazi	Game loop structure General encoding Image generation Game data storage Unix service	25
Diana Teka	Board set-checker Player scores display Help command “/howtoplay” Database structure	18

## Effort cont.

- Communicated using Discord and email
- Used clockify for timekeeping
- Used Jira to track tasks

Discord is a good platform for communication!

I wish we could have used the in-class time more effectively (more detail in next slide)

# Lessons Learned

- When environment only allows one user, allocate other tasks to other team members. Taking turns does not work
- We needed more time to consider our overall structure. In retrospect, due to our inexperience with the platform, this could not have been easily avoided

# What's Next?

- Database inserts and queries
- Hood pluto server hosting

In loving memory of Diana Teka