Verification and Validation Report: Sandlot

Team 29
Nicholas Fabugais-Inaba
Casra Ghazanfari
Alex Verity
Jung Woo Lee

April 4, 2025

1 Revision History

Date	**Version**	**Notes**
Mar. 3, 2025	1.0	TA Feedback
Mar. 10, 2025	1.1	Prior to Rev1
Mar. 3, 2025	1.0	TA Feedback
Mar. 10, 2025	1.1	Prior to Rev1
Apr. 4, 2025	1.2	Restatement of NFR evaluation criteria. Based on TA feedback: added explicit references to figures for section 4.
Apr. 4 2025	1.3	Based on TA feedback: Explanation of results and what constitutes a pass/fail for FR tests

2 Symbols, Abbreviations and Acronyms

symbol	description
T	Test
SRS	Software Requirements Specification
M	Module

Contents

List of Tables

List of Figures

This document showcases the evaluation of the tests from the VnV plan for Sandlot. The tests are categorized into functional and nonfunctional requirements that can be seen in more detail in the VnV plan.

3 Functional Requirements Evaluation

Test Name	Result	Analysis
Scheduling		
test-FR8	Passed	System can take availabilty data when creating team account.
test-FR10	Passed	Reschedule requests are able to be sent.
test-FR11	Passed	Reschedule requests can be accepted and update the database accordingly.
test-FR12	Failed	Feature must be added to notify team account users when their requests have been accepted.
test-FR18	Failed	Score data currently does not update when using admin features.
test-FR20	Passed	Schedule data is displayed if it exists in the database.
test-FR21	Passed	Team only schedule data is displayed if it exists in the database.
Accounts		
test-FR1-1	Passed	Season schedule is displayed to all users of all account types.
test-FR1-2	Passed	Season standings is displayed to all users of all account types.
test-FR3-1	Passed	Accounts are created and added to the system with the corresponding valid information.
test-FR3-2	Passed	Accounts are not created or added to the system with invalid data.

test-FR4	Failed	Feature must be added to change users' account details.
test-FR5-1	Failed	Feature must be added for a user to delete their account once account credentials have been entered.
test-FR5-2	Passed	Feature must be added for a user to delete their account once account credentials have been entered.
test-FR16-1	Passed	Login succeeds for multiple accounts of varying credentials and account types given correct credentials for that account.
test-FR16-2	Passed	Login fails for multiple accounts of varying credentials and account types given credentials that do not exist in the system.
Team Structure		
test-FR15	Failed	Feature must be added for a player level account user to join a team.

Table 1: Functional Requirements Test Results

For all of our FR tests, we had a separate database filled with placeholder data that was used to test the system. This means that we at all times knew what the database held before, during and after our functional tests. This is how we were able to determine if our tests passed or failed. Tests that passed matched the expected output of the database for a given set of queries while tests that failed did not. The actual output of the database for any given test varied greatly depending on the queries that were run.

4 Nonfunctional Requirements Evaluation

As outlined in the VnV Plan, a Microsoft Form was provided to testers of the system to validate specific system tests outline in the VnV Plan. Answers for each question ranged from either, a rating between 1 (Strongly Disagree) to 5 (Strongly Agree), 1 (Very Difficult) to 5 (Very Easy), or 1 (No, the system was difficult to use due to accessibility issues) to 3 (Yes, the system was accessible). A satisfactory result for each question should be scored a 3 or above.

4.1 Usability

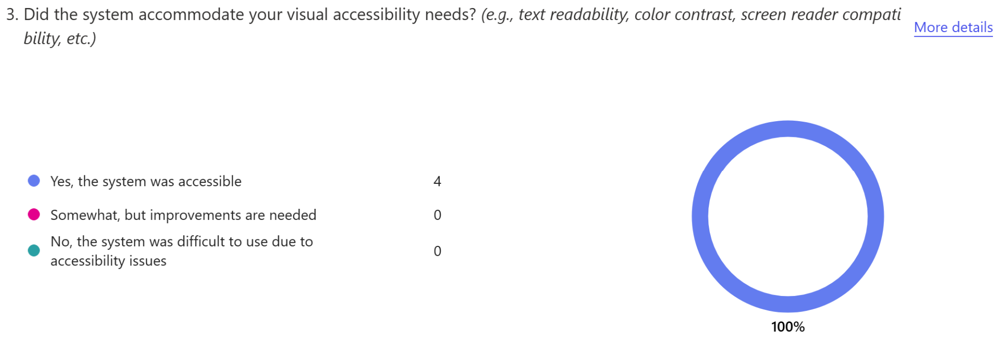
Sandlot's Usability Survey Link used for testing: Sandlot Usability Survey

