# Group Project
# Yahtzee
# Web Application

CIS-17B Spring 2023
Danielle F,
Ismael P,
Logan O
Date: 06/11/23

# Table of Contents

# Introduction

Title: Yahtzee

Choose a starting player by any method (oldest player, youngest player, highest roll of the dice, etc.)

Beginning with the starting player, players will take turns one at a time in clockwise order. The game consists of thirteen rounds, and at the end of the thirteenth round, the game will end. (All the categories on the players' scorecards will be completely filled in at that point.)The dice can be scored after any of the rolls, but scoring the dice ends the player's turn. Setting dice aside after one roll does not prevent one or more of them from being rolled again on any subsequent roll if the player so chooses.

Each player's goal is to try and score as high as possible in one of the thirteen categories shown on their scorecard.

To score the dice, the player selects one of the categories on their scorecard and writes the score into it. Each category can be scored only once per game (except for the Yahtzee category). Categories can be filled in any order. A player must score the dice on their turn even if it turns out that there are no good categories remaining to score in. Once a category is filled it may not be changed.

A player may write a score of zero in any category if they have rolled no point-generating results or simply choose to do so. After marking their score on their scorecard, the player's turn ends, and play proceeds to the player on their left.

# Summary

Project Size: ~ 2,496 lines

The User and Admin side of this project was done by Ismael, the Game portion of the project was done by Danielle, and the Admin controls were done by Logan. When first starting the program, it called the database in order to fill the cookies and local storage with the user information. There is a slight problem when filling the local storage and cookies where you need to click the start button twice until the alert that the game is ready to test comes up. After that, you are free to log in with an existing user or create a new one. The Admin information is (admin) for the username, and (password) for the password. For an example user, the login would be (Homer) for the username, and (Homer!23) for the password. When first logging in as the admin, it gives you a menu

that allows you to check all the users, find one by id, find one by email, delete a user, update the admin login information, or go back to the login screen. When logging in as a user, they are presented with a yahtzee game in which at the end of said game their high score is recorded and placed into the database.

# Description

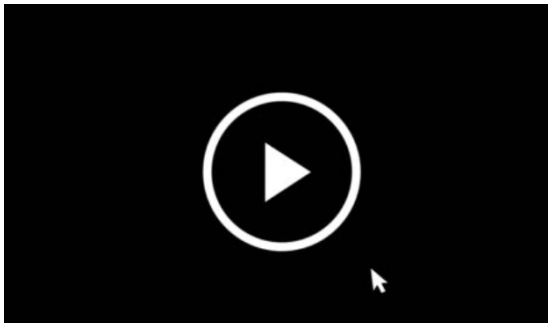
This is a Yahtzee Program, it allows you to play Yahtzee by yourself or play against a friend. Whoever gets the higher score wins, and if the person logged in gets a new personal high score, that score will be saved to the database.

# GitHub Links

- Danielle's Github: https://github.com/koa2019/yahtzee
    - Final/Latest version: yahtzee_php_v8
- Ismael's Github: https://github.com/Error1417/Yahtzee_Project
    - Final/Latest version: yahtzee_php_v8 https://github.com/Error1417/Yahtzee_Project/tree/main/yahtzee_php_v8

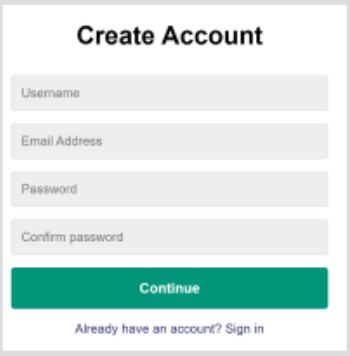
# Video Demo

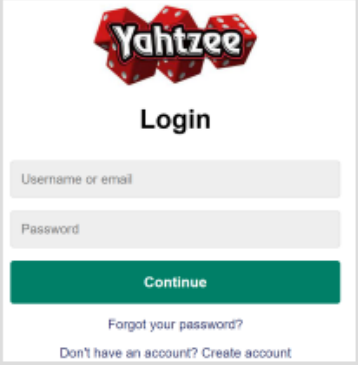
**This is a picture with a link attached to it. The tutorial was done on Mac, so some Windows features might also apply. Needed to keep the video under two minutes, so it's a bit fast.**

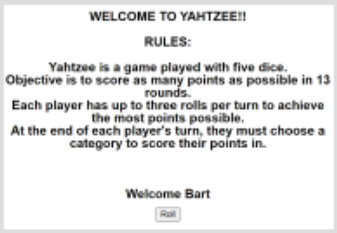


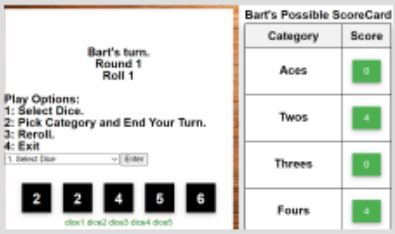
Run configuration picture demo available: See runConfig.pdf.

