

Team: DEWBED

Members: Bettina, Devika, Emily, William

Project: 2C - Post-Mortem Analysis

Status Meeting Minutes:

[https://github.com/bettinabenitez/introspectivespotify/blob/master/P2C%20-%20Final%20Architecture/minutes\\_2c.txt](https://github.com/bettinabenitez/introspectivespotify/blob/master/P2C%20-%20Final%20Architecture/minutes_2c.txt)

Time Management:

<https://docs.google.com/spreadsheets/d/1oINr5vyT0B59K9UyjJcFpJVJsEYvTNIFgNVoeYAGTYy/edit?usp=sharing>

Primary Author: Devika

### **Table of Contents**

[Creation of the Initial Architecture](#)

[Initial Refinement of the Architecture](#)

[Component Design Analyses](#)

[Creating Draft Architectural Description](#)

[Studying for Your Design Review](#)

[The Design Review Meeting \(as a reviewer\)](#)

[The Design Review Meeting \(as a reviewee\)](#)

[Writing up the Review Report and Working the Issues](#)

[Revising your Architecture to Address the Issues](#)

[The Planning and Ongoing Management of these Activities](#)

[The Overall Project as an Educational Exercise](#)

### **Creation of the Initial Architecture**

We initially found it difficult to begin the architecture due to our lack of experience. We decided to prepare some questions for Professor Kampe to ask during the lab. Professor Kampe was able to clarify some questions and even offer insight into what a potential Discord Bot prototype could look like. After meeting with our professor, we were able to hash out our ideas. As a group, we decided on the major modules and split up the tasks based on the modules and the creation of a prototype. Each member scoured the internet for APIs, documentation, and source code. In the end, although our initial proposal was not perfect, it laid a great foundation for our future tasks. Writing this proposal with a top-down and bottom-up approach taught us to understand our product initially at a high-level and then at a low-level.

### **Initial Refinement of the Architecture**

After our professor had given us our feedback for the initial architecture, we spent time refining and recreating the architecture to ensure it depicted a better picture for our review meeting. We initially found it difficult to find a time to meet due to timezone issues and we felt discouraged with our preliminary grade. One particular difficulty we had was determining how to describe the structure of "Introspective Spotify." We decided to reread McConnell section 3.5. This clarified how the architecture proposal should describe structure. We also reviewed Professor Kampe's comments and spent time individually to fix the architecture. The readings and feedback painted a better picture of how an architecture proposal should describe structure. To refine our architecture, we created a UML components diagram. During the creation of this diagram, we had a meaningful discussion about how we want our modules to interact with one another. What seemed like a daunting task at first, helped us build a much stronger understanding of our product.

### **Component Design Analyses**

Upon honing in on our components, we came up with all the commands and functions each major class should have. Whether a user needed to join a music session or whether a user wanted to analyze their music with their friends, there were a lot of commands. At first, we felt overwhelmed and researched for any more suggestions. Yet, when we came across the Spotify Web API, we realized that the engineers at Spotify had made their API very user friendly and had many of our components as classes in their documentation. We spent time each taking a component and writing out the different commands with their parameters, what they'd call, and what other components they'd interact with. Overall, the experience made us flesh out our components and forced us to ensure each command could be executed well.

### **Creating Draft Architectural Description**

After fleshing out our components, we focused on ensuring our architecture had a solid, well-rounded architectural description for our reviewing teams to look at. Since our initial architecture and our component design went through so many edits, it was easier to write our overall description after learning from our mistakes. We ensured to mention each component and addressed how our bot was connected to both Discord and Spotify. Additionally, we ensured we gave a brief rundown of the initial steps someone would take to start interacting with our bot. Lastly, we ensured we had screenshots from our initial prototype and even gave a picture of the executing code as well. Overall, this part of Project 2 was easier after we had to drastically change our overall architecture.

### **Studying for Your Design Review**

Finding another team was quite easy as one of our teammates knew another team with an international student. We were able to have virtual introductions through Facebook messenger and discussed potential times. A major setback was the difficulty of scheduling a meeting for eight different people, especially with multiple time zones. Something that relieved this

confusion was the creation of a “When to Meet” survey. Since the other team’s architecture was based on Dungeons and Dragons (D&D), we encountered difficulty due to our lack of understanding of D&D. Reflecting on this experience, we regret not taking the initiative to ask Discord Dragons to send us their initial proposal and requirements. We believe that would have given us the context to prepare better questions. However, we are happy with how diligently we each prepared for the meeting. Each DEWBED member took 48 hours to read through the other team's review and create their personal notes. After we submitted our individual notes to Professor Kampe, we created a collaborative document with all of our questions, and created a guide for our virtual review. Each member had a role; one member was a scribe and the rest was assigned an equal number of questions. The questions were ordered strategically to follow the structure of their submitted proposal. This guide proved to be very useful during the meeting because it ensured that we remembered to ask all of our questions. This experience taught us a lot about collaborating with teams who may conduct meetings at different times than us. We also learned the importance of preparing for inspection meetings, as we thought of the majority of our suggestions prior to the meeting. We are also glad we created new friendships in our class!