1. Readability: test-AP3, STY1, UP1
   Brief Description(How tests were performed): The user was provided the website and asked to read groups of text or images such as the schedule or account registration fields. They would then be provided the survey form, provided above, to fill out and provide their own feedback on their experience.
   Average Result: Yes, the system was accessible (Score: 3). Results graphed in Figure 1.
   Evaluation: After the tester(s) had read the Sandlot website's different texts and images, they found the website's readability to be up to a high quality, following a consistent stylistic formatting as evaluated from the resulting rating metric. This indicates that the text and images on the website are clear and easy to understand, which is crucial for user engagement and satisfaction. However, there is still room for improvement in ensuring that all text elements are uniformly accessible across different devices and screen sizes.

2. Registration/Login: test-EU2
   Brief Description(How test was performed): The user was asked to register and sign in to an account they had created, identifying any faults such as login failure, incorrect password verification, etc., throughout their experience. They would then be provided the survey form, provided above, to fill out and provide their own feedback on their experience.
   Average Result: Strongly Agree (Score: 5). Results graphed in Figure 2.
   Evaluation: After the tester(s) had registered and signed in to the account they had created on the Sandlot website, they had found the website's registration and sign in process to function appropriately (i.e. providing correct sign in feedback if their password was incorrect) based on the received resulting rating metric. This high satisfaction score indicates that the registration and login processes are user-friendly and

reliable. However, implementing additional security measures such as two-factor authentication could further enhance user trust and security. Features such as password strength checking and email confirmations would be beneficial to add on top of the current implementation.

3. Navigation: test-AP4, EU1, LR3
   Brief Description(How tests were performed): The user was provided the website and asked to navigate to certain functionalities/pages such as viewing the schedule (EU1, LR3), rescheduling a game, etc. They would then be provided the survey form, provided above, to fill out and provide their own feedback on their experience.
   Average Result: Very/Somewhat Easy (Score: 4.4). Results graphed in Figure 3.
   Evaluation: After the tester(s) had navigated the Sandlot website's different pages (Schedule, Home, Profile, etc.), they found the website's navigation including its menus and links to be easily accessible and straightforward from the resulting rating metric. This high ease-of-use score suggests that the website's navigation is intuitive and user-friendly. However, further improvements could be made by adding more contextual help and tooltips to guide users through complex tasks.



3. Did the system accommodate your visual accessibility needs? *(e.g., text readability, color contrast, screen reader compatibility, etc.)*

