Department of Computer Science & Engineering, University of Nevada, Reno

BETA Universe Systems Initiative Table Application

Andy Alarcon, Jacob Gayban, Mark Graham, Jacob Tucker, Griffin Wagenknecht

Dr. David Feil-Seifer, Dr. Devrin Lee, Vinh Le

John Molt

March 11, 2022

Table of Contents

Abstract

In tabletop role-playing games, players can be burdened by extraneous calculations and player order, which disrupt the flow and narrative of the game. The BETA Universe Systems Initiative Table Application is a mobile-optimized companion web application that aims to digitize and streamline gameplay components from the tabletop role-playing game “BETA Universe Systems.” Players can register for accounts, manage player characters, and participate in game sessions that feature real-time communication via a chat log and combat loop. Gamemasters can also register for accounts, manage the non-player characters and manage game sessions along with running them. This document outlines the project's design and specification.

Project Updates and Changes

Several changes were made in order to streamline the flow of the game and to reduce complexity for both users and us. The Intervention action (FR22, FR23, FR24, UC16)) has been removed. The UI button for the Defend action has been removed. Instead, players being attacked will be shown a prompt on their screen. Minor UI changes have also been made. Selected targets on the initiative table are now highlighted. All buttons within an active session can now be disabled during an attack turn. Color scheme format has been changed to a more appealing color palette.

User Stories and Acceptance Criteria

Action System

As a player, I want to attack other players in order to reduce their hit points.

- AC1. At least 2 players are in the same session.
- AC2. A player may select their target and launch an attack on them.
- AC3. The attacker may choose a weapon to attack with.
- AC4. After the attack information is confirmed, the defender will input their defense equipment.
- AC5. After the defense information is confirmed, the system shall determine the outcome of the attack.
- AC6. Calculated penalties will be applied to the defender.

As a game master, I want to prompt other players to make a move in order to progress the game.

- AC1. At least 2 players are in the same session, with one of them being a game master.

- AC2. The game master presses the “Start” button in order to start the game.
- AC3. The game master presses the “Go” button.
- AC4. A modal shall be displayed only on the screen of the player with the highest initiative.

Character System

As a user, I want to be able to create, edit, and delete characters for use in a session.

- AC1. A “New Character” button shall be displayed on the character page.
- AC2. An “Edit” and “Delete” button shall be displayed on each existing character.
- AC3. The “New Character” and “Edit” buttons shall display a form for filling in the character properties.
- AC4. A user cannot submit a form without completing all the mandatory fields.
- AC5. Upon submission, the character data will be saved into the database.
- AC6. After the user clicks the “Delete” button, the character shall be removed from the character page and from the database.

As a user, I want to be able to create and edit equipment for my characters to use within a session.

- AC1. A section for equipment shall be displayed on the character form.
- AC2. If weapons have been defined before, the form will show the values from last time.
- AC3. Upon submission, the weapon data will be saved into the database.

Authentication System

As a new user, I want to be able to register an account so I can play the game.

- AC1. Login page shall display a “Register” button that will navigate to the registration page.
- AC2. The registration page shall display a form for the new account information.
- AC3. A user cannot submit a form without completing all the mandatory fields.
- AC4. A six digit security code is sent to the user after submitting the form.

As a returning user, I want to be able to log into the site so I can see my characters.

- AC1. Upon navigating to the website, a login page shall be displayed.
- AC2. A user cannot submit a form without completing all the mandatory fields.
- AC3. If the user enters an incorrect email/password, a notification shall be displayed.
- AC4. If the user enters the correct login information, the Session/Character page shall be displayed.

- AC5. All characters associated with that user shall be displayed.

Real-Time Session System

As a player, I want to receive updates about the state of the game so I can strategize accordingly.

- AC1. A chat log and a chat box will be displayed on the session page.
- AC2. Earlier chats will be displayed within the chat box.
- AC3. After every player action, the system will add a relevant message to the chat.

As a player, I want to be able to communicate with other players so we can formulate a strategy.

- AC1. A chat log and a chat box will be displayed on the session page.
- AC2. Earlier chats will be displayed within the chat box.
- AC3. A dropdown menu will be provided.
    - Users can select individual players to message, or can message everyone in the session.
- AC4. Users can type in a message, and the applicable user(s) shall be able to see it in the chat box.

Web API System

The end user does not interact with the Web API system, rather it is used as a component of the technical architecture. The Web API system is very closely tied to the Character and Session Management systems; the acceptance criteria for those systems also requires this system to function, even if it isn't explicitly needed by the user.

Session Management System

As a player, I wish to be able to join a session so I can play with others.

- AC1. All available sessions are displayed on the sessions page.
- AC2. A "Join" button shall be displayed for each individual session.
- AC3. After a player clicks the "Join" button, they will be prompted to select a character(s) to join the session with.
- AC4. After a player chooses their character(s), the initiative table interface will be displayed.
- AC5. A player in a session will be able to see all other players in the session at that moment.

