

Requirements and Specification

Document Revision History

Rev. 1.1 2020-10-17 - Revised in format

Rev. 1.3 2020-11-13 - UserStory revised overally

- 1.0 : Initial version
- 1.1 : Revised in format
- 1.2 : UI & UserStory1(signin) Revised
- 1.3 : UserStory revised overally
- 1.4 : Delete Discord Button
- 1.5 : Delete setting page and invitation in chatroom

Project Abstract

shallWeGame provides a neat and efficient platform for gamers to share daily game routines. Users can freely show their identities as an active playing gamer by posting their game lives. Also, they can look for other gamers to play with, see posts about games they like, or that their followers have posted. In **shallWeGame**, users can send 'shallWe' privately to whom they hope to play with through posts. We expect our service to be a platform that users actively communicate with other gamers on specific games they like, and easily accessible to those who are interested in games.

Customer

shallWeGame targets ...

- General Customers:
 - People who like to see posts about games
- Specific Customers:
 - Gamers
 - who likes to share their game lives
 - who wants to find anonymous gamers to play with
 - who likes to see posts about games or gamers they like

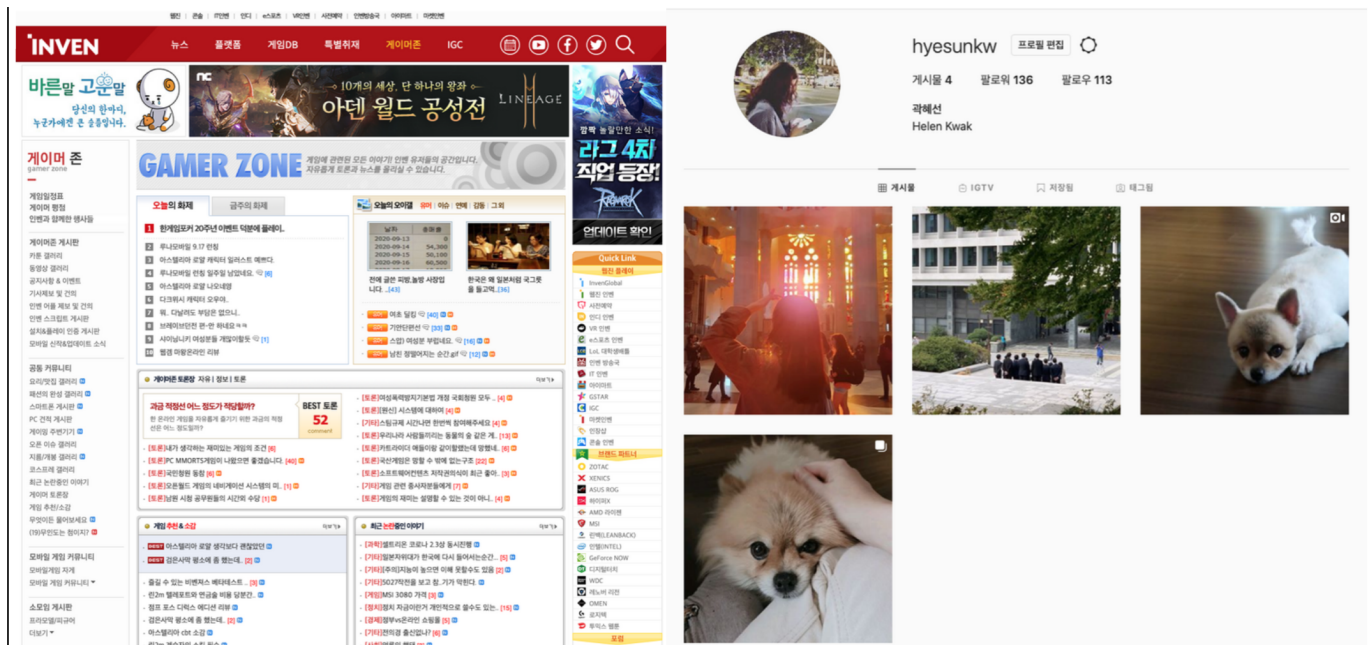
Competitive Landscape

shallWeGame is a microblog for gamers. Microblog is a kind of social network service(SNS), which delivers fragmented information with one or two sentences posted by individuals in real-time. In this section, we compare **shallWeGame** with two existing services: 'Instagram', a microblog service, and 'Inven', a non-microblog SNS for gamers.

The characteristics of the two services are specified below.

- Instagram

- Instagram is a microblog worldwide which focuses on sharing images.
- Users follow people who are their friends in real life.
- Users follow people who post images they are interested in.
- Hashtags are used to search posts related to certain topics.
- Some users posts images and follow others for commercial purposes
- Inven
 - Inven is a non-microblog SNS that is actively used by many gamers in Korea.
 - Bulletin boards are distinguished by games so that users can upload related posts.
 - Posts are relatively long and can contain many images.
 - Users can look for or join game parties by posting on bulletin boards.
 - Information and news about games are provided by journalists or normal users.



