SW Engineering CSC 648/848 Spring 2024

Section 02

Team 05

Team Members	
Team Lead	Johnny Kwon
Database Engineer	Abby Lin
Frontend Engineer	Zabiullah Niemati
Frontend Engineer	Zizo Ezzat
Github Master/Frontend	Ethan Ho
Backend Engineer	Nichan Lama
Backend Engineer	Fadee Ghiragosian

Team Lead email: hkwon4@sfsu.edu

Github Page:
https://github.com/CSC-648-SFSU/csc-648-02-spring24-team05

Document History	
Date Submitted	3/13/2024
Date Revised	

Functional Requirements - Prioritized

1. **Users (Students and Parents)**

(Reference: 1: must have, 2: desired, 3: opportunistic)

1.1. Users shall be able to register for an account using an email address and a secure password.

1.1.1. Priority Level: 1

1.2. Users shall be able to maintain personal, academic, and athletic profiles, which include contact information, academic history, athletic records, and personal statements.

1.2.1. Priority Level: 2

1.3. Users shall be able to set preferences for university program notifications and communications.

1.3.1. Priority Level: 2

1.4. Users shall have access to recommended universities and programs based on their profiles and preferences.

1.4.1. Priority Level: 3

1.5. Users shall have access to a resource center with articles and guidelines on college athletics and recruiting processes.

1.5.1. Priority Level: 3

1.6. Users shall be able to participate in discussions and chats hosted by various universities in the app.

1.6.1. Priority Level: 3

1.7. Users shall be able to participate in discussions in the app.

1.7.1. Priority Level: 3

1.8. Users shall be able to contact faculty.

1.8.1. Priority Level: 2

1.9. Users shall be able to book a schedule from faculty for consultation.

1.9.1. Priority Level: 3

1.10. Users shall be tracked by faculty members for their progress and updates

1.10.1. Priority Level: 3

1.11. Users shall be able to receive invitations from universities.

1.11.1. Priority Level: 2

2. University Admin

2.1. University Admin shall be able to post and update profiles.

2.1.1. Priority Level: 1

2.2. University Admin shall have access to analytics tools to monitor engagement and interest from users.

2.2.1. Priority Level: 1

2.3. University Admin shall be able to send personalized invitations to prospective students for recruitment purposes.

2.3.1. Priority Level: 2

2.4. University Admin shall have the ability to create and manage content such as news and program highlights.

2.4.1. Priority Level: 1

2.5. University Admin shall be provided with a reporting feature to generate reports on user engagement, application numbers, and platform usage statistics.

2.5.1. Priority Level: 1

2.6. University Admin shall be able to manage and assign tasks to faculty members through the platform.

2.6.1. Priority Level: 1

2.7. University Admin shall be able to post articles and guidelines in a resource center for users to access.

2.7.1. Priority Level: 3

3. Faculty (Including Coaches and Advisors)

3.1. Faculty shall be able to create and publish their schedules for availability to students for consultations or scouting.

3.1.1. Priority Level: 2

3.2. Faculty shall have the ability to track and monitor the progress and updates of athletes they are interested in.

3.2.1. Priority Level: 3

3.3. Faculty shall be able to initiate contact with users for recruiting purposes with the option to offer positions.

3.3.1. Priority Level: 1

3.4. Faculty shall be able to collaborate with other faculty members within the platform to strategize on recruitment and program development.

3.4.1. Priority Level: 2

3.5. Faculty shall have the ability to endorse athletes on their profiles, providing credibility and visibility to the users.

3.5.1. Priority Level: 3

3.6. Faculty shall be able to provide feedback on the platform to improve its functionality and user experience.

3.6.1. Priority Level: 2

3.7. Faculty shall be able to complete tasks assigned by the university admin.

3.7.1. Priority Level: 2

3.8. Faculty shall be able to post and update profiles.

3.8.1. Priority Level: 1

4. Communication and Engagement Tools

4.1. Communication tools shall support multimedia sharing, including images and videos, to aid in the recruitment process.

4.1.1. Priority Level: 2

4.2. The platform shall redirect users to schedule appointments or interviews with university admin or faculty.

4.2.1. Priority Level: 2

5. Search and Filter Functionality

5.1. The platform shall offer a recommendation system to suggest universities and programs to users based on their activities and interests.