As a game master, I wish to be able to create and delete sessions that will host my campaigns so that others will be able to know if they are active or not.

- AC1. A "New Session" button shall be displayed on the session page.

    - This button is only visible to game masters.
- AC2. An “Join” and “Delete” button shall be displayed on each existing session.
    - The “Delete” button is only visible to game masters.
- AC3. The “New Session” and button shall display a form for filling in the session properties.
- AC4. A user cannot submit a form without completing all the mandatory fields.
- AC5. Upon submission, the session data will be saved into the database.
- AC6. After the user clicks the “Delete” button, the session shall be removed from the session page and from the database.

Testing Workflow

Happy Path Workflows

1. User Registration
    a. From the home page click the register button.
    b. Fill out a valid user name, email and password and click the submit button.
    c. Security sent confirmation will display and click ok.
    d. Enter code sent to the email into the modal and click submit.

User Registration will be validated when the user clicks the submit button on the registration code modal. Test is validated if the user is logged in and brought to the session and character screen. This will also be confirmed by clicking on the top right hamburger button and verifying the correct user name and email are displayed at the top of that section.

2. Login
    a. From the home page, enter a valid registered email address.
    b. Enter the correct password for the account registered to that email.
    c. Click the submit button.

Login will be validated when the user clicks the submit button on the login page. Test is validated if the user is logged in and brought to the session and character screen and any previously created characters registered to that account are displayed. This will also be confirmed by clicking on the top right hamburger button and verifying the correct user name and email are displayed at the top of that section.

3. Create Character
    a. From the home page, log into a registered account.
    b. From the session and character page, click the + Character button.
    c. Enter valid information on the character creation modal and click save.

Create Character will be validated when the user clicks save on the character creation modal. The test will be validated by an alert rendering to the page that displays

"Adding The character was saved!" The test will also be confirmed by the rendering of a new card displaying the newly created character's name and PC/NPC status.

4. Join Session
    a. From the home page, log into a registered account.
    b. Click "Join" on one of the available session cards.
    c. Select one or more characters depending on GM status of account from the character selection modal and click Join

Join Session will be validated when the user clicks join on the character selection modal. The test will be validated by the user being brought to the internal session page with the correct session name in the top center title of the session. This test will also be confirmed by the chosen character(s) being rendered in the Initiative Table section on the left portion of the session.

5. Attack Button
    a. Preconditions
        i. Two or more users are logged into the same session.
        ii. One of the logged in users will have GM status.
        iii. The GM user clicks start and allows users to roll for initiative.
        iv. The GM clicks go and starts the user with the highest initiative's turn.
    b. On the start of a user's turn, select valid targets from the initiative table. The selected targets buttons will change color.
    c. Click confirm to lock in the targets. The initiative table will appear inactive.
    d. Click the Attack button to bring up the ATK/DFND Modal.

Attack Button will be validated when the user clicks the Attack button. The test will be validated by the ATK/DFND Modal rendering with the current characters registered weapons and armor displaying in the modal.

Unhappy Path Workflows

1. User Registration
    a. User enters an invalid email address in the email field. This will result in red text underneath the field indicating the email is invalid. Test is verified by red text rendering when the user enters an invalid email.
    b. User enters an invalid password. This will result in red text underneath the password field indicating the password is invalid. Test is verified by red text rendering when the user enters an invalid password.
    c. User enters an email address already in use. When the user enters the code sent to that email address, an alert message will be displayed indicating the email is already in use. The test is verified when the user submits the code from the email and the error alert is displayed.

2. Login
    a. User enters an unregistered email address. An alert message will be displayed indicating that an account with that email does not exist. The test will be verified when the user clicks submit and the error alert is displayed.
    b. User enters a registered email address with an incorrect password. This will result in an error message displayed indicating that the password is incorrect. The test is verified when the user clicks submit and the error message is displayed.
3. Join Session
    a. User tries to join a session without selecting a character. This will result in an error message displayed prompting the user to select a valid character. This test will be verified when the user clicks the join button without selecting a character and the error message is displayed.
4. Character Creation
    a. User clicks save on the character creation modal with invalid character name and group. Red text will appear below the field indicating constraints on valid entries. The test will be verified when the user clicks save and red text appears under the fields with incorrect data.
5. Attack Button
    a. User clicks the ATK Button from the internal session page without selecting any targets or confirming. This will result in an error message displayed prompting the user to select targets to attack and confirm selection. This test will be verified when the user clicks attack and the error alert is displayed.
    b. User selects targets, but does not click on confirm and clicks the ATK button. This will result in an error message displayed prompting the user to select targets to attack and confirm selection. This test will be verified when the user clicks attack and the error alert is displayed.
    c. User clicks the Confirm button without selecting targets and clicks the ATK button. This will result in an error message displayed prompting the user to select targets to attack and confirm selection. This test will be verified when the user clicks attack and the error alert is displayed.

