

# Team 22 : Memoji

## Sprint 3 Planning Doc

---

### Team Members

**Scrum Master - Jonathan Poholarz** ([jpoholar@purdue.edu](mailto:jpoholar@purdue.edu))

Maxwell Jones ([jone1268@purdue.edu](mailto:jone1268@purdue.edu))

Manoj Polisetti ([mpoliset@purdue.edu](mailto:mpoliset@purdue.edu))

Andrew Ring ([amring@purdue.edu](mailto:amring@purdue.edu))

Delun Shi ([shi272@purdue.edu](mailto:shi272@purdue.edu))

### Overview

Our previous sprint focused on implementing a majority of the gameplay features within the game. For this final sprint, we will be focusing on wrapping up the gameplay, polishing elements, and rigorous testing. Specifically, the plan includes developing the final round for players, the final results, starting a new game after completing a game without exiting the application, and handling player mid-game disconnections. In our polishing efforts, we also plan to go through a refactoring phase, redesigning different elements of the user interface to incorporate dynamic screen sizes and the application of theme files. This will undoubtedly lead to a significant batch of regression testing to ensure that all features of our application work as intended and look spotless going into our final presentation.

A significant challenge this sprint will involve player disconnection and reconnection during different states of gameplay. This will involve significant work on Networking for the server, host, and player. There are many cases to keep track of and unknowns that could complicate development. We intend to divide these scenarios into two subgroups: loss of internet connection and unexpected closing of the application. Loss of internet connection should usually preserve the application's state, reducing information lost. An unexpected close or crash will likely require retransmission of data from the server which we will need to plan for.

**Meeting Plan:** Wednesday (6:00 pm-7:30 pm), Friday (3:00 pm-6:00 pm)

## Sprint Details

Sprint Story #1 (F19) - As a player, I would like to receive instructions and prompts from the host computer screen during play			
#	Task	Time	Owner
1	Design Instructional Screen for game (point of the game, how to win, etc.)	4 hr	Andrew
2	Design Instructional Screen for answering prompts	4 hr	Andrew
3	Design Instructional Screen for voting on answers	2 hr	Andrew
4	Design Instructional Screen for scoring	3 hr	Andrew
5	Design Instructional Screen for final round	3 hr	Andrew
6	Implement option toggle in Host game setup to enable/disable instructions (on by default)	3 hr	Andrew
7	Implement instruction disable when a 2nd game is started in the same lobby	2 hr	Andrew
8	Implement sub-option toggle in Host game setup to force instruction prompts to show for every round	2 hr	Andrew
9	Implement transitions to instruction screens when appropriate, depending on the option's setting	4 hr	Andrew
10	Test that all instructional screens are displayed when option toggled on	1 hr	Andrew
11	Test that instructional screens continue to be displayed in the second game if the option to force this is enabled	1 hr	Andrew
12	Test that instructional screens are disabled in the second and later games if the option to force this it is disabled	1 hr	Andrew
#	Acceptance Criteria		
1	Given the instruction option is enabled, when a game session is started, then an instructional screen explaining the game will be shown before the first prompt answering round.		
2	Given the instruction option is enabled, when the game changes to a new type of round (such as answering prompts, voting on prompts), then instructional screens will be		

	shown.
3	Given the force instructions always on option is enabled, when the second game in the same Host session is started, then instructional screens will continue to be shown.
4	Given the force instructions always on option is disabled, when the second game in the same Host session is started, then the instructional screens will not be shown.
5	Given the instruction option is disabled, when a game session is played, then no instructional screens will be shown.
6	Given a Host has connected to the Server, when a game session is being set up, then toggleable instructional screen options will be shown to the user.

Sprint Story #2 (F41) - As a player, I would like to view the Prompt and the voting options for each round on the Player screen			
#	Task	Time	Owner
1	Implement ability to view correct Prompt for each round	2 hr	Manoj
2	Implement functionality to view the Emoji canvas of the options on the Voting screen of initial rounds	5 hr	Manoj
3	Test Prompt and Emoji canvas matching the prompt and emoji canvas on the Host	2 hr	Manoj
#	Acceptance Criteria		
1	Given that the Player is on the voting screen, when the Host sends the Player the prompt and voting options, then the Player can view the Prompt and the Emoji canvas to make a choice and vote		
2	Given that the Host sends a prompt and the voting options, when the Player is on the voting screen, then the options and prompt displayed on the Host matches the prompt and options displayed for the Player		
3	Given that the Player clicks on an Emoji canvas, when the player clicks submit, then the correct ID of the emoji canvas clicked is sent to the Host		

