CSC 648/848 SFSU 2024 Milestone 3

Review of functionality, UI, SW and planning for final product delivery

Includes Checklist for instructors and teams (Appendix I and II)

Objective and Overview

The objectives of Milestone 3 are to:

- **Define exactly what product you are delivering.** We will come to agreement on what the final application is going to look like in terms of functionality, especially which functional requirements are priority 1 (P1). This will be your commitment to the instructor/client to delivery by M5.
- **Ensure software development is on track.** We will verify that the all the SW components are installed and integrated and that most major functions work.
- **Provide feedback on all major UI screens and functionality**. This will be done through a review of the so-called horizontal or UI prototype.
- **Check software architecture** by reviewing the database and overall design at a high level UML class diagram and sequence diagram.
- **Check all algorithms** like search or machine learning component, whatever major algorithm in your application will be checked.
- **Identify and address all technical risks**
- **Ensure effective teamwork** by verifying that all team members have started implementation of their portion of the project.
- **Ensure software development is effective** by verifying your team's collaboration practices based on github branch policy and github review policy.

Milestone 3 will be in the form of two-part review:

Part 1 – each team presents to Prof. Song: Review of functionality, UI and general project status will be done during the meeting (with each team) of ~12 minutes. The meeting will happen during the class on the deadline day. In order to be efficient, teams must observe strict schedule and come fully prepared. For this, please prepare Appendix I (item 2,4) before the M3 demo. And please finalize Appendix I after the meeting based on Prof. Song's feedback. The Appendix 1 should be submitted to your M3 folder for review (Due : on the next day of M3 demo).
Part 2: SW review (in-emails): github usage, branch organization, code review practices by TA after Part 1 review.

M3 will be graded, and the feedback will be given if any. The teams will meet to analyze feedback and revise the design and implementation accordingly as well as definition of P1 features. After M3, the teams will have "feature freeze" e.g. the teams must focus on intense implementation of P1 features.

Part 1 Review – Functionality and UI feedback and general project status

What to bring to the meeting

Each team **must** prepare your ***Product prototype*** to the Milestone 3 Part 1 meeting:

- ***Product prototype*** has limited functionality
- You have to demonstrate the 5~6 key P1 functionalities for your product prototype for the meeting. For the key functionalities, you should connect back-end and front-end.
- The product prototype should provide UI implementation of 5~6 key P1 functionality.
    - The UI implementation should follow UX flow defined in M2.
- The current version of your SW should run on deployment server.

On the part1 meeting, the instructor will let each team to demonstrate major functionalities on real-time using your SW and will give you feedback. **You are requested to appoint a scribe who collects the feedback. Use Appendix I as a template to record feedback.**

After the M3 Part 1 meeting (recommended to do it immediately after the meeting): Team has to meet, analyze meeting feedback and revise M3 doc (Appendix 1), design and implementation as necessary. Team also must finalize P1 set of features. The instructor feedback as well as finalized P1 list MUST be written down using template as in Appendix I. You will submit it with M3 folder, by the next day of M3 part 1 meeting.

Part 2 review: SW review – to be done by TA after Part I review, by accessing your github repo. See Appendix II for details

Appendix I – Rubrics and checklist for Part 1 Milestone 3 review: Project Status and UI Review.

Section: 01 Team: 3 Date: 04/15/2024
Number of students present: 6

1. *UI and functionality feedback (P1 functions only)*

During the meeting, students will demonstrate to run your SW from deployment server**:**
- Test 5~6 P1 features
- Show UI and usability: adherence to the feedback on UI mockup at M2, layout, flow, clarity, functionality etc.

Instructor will
- Check functionality and record issues/observe bugs
- Share comments on key UI and functional implementation
- Verify enough web pages are implemented and connected
- Verify Performance of web page

Students must record the instructor's comment in the below. Then the team should meet to analyze feedback, prioritize and revise and plan to implement changes accordingly**. Note that immediately after the review meeting with the instructor, the team must finalize P1 set of features and focus only on those from then on.**

- Instructor's comments on UI/functionality for your demo (should be during the class of M3 demo)

- Your Plan for the comments

2. List of P1 features committed for delivery

Write down the candidate list of features before the demo and, verbally explain it during the meeting if time is allowed. Based on the instructor's review comment, finalize your committed

features. Once you commit at M3, you can not change during the rest of the semester. You should implement the committed features by M5.

