1. UI and Functionality Feedback and Responses

- Use postman to debug network calls
    - For this issue, we will do as instructed and use postman to have an easier time debugging the connection between our server and our code
- Fix “refusing frame” of images on develop
    - This issue will be fixed hopefully by switching to an s3 bucket in AWS for storing images instead of google drive.
- Builder tool display of each subject-type list working
    - We have set up more rigorous completion plan to keep this goal on track
- Favorites function of builder tool
    - Same as above comment
- Progress of info page and builder tool UI elements
    - These are being tracked now by our more available member so that they can make it look nice and functional

As instructed, next week and two weeks after that, we will present our progress to make sure we are on track.

2. P1 Features Committed for Delivery

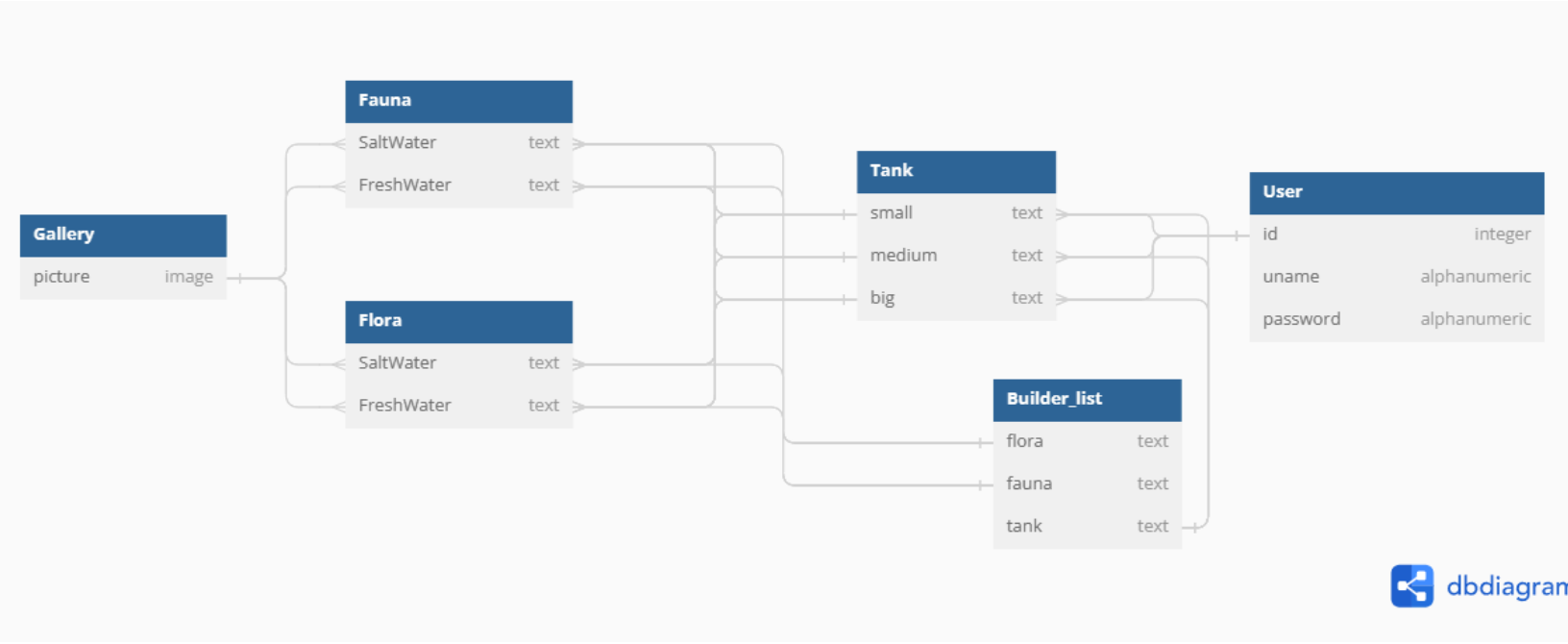
Our planned P1 features to be implemented before the end of M3 are as follows:

1. Search function working properly and fully
2. Gallery displaying all pictures
    a. Pictures clickable go to info page of item
    b. Pictures have name and scientific name
    c. Pictures have button to add to favorites
    d. Pictures have button to add to build
3. Builder tool working at lowest capacity
    a. Tabs for different types (plant, fish, tank, favorite)
    b. Search bars for each type (like mini galleries)
    c. List showing all chosen elements
    d. Compatibility function to disallow selection of incompatible livestock
4. Information pages for each item in the db that show information on the subject

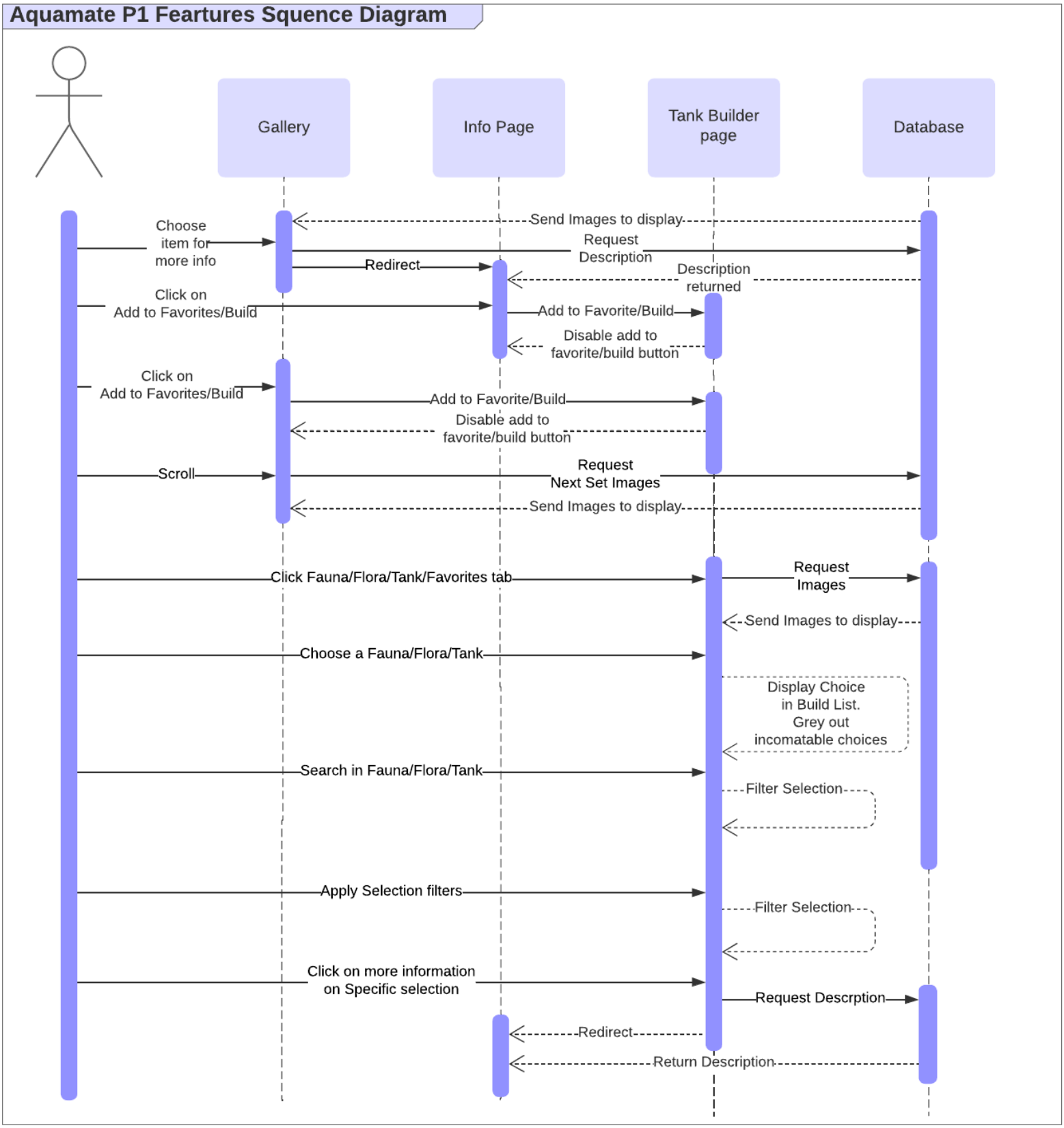
Many of the sub items on the list also count as P1 functions. After these items, there will be very few features to add and will be mostly css and testing/QA