**Sprint Story #3 (F5) - As a host, I would like to ignore/skip players who have not submitted answers to prompts within the allotted time and continue the game**

Sprint Story #4 (F32) - As a player, I would like to view the remaining time for a question on the host screen

**(Story basically completed already aside from implementation which lies more in Sprint Story #2)**

#	Task	Time	Owner
1	Implement host timeout message handling in Player Networking	2 hr	Delun
2	Define timeout message signal to Player GameStateManager	1 hr	Delun
3	Implement handling blank prompts in the Host (such as from players that did not answer in time)	2 hr	Delun
4	Implement handling of signal to force prompt submission of WIP or blank prompts	2 hr	Delun
5	Implement handling of signal to force end voting if a player did not vote in time	2 hr	Delun
6	Test timeout force-advancing a player during the prompt response phase	2 hr	Delun
7	Test timeout force-advancing a player during voting phase	2 hr	Delun
8	Test that blank prompt responses are handled appropriately	2 hr	Delun
9	Test that started prompt response is forwarded as-is to the Host and displayed appropriately	2 hr	Delun
10	Test that un-submitted votes are not counted in tally	2 hr	Delun
#	Acceptance Criteria		
1	Given a prompt answering round has started, when the prompt answering round timer expires, then a timeout message will be sent to each Player still answering.		
2	Given a Player is answering a prompt, when a timeout message is received, then the Player's answer will be force submitted as-is and sent back to the host.		
3	Given a Player has not started answering a prompt, when a timeout message is received, then the Player's blank answer will be force submitted and handled as a blank response by the Host.		
4	Given a voting round has started, when the voting round timer expires, then a timeout message will be sent to each Player still voting.		

5	Given a Player is voting on prompt responses, when a timeout message is received, then the Player's lack of a vote will be ignored in the vote tallying.
---	--

Sprint Story #5 (F30) - As a player, I would like to reconnect to a running game after disconnecting and be able to continue playing as the same player with the same username and same avatar			
#	Task	Time	Owner
1	Implement saving/loading player ID to file in case of disconnects/crashes that would need to recover it (possibly with the lobby code in addition)	3 hr	Jonathan
2	Implement reconnect to stored lobby code with associated ID upon relaunch of the game (as a popup so user could say no if they quit the game legitimately)	2 hr	Jonathan
3	Implement icon to recover the pop-up on the Launch Screen if the user unintentionally closed it	1 hr	Jonathan
4	Implement/verify that all prompts/votes/etc. are stored in the Host in case of retransmission requests	1 hr	Maxwell
5	Implement networking signals for retransmission requests (given a playerid and lobbycode)	2 hr	Maxwell
6	Implement network disconnected error pop-ups when trying to send a message without an internet connection (in the Player)	5 hr	Maxwell
7	Implement network disconnected error pop-ups when trying to send a message without an internet connecting (in the Host)	5 hr	Maxwell
8	Test that error pop-up appears when sending prompt responses without internet connection	2 hr	Manoj
9	Test that error pop-up appears when sending vote responses without internet connection	2 hr	Manoj
10	Test that player ID and lobby code are stored successfully to external file for each game session played (only store one)	0.5 hr ea.	All members
11	Test that player ID and lobby code are removed when a game session is closed normally by the host or quit by a player from the game start wait screen	0.5 hr ea.	All members

12	Test that reconnect to previous game session pop-up appears on Player Launch Screen after unexpected close	0.5 hr ea.	All members
13	Test that previous game session pop-up leads to successful reconnect to Host game session without adding a new player or audience member	0.5 hr ea.	All members
14	Test that reconnecting player can request and receive prompts or votes that are currently going on	0.5 hr ea.	All members
15	Test that Players sending messages without internet connections receive network disconnected error pop-ups and wait for the connection to be restored before advancing	0.5 hr ea.	All members
16	Test that Hosts sending messages without internet connections receive network disconnected error pop-ups and wait for the connection to be restored before advancing the game state	0.5 hr ea.	All members
<b>#</b>	<b>Acceptance Criteria</b>		
1	Given a Player has connected to a Host, when the Player receives a connection confirmation containing the game lobby code and player ID, then the Player will store the lobby code and player ID in order to reconnect in case of a crash.		
2	Given a Player's application has closed unexpectedly, when the Player relaunches the application, then a reconnection request pop-up will be shown on the Title Screen to ask the Player about reconnecting.		
3	Given a reconnection request pop-up has been shown to the Player, when the Player accepts the request, then the application will attempt to reconnect with the Host using the stored lobby code and player ID.		
4	Given a reconnection request pop-up has been shown to the Player, when the Player declines the request, then the application will minimize the pop-up to an icon on the screen.		
5	Given a Player has declined a reconnection request, when the Player interacts with the reconnect icon on the screen, then the reconnection request pop-up will be restored.		
6	Given a Host game session is running, when prompts are generated, then the prompts are stored on the Host in some sort of data structure in case of a retransmission request from a Player.		
7	Given a Player has closed their game session unexpectedly and disconnected, when the Player sends a valid reconnection request, then the Host will retransmit information for the Player in the current game state.		
8	Given a Player has disconnected from their Internet connection, when the Player tries		