### **The Design Review Meeting (as a reviewer)**

This review meeting was a bit difficult to facilitate due to our inexperience with D&D. As aforementioned in the design review section, we regret not taking more initiative to ask for Discord Dragon’s proposal and requirements document. The majority of the conversation related to clarifying questions about D&D. However, towards the end of the meeting, we were able to gain a better understanding and ask some architecture-related questions. Despite our initial confusion, we are glad that we had a productive call where all members participated calmly and respectfully. We are happy that we reassured the other team with positive affirmations at the beginning and end of the meeting. We even personally messaged the

Discord Dragons team after the meeting to congratulate them on their product! Overall, we learned the importance of understanding that the product you are reviewing and the importance of having a positive, safe environment to discuss.

### **The Design Review Meeting (as a reviewee)**

Due to our inherent bias, it was difficult to objectively review Introspective Spotify ourselves. So, we were excited to have an opportunity to have others review our architecture. This meeting proved to be a great resource. We were able to hear helpful suggestions (i.e. having lowercase commands) and helpful concerns (i.e. trolling concerns). These notes were not ones that we thought of ourselves, so we are grateful others brought them to our attention. This design review meeting emphasized how essential non-biased reviewers are to software development and quality assurance.

### **Writing up the Review Report and Working the Issues**

After our meeting, we quickly met to debrief and have our scribe, Bettina, go over her notes from the meeting. We then took her notes and assigned different categories to display issues raised, must-haves, suggestions, and comments. While creating the document, we took the time to re-read any architecture from the other team and have quick discussions about each category. In the end, we discovered new questions and had solid ideas for the reviewed team. We ensured that the other team was able to receive our comments and even allowed them to ask any other questions they may have. In the end, the reviewed team made this process easier as their architecture did not have many issues and they answered questions with solutions to any defects, issues, and suggestions.

### **Revising Your Architecture to Address the Issues**

After Discord Dragons sent us their report, we discussed issues individually. We decided to focus on fixing the “must-fix” issues first. We divided the issues among ourselves; each member reviewed the suggestions, performed additional research, and added it to the final proposal (when appropriate). Some suggestions could not be implemented, such as dequeuing a song, as the Spotify API does not have that feature, yet. We also delegated some suggestions, such as emoji voting to skip songs, for version 2.0 to make room for growth. While revising our architecture, we were reminded of how critical these revisions are. If we implemented Introspective Spotify without revising it first, we would likely run into large issues down the road that would be very expensive to fix. This step also taught us not to get defensive about criticism, but instead to use them as helpful tips to make our product better!

### **The Planning and Ongoing Management of these Activities**

When we first saw all the tasks for Project 2, we felt extremely overwhelmed. With the impending slam of midterms and the large deliverables due, we took the time during Project 2A to address any team members concerns about partitioning work and timezone difficulties. This project, we spent more time individually working rather than working on large team calls as we did for Project 1. We ensured that we kept each other updated with Google Doc comments and even continued to check in with members through Facebook Messenger. In the end, we were able to manage our time well with lab sessions and late night calls. Even though we had initial concerns about the overall time and timezones, we were able to commit to our plan and execute all deliverables with our hardest work by being adaptable to late night calls and scheduling our calls in advance.

### **The Overall Project as an Educational Exercise**

Overall, Project 2 helped our team understand the inner structure of what Introspective Spotify would be. While Project 1 made us feel excited about the potential of a Discord Bot, Project 2 displayed to us that it could be possible through our research. Initially, our team had a hard time

with the initial architecture, but with the help of the Professor's constructive criticism, we were able to revise it for our review meeting. Even the review meeting helped our team members understand our product even better through the reviewing team's questions. It was a great experience reviewing another team's architecture as well, as it made us feel more connected to our virtual classmates. We even learned some D&D through the process! Creating the final architecture was surprisingly easy as the reviewing team had created a wonderful review report, which was easy to go off of. While Project 2 started off on the wrong foot, we took it as a learning opportunity and embraced our mistakes with open minds. Learning from Project 1, we addressed all concerns in the beginning and ensured every team member felt on board. We can't wait to execute Project 3!