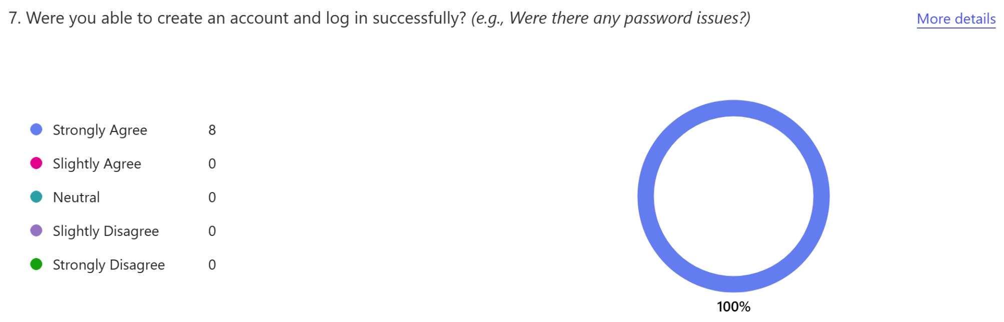
More details

● Yes, the system was accessible 4

● Somewhat, but improvements are needed 0

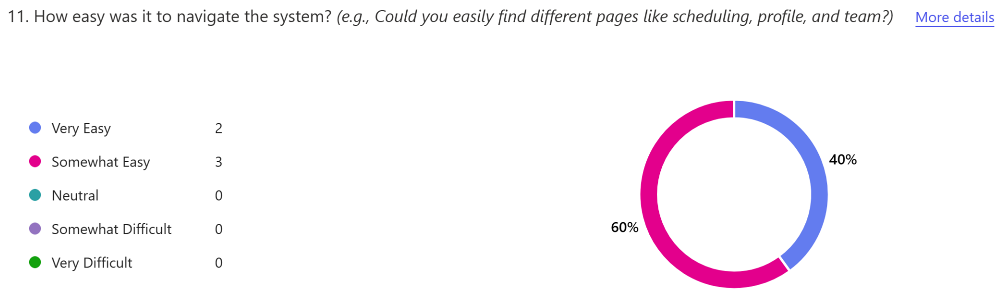
● No, the system was difficult to use due to accessibility issues 0

Figure 1: Readability Response Metrics



7. Were you able to create an account and log in successfully? *(e.g., Were there any password issues?)*

Figure 2: Registration/Login Response Metrics



11. How easy was it to navigate the system? *(e.g., Could you easily find different pages like scheduling, profile, and team?)*

Figure 3: Navigation Response Metrics

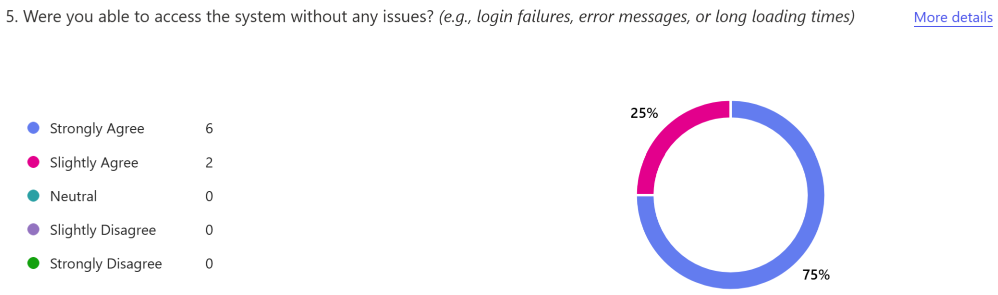
4.2 Security

1. General Access: test-AS1-1, AS1-2, PV1
   Brief Description(How tests were performed): The tester was given general access to the website, in which they were not logged in to any account. They were then prompted to access both the season schedule (AS1-1) and standings (AS1-2), responding if each were both displayed correctly (i.e. all games showed up on the schedule, all teams showed up in the standings with their respective records). Also, the tester would attempt to locate sensitive information (PV1) (i.e. contact information of players) across the website and confirm if no such information was able to be observed.
   Average Result: Strongly/Slightly Agree (Score: 4.75). Results graphed in Figure 4.

Evaluation: The tester(s) had successfully accessed the Sandlot website's season schedule and standings based on the resulting rating metric, without being logged in to an account. Additionally, they were not able to obtain contact information when browsing the website, securing sensitive information saved by the system from account users. The high rating score indicates that the website's general access is operating appropriately with little to no errors. However, improvements could be made to the loading times of the system as one comment made from the usability survey stated on initial loadings of a page, the initial time to view the page would take a little longer than expected.

2. Request Validation: test-AS6-1, AS6-2, IG1-2
   Brief Description(How tests were performed): The tester was asked to sign in to a given account (player, team, commissioner), inputting valid (AS6-1) and invalid (AS6-2) account information. The system should then either log the tester into the account or deny their log in request due to invalid account information. When logged in with valid credentials, the tester should then submit a reschedule request (IG1-2) and accept/deny the request on the receiving end (the opposing team) by logging into their account. They will then confirm if the request had either been successfully denied or accepted. If the reschedule request was successfully accepted, the tester should confirm that the updated game day/time is reflected upon viewing the season schedule.
   Average Result: Slightly Agree (Score: 4). Results graphed in Figure 5.
   Evaluation: The tester(s) had successfully logged into their account with valid credentials and were denied access with invalid credentials based on the resulting rating metric. However, the reschedule request process was not as straightforward as expected, with some testers still being able to view their reschedule request even after they accepted or denied it. This indicates that the reschedule request process requires improvements to ensure that feedback is presented appropriately to users on their rescheduling actions (accept/deny). Moreover, the data gathered from the usability testing suggests that the system's account verification is secure, ensuring that only authorized users can access the system as well as their own personal information pertaining to their account.