	to send a message over the network, then a networking connection error pop-up will be shown, and the message transmission will wait for a retry.
9	Given a Host has disconnected from its Internet connection, when the Host tries to send a message over the network, then a networking connection error pop-up will be shown, and the message transmission and game advancement will wait for a retry.

Sprint Story #6 (F26) - As a player, I would like to select multiple answers during the final round when needed			
#	Task	Time	Owner
1	Design Final Round Prompt Screen for Host with unique design from the other rounds	3 hr	Manoj
2	Design Final Round Voting Screen that allows for selecting multiple answers to a prompt	3 hr	Manoj
3	Implement ability to select up to three answers to the final round prompt	4 hr	Manoj
4	Implement on-screen indicators to signify which prompts are given the primary vote, secondary vote, and tertiary vote (such as a gold, silver, bronze indicator)	3 hr	Manoj
5	Implement ranked voting response transmission to the Host from the Player	2 hr	Manoj
6	Implement ranked voting message receiving on the Host	2 hr	Delun
7	Implement ranked voting scoring and tallying	4 hr	Delun
8	Design Final Round Results screen showing all Player responses, gold/silver/bronze votes received, and points allocated	2 hr	Delun
9	Test that Player can select up to three responses to the final round prompt.	3 hr	Manoj
10	Test that on-screen indicators match rankings selected by the Player for their primary vote, secondary, and tertiary.	3 hr	Manoj
11	Test that ranked votes are processed in the same order on the Host as they were selected by the Player.	3 hr (1.5 ea.)	Manoj/Delun
12	Test that points awarded in the Final Round Results Screen	3 hr	Manoj/Delun

	are tallied correctly and point distribution correlates.	(1.5 ea.)	
13	Test that all Player ranked responses are shown on the Final Round Results Screen.	2 hr	Manoj
<b>#</b>	<b>Acceptance Criteria</b>		
1	Given the Host is running a game, when the game reaches the final round, then the Host screen will advance to the Final Round Prompt Screen.		
2	Given the Host is running a game, when the game reaches the Final Round Prompt Screen, then the prompt will be sent to all Players.		
3	Given the Host has sent the final round prompt responses to the Players, when the Players receive the prompt responses, then they will navigate to a Final Round Prompt Voting screen.		
4	Given the Player has transitioned to the Final Round Prompt Voting Screen, when the Player votes for responses, then the Player can select up to three responses, ranking them as their first, second and third place vote.		
5	Given the Player has selected their three vote responses, when the Player submits their ranked votes, then the votes are sent to the Host to be tallied.		
6	Given the Host has received ranked votes from all Players, when the Host processes and tallies the votes, then they are awarded to the correct players and of the correct tier as selected by the Players.		
7	Given the Host has tallied the votes, when the Final Round Results Screen is shown, then the screen displays the answers voted for with the correct gold/silver/bronze tier for each player's votes.		

Sprint Story #7 (F8) - As a host, I would like to display a final results screen and winner/credits screen after the game completes			
#	Task	Time	Owner
1	Design credits screen	1 hr	Andrew
2	Implement updated functionality to a final results screen, including accounting for ties	2 hr	Andrew
3	Implement new game and game reset functionality in final screen	2 hr	Andrew
4	Test that the final results screen displays correctly	1 hr	Andrew



5	Test that the final results are ordered correctly, accomodating for ties	1 hr	Andrew
6	Test that the credits screen displays correctly	1 hr	Andrew
7	Test that a new game can be made with the same players	1 hr	Andrew
8	Test that a new game can be made and completely reset	1 hr	Andrew
#	<b>Acceptance Criteria</b>		
1	Given that the game is running, when the game completes the final round of voting, then the Host will display the final results screen and credits.		
2	Given that the game is displaying the results screen, when there is a tie in points between players, then the display will account for the tie(s).		
3	Given that the game is displaying the final results screen, when the “new game/reset” button is pressed, then the Host will proceed with preparing to start a new game.		
4	Given that the Host is starting a new game, when the game returns to the lobby screen, then the players from the previous session are still connected.		
5	Given that the Host is starting a new game, when the game returns to the lobby screen, then the Players are taken to their lobby screens where they may disconnect, change username, and change avatar.		

