Team Name: Catsino
Roster: Haowen Xiao, Ashley Li, Natalie Keiger, Bogdan Sotnikov
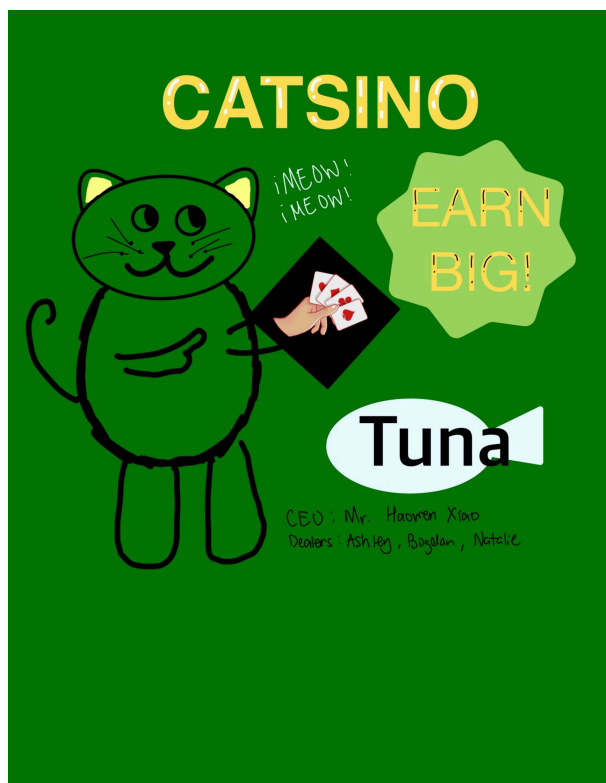**Website name**: Catsino
Numbers: ashley 7182230447, natalie 9173400955, haowen 7189058986, bogdan 9293024747
Usernames: .natalien, ichikami, _ashleee, sotb
TARGET SHIP DATE: {2025-12-23}

Project Description: using **SQLite3, Jinja2, Flask,** and **Bootstrap**, we will build a website where users can log in/create accounts to play casino games (roulette, blackjack, slot machines, and poker). These games will reward them tuna (our currency), which they can use to collect cat icons in the shop.

Task Name	Description	Name	Due	Done
Site map		Ashley	12/2/25	ye ▾
List of program components + description		Natalie & Bogdan Maybe Haowen	12/2/25	ye ▾

API descriptions			Bogdan	12/2/25	ye ▾
Component map			Natalie	12/2/25	ye ▾
DB organization			Ashley	12/2/25	ye ▾
Readme			Haowen	?	no ▾
Roulette	Each number has attributes like value, color, and whether it's even or odd. The values will be stored in a table. Users will have the choice to bet on value, even/odd, and color. A randomizer will select a number from 1 to 36, and the chosen number will be checked for any matches to the bet. Bet on even/odd, red/black/green(1/36), also individual numbers		Ashley		no ▾
Slots			Natalie		no ▾

Roles	Name	Done?
Blackjack	Haowen Xiao	
Roulette	Ashley	
Write poker	Bogdan	
Frontend	Ashley	
Slot machines	Natalie	

Component Descriptions:
★ Sign up:
- Allows users to create a new account
- Checks if user 1) entered both fields and 2) already exists, then allows for account creation
- Checks if account already exists, if not allows for creation and redirects to login page