5.1.1. Priority Level: 3

5.2. All users shall be able to filter searches based on specific sporting events and relevant athletic accomplishments.

5.2.1. Priority Level: 1

5.3. Searches can be done both through keywords and user manual input.

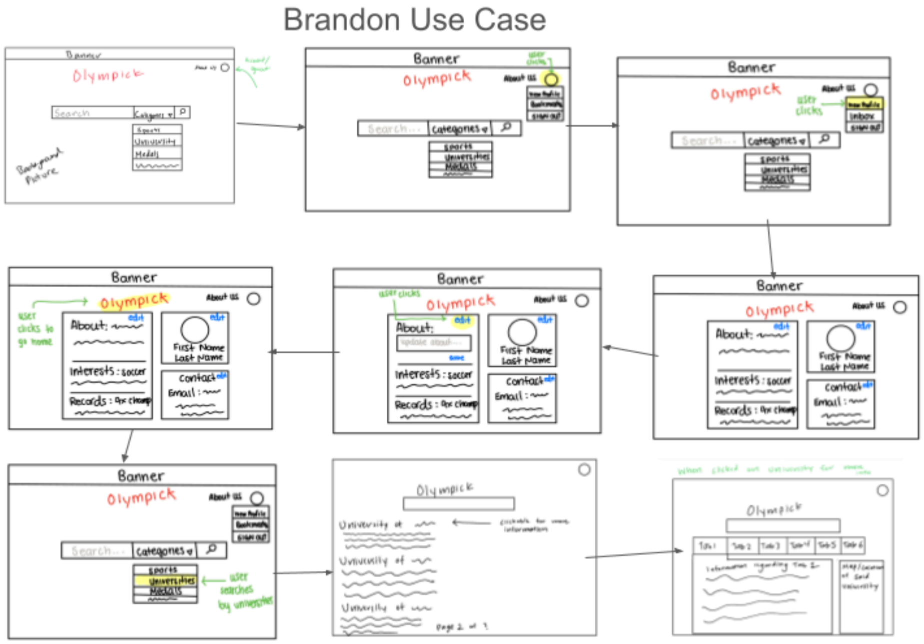
5.3.1. Priority Level: 1

5.4. Filters shall include the ability to sort by academic and athletic scholarships offered, acceptance rates, and specific sports ranking within the university.