Testing Strategy

We will be conducting automated integration testing. We will be setting up automated tests for each of our main functions. We have it designed so that when we run a single test instance, it tests all of our main functionality; we just need to add the new cases to the end of the testing code when new features are added. Griffin will be in charge of the automated testing for most of the project and will begin starting now. Others are welcome to test as well.

We will be handling defects found by posting whatever error we find to our team's Trello board to report it. The report must include where the issue occurred and steps to recreate it. Then depending on where the error could have originated from, the person in charge of that feature will be responsible for identifying it and fixing it. We will then re-run tests a few times to make sure that the defect does not occur. The spot where the defect was will then have a flag on it to remember where it was and make sure there is nothing wrong with future testing.

We will determine that our project is complete when all our code is complete. Our systems have passed the acceptance criteria from our external advisor and us. We have finished twenty error-free rounds of the attack loop in a row and have not received any bugs. This will ensure that it should run error-free, even if run with multiple instances at once or in quick succession. In addition, this tests our other systems since characters and sessions are heavily involved in the attack loop.

Test Plan

Test No.	Test Type	Test Screen and Purpose	Test Situation	Expected Result
1	Automated Integration Test	Registration Screen : Tests that the registration page is working as intended.	1. Blank data is used to fill out the form. 2. An invalid email is entered. 3. The email is already in use for another account. 4. Form is valid and email is valid to use. 5. The six digit security code entered is invalid. 6. The six digit security code entered is valid.	1. The user is not allowed to submit until fields are correct. 2. The user is not allowed to submit until the email is corrected. 3. An error message is rendered on the screen to inform the user that email is already in use. 4. A success message is rendered on screen and the user is sent a security code email. 5. An error message is rendered on the screen to inform the user that code was invalid. 6. A success message is rendered on screen and the user is registered and logged in.
2	Automated Integration Test	Login Screen : Tests that the login page is working as intended.	1. Blank data is used to fill out the form. 2. An invalid email is entered. 3. The email and/or password are invalid. 4. Form is valid and email is valid to use.	1. The user is not allowed to submit until fields are correct. 2. The user is not allowed to submit until the email is corrected. 3. An error message is rendered on the screen to inform the user that email and/or password is invalid. 4. A success message is rendered on screen and the user is logged.

3	Automated Integration Test	Home Screen : Tests that the create session option is working as intended.	1. User is logged in as a GM. 2. User is not logged in as a GM. 3. Blank or invalid data is used to fill out the session form. 4. Valid data is used to submit the session form. 5. An error occurs when saving the data.	1. The create button should be visible. 2. The create button should not be visible. 3. The user is not allowed to submit until fields are corrected. 4. A success message is rendered on screen to inform the user that the data was saved successfully. The UI will display the new data. 5. An error message is rendered on the screen to inform the user that an error occurred while saving the data and to try again.
4	Automated Integration Test	Home Screen : Tests that the delete session option is working as intended.	1. User is logged in as a GM. 2. User is not logged in as a GM. 3. A session is successfully deleted. 4. An error occurs when deleting the session.	1. The delete button should be visible for each session. 2. The delete button should not be visible for each session. 3. A success message is rendered on screen to inform the user that the session was deletedsuccesfully. No longer appears on screen. 5. An error message is rendered on the screen to inform the user that an error occurred while deleting the session and to try again.
5	Automated Integration Test	Home Screen : Tests that the join session option is working as intended.	1. User has no characters. 2. User has characters, but does not select any to join with and attempts to join. 3. User has characters, selects one to join with and attempts to join. 4. An error occurs when joining the session.	1. An error message is rendered on the screen to inform the user that an user does not have characters. 2. An error message is rendered on the screen to inform the user that they need to select a character and to try again. 3. The page is redirected to the internal session view and the user is connected to the session they selected. 4. An error message is rendered on the screen to inform the user that an error occurred while joining the session and to re-try.