★ Login
  ● Checks if users 1) entered both fields and 2) already exists, then allows for account login
  ● Redirects to home page, also has a button to lead to signup page
★ Home
  ● Directs to all games (poker, slot, blackjack, roulette) plus logout and store
  ● Welcomes user and displays stats (wins, tuna amount, etc)
★ GAMES: use evil insults API when players lose tuna
    BLJ (Blackjack):
      ● Allows users to play blackjack
      ● Uses card API
    Slots:
      ● Allows users to play on a slot machine
    RL (roulette):
      ● Allows users to play roulette
      ● Players pick between options to bet on: value, color, even/odd
      ● The tuna reward will vary depending on how specific the bet is
          ● for example, value would give the most tuna as a reward because it's the most specific
      ● Losers will receive an evil insult
    Poker:
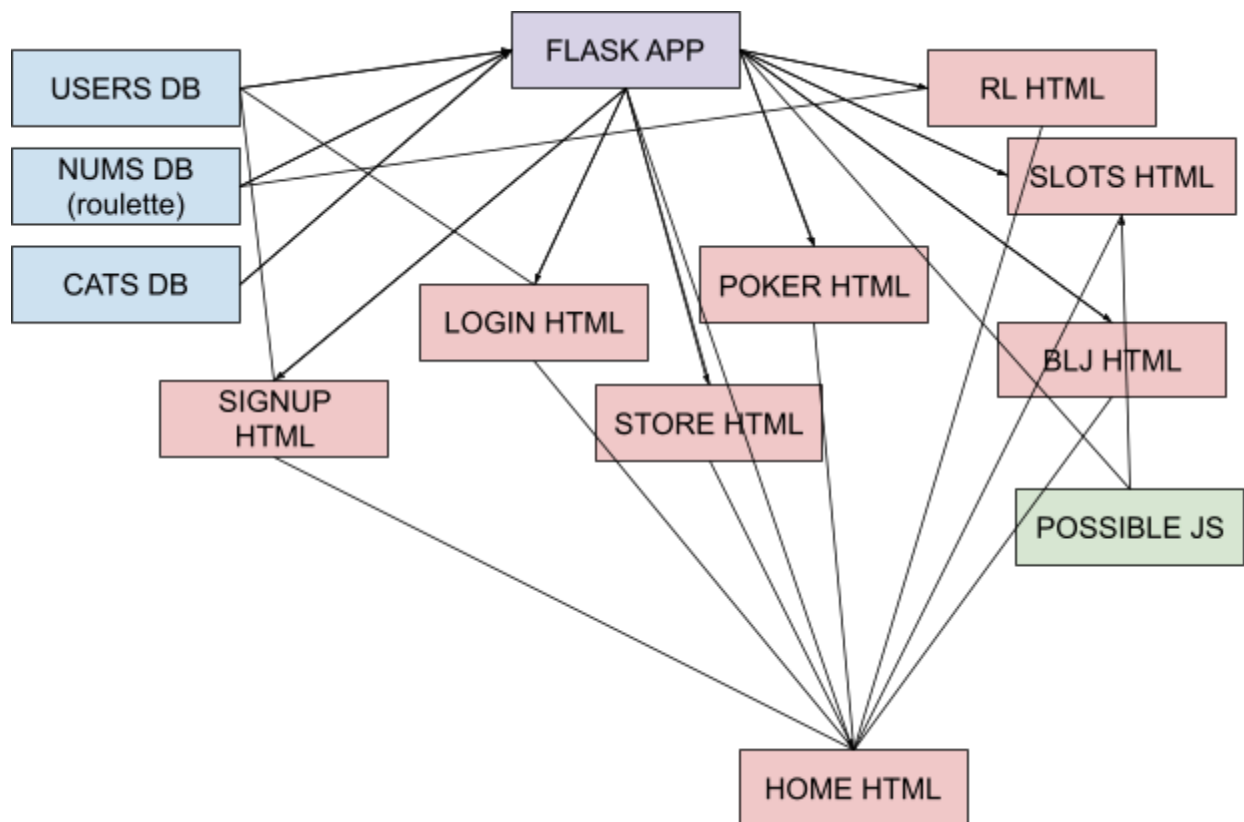   ● Allows users to play poker against other players
      ● Uses card API
★ Store:
  ● Users can purchase cat icons with tuna won from games
  ● Having a new icon costs a set amount (coming from cat API)
★ Currency (tuna):
  ● Users start with a set amount to bet
  ● Allowed to be used in the store to change player icon

Component Map:

APIS:

[Deck of Cards API](#)

Can create multiple decks of cards at a time, draw, and reshuffle them. Can also create piles and partial decks. This will be used to draw cards for Blackjack and Poker. The piles feature can also be used as the different players' hands. Cards can be moved between piles and decks. Interaction with the API is done entirely through GET.

[The Cat API](#)

Returns images and statistics of cats by breed. Can be used to get cat pictures for profiles/players/dealers. Also useful for a potential collectible cat card feature.

[Evil Insult API](#)

Returns an insult and its source by language. Can be used to insult the user for losing.

DB Organization

USERS DB

ID (int)	Username (txt)	Password (txt)	Tuna (int)	Main cat id (int)	Cats (txt)
1	user	pass	12	3	"3 4 5"

CATS DB

id (int)	img (txt)	Rarity (txt)	name (txt)	Cost (int)
3	https://external-content.duckduckgo.com/iu/?u=https%3A%2F%2Fimages.pexels.com%2Fphotos%2F20787%2Fpexels-photo.jpg%3Fcs%3Dsrgb%26dl%3Danimal-cat-adorable-20787.jpg%26fm%3Djpg&f=1&nofb=1&ipt=8d29bf65551e1f35e585ea7cc8e32cf6e5ade45f7cfb08a2a8948bcbcf83dbf8	ultrarare	Jeff	40

Roul DB

This database is created so that each number has assigned attributes. The attributes will later be used in order to check if the player has won the bet based on what they chose to bet on.

ID (int)	Color (txt)	Val (int)
1	red	odd

Site map (BootStrap)