5.4.1. Priority Level: 3

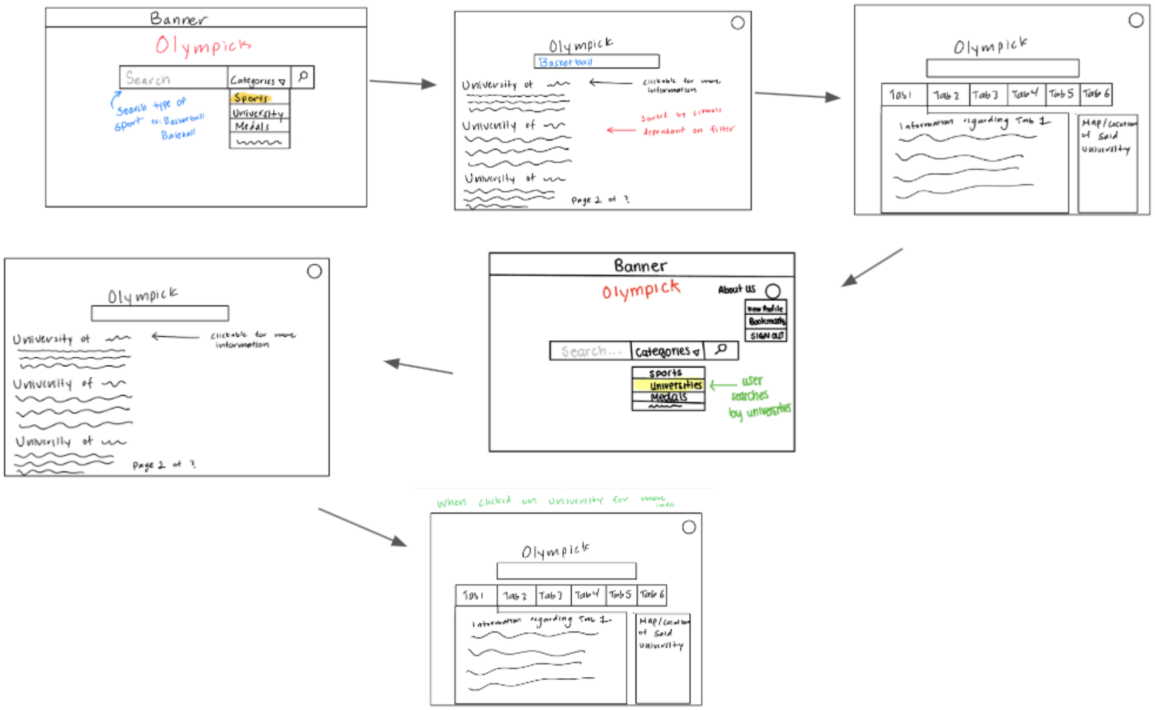
UI Mockups and Storyboards

Use Case: Brandon wants to join a university with a strong track and field team. In order to do this, he would need to create a profile where he can enter his current stats and then use a search feature and try to filter through the list of options. He would also need to be able to contact the coaches and universities to learn more about their offerings.

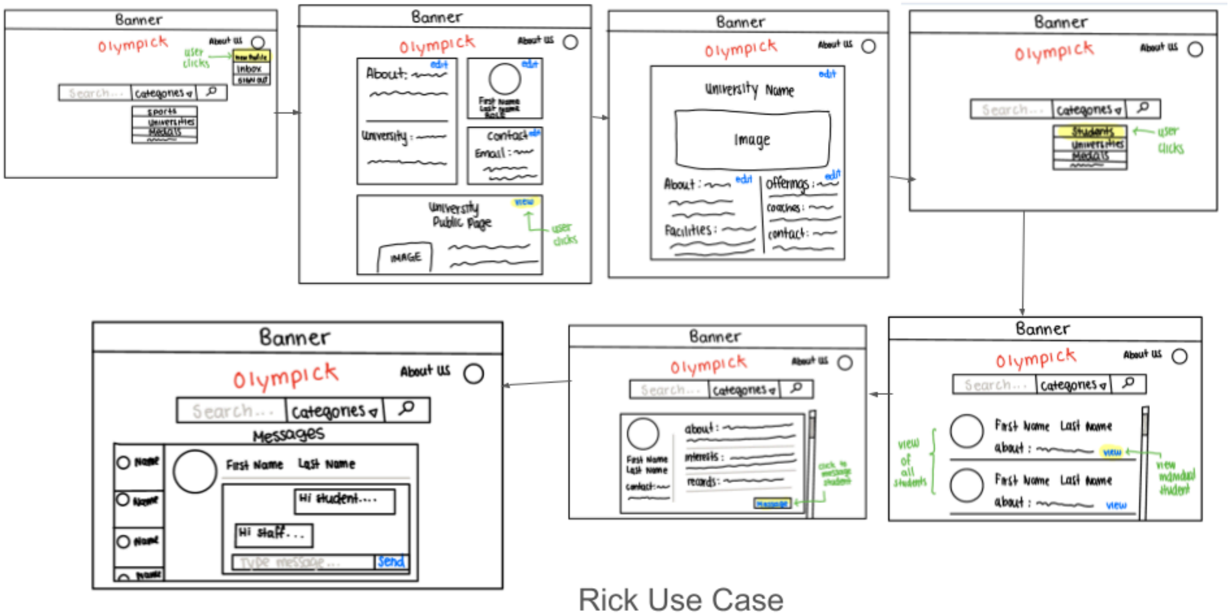

Brandon Use Case

Use Case: Joy wants to find the right athletic program for her. She would need to be able to search through what programs the school has to offer in order to find the right program using the search filters. She will also search what type of athletes won medals in the school to choose which program she will go into. To take it one step further, she could contact the alumni and seek advice.