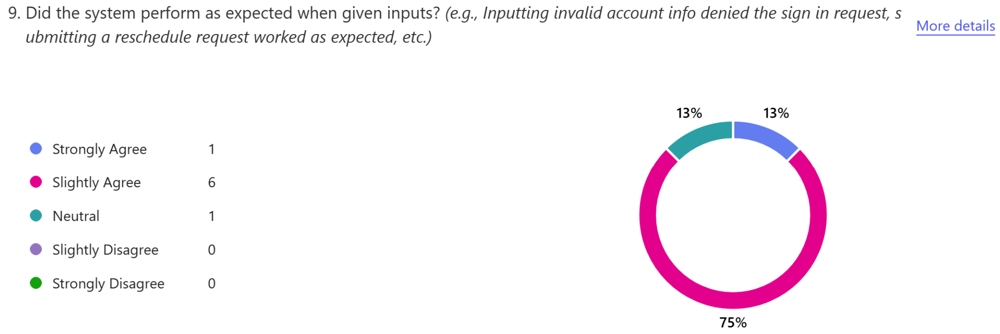
3. Scheduling: test-IG1-1
   Brief Description(How test was performed): The tester will be provided with sample scheduling data and be asked to generate a season schedule. This season schedule that is created should not have any conflicting scheduling conflicts that exist.
   Tests Conducted: 50
   Pass/Fail Rate: 43/50 (86%)
   Evaluation: The tester(s) successfully generated a season schedule without any conflicting scheduling conflicts in 43 out of 50 attempts. The seven failures occured randomly and would be resolved when the website was reloaded. This high pass rate indicates that the scheduling algorithm is robust and reliable for most scenarios. However, improvements on the front end need to be investigated to ensure that all possible errors are resolved. These errors could occur from when the front end retrieves the scheduling data from the database or how it handles the data on the front end. The feedback from the usability testing also highlighted the need for better error messages and guidance in the display of the schedule, to help users understand and resolve issues more effectively.



5. Were you able to access the system without any issues? *(e.g., login failures, error messages, or long loading times)*

● Strongly Agree	6
● Slightly Agree	2
● Neutral	0
● Slightly Disagree	0
● Strongly Disagree	0

Figure 4: General Access Response Metrics



9. Did the system perform as expected when given inputs? *(e.g., Inputting invalid account info denied the sign in request, s ubmitting a reschedule request worked as expected, etc.)*

More details

● Strongly Agree	1
● Slightly Agree	6
● Neutral	1
● Slightly Disagree	0
● Strongly Disagree	0

Figure 5: Request Validation Response Metrics

5 Comparison to Existing Implementation

The current website used by the league, GSA Softball, provides basic functionalities for managing the McMaster Graduate Softball league, including viewing schedules, team standings, weather information, etc. However, there are many links or functionalities with the current website that are not operational, providing 404 Not Found errors. The new Sandlot website introduces several improvements and additional features aimed at enhancing the overall user experience and operational efficiency.

One of the significant improvements in the Sandlot website is the user interface design. The new design provides a more modern and intuitive user experience, making it easier for users of the system to navigate through different pages on the website. Additionally, the layout is much cleaner, using a consistent stylistic formatting which enhances the readability and accessibility of the system's text and images including the schedule, directory information such as parking or weather information, and more. This is a notable improvement over the existing website, which has a more dated and less user-friendly interface.

The Sandlot website also introduces advanced functionalities that are designed to streamline league management and reduce the administrative burden on league organizers. For example, the league organizers will be privy to multiple configuration options like division management and general scheduling decisions such as a season's start and end date or minimum games played by a team.

Another key enhancement is the integration of user accounts and personalized dashboards. The Sandlot website allows players and captains to create accounts and access personalized information relevant to their roles. This includes viewing their team's schedule and managing player rosters. The current website lacks this level of personalization and user-specific functionality.

In summary, the Sandlot website offers a comprehensive and user-friendly platform that addresses many of the limitations of the current GSA Softball website. Improvements to the UI, account management, and league configuration options provide a significant upgrade, making the Sandlot website a more efficient and effective tool for managing league activities.

