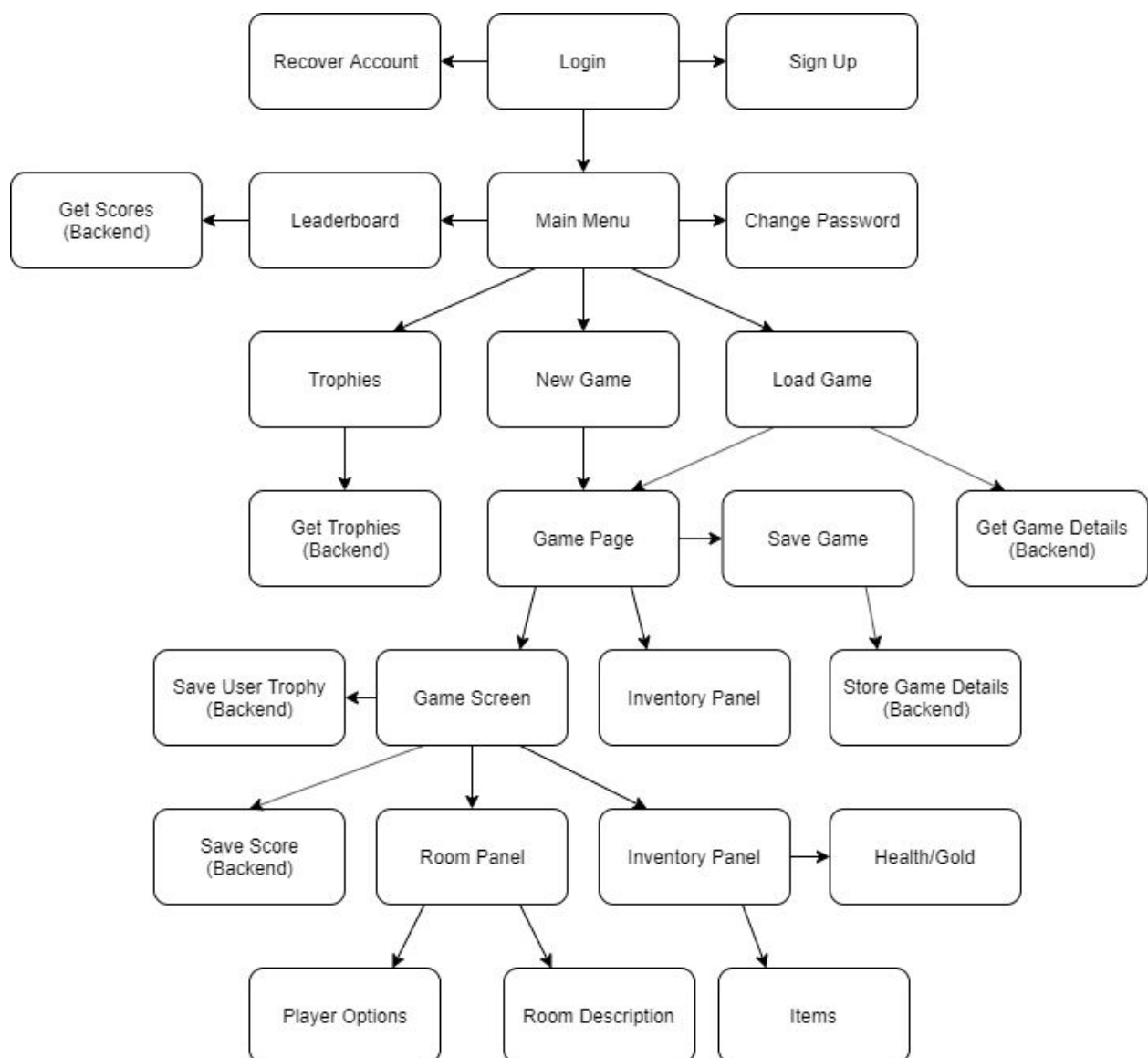


Sprint 2: Incremental and Regression Testing

Julian Ng, Cole Compton, Adrian Taubner, Christian McKain, Jacob Riggs

Note: **Sprint 2 changes and additions will be colored in red**

Component Structure



Modules

Login

- The login module will authenticate users through the backend and allow access to the rest of the site

Sign Up

- The Sign Up module will handle new user creation through the backend and enable users to access the rest of the site through the login module

Recover Account

- The recover account module will send you an email to reset your password given an email

Main Menu

- This module will allow the user to choose other modules to navigate to, including the change password page, the leaderboard, the trophy room, their saved game, and new game.