- Authentication
- User Settings
- Create Post
- Post Page
- AI Recommendation
- Home Page - Public feed of all posts that users can interact with
- Profile Page
- Search Functionality
- Comments on Post
- Spotify Button for each song in the AI Recommendation

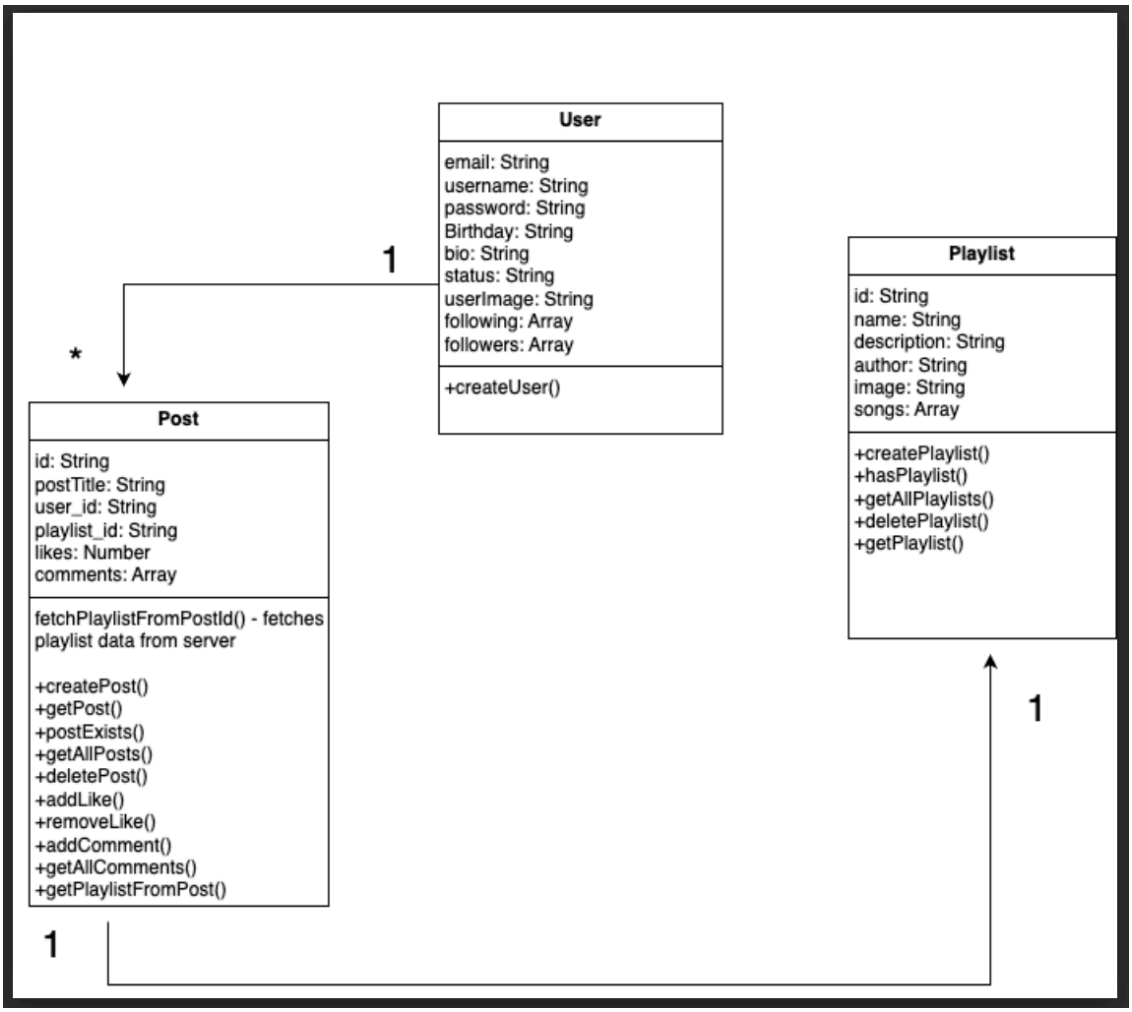
3. Architecture

Familiarize yourself with Unified Modeling Language (UML).