3. Architecture

Class Diagram:



Sequence Diagram:



4. Project Status/Risks

Identification of possible Risk for this web app

Skills Risks

Risk: Specific knowledge gaps may exist, despite full attendance and training sessions.

Mitigation Plan: Offer targeted training during Sunday sessions to address known gaps. Encourage team members to take on tasks related to these new skills under supervision to solidify learning.

Schedule Risks

Risk: Adjusting the project timeline for training sessions may affect overall delivery deadlines.

Mitigation Plan: Integrate training into the project schedule without impacting critical tasks. Use Asana to monitor task dependencies and adjust due dates proactively.

Teamwork Risks

Risk: Despite 100% attendance, there may be risks of unequal contribution levels.

Mitigation Plan: Continuously assess individual contributions and adjust task allocations to ensure a balanced workload. Encourage peer support and pair less experienced members with those more knowledgeable.

Legal/Content Risks

Risk: Proper citation and legal use of data and images need to be consistently managed.

Mitigation Plan: Develop a standard operating procedure for using and citing external content. Assign a team member to oversee compliance with this procedure, and regularly review Asana tasks for adherence.

Updated information for Milestone 3

1. **Enhance Documentation:** As you resolve technical challenges like the NGINX-backend integration, ensure that these solutions are well-documented within Asana or a dedicated knowledge base. This can speed up future troubleshooting and serve as a learning tool for new team members.

2. **Skill Development Tracking:** Implement a system within Asana for tracking skill development and knowledge gaps. After each Sunday session, document what was learned and by whom. This can help in planning future training sessions more effectively and ensuring that all team members are progressing.
3. **Pre-Meeting Preparation:** To make the most out of your Sunday meetings, consider having team members submit questions or topics of interest in advance. This allows for a more structured meeting where time can be allocated efficiently to different issues, including training on specific parts of the website.

4. **Task Prioritization and Dependency Management:** Continue to refine how tasks are prioritized and dependencies managed within Asana. This may involve more detailed planning of which tasks can be parallelized and which must be completed sequentially. It could also help in better adjusting the project timeline when new training needs arise.

5. **Technical Risk Analysis:** For ongoing technical challenges, such as integration issues, perform a risk analysis to identify potential future problems. This can include looking at compatibility issues, potential bottlenecks, or security vulnerabilities. Planning for these in advance can reduce downtime and streamline development.

6. **Feedback Mechanism:** Implement a structured feedback mechanism for after each Sunday session. This can help in understanding what aspects of the training are working well and what needs improvement. It can also gauge the team's overall morale and identify any undercurrents of dissatisfaction early.

7. **Cross-training Strategy:** Develop a strategy for cross-training team members in different aspects of the project. This not only helps in reducing the risk associated with skill gaps but also improves team cohesion and understanding of the project as a whole.

8. **Compliance and Legal Review:** Given the importance of legal compliance, especially concerning content and data use, schedule regular reviews of your standard operating procedures. These reviews can be part of your Sunday sessions or a separate meeting, focusing on ensuring that all aspects of the project are compliant with relevant laws and regulations.