Joy Parks Use Case



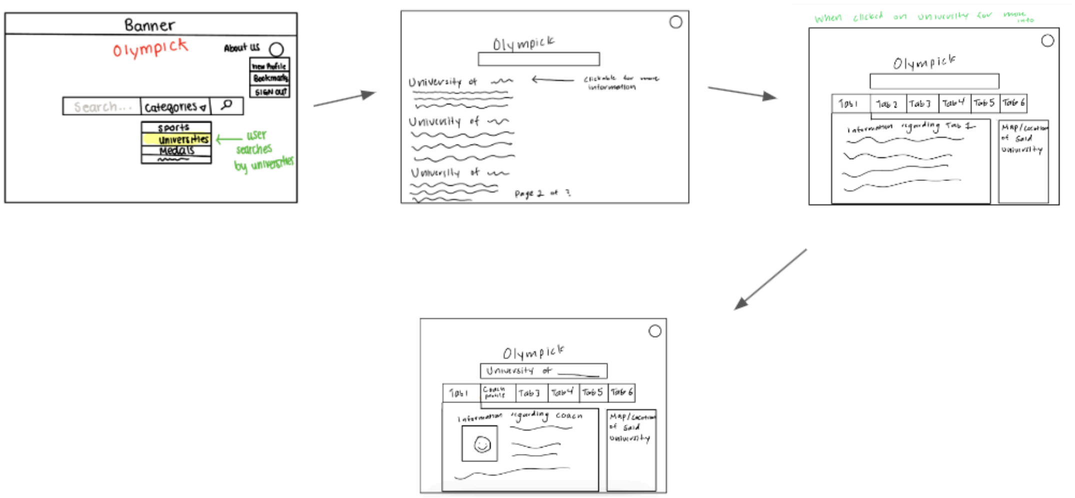
Use Case: Rick wants to build up his university's athletic programs with talented students. He would need to set up an account and public page for his university to highlight its offerings to potential students. He would also want to use this platform to search through students and proactively recruit options. Additionally, he would need to communicate with applicants through the app.



Rick Use Case

Use Case: Sarah is tasked with conducting the initial exploration of her child's journey. She needs to thoroughly examine comprehensive university profiles, programs, and coaching staff. To engage with university representatives and collect information about the program while facilitating areas for the athletes, she plans to utilize a dedicated app. Lastly, she will make an informed decision and initiate the application process.

