1. **Team Number:** 014-5

2. **Team Name:** Jicama B

3. **Team Members:**
   - James Nguyen, nikuteh, jang5198@colorado.edu - back end
   - Cara Wang, wangcara, cawa5085@colorado.edu - front end
   - Benedikt Safigan, bsafig, besa8557@colorado.edu - database
   - Mason Chansamone, mchansamone1, mach3999@colorado.edu - back end/database
   - George Fisher, georgefisher1024, gefi6263@colorado.edu - front end

4. **Application Name:** Spotigang

5. **Application Description:**

   Our application is a music-sharing platform that will be built using Spotify's free API. Users will be able to connect their Spotify accounts and view what their friends are listening to in real time. Users will also be able to post reviews to specific songs or albums.

   The features include being able to view the songs your friends are listening to, view your profile, view your friends list through your profile, rate an album or song, comment on others' ratings, and view your friends' Spotify wrapped. Additionally, there will be a social feed, like the feed from Instagram, that users can scroll through, to see all the reviews their friends have posted.

6. **Audience:** Spotify users who want to connect more with their friends' music.

7. **Vision Statement:**

   For music-lovers who want a better way to share reviews of music. Spotigang is a social media app that makes it easy to post and view reviews of songs and albums.

8. **Version Control:**
   Git Hub Link: https://github.com/CU-CSCI3308-Fall2025/group-project-bsafig

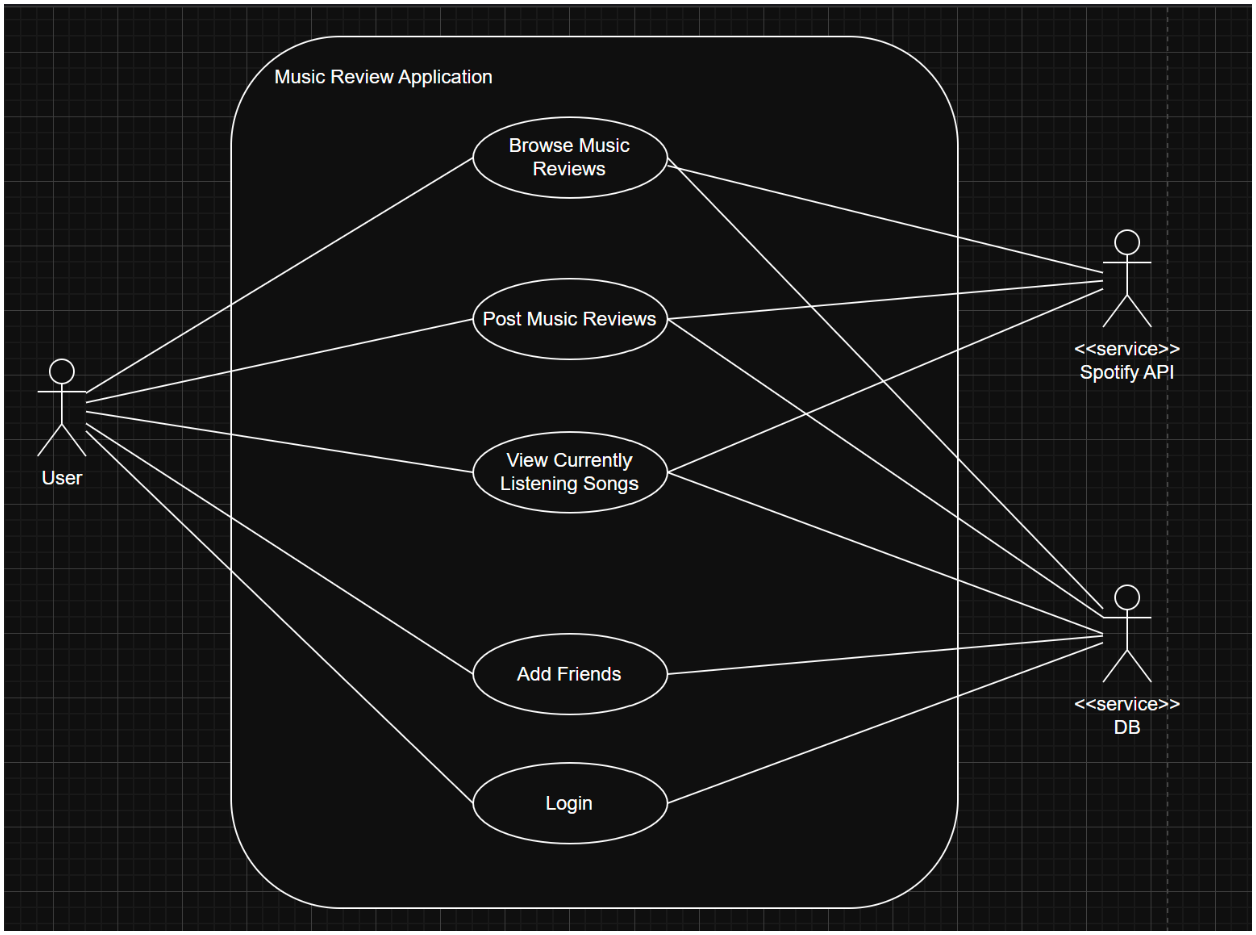
9. **Development Methodology:**

We will engage in weekly in-person scrum meetings every Monday in addition to our TA meetings. We do not plan to use Kanban boards, but we plan to use Git issues and milestones to organize our goals and progress.