Leaderboard Module

- The leaderboard will display a certain number of the highest scores so far achieved by any user along with their username and other data about their game.
- Users will be able to see their name highlighted in the list if they appear on the leaderboard
- If a user does not appear on the leaderboard, they will appear below it with their rank out of all runs.

Change Password

- The Change Password module will allow a user to modify their login password.

Trophy Room

- The Trophy Room module will act as a way to show users special achievements that they have earned while playing the game. Trophies will have a name, image and description associated with them which will be stored on the frontend and displayed based on a response from the backend listing which trophy ids have been earned by a given user.

New Game

- The New Game module will provide a prompt for the user to enter information for a new game

Load Game

- The Load Game module will allow users to continue a saved game by retrieving it from the database and displaying the saved inventory and current event right where it had been left off.

Game Module

- The Game Page has two child components, the game panel and the inventory panel.
- Game Panel
 - Players can view their current HP and Gold values
 - For each game scenario, the player will be provided with a description of the environment and a list of possible choices to make
 - Once the player chooses a course of action, the client sends the choice to the server
 - The server will send a response with the outcome of the player's choice and the updated player values (HP and Gold)
 - The backend will save current score to the database after every scenario
- Inventory Panel
 - This is where the items users have collected will be stored.
 - The backend will be responsible for saving the contents of the inventory whenever a user chooses to save their game.
 - Items may affect player stats or allow certain choices in events.

Testing Approach

We decided to use top-down testing for our application because most of our work the first sprint was focused on the upper-level modules, such as creating an account, logging in, or seeing trophies and the leaderboard. This way, when we later added modules onto these, we were able to continue to incrementally test from the top down.

Incremental & Regression Testing

Module: Login

Incremental Testing

Defect No.	Description	Severity	How to Correct
1	When logging in, the message 'Incorrect login, please try again' shows up, even if the username and password is correct	Critical	Check how variables are being passed from the forms field to the method call and vue store
2	When logging in with static values page will not change to the main page	Critical	Check how the response from server is being handled.
3	Default page is not the login page	critical	Edit routing table in main.js to add a catch all redirect field when attempting to visit non described routes

Regression Testing

Defect No.	Description	Severity	How to Correct
1	Post request changed into option	critical	Add a CORS ignore to the server as the

	request preventing the server from responding		front and backend do not share a domain
2	Failed logins cause the page to redirect to /login?	low	Set force redirect to false of a failed login to prevent url change

Module: Sign Up

Incremental Testing

Defect No.	Description	Severity	How to Correct
1	When signing up, email account without "@"-sign can be used	Important	Check email input when signing up
2	When signing up, you will stay on the same page	Critical	Redirect to new page when signing up
3	When creating an account, the error message 'Incorrect login. Please try again' would always appear	Critical	Replace incorrectly named variables this.username and this.usernameAgain to this.password and this.confirmPassword.
4	When signing up, make sure to check for duplicate email	Critical	Do not allow the new account to be created
5	When signing up, make sure to check for duplicate username	Critical	Do not allow the new account to be created

Regression Testing (Automated using Mocha, Chai, and Postman)

Defect No.	Description	Severity	How to Correct
------------	-------------	----------	----------------

1	Checking to make sure an account can be created with an email, username, and password	Critical	Run automated tests that post http /newuser requests with test user information
---	---	----------	---

Module: Server

Incremental Testing

Defect No.	Description	Severity	How to Correct
1	Checking login information by attempting to encrypt the password sent by client and compare it to password held in database.	Critical	Hash passwords instead and compare a hash of password from client to hash of password in the database.
2	Attempted to encrypt email addresses and retrieve emails addresses.	Workaround	Hash emails instead instead of using encryption.
3	Did not consider CORS for receiving requests.	Critical	Imported the cors library for node.js.
4	Requests not being read correctly as a JSON.	Workaround	Use the body-parser library for node.js.
5	Inserted values out of order when creating a new user.	Critical	Fixed the order of the inputs to match that of the database attributes for the User table.
6	Was ordering scores incorrectly when passing them	Workaround	Used the mysql ORDER BY to query the

	to the client.		information in the correct order.
7	Did not change the username for every table value that was related to a username that was changed e.g. (score, trophies)	Important	Also UPDATED the values in the Scores and UserTrophies tables for usernames.

Regression Testing (Automated using Mocha, Chai and Postman)

Defect No.	Description	Severity	How to Correct
1	Checking that the login request is doing the correct checks to filter the correct user-password combination.	Critical	Run automated tests that post http login requests with test user information
2	Checking that changing usernames does not affect any interactions with both the front end and database.	Important	Test using Postman that makes a change username request and then analysis of the database and front end afterwards.