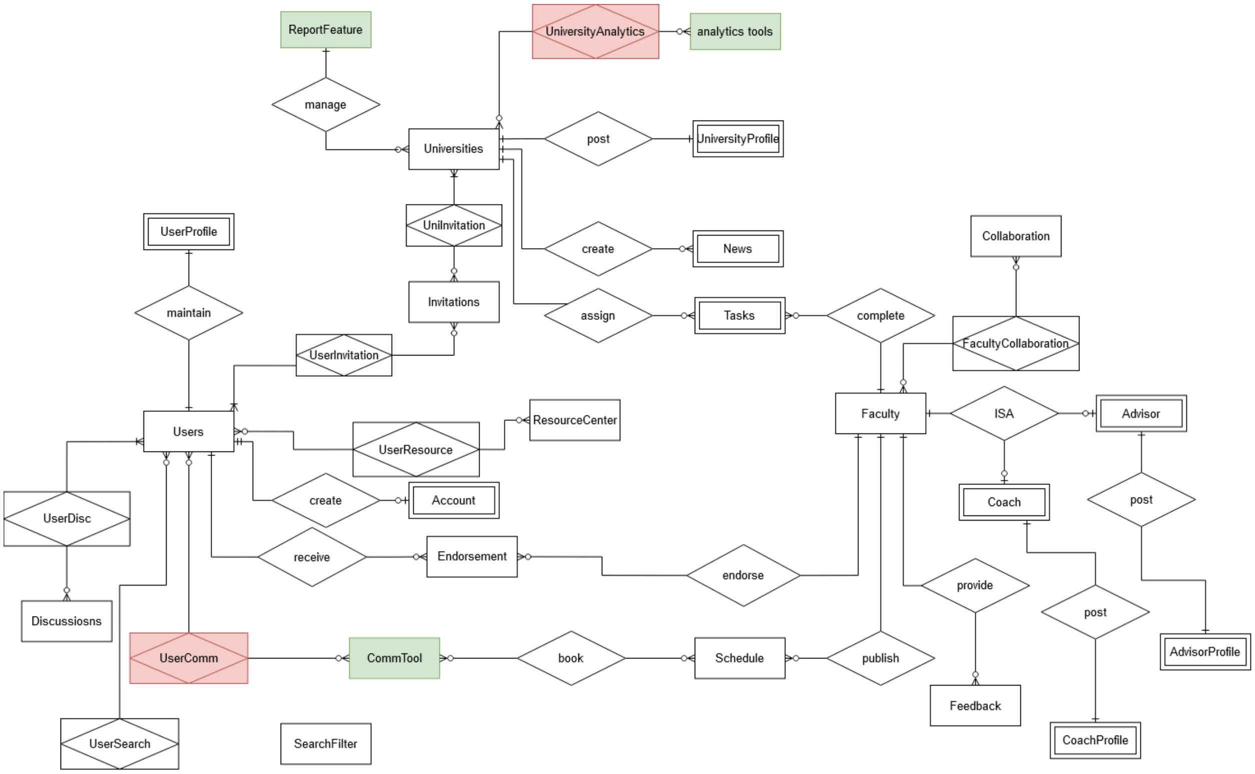
Sarah Williams Use Case



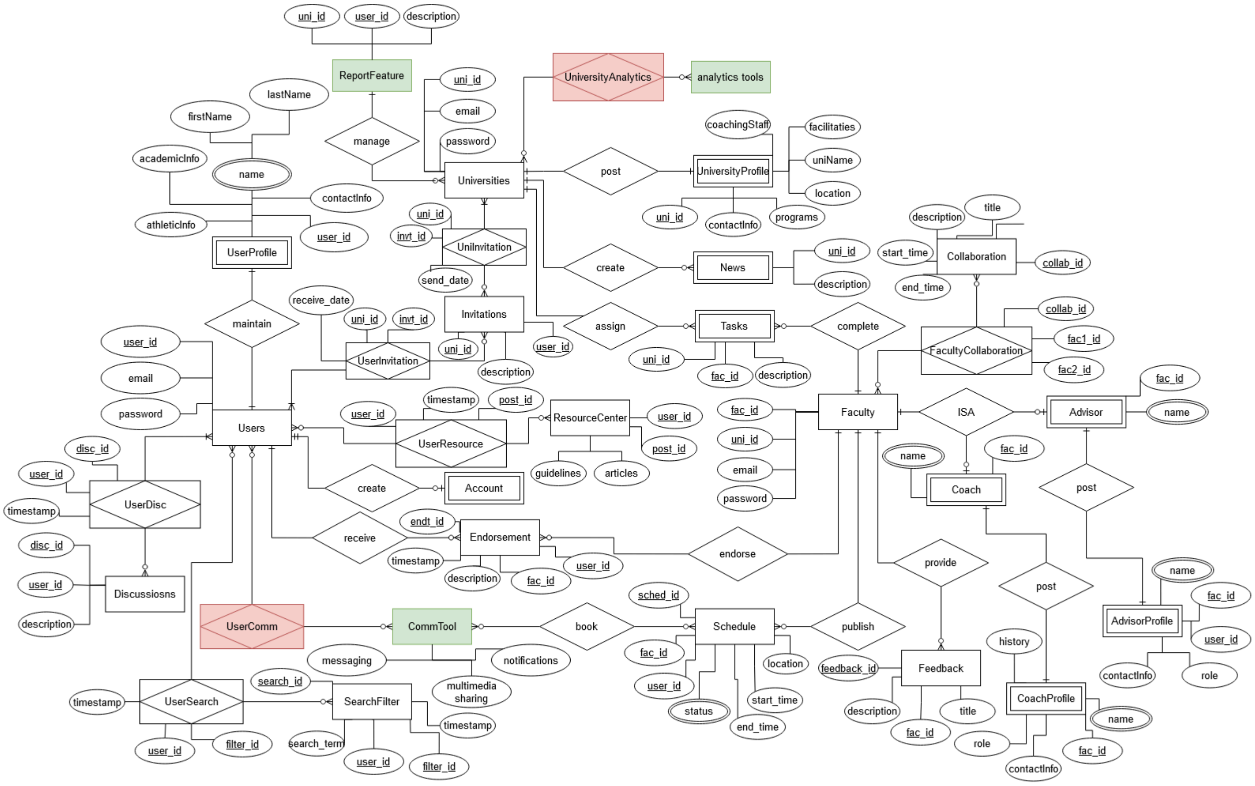
High-level Architecture, Database Organization

Entity Relationship Diagrams

Simplified General Overview



Detailed View (with all attributes)



List of Entities and their Attributes

1. Users (Students and Parents)

    1.1. user_id: integer, PK

    1.2. email: string

    1.3. password: string

2. User Profile

    2.1. name: composite, string

    2.2. firstName: string

    2.3. lastName: string

    2.4. user_id: integer, FK (reference: Users.user_id)