<b>Sprint Story #8 (F9) - As a host, I would like to play additional games after the first game without forcing all players to disconnect and reconnect to a new lobby</b>			
#	Task	Time	Owner
1	Implement resetting the game state when a game has completed and a new game has been started	2 hr	Delun
2	Test that the Host has all the same players still connected from last game	3 hr	Delun
3	Test that the Host’s variables and game state have been cleared/reset (Variables for prompts, scores, current screens)	1 hr	Delun
4	Test that the Host sends the correct messages to the server	2 hr	Delun
5	Test that the Host can accept new connections for players (if space for players exists) and audience (if space for players is full)	2 hr	Delun

6	Test that the Player is brought back to the initial “waiting for game to start” screen	1 hr	Delun
7	Test that the Player is able to change avatar and username	1 hr	Delun
8	Test that the Player is able to disconnect and reconnect to the game	2 hr	Delun
9	Test that the Player can still send messages to the server	1 hr	Delun
#	<b>Acceptance Criteria</b>		
1	Given that the Host is at the end of a game, when the Host calls to start a new game, then the Host will proceed to reset and prepare to start a new game.		
2	Given that the Host is at the end of a game, when the Host calls to start a new game, then the Players previously connected to the Host will stay connected.		
3	Given that the Host is at the end of a game, when the Host calls to start a new game, then the Host’s variables for scores, prompts*, current screens, etc. will be reset/cleared. (* prompts generated already are not cleared)		
4	Given that the Host is at the end of a game, when the Host calls to start a new game, then the Host sends a message to the Server to notify the Players connected that the game is resetting.		
5	Given that the Host calls to start a new game, when the Host is on the lobby screen, then the Host can accept new connections for Players (if there is room for more Players to connect) and/or audience (if there is no room for more Players to connect).		
6	Given that the Host is at the end of a game, when the Host calls to start a new game, then the Players will transition to the initial “waiting for game to start” screen.		
7	Given that the Host calls to start a new game, when the Host is on the lobby screen, then a Player is able to disconnect and reconnect to the game.		
8	Given that the Host has started a new game, when a Player or Host sends a message to the Server, then the Server receives the messages.		

<b>Sprint Story #9 (NF10) - As a developer, I would like to minimize tech debt from the first two sprints.</b>			
#	Task	Time	Owner
1	Refactor Host screen design to account for dynamic screen resolutions	5 hr	Jonathan

2	Refactor Player screen design to account for dynamic screen resolutions	3 hr	Jonathan
3	Refactor Player code to use onready keyword and variables to node paths instead of direct references to other nodes	2 hr	Jonathan
4	Refactor Host code to use onready keyword and variables to node paths instead of direct references to other nodes	2 hr	Jonathan
5	Refactor Player file structure to organize screen files and their elements into sub folders	4 hr	Jonathan
6	Refactor Host file structure to organize screen files and their elements into sub folders	4 hr	Jonathan
7	Refactor Host screen design to strictly use elements that can be altered by a theme resource file	2 hr	Jonathan
8	Refactor Player screen design to strictly use elements that can be altered by a theme resource file	2 hr	Jonathan
9	Refactor Player join game process to simplify code structure	2 hr	Jonathan
10	Fix Player ScreenManager not queueing old screens for deletion and wasting resources	0.5 hr	Jonathan
11	Create theme file(s) for use in Host and Player screens	4 hr	Jonathan
12	Regression test Host screens after refactor	3 hr	Jonathan
13	Regression test Player screens after refactor	3 hr	Jonathan
#	<b>Acceptance Criteria</b>		
1	Given that the application is launched, when the Host screens are viewed on different screen sizes, then the application screens will draw dynamically based on the screen resolution		
2	Given that the application is launched, when the Player screens are viewed on different screen sizes, then the application screens will draw dynamically based on the screen resolution		
3	Given that the application is launched, when the screens are viewed, then the screens will use theme files rather than preset images and color rects.		
4	Given that the Player is running the application, when the Player screen changes, then the old screen will be queued for deletion instead of being drawn behind newer screens.		