Module: Database

Incremental Testing

Defect No.	Description	Severity	How to Correct
1	No relationship between users and their trophies established.	Workaround	Made a table UserTrophies that connects users with the trophies they've earned.
2	Scores in the score	Workaround	This made it so

	table had a primary key of user.		only 1 user score was displayed per user, but changed the table to have the user and score attributes be key attributes.
3	Trophies table was keeping track of image files.	Workaround	Not enough space in the database we were using until we had to pay money so instead we saved references to these images.

Regression Testing (Automated using Postman)

Defect No.	Description	Severity	How to Correct
1	Adding users to the database and duplicates were allowed.	Workaround	Make a check to the database before adding to the database. Request in Postman setup to check for successful adding of user.
2	Show all of the current scores of all users in the scoreboard.	Workaround	Would like to only send the top 5 or so high scores of each user. Postman setup for testing score return results.
3	Show all scores of a specific user.	Workaround	Due to only the top 5 scores of each user being shown usually, instead we need to display all scores of a user. Postman setup for user score return

			results.
4	Password changes need to be stored hashed rather than as a string.	Important	Postman testing of changing a password and then attempts on login. Also checking database to make sure that the new passwords are hashed and kept secure.
5	No functionality to add trophies to test the Trophies table in the database.	Workaround	Created a temporary method used to add trophy entries so we are able to test with user relation to trophies.
6	No functionality to delete entries from the database.	Workaround	When testing with the database, a lot of information is stored that is purely for testing and we need a way to get rid of it. We use the username as the way to delete entries from multiple tables.

Module: Leaderboard

Incremental Testing

Defect No.	Description	Severity	How to Correct
1	When arriving on leaderboard page, no scores are shown	Critical	Check the database to see if there are any users in the scores table

			and check to see how the server retrieves the scores from the database and sends them to the client
2	Leaderboard scores are not sorted	Critical	Check how the client receives and sorts the scores before displaying them
3	Username are not listed	Critical	Check the messages being sent by the server to make sure all the information is there

Regression Testing

Defect No.	Description	Severity	How to Correct
1	Test functionality with test data	Important	If there is no information in the scores table on the database, create test scores and check the functionalities of the leaderboard

Module: Trophy Room (No Defects)

Module: Account Recovery

Incremental Testing

Defect No.	Description	Severity	How to Correct
1	Unable to reset passwords by decrypting and password back	Critical	Corrected by changing recover from asking for a username to asking

			for an email and sending an email with a newly generated email
2	The recovery was not sending back emails beyond a sentence length	important	Increase the max size on emails in the server to a reasonable amount (50 Char)

Regression Testing

Defect No.	Description	Severity	How to Correct
1	Create test user with email and recover password	Important	Check test email to see if it was sent correctly

Module: Main Menu

Incremental Testing

Defect No.	Description	Severity	How to Correct
1	Call to continue game request on click of continue game button returns an undefined game object	Critical	Send the correct header in the request by replacing old username getter with reworked vue store getUsername function
2	Console prints that vue does not recognize newGame and continueGame as functions, clicking the respective buttons does nothing	Critical	Include the mapActions for these tasks within the methods section of the MainMenu component

Regression Testing

Defect No.	Description	Severity	How to Correct
1	When logged and immediately logging out main page would stay open	important	Force redirect after logging out

Module: New Game

Incremental Testing

Defect No.	Description	Severity	How to Correct
1	Game object would not correctly populate with default values	Critical	Re-factor the vuex store to correctly use actions and mutations so that API requests respond on time

Regression Testing

Defect No.	Description	Severity	How to Correct
1	Click on "Start Game" button on the main menu	Critical	Ensure that the game starts with correct player environment

Module: Load Game

Incremental Testing

Defect No.	Description	Severity	How to Correct
1	Continue game button simply functions as if a new game had been started	Critical	Migrate API request for continuing game from within the vuex store to the MainMenu component

Regression Testing

Defect No.	Description	Severity	How to Correct
1	Start a game and then exit out of game page	Important	Click continue game button and ensure that player environment is consistent

Module: Game Module

Incremental Testing

Defect No.	Description	Severity	How to Correct
1	Room name and details do not display at all	Important	Correctly access the fields of the API's /enter response by accessing the properties of the first item in the result list
2	Enter and exit room messages stay displayed after leaving a room	Workaround	Upon exiting a room, clear the enterMessage, and upon entering one, clear the exitRoom message
3	User is unable to continue to the next room	Critical	Modify the for-each-in loop of the cycleRoom function to so it correctly accesses the rooms in the potentialRooms array

Regression Testing

Defect No.	Description	Severity	How to Correct
------------	-------------	----------	----------------

1	Make sure that all instances of game are deleted upon a game finishing	Important	Delete every instance of a game in the database by using the username
---	--	-----------	---