    2.5. contactInfo: string

2.6. athleticInfo: text

2.7. academicInfo: text

3. Communication Tools (concept design in progress)

3.1. messaging interface:

3.2. notifications: text

4. Resource Center

4.1. articles: text

4.2. guidelines: text

4.3. post_id: integer, PK

4.4. uni_id: integer, FK (reference: Universities.uni_id)

5. Discussions

5.1. description: text

5.2. disc_id: integer, PK

5.3. user_id: integer, FK (reference: Users.user_id)

6. Universities

6.1. uni_id: integer, PK

6.2. email: string

6.3. password: string

7. UniversityProfile

7.1. uniName: string

7.2. uni_id: integer, FK (reference: Universities.uni_id)

7.3. location: string

7.4. programs: text

7.5. contactInfo: string

7.6. coachingStaff: text

7.7. facilities: string (depends on length)

Analytics tools (concept in progress)

8. Invitations

8.1. invt_id: integer, PK

8.2. uni_id: integer, FK (reference: Universities.uni_id)

8.3. user_id: integer, FK (reference: Users.user_id)

8.4. description: text

9. News

9.1. uni_id: integer, FK (reference: Universities.uni_id)

9.2. description: text

10. ReportFeature (concept work in progress)

11. University Admin shall be provided with a reporting feature to generate reports on user engagement, application numbers, and platform usage statistics.

11.1. uni_id: integer, FK (reference: universities.uni_id)

11.2. user_id: integer, FK (reference: users.user_id)

11.3. description: text

12. Tasks

12.1. uni_id: integer, FK (reference: Universities.uni_id)

12.2. fac_id: integer, FK (reference: Faculty.fac_id)

12.3. description: text

13. Faculty

13.1. fac_id: integer, PK

13.2. uni_id: integer, FK (reference: Universities.uni_id)

13.3. email: string

13.4. password: string

14. Coach

14.1. fac_id: integer, PK, FK (reference: Faculty.fac_id)

14.2. name: composite, string

14.3. firstName: string

14.4. lastName: string

15. Advisor

15.1. fac_id: integer, PK, FK (reference: Faculty.fac_id)

15.2. name: composite, string

15.3. firstName: string

15.4. lastName: string

16. Schedule

16.1. sched_id: integer, PK

16.2. fac_id: integer, FK (reference: Faculty.fac_id)

16.3. user_id: integer, FK (reference: Users.user_id)

16.4. start_time: datetime

16.5. end_time: datetime

16.6. location: string

16.7. status: string (available, booked, canceled)

17. Collaboration

17.1. collab_id: integer, PK

17.2. title: string

17.3. description: text

17.4. start_time: datetime

17.5. end_time: datetime

18. Endorsement

18.1. endt_id: integer, PK

18.2. fac_id: integer, FK (reference: Faculty.fac_id)

18.3. user_id: integer, FK (reference: Users.user_id)

18.4. description: text

18.5. timestamp: datetime

19. Feedback

19.1. feedback_id: integer, PK

19.2. fac_id: integer, FK (reference: Faculty.fac_id)

19.3. title: string

19.4. timestamp: datetime

19.5. description: text

20. AdvisorProfile

20.1. fac_id: integer, PK, FK (reference: Faculty.fac_id)

20.2. name: string

20.3. role: string (advisor)

20.4. contactInfo: string

20.5. user_id: integer, FK (reference: Users.user_id) : list of students advised

21. CoachProfile

21.1. fac_id: integer, PK, FK (reference: Faculty.fac_id)

21.2. name: string

21.3. role: string (coach)

21.4. contactInfo: string

21.5. history: text

22. SearchFilter

22.1. filter_id: integer, PK

22.2. name: string

22.3. criteria: text / JSON

22.4. Associative Entities

23. UserDisc

23.1. user_id: integer, FK (reference: Users.user_id)

23.2. disc_id: integer, FK (reference: Discussions.disc_id)

23.3. timestamp: datetime

24. FacultyCollaboration

24.1. collab_id: integer, FK (reference: Collaboration.collab_id)

24.2. fac1_id: integer, FK (reference: Faculty.fac_id)

24.3. fac2_id: integer, FK (reference: Faculty.fac_id)

25. UserSearch

25.1. user_id: integer, FK (reference: Users.user_id)

25.2. filter_id: integer, FK (reference: SearchFilter.filter_id)