6 Unit Testing

Test Name	**Result**	**Analysis**
Reschedule Module		
test-URS1	Passed	Backend reschedule function correctly updates database.
Database Module		
test_get_schedule_games	Passed	Backend correctly retrieves scheduled game data.
test_team_games	Passed	Backend correctly retrieves all of the specified team's games.
test_create_RR	Passed	Backend properly creates a new reschedule request with the specificied data in the database.
test_get_team_RRs	Passed	Backend correctly retrieves all reschedule requests adressed to a specific team.
test_delete_reschedule_request	Passed	Backend properly deletes a specified reschedule request from the database.
test_get_standings_data	Passed	Backend properly retrieves completed game scores and calculates the standings of teams based on them.

test_get_player_data	Passed	Backend correctly retrieves the list of players that belong to a specified team including their information.
test_get_team_data	Passed	Backend correctly retrieves all teams in the databse and their divisions.
test_create_player_account	Passed	Backend properly creates a player account in the database with the player information provided to it.
test_create_team_account	Passed	Backend properly creates a team account in the database with the team information provided to it.
test_get_player_account	Passed	Backend correctly retreives a player's account information based on the player's email.
test_get_team_account	Passed	Backend correctly retrieves a team's account information based on the team's username.
test_delete_player	Passed	Backend properly deletes a player account from the database based on the information provided.
test_delete_team	Passed	Backend properly deletes a team account from the database based on the information provided.
Season Scheduler Module		
test-USS1	Passed	Scheduler correctly distributes games between inputted start and end dates.
test-USS2	Failed	Function that creates initial set of games must be modified to create the correct number of games per team.
test-USS3	Passed	Scheduler correctly generates schedule for a variety of teams and division distributions.

Table 2: Unit Test Results

The system has proven to be fairly robust, with a majority of our unit tests passing successfully. This shows its stability, reliability, and ability to perform as expected in these specific areas.

7 Changes Due to Testing

7.1 Feedback from Rev 0 demo

- **User Role Implementation:** The demo showcased the user role functionality. However, the commissioner role was not yet available. The requirements for the commissioner role, as outlined in the SRS, were implemented for accessing league configuration and management options and other admin-level functionalities including alerts and generating a season schedule.

- **Scheduling Algorithm:** The team developed their own algorithm for scheduling, which was deemed reasonable. Configuration options were added to make the software general enough for other users while satisfying the specific requirements for Dr. Nease. Default values were set to meet Dr. Nease's requirements.

- **Rescheduling Functionality:** The team verified that rescheduling can be done and that it is straightforward. Additional features were added to enhance the rescheduling process, such as providing clear messages when a game cannot be rescheduled and explaining how the reschedule request works.

- **Usability Testing:** Usability testing was emphasized as very important for the project. The team conducted extensive usability tests and made several improvements based on the feedback received. This can be seen below in the user feedback section.

7.2 Feedback from the Supervisor

- **Informataion Access:** Players can now see the email addresses of players on the same team but cannot access the email addresses of players on other teams' rosters. This ensures privacy while providing necessary contact information for a player's current team.

- **Team Schedule Location:** The team schedule is now prominently displayed on the schedule page, making it easily accessible to users.

The entire season schedule is still available for view on the schedule page, if a user would like to access it.

- **Waiver Management:** The waiver process needs to be streamlined by integrating it into the website. Players should be able to view their signed waivers directly on the site, and a PDF version should be available for a player to download.
- **Join Request Limit:** The system should allow for configurable limits on the number of join requests a team can receive, providing flexibility for different league requirements.
- **Configuration Options:** Toggle options for reschedulability, adding fields and timeslots, choosing available days, and setting constraints should be implemented to make the system adaptable for other leagues.
- **Alerts/Announcements:** A bulk email feature needs to be added for sending alerts to specific groups of users. Announcements should be be dated and include headers and content for better communication.
- **Team Page Improvements:** Pie charts for scores, upcoming games, and team profile pictures would be desireable bonus features to add that could enhance the team pages.
- **Captcha and Security:** Captcha is desired to prevent automated sign-ups and enhance security.

7.3 User Feedback

- **UI & Navigation:**
  - The GitHub logo, Documentation, and Github links were removed and a login/signup button was added to all pages, providing enhanced clarity to the registration/sign in process.
  - Notices should be added to inform users to join a team to view the schedule if they are not part of a team.
  - Messages explaining why a game cannot be rescheduled should be added.
  - A legend for field colors needs to be added to the schedule.

