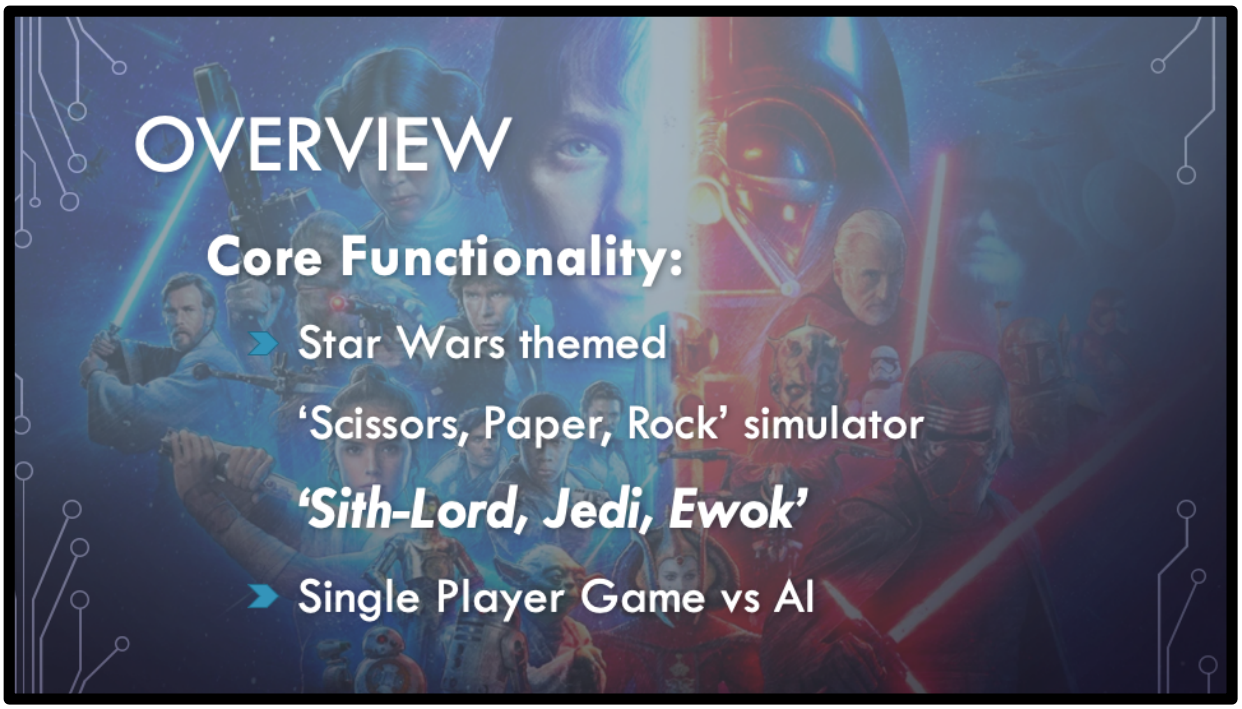
Hi gang,
I’m a Waldow, this is a presentation, and I built an app.

I’m going to cover these three topics today, so strap in.



So what did I build?

Due to the fact I am a huge Star Wars nerd, I built a Star Wars themed 'Scissors, Paper, Rock' simulator, called:

'Sith-Lord, Jedi, Ewok'