25.3. timestamp: datetime

26. UserResource

    26.1. user_id: integer, FK (reference: Users.user_id)

    26.2. post_id: integer, FK (reference: Resource Center.post_id)

    26.3. timestamp: datetime

27. UniInvitation

    27.1. uni_id: integer, FK (reference: Universities.uni_id)

    27.2. invt_: integer, FK (reference: Invitations.invt_id)

    27.3. send_date: datetime

28. UserInvitation

    28.1. user_id: integer, FK (reference: Users.user_id)

    28.2. invt_: integer, FK (reference: Invitations.invt_id)

    28.3. receive_date: datetime

Media Storage

Media such as pictures, video, and audio will be stored in the server's file system and will be pointed to by the database. File sizes will be limited to 5MB across all types due to limited server storage space.

Search/Filter Architecture and Implementation

Our database terms essential for search include universities, academic programs, faculty, coaches, sporting events, etc. ensuring a broad spectrum of searchable words.

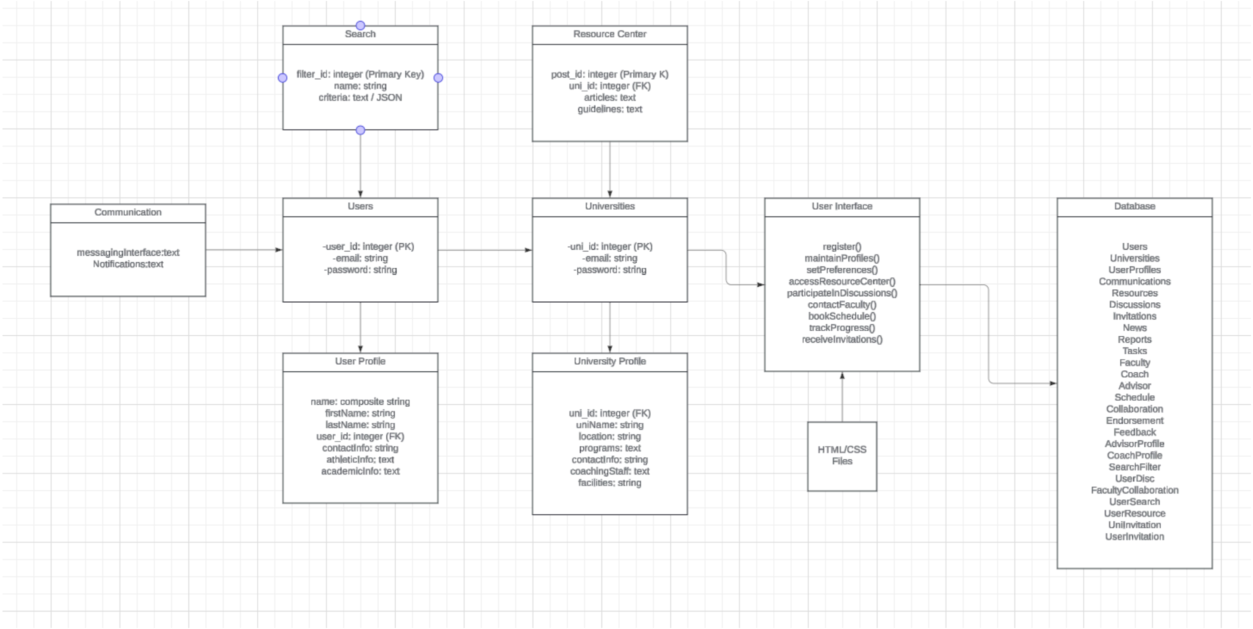
Search algorithms allow users to input keywords related to our Database terms primarily universities, students, etc. facilitating a comprehensive exploration of available data.

SQL queries are structured to accommodate this functionality, utilizing placeholders like ‘%keyword%’ where a keyword is the search term.

In addition to keyword-based searches, filtering SQL queries enhance the user experience by providing access to data directly relevant to their needs, fostering a more efficient and targeted search process.