- **Rescheduling:**
  - The reschedule request process should be clarified with a popup or description on the page, indicating how the rescheduling of a game works.
  - Already played games should be greyed out/removed on both the reschedule menu and the regular schedule menu.
  - The proposed dates section of a received reschedule request should clearly indicate who the request is coming from as currently no team information is visible unless a user were to verify the game themselves by navigating to the schedule page.
  - Dates in reschedule requests should be reformatted to be more readable (e.g., "Tuesday, June 14th").
- **Registration/Login:**
  - A notice encouraging users to join a team should be added after signing up as a player.
  - Sign-up email confirmations should be implemented once an account has been successfully registered by the user.
  - Password checking on sign-up submission should be considered to ensure password strength.
  - Two-factor authentication was considered for sign-up.
- **General Improvements:**
  - Preferred division field on team sign-up can be explained better with a tooltip.
  - Offdays that appear in the registration process should be only added to the preferred off-day list, if the admin had configured the league to be played on those days. Also, if the league were to be scheduled on only one day of the week, the off-day selection should be disabled.
  - Multiple off-days being selected was considered.
  - Registering a new account should automatically sign the user in to their created account.

- Reschedule options need to be made more obvious with a dotted border and lighter color.
- Game start times should be highlighted as more important than end times. Considered removing end times altogether.
- Different colors, along with a legend, should be used to highlight games in the reschedule menu based on their status.

8 Automated Testing

Currently, unit tests for the database, season scheduler, and reschedule modules are automated and continuously integrated using GitHub actions to execute them on every push and pull reqeust to our main branch. This ensures our code is as robust as possible when pushing to production branches while also saving us the hassle of manually running the tests ourselves. In the future we plan to extend our automated testing environments to produce visual test reports in the form of HTML documents after each execution as well as provide code coverage metrics to help bolster our testing data.

9 Traceability between Requirements and Modules

M1: Hardware Hiding Module

M2: Account Module

M3: Player Module

M4: Team Module

M5: Commissioner Module

M6: Account Structure Module

M7: Team Structure Module

M8: Schedule Structure Module

M9: Standings Structure Module

M10: Season Scheduler Module

M11: Reschedule Module

M12: Alerts Module

M13: Database Module

Req.	Modules
FR-1	M9, M10,
FR-3	M2, M6
FR-4	M2, M6
FR-5	M2, M6
FR-8	M3,
FR-10	M3, M11
FR-11	M3, M11
FR-12	M3, M11
FR-15	M3, M4
FR-16	M2
FR-18	M5, M10
FR-20	M8, M10
FR-21	M7, M8, M10

Table 3: Trace Between Functional Requirements and Modules

Appendix — Reflection

The information in this section will be used to evaluate the team members on the graduate attribute of Reflection.

The purpose of reflection questions is to give you a chance to assess your own learning and that of your group as a whole, and to find ways to improve in the future. Reflection is an important part of the learning process. Reflection is also an essential component of a successful software development process.

Reflections are most interesting and useful when they're honest, even if the stories they tell are imperfect. You will be marked based on your depth of thought and analysis, and not based on the content of the reflections themselves. Thus, for full marks we encourage you to answer openly and honestly and to avoid simply writing "what you think the evaluator wants to hear."

Please answer the following questions. Some questions can be answered on the team level, but where appropriate, each team member should write their own response:

1. What went well while writing this deliverable?

2. What pain points did you experience during this deliverable, and how did you resolve them?

3. Which parts of this document stemmed from speaking to your client(s) or a proxy (e.g. your peers)? Which ones were not, and why?

4. In what ways was the Verification and Validation (VnV) Plan different from the activities that were actually conducted for VnV? If there were differences, what changes required the modification in the plan? Why did these changes occur? Would you be able to anticipate these changes in future projects? If there weren't any differences, how was your team able to clearly predict a feasible amount of effort and the right tasks needed to build the evidence that demonstrates the required quality? (It is expected that most teams will have had to deviate from their original VnV Plan.)

Team Reflection

1. Which parts of this document stemmed from speaking to your client(s) or a proxy (e.g. your peers)? Which ones were not, and why?

   The parts of this document that stemmed from speaking directly to clients or peers include the changes due to testing section that had condensed all of the feedback received from the Rev 0 demo, the supervisor, and selected users. Additionally, the usability testing conducted, along with the analysis of the data collected was gathered from speaking to client(s) or a proxy. The parts of the document that did not come from client(s) or a proxy was the unit testing as the team had written multiple sets of unit tests that were used to test certain functionalities of the system such as the scheduling algorithm. These were based on best practices and our team's internal discussions to ensure the robustness and successful operation of the system.