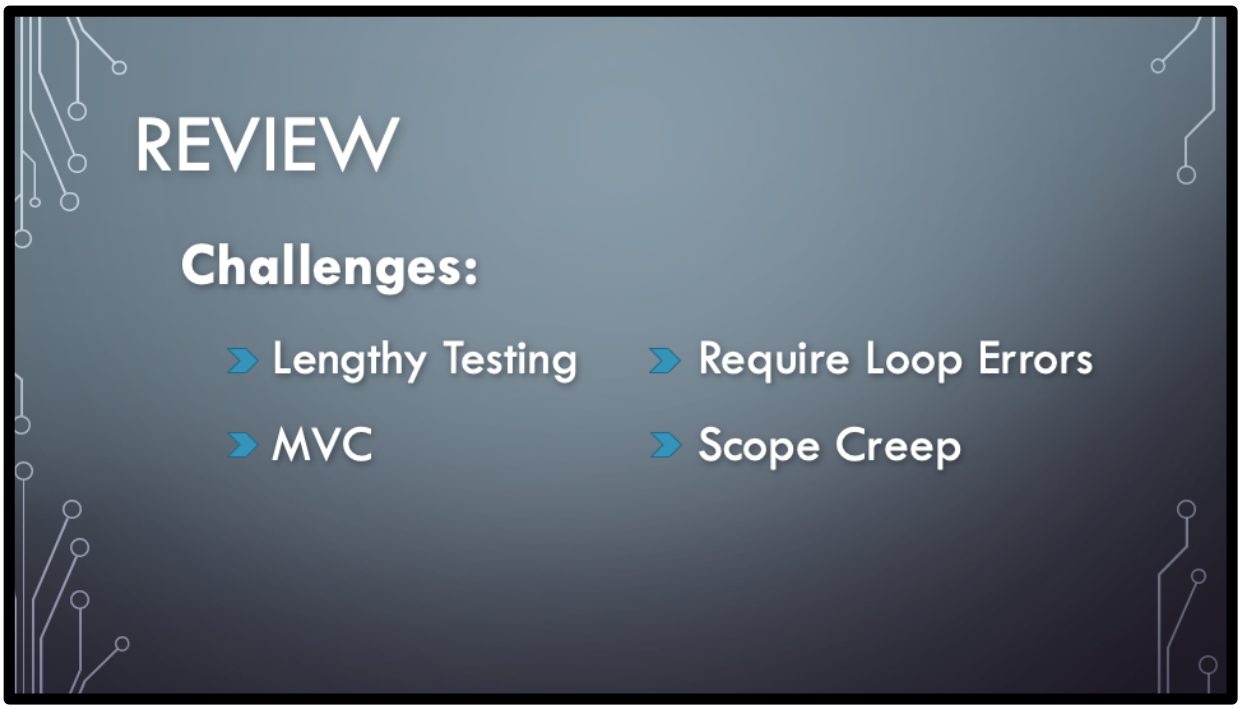
Which works as a single play game, with the player competing against the computer

Features:

From the Main Menu of the game, players can select one of 4 options:

- Play Game: This starts a new game of 'Sith-Lord, Jedi, Ewok'
- Read Rules: Displays the rules and scoring for the game
- View Leaderboard: Displays the top 10 scores for previously played games. This includes a save feature which persists data in a YAML file after the application is quit so players always have access to their previous high scores on their local machine
- Exit: Before exiting players are asked if they are sure they want to quit

Here is a list of all the additional features I implemented to enhance the user experience while using the app or playing the game



Challenges:

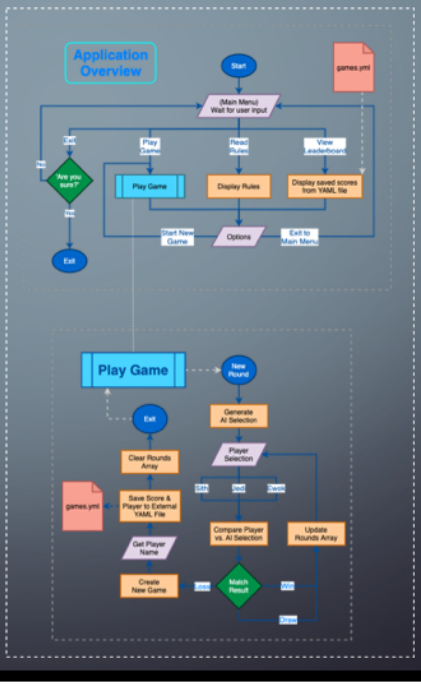
- Lengthy Testing:
  - I initially tried to build my app according to Test Driven Development principles, but this slowed my progress significantly as I wasn't familiar with this practice
  - I quickly abandoned this approach and just started building, which initially appeared to be saving a lot of time, but whenever I wanted to test a feature, I had to do it manually which ultimately would have taken longer than writing tests first
- MVC:
  - I also tried building the app in an MVC structure from my first line of code, but I began running into some serious trouble getting it to work and was wasting too much time with file structure and not building anything
  - Again I abandoned the 'best practice' for app development temporarily to just get my code working which meant most of my code was being written in the controller and nothing was being ported to any view files
- Require Loop Errors:
  - Once I had most of the basic functionality built I started moving all of the input/output tasks into view files. At first this was pretty simple as I could