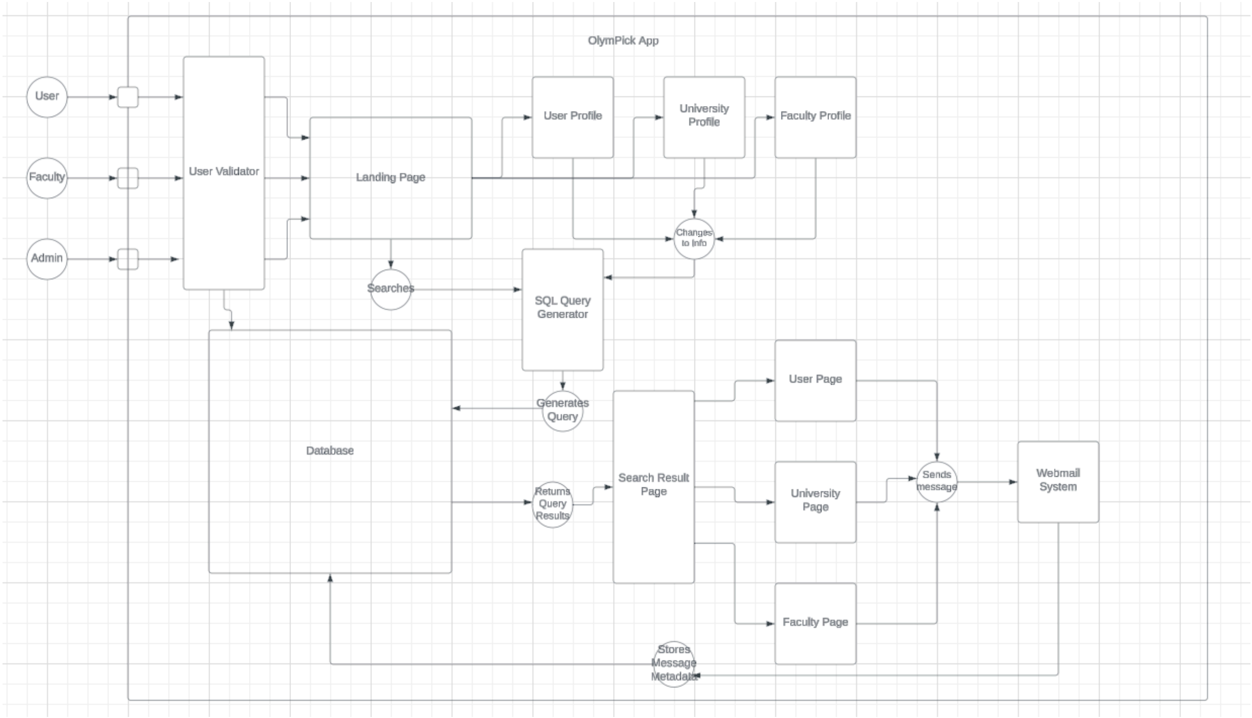
High-level UML Diagrams

High Level Class Diagram



Link:
https://lucid.app/lucidchart/7ee5975a-3328-47c8-8be5-9d020782b67a/edit?viewport_loc=-376%2C-366%2C3246%2C1602%2CHWEp-vi-RSFO&invitationId=inv_62a4ffc1-957b-4599-9672-bf424fb87934

Component Diagram



Key Risks

1. Skills Risks
    1.1. All members are using Flask backend development for the first time
        1.1.1. Resolution: We have all shared collective resources and are looking for new resources to learn the proper implementation of Flask
    1.2. All members are new to using NGINX web framework
        1.2.1. Resolution: Assigned tutorials and readings are given to all backend developers

2. Schedule Risks

   2.1. Other classwork and work schedules are affecting collaborative working opportunities

      2.1.1. Resolution: Prioritize collaborative tasks and plan ahead to allocate specific times for working together.

3. Technical Risks

   3.1. Compatibility issues, software bugs, or dependency conflicts can cause application failures or security vulnerabilities.

      3.1.1. Resolution: Performing regular testing and staging environments to validate software changes before deployment and maintaining documentation and version control for software configurations and dependencies.

4. Teamwork Risks

   4.1. Lack of experience with working with groups on large-scale projects

   4.1.1 Resolution: Implement paired programming partners to frontend/backend to share and learn faster.

5. Legal/Content Risks

   5.1. Some records and information, despite being public, may require additional permissions or crediting.

      5.1.1. Resolution: Conduct a thorough review of all content, records, and information used and identify any content that requires permissions.

Project Management

Our project will be managed with the use of the Jira software by Atlassian. Using the scrum project management tool, we will be able to define our tasks and their associative timelines for each team member.