**Sprint Story #10 (F31) - As a player, I would like to experience a diverse set of prompts such that at least two consecutive games have little overlap in the prompts chosen**

#	Task	Time	Owner
1	Brainstorm additional prompts to increase game variety	10 hr (2 ea.)	All team members
2	Add approved prompts to the prompt file and resources used by the applications	1 hr	Jonathan
3	Implement prompt parsing to use tagged elements such as <username> and insert the necessary data	0.5 hr	Maxwell
4	Test that newly added prompts appear in the game generated prompts	1 hr	Jonathan
5	Test that prompts with <username> are able to splice currently connected Player usernames into the prompts	1 hr	Maxwell
6	Test that the same prompts don't appear in a second game session by the same Host	1 hr	Jonathan
#	Acceptance Criteria		
1	Given multiple games are played in the same Host lobby, when new prompts are generated from the prompt file(s), then the prompts will be unique from already seen prompts until all of the prompts are exhausted.		
2	Given prompts are being generated, when a prompt containing a tag such as <username> is encountered in a prompt, then the tag is parsed such that the generator knows what information should replace the tag in the prompt.		
3	Given that a prompt contains a tag, when the generator determines what information should replace the tag in the prompt, then the information is successfully inserted into the prompt when it is sent to Players and stored on the Host.		
4	Given new prompts are added to our resource files, when prompts are generated in game, then the newly added prompts also appear in the prompts generated.		
5	Given the game has advanced to a prompt answering stage, when prompts are generated, then they are chosen randomly from the resource file (i. e. different from session to session)		

**Sprint Story #11 (F38) - As a developer, I would like to remove inactive/unresponsive lobbies from the table of hosts on the server**

#	Task	Time	Owner
1	Implement Server pinging hosts that are connected to it.	1 hr	Maxwell
2	Implement changing Host “last ping” timestamp when server receives ping response	1 hr	Maxwell
3	Implement comparing Host “last ping” timestamp on second time interval to see what Host should be removed.	0.5 hr	Maxwell
4	Test Server sends ping to host on set time intervals	1 hr	Maxwell
5	Test Host sends ping response	0.5 hr	Maxwell
6	Test Inactive/Unresponsive hosts are removed properly	1 hr	Maxwell
#	Acceptance Criteria		
1	Given that a Host is running and connected to the server, when the Server sends a ping message to the Host, then the Host responds to the Server.		
2	Given that the Server is running, when a Host responds from a ping message, then the Server updates the Host’s “last ping” value.		
3	Given that the Server is running, when the Host timeout interval expires, then the Server removes Hosts with a “last ping” value that is too high.		

Sprint Story #12 (F39) - As a developer, I would like to handle mid-game disconnects such that players can resume their games			
#	Task	Time	Owner
1	Implement Player reconnection.	5 hr	Maxwell
2	Implement Player reconnection on Server side	3 hr	Maxwell
3	Implement pinging Players to check on state of Player connection	1 hr	Maxwell
4	Implement Server functionality to send to host that a player disconnected.	1 hr	Maxwell
5	Implement Host functionality to handle a Player disconnection during mid-game.	5 hr	Maxwell
6	Test Player reconnection when Player client intentionally disconnects	2 hr	Maxwell

7	Test Player reconnection when Player client crashes	1 hr	Maxwell
8	Test Player reconnection when Player client loses connection to the server	1 hr	Maxwell
9	Test Host handles mid-game Player disconnections to not disrupt gameplay	1 hr	Maxwell
#	<b>Acceptance Criteria</b>		
1	Given that a game is currently in progress, when a Player disconnects, then the corresponding Host is acknowledged and handles the disconnection properly by updating variables based on current game state.		
2	Given that a game is currently in progress, when a Player disconnects, then the Player saves its Player UUID.		
3	Given that a Player has disconnected from a game, when a Player attempts to reconnect to a Host, then the server maps the player with the UUID stored and corresponding Host.		
4	Given that a Player has disconnected from a game, when a Player attempts to reconnect to a host, then the Host handles the reconnection.		
5	Given that a game is in progress, when the Server sends a ping message to the Players, then the Players will respond to the ping and the Server will update stored Players based on results.		