2. In what ways was the Verification and Validation (VnV) Plan different from the activities that were actually conducted for VnV? If there were differences, what changes required the modification in the plan?

Why did these changes occur? Would you be able to anticipate these changes in future projects? If there weren't any differences, how was your team able to clearly predict a feasible amount of effort and the right tasks needed to build the evidence that demonstrates the required quality? (It is expected that most teams will have had to deviate from their original VnV Plan.)

The VnV Plan was different and required several changes before conducting actual Verification and Validation for reasons including access to actual users of the system and the constant iteration that had occured from when the VnV Plan was first written. These changes occurred because real-world testing often uncovers issues that are not apparent during the planning phase. The iterative feedback from users and the supervisor helped us identify and address these issues, however this prompted changes needing to be made for the initially written VnV Plan. Specifically, some of these changes included reducing the amount of tests that would proceed through the VnV process or tests that the team had deemed unnecessary and thus removed. In future projects, we would anticipate such changes by incorporating more flexible and adaptive planning processes, allowing for iterative testing and feedback loops to refine the VnV activities continuously. Despite these deviations from the original plan, our team was able to predict a feasible amount of effort and the right tasks needed to build enough data for a proper analysis of the system. This was achieved through team alignment with the project's goals and requirements, ensuring the scope of the project was feasible for completion within the given time frame.

Nicholas Fabugais-Inaba – Reflection

1. What went well while writing this deliverable?

   The organization of each team member working on a particular section was something that went very well as part of this deliverable. This allowed us to fill out the various sections of this deliverable with a high level of detail as each person could focus on a section that they felt most comfortable in writing about. Additionally, with the sections completed, our team was able to communicate effectively and provide feedback on each other's sections, ensuring that the deliverable would be written up to all of our standards.

2. What pain points did you experience during this deliverable, and how did you resolve them?

   When conducting the usability testing, it was difficult to get a large number of users to test the system and provide enough feedback for the team to conduct an extensive analysis. Although there was a limited amount of data collected, the team was still able to utilize the data and provide a proper analysis. This was done by identifying consistent themes in the users' feedback as this would let us know areas of the system that either provided a strong user experience or parts of the system that required improvement.

Jung Woo Lee – Reflection

1. What went well while writing this deliverable?

   Having the VnV Plan allowed this deliverable to be done smoothly as it mostly involved performing the required tests outline in that document. Moreover, the team was able to stay organized and work on this document in a timely manner by divvying the sections and working simultaneously.

2. What pain points did you experience during this deliverable, and how did you resolve them?

   The usability testing was a bit difficult to conduct as it was hard to get a diverse and more extensive set of users to test the system. But even with the limited number of testers, the team was still able to gain a large quanitity of feedback that was used to analyze the system. This was done by identifying consistent themes in the users' feedback as this would let us know areas of the system that either provided a strong user experience or parts of the system that required improvement.

Casra Ghazanfari – Reflection

1. What went well while writing this deliverable?

   Before working on this document, we had a good idea on how we wanted to structure and automate our unit tests, and because of our pre-existing ideas we were able to implement them extremely quickly and easily. Even though they were implemented quickly we made to sure not to sacrifice quality and made sure that the robustness of our system was thoroughly checked through our tests. Overall the unit testing portion of the deliverable went very well due to our team's preplanning.

2. What pain points did you experience during this deliverable, and how did you resolve them?

   We found the usability testing to be a bit of a challenge due to our project not being fully complete. It made it difficult for us to decide how we would even conduct the usability testing if users could not easily access the project. However, we came up with a clever solution to this problem by having users direct us how they would navigate the website allowing us to emulate their user experience without needing them to have access to the website itself. This allowed us to gather valuable usability feedback we would not have been able to gather had we not come up with this solution.

Alex Verity – Reflection

1. What went well while writing this deliverable?

   The usability tests went the best, as we had quite a few people easily accessible and recieved lots of good results. We were also able to improve our feedback form using their feedback which will help with future usability tests.

2. What pain points did you experience during this deliverable, and how did you resolve them?

Unit tests and automated testing were difficult to implement as we had to write validation code for things like the database. We were able to resolve in the end but it was the most difficult part for me.