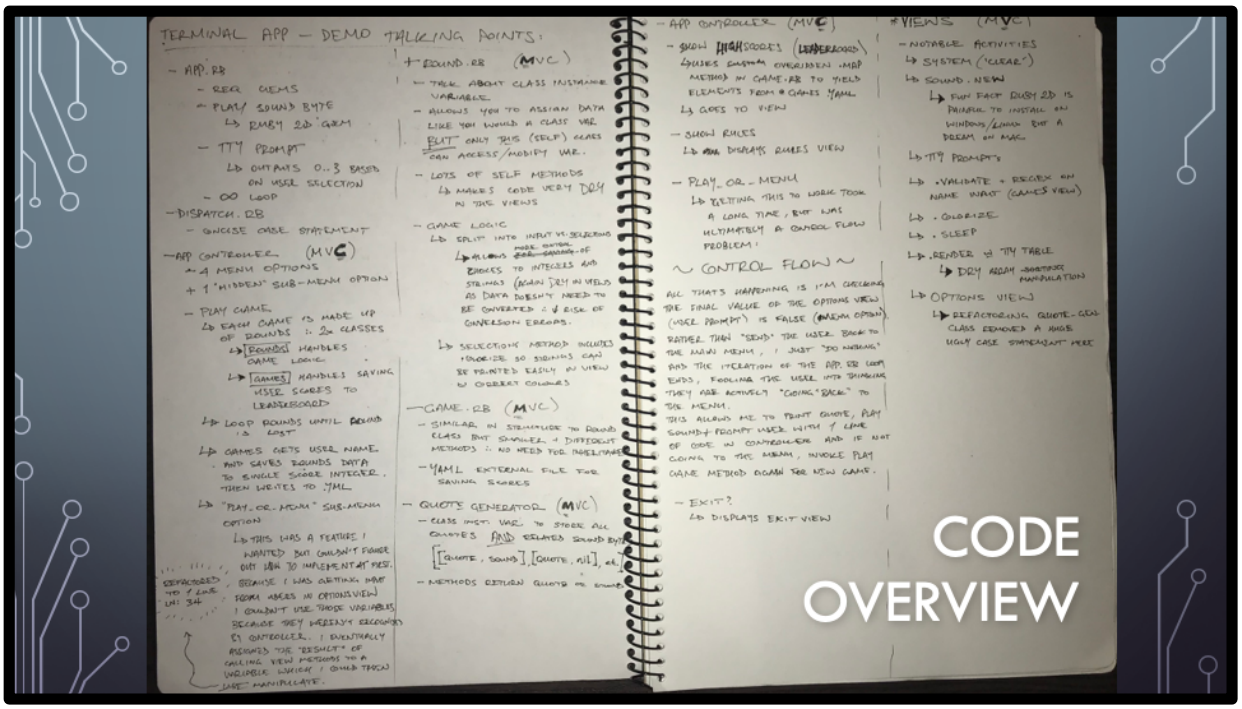
cut and paste chunks of code into new files and just point to them with require_relative.

    - What was challenging, however, was that I didn't know how to hand data back to the controller correctly so I just tried to point the view files back at the controller which ended up creating an infinite loop and crashing the app.
    - I eventually realized I could just save the results of my View files to a local variable and then I was able to easily manipulate the data back in the controller
    - Once I started refactoring my codebase I realized I didn't even need to do this ^ and came up with an even better solution
- Scope Creep:
    - Once I had the basic functionality built, I had buckets of time before the deadline, and was avoiding having to work on the documentation, so I started building tonnes of small features to enhance the user experience. Note to self, don't do this again in future.

Now that’s out of the way, who wants to see a demo?

FLOWCHART


Application Overview
Start
games.yml
(Main Menu) Wait for user input
Play Game
Display Rules
Display saved scores from YAML file
Are you sure?
Exit
Start New Game
Options
Exit to Main Menu
Play Game
New Round
Exit
Generate AI Selection
Player Selection
Clear Rounds Array
Save Score & Player to External YAML File
games.yml
Compare Player vs. AI Selection
Update Rounds Array
Get Player Name
Create New Game
Match Result