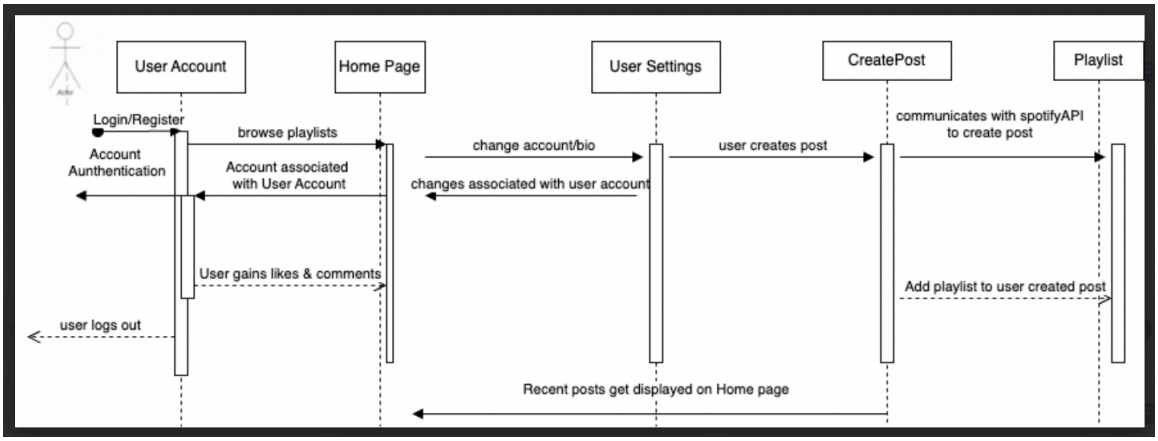
To implement your committed features, please provide

- High-level UML class diagrams for implementation classes of core functionality, Focus on main high-level classes only (one or at most two levels deep). This must reflect an OO approach to implementing your web site. For UML, you could find many references including http://edn.embarcadero.com/article/31863.

- High-level sequence diagrams: for ~5~6 functional requirements, please develop UML sequence diagram.



UML sequence diagram



4. ***Project status – <u>write down the items before the demo</u> and verbally explain it during the meeting if time is allowed***
    - Risks: all actual (not hypothetical) risks (schedule, team work, technical, skills etc.) should be identified and either resolved or plans should be made to resolve them.

- **Skills risks and mitigation plan**
  - ❖ Do you have a proper study plan to cover all the necessary technologies?

  - ➢ In addressing the skills risks associated with our project, we have implemented a comprehensive study plan covering all the necessary technologies. Each technology vital to the project has been meticulously outlined, including designated team members responsible for learning and mastering it. We have prioritized a well-rounded approach, ensuring that every team member possesses a foundational understanding of all project-relevant technologies.

  - ➢ Our study plan includes detailed timelines for each technology, allowing us to pace our learning effectively and allocate resources efficiently. By setting deadlines for mastering specific concepts within each technology, we aim to maintain a steady progression without compromising depth or breadth of knowledge.

  - ➢ Furthermore, we have compiled a list of key concepts for each technology, providing clear objectives for our learning efforts. This structured approach enables us to focus on essential skills and competencies necessary for project success.

  - ➢ Our proactive approach to skills development mitigates potential risks associated with knowledge gaps, ensuring that our team is well-equipped to tackle the challenges ahead.

- **Schedule risks**
  - ❖ Does your team have a team schedule for every member including their detailed task?