6	Automated Integration Test	Home Screen : Tests that the create character option is working as intended.	1. Blank or invalid data is used to fill out the character form. 2. Valid data is used to submit the character form. 3. An error occurs when saving the data.	1. The user is not allowed to submit until fields are corrected. 2. A success message is rendered on screen to inform the user that the data was saved successfully. The UI will display the new data. 3. An error message is rendered on the screen to inform the user that an error occurred while saving the data and to try again.
7	Automated Integration Test	Home Screen : Tests that the edit character option is working as intended.	1. Blank or invalid data is used to fill out the character form. 2. Valid data is used to submit the character form. 3. An error occurs when saving the data.	1. The user is not allowed to submit until fields are corrected. 2. A success message is rendered on screen to inform the user that the data was saved successfully. The UI will display the edited data. 3. An error message is rendered on the screen to inform the user that an error occurred while saving the data and to try again.
8	Automated Integration Test	Home Screen : Tests that the delete character option is working as intended.	1. A character is successfully deleted. 2. An error occurs when deleting the character.	1. A success message is rendered on screen to inform the user that the character was deleted successfully. No longer appears on screen. 2. An error message is rendered on the screen to inform the user that an error occurred while deleting the character and to try again.
9	Automated Integration Test	Internal Session Screen: Tests that the edit character option is working as intended.	1. User does not select any character to edit and attempts to edit. 2. Blank or invalid data is used to fill out the character form. 3. Valid data is used to submit the character form.	1. An error message is rendered on the screen to inform the user that they need to select a character and to try again. 2. The user is not allowed to submit until fields are corrected. 3. A success message is rendered on screen to inform the user that the data was saved successfully. The UI will display the edited data. 4. An error message is rendered on

			4. An error occurs when saving the data.	the screen to inform the user that an error occurred while saving the data and to try again.
10	Automated Integration Test	Internal Session Screen: Tests that the chat function is working as intended.	1. User attempts to send a message (direct or global) but it fails to deliver. 2. User attempts to send a message (direct or global) and it successfully delivers.	1. An error message is rendered on the screen to inform the user that an error occurred while sending the message and to try again. 2. The message is added to the chat log. Global messages will appear in white, Direct messages will be highlighted in yellow.
11	Automated Integration Test	Internal Session Screen: Tests that the target selection and confirmation function is working as intended.	1. User clicks on a target in the initiative table. 2. User attempts to re-click an already selected target. 3. The user confirms their selection. 4. The user re-confirms their selection after having confirmed.	1. The target will become red to signify it is selected. 2. The target will revert back to it's previous color to signify it is no longer selected. 3. The initiative table section will become disabled. Not allowing any target selection changes. 3. The initiative table section will become re-enabled. Allowing any target selection changes.
12	Automated Integration Test	Internal Session Screen: Tests that the attack/defend function is working as intended.	1. User clicks to attack and enters values in the attack modal incorrectly and submits. 2. User clicks to attack and enters values in the attack modal correctly and submits.	1. An error message is rendered on the screen to inform the user that the data is in an invalid format and to try again. 2. A success message is rendered on screen to inform the user that they must roll to attack. Following the roll the defender is prompted to defend (same procedure as attack). After defender submits their values, attack loop continues. Chat reflects the actions taken along the way.

13	Automated Integration Test	Internal Session Screen: Tests that the start function is working as intended.	1. User is logged in as a GM. 2. User is not logged in as a GM. 3. The user clicks the start button.	1. The start button should be visible for each session. 2. The start button should not be visible for each session. 3. All players are prompted to roll for initiative. UI is updated to reflect each roll.
14	Automated Integration Test	Internal Session Screen: Tests that the end function is working as intended.	1. User is logged in as a GM. 2. User is not logged in as a GM. 3. The user clicks the end button.	1. The end button should be visible for each session. 2. The end button should not be visible for each session. 3. All players initiatives are set back to zero. UI is updated to reflect the change.
15	Automated Integration Test	Internal Session Screen: Tests that the go function is working as intended.	1. User is logged in as a GM. 2. User is not logged in as a GM. 3. The user clicks the go button.	1. The go button should be visible for each session. 2. The go button should not be visible for each session. 3. The player with the highest initiative is prompted that it is their turn. UI is enabled for that player. Chat reflects the action taken.
16	Automated Integration Test	Internal Session Screen: Tests that the move/hold /wait functions are working as intended.	1. The user clicks the hold button. 2. The user clicks the wait button. 3. The user clicks the move button.	1. The player that selected this action has his initiative reduced. UI is updated to reflect the change. Chat reflects the action taken. 2. The player's turn that selected this action ends. Chat reflects the action taken. 3. The player that selected this action has his initiative reduced. UI is updated to reflect the change. Chat reflects the action taken.

17	Automated Integration Test	Internal Session Screen: Tests that the leave function is working as intended.	1. The user clicks the leave button.	1. The player that selected this action leave's the session. UI is updated to reflect the change. Chat reflects the event.

Table 1: The table above shows our test plan for the acceptance criteria and workflow items.

Work Contribution

	Andy Alarcon	Jacob Gayban	Mark Graham	Jacob Tucker	Griffin Wagenknecht
Project Assignment 3 Paper (Writing sections and formatting)	2.0	1.5	2.0	1.5	3.5
Total	2.0	1.5	2.0	1.5	3.5

Table 2: The table above shows the amount of time spent by each team member on each activity.