TERMINAL APP – DEMO TALKING POINTS:
- APP.RB
- REQ GEMS
- PLAY SOUND BYTE
↳ RARE 2D GEM
- TTY PROMPT
↳ OUTPUTS 0..3 BASED ON USER SELECTION
- OO LOOP
- DISPATCH.RB
- CONCISE CASE STATEMENT
- APP CONTROLLER (MVC)
- 4 MENU OPTIONS
+ 1 "HIDDEN" SUB-MENU OPTION
- PLAY GAME
↳ EACH GAME IS MADE UP OF ROUNDS :. 2x CLASSES
↳ [ROUNDS] HANDLES GAME LOGIC
↳ [GAMES] HANDLES SAVING USER SCORES TO LEADERBOARD
↳ LOOP ROUNDS UNTIL ROUND IS LOST
↳ GAMES GETS USER NAME AND SAVES ROUNDS DATA TO SINGLE SCORE INTEGER. THEN WRITES TO YML
↳ "PLAY_OR_MENU" SUB-MENU OPTION
↳ THIS WAS A FEATURE I WANTED BUT COULDN'T FIGURE OUT HOW TO IMPLEMENT AT FIRST. BECAUSE I WAS GETTING INPUT FROM USERS IN OPTIONS VIEW I COULDN'T USE THOSE VARIABLES BECAUSE THEY NECESSARILY BELONGED TO CONTROLLER. I EVENTUALLY ACHIEVED THE RESULT OF CALLING NEW METHODS TO A VARIABLE WHICH I COULD THEN USE MANIPULATE.
REFACTORED TO 1 LINE OF 34
- ROUND.RB (MVC)
- TALK ABOUT CLASS INSTANCE VARIABLES
- ALLOWS YOU TO ASSIGN DATA LIKE YOU WOULD A CLASS VAR. BUT ONLY THIS (SELF) CLASS CAN ACCESS/MODIFY VAR.
- LOTS OF SELF METHODS
↳ MAKES CODE VERY DRY IN THE VIEWS
- GAME LOGIC
↳ SPLIT INTO INPUT VS. SELECTING
↳ ALLOWS MORE OPTION BECAUSE OF BOOLEANS TO INTEGERS AND STRINGS (AGAIN TRY IN VIEWS AS DATA DOESN'T NEED TO BE CONVERTED :. 0 RISK OF CONVERSION ERRORS.
↳ SELECTIONS METHOD INCLUDES COLORIZE SO SQUARES CAN BE PRINTED EASILY ON VIEW W CORRECT COLOURS
- GAME.RB (MVC)
- SIMILAR IN STRUCTURE TO ROUND CLASS BUT SMALLER + DIFFERENT METHODS :. NO NEED FOR INHERITANCE
- YAML EXTERNAL FILE FOR SAVING SCORES
- QUOTE GENERATOR (MVC)
- CLASS INST VAR TO STORE ALL QUOTES AND RELATED SOUND BYTE [[QUOTE, SOUND], [QUOTE, nil], etc]
- METHODS RETURN QUOTE OR SOUND
- APP CONTROLLER (MVC)
- SHOW HIGH SCORES (LEADERBOARD) USES SAME OVERRIDEN .MAP METHOD IN GAME.RB TO YIELD ELEMENTS FROM 0 GAMES YAML
↳ GOES TO VIEW
- SHOW RULES
↳ DISPLAYS RULES VIEW
- PLAY_OR_MENU
↳ GETTING THIS TO WORK TOOK A LONG TIME, BUT WAS ULTIMATELY A CONTROL FLOW PROBLEM!
~ CONTROL FLOW ~
ALL THAT'S HAPPENING IS I'M CHECKING THE FINAL VALUE OF THE OPTIONS VIEW (USER PROMPT) IS FALSE (MENU OPTION) RATHER THAN "SEND" THE USER BACK TO THE MAIN MENU, I JUST "DO NOTHING" AND THE ITERATION OF THE APP.RB LOOP ENDS, FOOLING THE USER INTO THINKING THEY ARE ACTIVELY "GOING BACK" TO THE MENU.
THIS ALLOWS ME TO PRINT QUOTE, PLAY SOUND + PROMPT USER WITH 1 LINE OF CODE IN CONTROLLER AND IF NOT GOING TO THE MENU, INVOKE PLAY GAME METHOD AGAIN FOR NEW GAME.
- EXIT?
↳ DISPLAYS EXIT VIEW
- NOTABLE ACTIVITIES
↳ SYSTEM ('CLEAR')
↳ SOUND.NEW
↳ FUN FACT RUBY 2D IS PRIMARILY TO INSTALL ON WINDOWS/LINUX BUT A DREAM ON MAC
↳ TTY PROMPT'S
↳ .VALIDATE + REGEX ON NAME INPUT (GAMES VIEW)
↳ .COLORIZE
↳ .SLEEP
↳ .RENDER W TTY TABLE
↳ DRY AGAIN - SHORTHAND MANIPULATION
↳ OPTIONS VIEW
↳ REFACTORING: QUOTE_GEN CLASS REMOVED A HUGE UGLY CASE STATEMENT HERE
CODE OVERVIEW