  - ➢ We conduct two team meetings every week to facilitate communication, collaboration, and progress tracking. The first meeting takes place during the second half of our software engineering class on Mondays, providing an opportunity to reflect on the previous week's accomplishments, challenges encountered, and plans for the upcoming weeks. Additionally, we hold a second meeting on Sundays from 12 pm to 1 pm, allowing for in-depth discussions and problem-solving sessions.

- ➢ During these meetings, each team member shares their achievements, challenges, and future plans, fostering transparency and accountability within the team. We actively engage in addressing challenges by collectively brainstorming solutions and offering assistance to overcome obstacles. This collaborative approach strengthens team cohesion and ensures that all members are supported in their endeavors.

- ❖ If change happens, does it update transparently? Does your team use project management tools (e.g. Jira, Trello)?

- ➢ In conjunction with regular meetings, we utilize Trello as our project management tool to streamline task allocation, tracking, and updates. Our scrum master oversees the organization of tasks within Trello, assigning them to team members based on their skills and expertise. Clear deadlines are set for each task to maintain momentum and accountability.

- ➢ Our scrum master diligently updates the Trello board to reflect task progress, ensuring transparency and real-time visibility into project developments. This centralized platform serves as a repository for tasks, milestones, and updates, facilitating effective coordination and communication among team members.

- ➢ Our proactive approach to scheduling and project management minimizes schedule risks and enhances the overall efficiency and effectiveness of our project execution.

- **Teamwork risks** (any issues related to teamwork)
  - ❖ Everybody is in the meeting regularly?

  - ➢ All team members demonstrate a high level of commitment by consistently attending all scheduled meetings. During these gatherings, each member actively participates by sharing updates on their assigned tasks, challenges encountered, and progress made. This regular exchange of information fosters transparency, accountability, and alignment among team members.

  - ❖ Everybody keeps his/her pace? If not, what is your plan to mitigate the risks?

- ➢ Our team excels in maintaining individual pace while ensuring collective progress toward project goals. Every member takes ownership of their responsibilities and strives to meet deadlines effectively. In instances where a team member encounters difficulty or falls behind schedule, the team rallies together to provide support and guidance. Our scrum master plays a pivotal role in monitoring individual progress and facilitating discussions to address any roadblocks encountered.

- ➢ Furthermore, our team culture prioritizes camaraderie and well-being. Our scrum master goes the extra mile by bringing snacks to each meeting, creating a relaxed and enjoyable atmosphere conducive to productive collaboration. This thoughtful gesture fosters team cohesion and alleviates stress, allowing us to work harmoniously towards achieving our objectives.

- **Legal/content risks**
  - ❖ Can you obtain content/SW you need legally with proper licensing, copyright)?

  - ➢ Our project adopts a strategic approach by predominantly leveraging open-source software and freely available content. By utilizing resources that are openly accessible and distributed under permissive licenses, we minimize the risk of encountering copyright infringement or licensing issues. Open-source software provides us with the flexibility, transparency, and legal certainty necessary to develop and deploy our project without constraints.

  - ➢ Furthermore, our adherence to open-source principles aligns with our commitment to fostering collaboration, innovation, and community-driven development. By embracing open-source methodologies, we contribute to a culture of sharing knowledge, code, and resources for the benefit of the broader software engineering community.

  - ➢ Our deliberate choice to prioritize open-source software and free content mitigates legal and content-related risks, ensuring compliance with licensing requirements while fostering a culture of openness, collaboration, and innovation.

Appendix II– Rubrics and checklist for Part 2 Milestone 3 review: SW Review (to be done off-line by TA after Part 1 review)

Section: **Team:** **Date:**

Instructor/TA to Check and comment below:

- Git/Github organization (e.g. organization of branches)
    - To setup Dev branch and Feature branches are strongly recommended.
    - Grading check points : Dev and feature branches are properly setup and used.

- Git/Gith, git hub usage: code review practices (to see if the review comments are proper and enough)
    - Grading check points :
        - how many code reviews are being done since M3 announcement
        - how meaningful code reviews are. (ex. “look good” does not have much meaning)

- Frameworks (back end front end) deployed correctly