Cloud Music Player

-Team 30-

Guojie Wen
Dai Zhong
Pengyu Chen
Zhe Chen

Sprint 2 Retrospective

Nov. 6th, 2017

What went well?

	User story	Design features	Implementation members	Hours
3	As a returning user, I would like to be able to view the music interface after login	Be able to show user's music playlists on the sidebar UI	Guojie Wen	6
	See above	Set up Firebase to be able to pull playlists	Guojie Wen	6
	See above	Be able to show music in the playlists	Guojie Wen	6
	See above	Test Firebase playlists listing	Guojie Wen	2

Completed

Firebase is completely integrated to our application and functional. Our application can fetch users’ playlists, and user is able to view their playlists from firebase.

	User story	Design features	Implementation members	Hours
4	As a user, I would like to be able to navigate and browse all my Google Drive file directories	Implement logic to navigate through Google Drive	Guojie Wen Dai Zhong	6
	See above	Implement "Go to parent" buttons	Guojie Wen	3
	See above	Test navigating UI	Dai Zhong	1

<u>Completed</u>

User is able to view user's Google Drive files from Google Drive API on the UI. User is able to click on a folder and see the content within. And user is able to go back to the previous directory.

	User story	Design features	Implementation members	Hours
8	As a user, I would like to be able to go to the music interface after I finish selecting music files from Google Drive	Implement functionality to go to the music interface	Zhe Chen	4
	See above	Implement ability to save selected music to the back end	Zhe Chen Guojie Wen	5
	See above	Test the UI design for going back page	Zhe Chen	3
	See above	Test the Firebase data save for going back page	Zhe Chen	3

Completed

When user chooses to go to the music interface after finishing importing music files, then they will see the music play list that they have created when the interface appears.

As selecting the music that they need, the music files that are clicked will be added to the database of the playlist. Besides, when the user clicks the music file name, they will be able to notice the color of music files' names have been changed.

	User story	Design features	Implementation members	Hours
9	As a user, I would like to be able to view all the playlist names I created on the left side of the screen	Be able to show all the playlist names	Pengyu Chen	10
	See above	Be able to show the name in bold	Pengyu Chen	1
	See above	Testing all playlist names are correctly shown	Pengyu Chen	2

Completed

User is able to see the playlist names in bold on the sidebar. Also, user can see the playlist names are the same as these names they created in the firebase, our database back end.

	User story	Design features	Implementation members	Hours
10	As a user, I would like to be able to click the playlist name to expand and view the entire playlist	Be able to expand the playlist	Pengyu Chen	10
	See above	Be able to view the songs in playlist	Pengyu Chen	10
	See above	Test the playlists can be expanded	Pengyu Chen	2
	See above	Test the songs in the playlists are correct	Pengyu Chen	2

Completed

User is able to expand the playlist by clicking the playlist name. User can see that the expanded sections have the correct names of the songs. Also, the expanded section can disappear by clicking the playlist name again.

	User story	Design features	Implementation members	Hours
11	As a user, I would like to be able to left click on a song, play the song, and highlight the currently playing song in the playlist.	Be able to left click on a song	Zhe Chen	5
	See above	Be able to play the song in the music player interface	Zhe Chen Guojie Wen	4
	See above	Implement the UI highlight for the current song	Zhe Chen Guojie Wen	4
	See above	Testing UI interface and functionality for playing the music	Zhe Chen	3

Completed

Left-clicking, highlighting, and playing the song are implemented and completely functional in the music player page. When user left-clicks the song in the play list, he or she can hear that song immediately and at the same time they can see the song playing in highlight.

	User story	Design features	Implementation members	Hours
19	As a user, I would like to be able to see the song's name, singer name, and album name in the player interface.	Create data structures to contain songs in Redux	Dai Zhong	5
	See above	Create UI to render song's information	Guojie Wen	3
	See above	Implement logics to renew data in Redux	Dai Zhong	5
	See above	Implement logics of rendering songs	Guojie Wen Dai Zhong	6
	See above	Test showing song's information	Guojie Wen	1
	See above	Test renewing data in Redux	Dai Zhong	1

Completed

When user clicks a song in the playlist to play, he or she is able to view the song’s name, the singer’s name, and the album name in the right side of the music player page. Also, the song’s picture can be shown with the designed style in the design document.

	User story	Design features	Implementation members	Hours
22	As a user, I would like to logout my Google Account when I want to do it.	Develop Google user sign out function	Guojie Wen	2
	See above	Finish up navbar styling	Guojie Wen	6
	See above	Test navbar UI	Guojie Wen	1
	See above	Test Google user sign out	Guojie Wen	1

Completed

User is able to view navbar, which is responsive, after they sign in, and user is able to log out of both Google API and Firebase once they click on the sign out button.