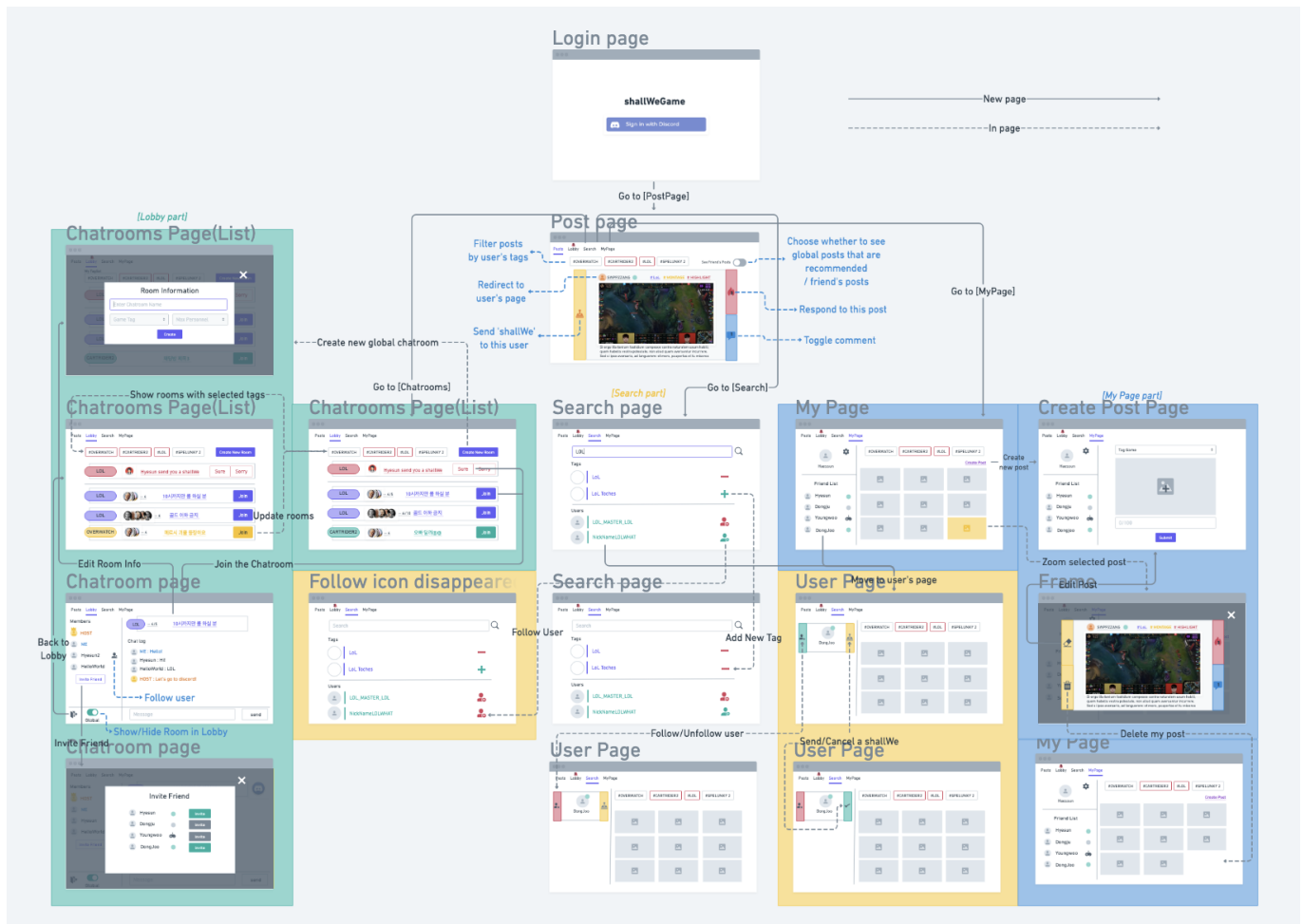
shallWeGame is similar to Instagram in the form of a microblog. Users can share posts with short texts and images. Also, **shallWeGame** borrows the concept of 'follow' in Instagram, so that users form and reinforce social relationships through posts and communication. We expand the use of this functionality for gamers to play games together. If users follow other users, they not only see posts of followers but also invite them to the chatroom.

shallWeGame is alike Inven as SNS for gamers. However, **shallWeGame** limits the form of posts to show posts in a catchy way. Also, unlike Inven, **shallWeGame** provides private space for individuals such that they can express their identity as gamers and share their own gaming lives. Also, users can find other users to play with among their followers or even globally.

