Sprint Review (Retrospective) Meeting

18/11/2024 – 01/12/2024

Group ID: **02**
Project Name: Beatify
Prepared by: **Võ Ngọc Khoa**

Date: 30/11/2024
Attendees:
22127152 – **Lê Gia Huy**
22127114 – **Phạm Hà Hiếu**
22127115 – **Trần Trung Hiếu**
22127203 – **Võ Ngọc Khoa**
22127031 – **Nguyễn Duy Bảo**

Sprint Outcomes

1. What Went Well

- The integration of the song recommendation system with the frontend was smooth, and the feature functions as expected.
- Both the Artist and Playlist pages were completed, with the user interface finalized according to the design specifications.
- The backend playlist management system was developed effectively, allowing seamless interaction between the frontend and database.
- The team communicated effectively, ensuring all tasks were completed on time and meeting quality standards.

2. What Went Wrong

- The initial integration of the song recommendation algorithm faced challenges with ensuring the accuracy, as some recommendations didn't align well with user preferences.
- Some unexpected design tweaks were needed for the Artist and Playlist pages, causing delays in finalizing the layout.
- The "like song" feature required additional testing and debugging due to minor inconsistencies between the frontend and backend functionality.

3. Problem Analysis
- The recommendation algorithm lacked sufficient data for precise predictions, and the integration process introduced some bugs that impacted its accuracy.
- The need for late-stage UI design adjustments stemmed from last-minute changes that were not anticipated earlier.
- The communication between frontend and backend was not flawless, especially with syncing data for the "like song" feature.

4. Improvements
- Better data validation and improved testing will ensure more accurate song recommendations in future sprints.
- More thorough planning in the design process will lead to fewer late-stage changes and help avoid delays.
- Greater focus on integrating frontend and backend testing earlier in the sprint will help identify issues sooner.
- Improved documentation for backend interactions and APIs will prevent confusion during integration.

5. Lessons Learned
- Early testing of algorithms, features, and UI designs helps identify potential issues ahead of time, reducing last-minute adjustments.
- Regular check-ins and clearer communication kept the team on track, but involving everyone earlier in the design process would have been beneficial.
- Prioritizing frontend and backend integration throughout the sprint would help synchronize the features and avoid bugs during development.