	User story	Design features	Implementation members	Hours
23	**As a user, I would like to hide the playlists on the left when the screen size of the browser gets small.**	**Create sidebar**	Guojie Wen	6
	See above	**Make sidebar autohide at small screen**	Guojie Wen	6
	See above	**Make toggle on navbar to show sidebar**	Guojie Wen	2
	See above	**Test UI functions**	Guojie Wen	1

Completed

User is able to view sidebar at big screen, user is able to toggle the sidebar to open at small screen, and user is able to sidebar can auto hide when screen shrinks.

What did not go well?

This time we finished all of our user stories planned for sprint 2 on time with good quality. However, below are some of the not well going fact that we would like to share among our team members after sprint 2 ends.

- ★ The dependencies between several user stories may contribute to the delay of some team work.
  - ○ As the team based project, some of the user stories were built upon the successful achievement of other user stories. Therefore, if a team member did not finish their part first, then the rest of the user stories will be delayed and unable to carry on.
- ★ Some backend logic issues were still existing but not affecting the overall user experience.
  - ○ In the audio files selecting page, if the user selected a file, then the rest of the music files might displace from previous places because of reordering according to alphabetical order. The functionality of selecting the song in the import page is achieved, but it might not provide perfect user experience in all circumstances.
- ★ Coding standard may have conflicts concerning different level of team members' coding experience.
  - ○ It was a common that during sprint 2, when one of the team members was about to finish their user stories, however another team member, with a higher level of coding experience, might later on changing some of the

codes, to reach their standards of user story features, which took unnecessary time and efforts to completely finish one functionality.

- ★ Basic syntax problems can still be a great challenge to coding process
    - ○ Since most of the team members are relatively new to the programming language like Javascript, CSS, and HTML, some basic syntax problems made by them can be time consuming to figure out on their own. For example, for user story 19, the programmer responsible for it got stuck for over 3 or 4 hours to figure out how to get metadata of the song by just URL. In the end, it was just that the way to do the http request was wrong.
- ★ Styling web application is painful for team members except for our leader
    - ○ One going through our commits history will notice that nearly all the styling jobs were done by Guojie Wen, our leader. The other members are very new to HTML and CSS, so styling web page can be ten times slower if jobs concerning it are left to other team members.
- ★ Lack of understanding the project causes the delay of the progress
    - ○ Without fully understanding of the project, sometimes we may miss something, and cannot find a better way to implement the user stories. Especially when we are doing logic stuff, if we create a new variable that we already have, that will be a fatal mistake.

How should you improve?

- **Deeper understanding in web development are necessary, but web development skills take time to master. Web development consists of HTML, CSS, JavaScript, and other non-development related skills, such as usage of libraries, development tools, architecture design, and so on. These skills take time and practice to develop, therefore in a short amount of time, it is difficult to improve, but we still need to work on these skills.**

- **More patience in research is needed. In this sprint, some members gave up on their tasks easily, did not spend enough time researching for what they are supposed to figure out, and handed their tasks to members that are more knowledgeable. Researching, analytical, and resource gathering skills are critical for software development, and these skills take time and experience to develop. Therefore, while it is difficult to develop these skills in time, having more patience in research, improving Google search queries for better results, and more critical thinkings for fitting search results into our application could be good starting points when it comes to developing these skills.**

- **Some team member may start their part of the project first without waiting other team members to finish their user stories. For example, one team member would write some onclick functions to handle the button clickings conditions on the top of UI buttons and even test the results through console in google chrome interface. It will save a considerable amount of time and make sure every member in the team is able to manage their time to finish the user stories without the impacts of unsuccessful functionalities in their user stories. On the other hand, we can finish our sprint at the faster pace and eventually enhance a better user experience for our customers.**

- **We need to have more communication with our project coordinator. In this sprint, we do not recognize that we need to add some test cases to test our user stories. However, in our project, it is hard to write a test case because we basically can use the data from firebase to visually display our progress of the user stories. Thus, we need to talk to our project coordinator more to let him know our situation.**

- **We need to be more efficient to work on the project. Since some teammates spend too much time working on some minor detail stuff and delay the progress of the whole project, we need to find the focus of the current sprint to improve the efficiency. Also, the better understanding of our project can also help us improve the efficiency.**

- **Removing dependencies in user stories allocation is very important. It allows team members to work concurrently without waiting for others to finish the parent stories. Of course, sometimes, to distribute jobs evenly among team members and to allocate user stories according to members preferences, dependencies are inevitable. But at least, we should let people whose personalities and skills are appropriate to finish the user stories which have role as parents.**