To sum up, **shallWeGame** is distinct from the existing services by the following properties:

- **shallWeGame** provides individual spaces for gamers to post their own gaming lives.
- **shallWeGame** provides gamers to find users to play certain games among their followers or globally.
- In **shallWeGame**, users can propose other users to play games with through posts they like.

User Interface Requirements



User Stories

Common Features

Story 1

Feature: User wants to watch posts only with clicked tags

Actors: Any users

Precondition:

The user must be registered and logged in
User is recently on Post page or My Page, or the other user's page

Trigger: User clicks unclicked tag

Scenario:

1. User wants to watch posts with certain tag
2. User clicks that certain tag
3. Posts that contains clicked tags are shown to user

Exceptions: None

Acceptance Test:

Given that certain tag is recently disabled
When user clicks that tag
Posts that contains that tag are shown to user

Story 2

Feature: Users can send "shallWe"

Actors: Any users

Precondition:

The user must be registered and logged in
"shallWe" receiver must have enabled "shallWe"

Trigger: User clicks "shallWe" either on receiver's post or page

Scenario:

1. User wants to play with someone who is not in user's friend list
2. User clicks "shallWe" button which is available in the receiver's post or page
3. An empty chatroom is created which is only visible to sender and receiver.
4. If receiver enters the chatroom, user enters a chatroom automatically. -(1)

Exceptions:

(1) Receiver and sender does not enters chatroom if he/she rejects "shallWe"

Acceptance Test:

Given that receiver enabled "shallWe"
When the user clicks the "shallWe" button
Then empty chatroom is created

When receiver accepts "shallWe" and enters chatroom
User enters the chatroom automatically

When user rejects "shallWe"
Nothing happens

Login page

Story 1

Feature: User can sign in

Actors: Users who not logged in

Precondition: The user must be registered and not logged in

Trigger: User clicks 'Login with Discord' button

Scenario

1. User wants to sign in
2. User clicks 'Login with Discord' button
3. Redirects to Discord Page

Exceptions: None

Acceptance Test:

Given user entering ID and password and clicking 'sign in' button
User should sign in and redirected to main-page

My Page

Story 1

Feature: User can see post with its tags and contents from post thumbnail

Actors: Any users

Precondition:

The user must be registered and logged in
User is recently on My Page or Friend's Page

Trigger: User clicks post

Scenario:

1. User wants to see post with its tags and contents
2. User clicks certain post
3. Clicked post are zoomed to show its tags and contents
4. User clicks zoomed post to close zoomed post

Exceptions: None

Acceptance Test:

Given that User is recently on My Page or Friend's page
When user clicks certain post
And post is zoomed

Friend's page

Story 1

Feature: User wants to follow other user

Actors: Any users

Precondition:

The user must be registered and logged in
User is recently on the other user's page

Trigger: User clicks follow button on the other user's page

Scenario:

1. User wants to follow the other user
2. User clicks the Follow button on the other user's page – (1)
3. The other user should be on the user's friend list

Exceptions:

(1) If the other user is recently on user's friend list, Follow button for that user is disabled

Acceptance Test:

Given that the other user is not recently on user's friend list
When user clicks Follow button on the other user's page
And the other user is added to user's friend list

Chatroom List

Story 1**Feature:** User can accept "shallWe"**Actors:** Any users**Precondition:**

The user must be registered and logged in
User is online and has received more than one "shallWe"

Trigger: User clicks Sure button**Scenario:**

1. User received "shallWe" from the other user – (1)
2. If user accepts "shallWe" by clicking Sure button,
3. User enters the chatroom

Exceptions:

(1) If user is already on the other chatroom, user can't check out received "shallWe" until user exits recent chatroom

Acceptance Test:

Given that user is online, received "shallWe" and recently not in chatroom
When user clicks Sure button
User enters chatroom

Story 2**Feature:** User creates new chatroom

Actors: Any users

Precondition:

The user must be registered and logged in
User must be on Lobby page

Trigger: User clicks "Create New Room" button

Scenario:

1. User wants to create a new chatroom
2. User clicks "Create New Room" button
3. User enters a new chatroom

Exceptions: None

Acceptance Test:

Given that user is recently on Lobby page
When user clicks "Create New Room" button
New chatroom with the user is created

Post Page

Story 1

Feature: User can toggle to watch Friend's posts or Global Post

Actors: Any users

Precondition:

The user must be registered and logged in
User is recently on Post Page and watching Friend's posts

Trigger: User toggles to Global posts

Scenario:

1. User wants to watch Global posts
2. User toggles
3. User is now watching Global posts which are recommended automatically

Exceptions: None

Acceptance Test:

Given that User is now watching Friend's posts
When user toggles
User is watching recommended Global posts on Post page