# Sample Outputs

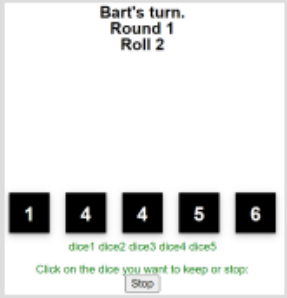



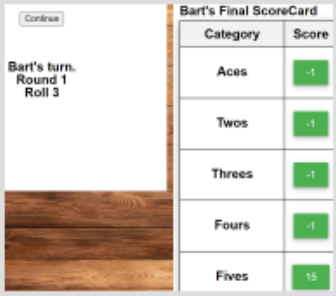




Play Menu

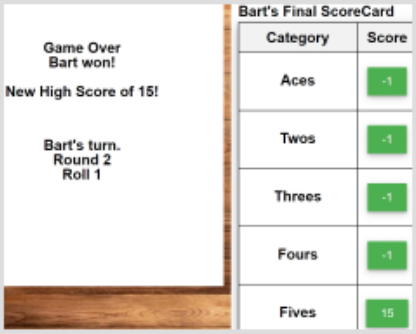



Select Dice


Select a category

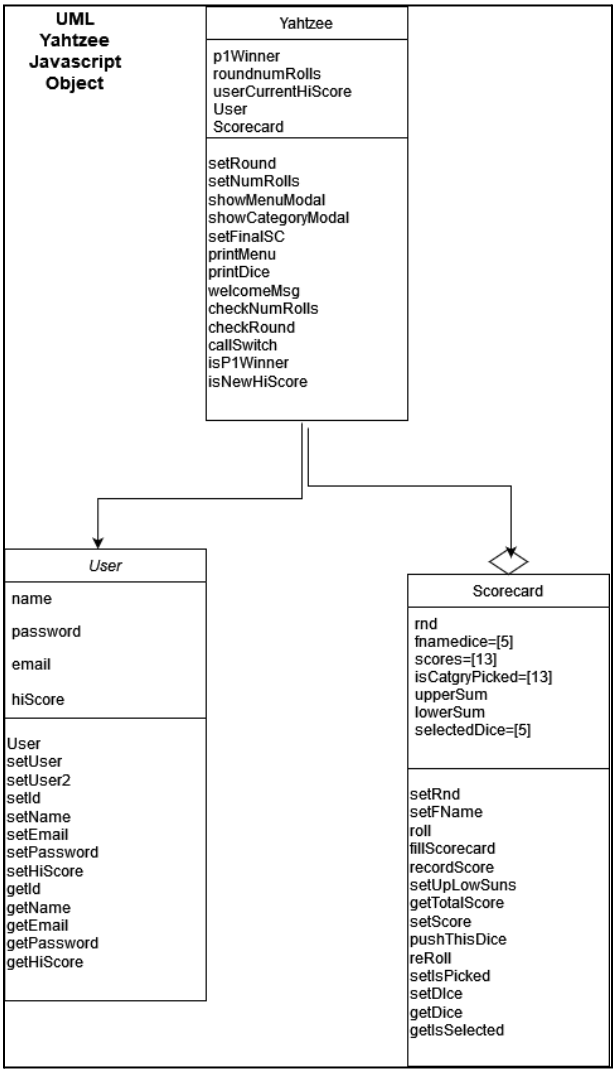



View selected category on final scorecard

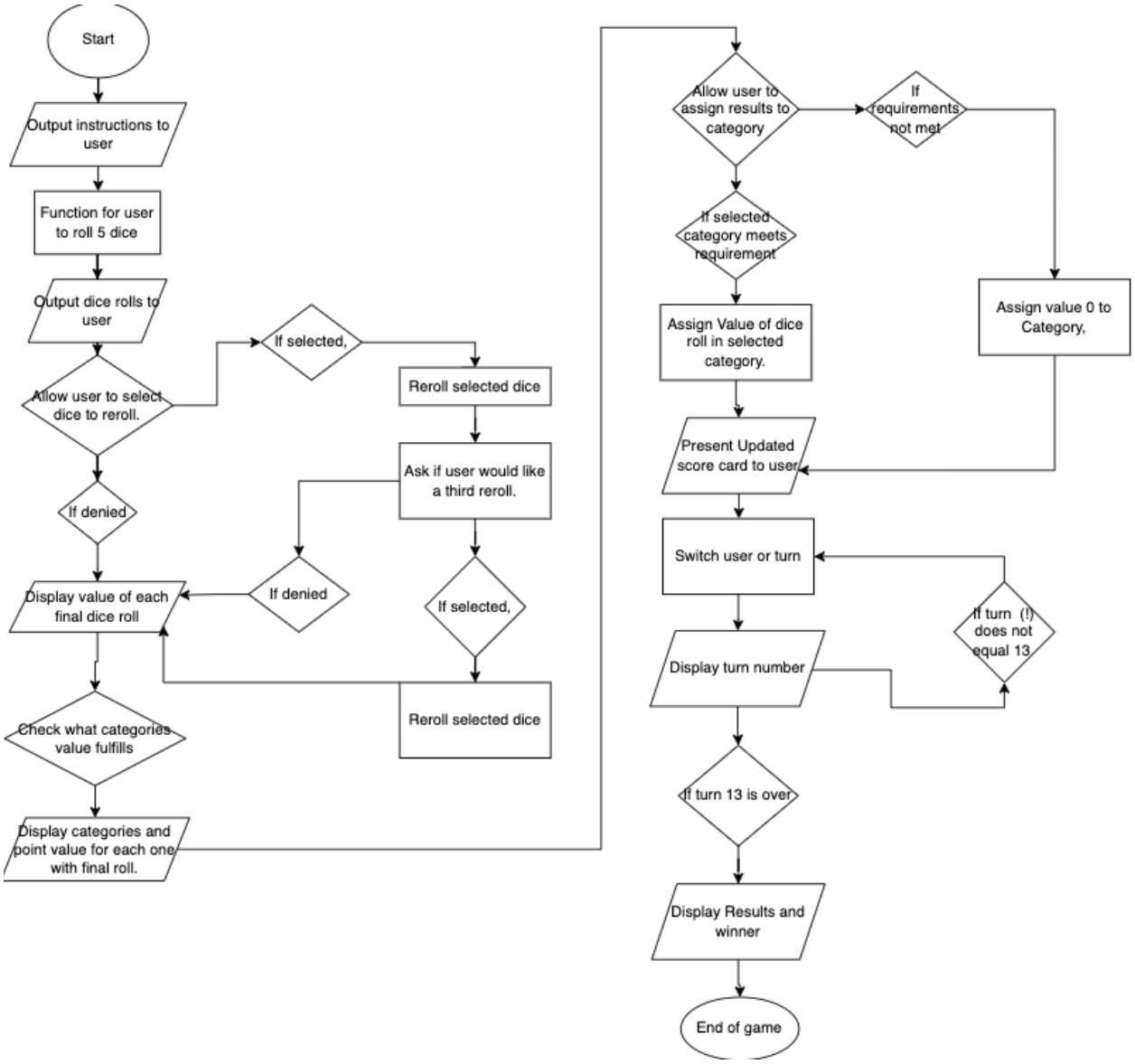



If Player 1 has won and has new high score, then update their profile.

# UML Chart


UML
Yahtzee
Javascript
Object
Yahtzee
p1Winner
roundnumRolls
userCurrentHiScore
User
Scorecard
setRound
setNumRolls
showMenuModal
showCategoryModal
setFinalSC
printMenu
printDice
welcomeMsg
checkNumRolls
checkRound
callSwitch
isP1Winner
isNewHiScore
User
name
password
email
hiScore
User
setUser
setUser2
setId
setName
setEmail
setPassword
setHiScore
getId
getName
getEmail
getPassword
getHiScore
Scorecard
rnd
fnamedice=[5]
scores=[13]
isCatgryPicked=[13]
upperSum
lowerSum
selectedDice=[5]
setRnd
setFName
roll
fillScorecard
recordScore
setUpLowSuns
getTotalScore
setScore
pushThisDice
reRoll
setIsPicked
setDice
getDice
getIsSelected

# Yahtzee C++ FlowChart



# Where to find Concepts

1) MVC - How you delineated your objects

Scorecard.html is dependent on a Yahtzee object. An instance of a Yahtzee object must be created within Scorecard.html's script tag. This creates a global variable for all of the HTML event listeners to reference when they are set and triggered.