Claimed Hour Totals For Sprint 3	
Teammate	Hours
Jonathan Poholarz	49
Maxwell Jones	45
Manoj Polisetti	44.5
Andrew Ring	45.5
Delun Shi	45.5

## Backlog

(Stories from Sprint 1)

(Stories from Sprint 2)

(Current Stories for this sprint)

### Functional Requirements:

1. **As a host**, I would like to create a game lobby and display the lobby's server generated letter code on screen
2. **As a host**, I would like to begin the game once enough players have joined the lobby
3. **As a host**, I would like to display a lobby screen while waiting for players to join
4. **As a host**, I would like to display a waiting screen while players are inputting answers to prompts
5. **As a host**, I would like to ignore/skip players who have not submitted answers to prompts within the allotted time and continue the game
6. **As a host**, I would like to display a results screen for each prompt of each round
7. **As a host**, I would like to display a total results screen at the end of each round
8. **As a host**, I would like to display a final results screen and winner/credits screen after the game completes
9. **As a host**, I would like to play additional games after the first game without forcing all players to disconnect and reconnect to a new lobby
10. **As a host**, I would like to exit the game properly, disconnecting all players and blocking further connections to the defunct game lobby
11. **As a host**, I would like to allow audience members to join at any time after the main players have connected
12. **As a host**, I would like to set up additional game modes (if time permits)
13. **As a host**, I would like to hear sound effects throughout the game (if time permits)
14. **As a host**, I would like to experience animations that add a personality to the game (if time permits)
15. **As a host**, I would like to host the game via a web-player (if time permits)
16. **As a player**, I would like to connect to a game lobby using my phone through the game application
17. **As a player**, I would like to choose my in-game username and avatar emoji
18. **As a player**, I would like to quit the game in-between rounds if desired, releasing my spot in the player list for a new player to join
19. **As a player**, I would like to receive instructions and prompts from the host computer screen during play
20. **As a player**, I would like to view a prompt response screen on my device for each assigned prompt in each round
21. **As a player**, I would like to view a voting screen on my device for each round of prompt voting
22. **As a player**, I would like to view a waiting screen on my device in-between tasks in which the game requires no further inputs from me
23. **As a player**, I would like to input emojis in my responses to prompts easily via a straightforward GUI
24. **As a player**, I would like to edit my response before submission - adding, moving, or deleting emojis with the GUI

25. **As a player**, I would like to vote for answers to the prompts using the GUI
26. **As a player**, I would like to select multiple answers during the final round when needed
27. **As a player**, I would like to view the results of the votes on the host screen
28. **As a player**, I would like to view round summaries on the host screen
29. **As a player**, I would like to view visual confirmation on the host screen when my inputs and votes are properly received
30. **As a player**, I would like to reconnect to a running game after disconnecting and be able to continue playing as the same player with the same username and same avatar
31. **As a player**, I would like to experience a diverse set of prompts such that at least two consecutive games have little overlap in the prompts chosen
32. **As a player**, I would like to view the remaining time for a question on the host screen
33. **As a player**, I would like to view a synced-with-the-host remaining time counter during answer and voting rounds on my device (if time permits)
34. **As a player**, I would like to connect to the game via a web-player (if time permits)
35. **As a developer**, I would like to maintain a server for which host computers can obtain a letter code for their game lobbies
36. **As a developer**, I would like to maintain a server to facilitate communication between the host computer and player devices in each lobby
37. **As a developer**, I would like to maintain a server which will connect player phones to the correct host computer lobby for each respective letter code
38. **As a developer**, I would like to remove inactive/unresponsive lobbies from the table of hosts on the server
39. **As a developer**, I would like to handle mid-game disconnects such that players can resume their games
40. **As a developer**, I would like to view crash messages in case of errors when possible  
===== Extra Stories we didn't have before =====
41. **As a player**, I would like to view the prompt along with the emoji responses on my device while voting

### Nonfunctional Requirements:

1. Code should be well organized and commented where necessary.
2. Scalability for each individual game session should not be an issue by setting an upper limit for how many players can participate.
3. Game will support Windows/OSX/(Linux if time permits) and Android/iOS
4. Application-server communication should operate responsively with low enough latency to not impact game participation (i.e. it doesn't take too long for answers to be sent)
5. Usability should be simple so that as wide an audience as possible is reached.
6. For security, inputs should be sanitized to prevent communication or usage of ill-formed data which may result in unintended effects
7. There should be a unifying theme across the applications and ports (if time permits)  
===== Extra Stories we didn't have before =====



8. **As a developer**, I would like to become familiar with Godot
9. **As a developer**, I would like to host a server in the cloud.
10. **As a developer**, I would like to minimize tech debt from the first two sprints.