10. **Communication Plan:** We will be communicating over Slack, with in person meetings weekly. We will use Git issues to communicate goals.

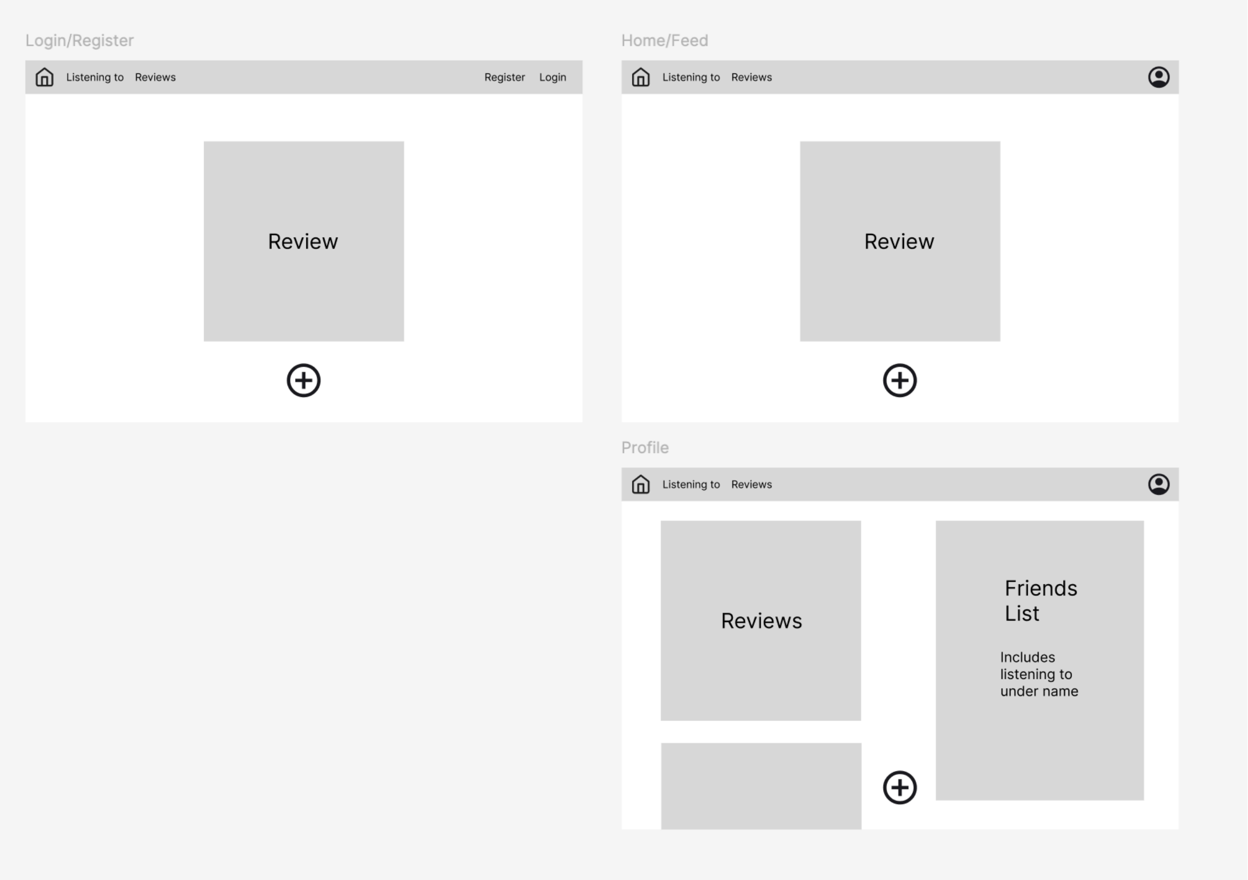
11. **Meeting Plan:**
    - Team Meeting: Every Monday, 4:25 PM, meeting In Person after class
    - Weekly Meeting with TA: Every Wednesday, 4:40 PM, meeting Online

12. **Use Case Diagram:**



13. **Wireframes:**
    - Home page
    - Listening to
    - Reviews
    - Profile

- Friends list
- Recently played
- Top 5 artists
- Top 5 songs
- Currently listening to



14. Potential risks:

- Data breach — SEVERE — data is stored in raw text. To mitigate risks from a breach we could hash all data, so even if the data is obtained, it's not readable without the hash keys

- Code Injection/SQL Injection, We need to ensure that any code a user may write can't be injected into our application which may enable them to steal data or harm our database.

-