The Yahtzee object aggregates the User object to represent a player's profile saved in the database. It provides the game with the user's name and current high score. The Yahtzee object also aggregates two instances of the Scorecard object. One instance represents the player's possible points and the other is their actual scorecard. The Scorecard object handles setting and testing the seventeen different points categories.

| | |
|---|---|
| MVC - How you delineated your objects | ScoreCard.html<br>Lines: 10 - 12 |
| Objects - Javascript/PHP - Serialization | getForm.html<br>Lines: 18, 66 |
| Reading/Writing Files/Local Storage | getForm.html<br>Lines: 41- 56<br><br>getForm.js<br>Lines: 6 - 16 |
| Databases SQL | user_yahtzee_3.odb<br>test_entity_user_yahtzee_2_-_Copy.sql |
| Form Validation | login.js<br>Lines: 30 - 56 |
| User-Admin-Login | login.html<br>getForm.html<br>AdminMenuPage.html |
| Cookies - Sessions - Securing Pages | getForm.html<br>Lines: 18 - 39 |

# Version Descriptions - Ismael

## Version 1- “Html”

- This version of the game is a very bare bones version of the login screen, there is no input validation, and it doesn’t save any information with the local storage. It basically only redirects you to other screens.

## Version 2- “Html_v2”

- This version I added the local storage that saved whatever user you created. We had a problem with saving the @ sign in this version that got fixed in later versions.

## Version 3- “Html_v3”

- This version of the game just focused on making the game look better with css.

## Version 4- “Html_v4.1”

- This version of the game added a lot, we combined the full yahtzee game with the user and admin code. We also added the admin menu and a couple other features.

## Version 5- “Html_v5”

- This version of the code just added input validation among other small features.

## Version 6- “Html_v6.1”

- This version of the code we started adding the php aspects, and I believe this was around the time we started testing with the database. This version also fixed the @ problem with saving to the local storage.

## Version 7- “yahtzee_php_v7”

- This was another big version which added the DBFill feature and the DBSelect feature along with cookies.

## Final Version- “yahtzee_php_v8”

- The final version of the code cleaned up the rest of the code, added the Update php, made a new database and sql that had users already in it so its easier to run for the professor.

# Version Descriptions - Danielle

For all my version comments please see:
danielle_yahtzee_github > docs_html > yahtzee_html_version_notes.txt.

## Yahtzee_html_v6.1-6.2

I finally figured how to aggregate ScoreCard class object within Yahtzee’s constructor and be able to access the instance of scorecard and finalSC throughout

Yahtzee.js. The game utilizes prompts for user inputs, but the html won't display on the browser. I can see everything in the console, but not on Scorecard.html until all prompts have finished running

## yahtzee_html_v7.1

I removed all the prompts. Instead I created divs for user inputs and printed them to Scorecard.html dynamically. The game is not stopping and getting the input from printMenu(), selectDice(), or pickCategory(). It runs through all thirteen rounds in the console without waiting for user input. If I put prompts in, the game will stop when it's supposed to, but it won't display the dice or scorecard on the html.

Event listeners, onclick, need to reference the scorecard objects in order to set the final Scorecard's scores array. To do this there must be a global reference variable to Yahtzee class in Scorecard.html's script tag.

- i.e. setAttribute('onclick', 'yahtzee.diceClick(yahtzee)').

## yahtzee_html_v7.2

Removed all loops and replaced them with if conditionals and put them in their own functions,  added checkRounds() and checkNumRolls() to yahtzee.js. This stopped the game from running through all 13 rounds. Added buttons to control game play. These buttons triggered different modals to appear or disappear on Scorecard.html.

## yahtzee_html_v9

I copied Ismael's yahtzee_v8 and added my yahtzee_v8 to merge the game. Ismael's code contains local storage, cookies, and PHP code to connect with MySQL.

# References


1. Lehr, Mark. "2023_Spring_CSC_CIS_17B · ml1150258/2023_spring_csc_cis_17b." GitHub, 2023, https://github.com/ml1150258/2023_Spring_CIS_CSC_17B.
2. Nixon, Robin. *Learning PHP, MySQL & Javascript: With jQuery, CSS & HTML5*. 5th ed., O'Reilly Media Inc., 2018.