Search page

Story 1

Feature: User can search Friends

Actors: Any users

Precondition: The user must be registered and logged in

Trigger: User clicks on the search tab in menu bar

Scenario:

1. Given user wants to search friend
2. User fills search bar with friend's name
3. User clicks on the 'search' icon
4. Then user get the friend's profile

Exceptions:

Nothing happens if search word is empty
Show nothing if there is no result

Acceptance Test:

Given that the user types name in the 'search bar' input
User clicks on the the 'search icon'
User should get the list of friends containing input

Story 2

Feature: User can add Friend

Actors: Any users

Precondition:

The user must be registered and logged in
The target friend must be searched

Trigger: User search friend or click search tab in menu bar

Scenario:

1. Given user wants to add friend
2. User clicks on the 'add friend' icon next to the target friend
3. Then user add the target as the new friend
4. Then 'add' button of target is replaced with 'delete' button

Exceptions:

None

Acceptance Test:

Given that the user clicks on the 'add friend' icon
User add the target as the new friend
Then 'add' button next to friend profile on search page is replaced with 'delete' button

Story 3

Feature: User can search game tags

Actors: Any users

Precondition: The user must be registered and logged in

Trigger: User clicks on the Search tab in menu bar

Scenario:

1. Given user wants to search game tag
2. User fills search bar with name of the game tag
3. User clicks on the 'search' icon
4. Then user get the profile of the game tag

Exceptions:

Nothing happens if search word is empty
Show nothing if there is no result

Acceptance Test:

Given that the user types name in the 'search bar' input
User clicks on the 'search icon'
User should get the list of game tags containing input

Story 4

Feature: User can add game tag

Actors: Any users

Precondition:

The user must be registered and logged in
The target game tag must be searched

Trigger: User search friend or click search tab in menu bar

Scenario:

1. Given user wants to add game tag
2. User clicks on the 'add game tag' icon next to the target game tag
3. Then user add the target as the new game tag.
4. User see the popup message "New tag 'tag_name' added!" and list of existing game tags

Exceptions:

None

Acceptance Test:

Given that the user clicks on the 'add game tag' icon
User add the target as the new game tag

Then user see popup message "New tag 'tag_name' added!"
Then user should be on search page with 'delete' button next to added tag profile

Chatroom page

Story 1

Feature: User can chat with other members in the chat room

Actors: Any users

Precondition:

The user must be registered and logged in
The user must be joined in chatroom

Scenario:

1. Given user wants to chat with other members
2. User writes a message in the chat input
3. User clicks on the 'send' button
4. Then all of the members in the chat room should see the message in the chat box

Exceptions:

Send button is disabled if chat input is empty

Acceptance Test

Given that the user have empty chat input
User should not able to click send button because it is disabled
When User writes message on chat input
Then send button is enabled
When User clicks send button
Then all of members in the chat room should see the message in the chat box

Story 2

Feature: User can add other members in the chatroom as friends

Actors: Any users

Precondition:

The user must be registered and logged in
The user must be joined in chatroom
There should be other member in the chatroom who is not friend

Scenario:

1. Given user wants to add other member in the chatroom as friend
2. User clicks on the 'add' button next to the target member
3. Then the target member is added to friend list

Exceptions:

'add' button is disabled if all of members in the chatroom is already friend

Acceptance Test

Given that the user see the 'add' button next to the members who is not friend
When user clicks on the 'add' button next to the target member,
Then user add target member to friend
User should stay in the chatroom after adding friend

Story 4

Feature: User can open chatroom to public

Actors: Any users

Precondition:

The user must be registered and logged in
The user must be the host of the chatroom

Scenario:

1. Given user is the host of the chatroom not opened to the public
2. User wants to open the chatroom to the public
3. User toggle switch to 'global' from 'private'
4. Other users not joined in chatroom can see the user's chatroom in Lobby page

Exceptions:

'private/global' toggle switch is disabled if the user isn't the host of the chatroom

Acceptance Test

Given that user first enters the chatroom, 'private/global' switch toggle is 'private'
Other users not joined in the chatroom can't see this chat room
When user toggles the switch to 'global'
Then other users not joined in the chatroom can see this room in Lobby page

Story 5

Feature: User can exit the chatroom

Actors: Any users

Precondition:

The user must be registered and logged in
The user must be joined in chatroom

Scenario:

1. User wants to exit the chatroom
2. User clicks on the 'exit' button – (1)
3. User is redirected to Lobby page

Exceptions:

(1) If the user is host of the room, another member of the chatroom who joined right after the user will be the host

Acceptance Test

Given the user first enters the chat room, user can see 'exit' button
When user clicks on the 'exit' button,
Then user should be on the Lobby page
When user who exits was the host of the chatroom,
Then the member who entered the chatroom right after the exited user will be the host