**Academic Pathway Optimization**

Goodfellas

IS436 - 03

Deliverable 1

February 20th, 2019

**Project Members**

Paul Lyon

Tanner Yatsko

Jas Singh

Zach Jaquet

Benjamin Wang

Densu Kurian

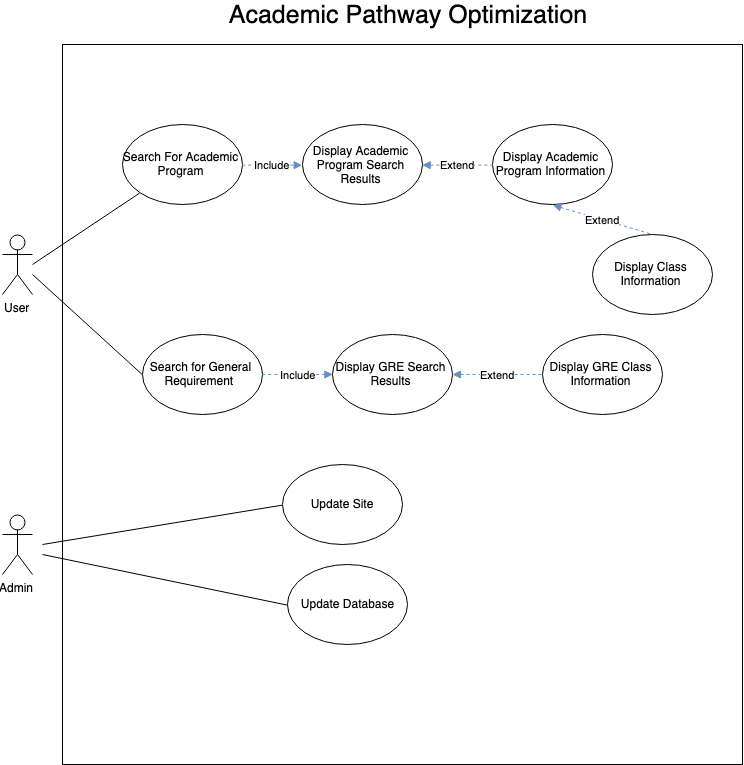
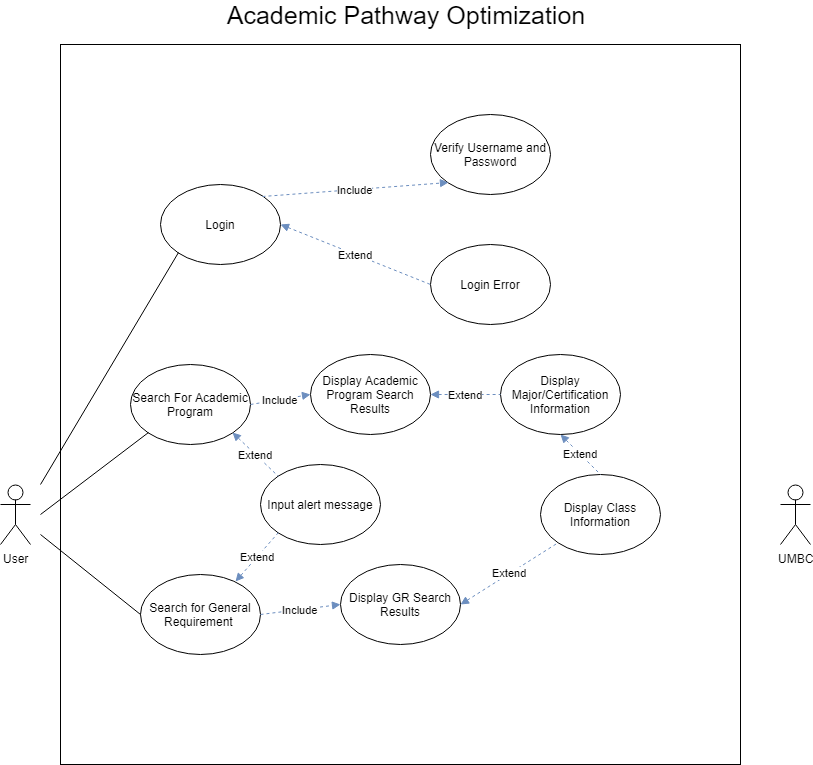
William Chanmugam

Niels Verhoeven

**Use Case Updates**

**Use Case Diagram Update:**

After careful analysis of our prior use case diagram, we found that the “UMBC” entity did not have any function and therefore should be removed. Instead, we added the entity “administrator” who has the power to modify the website and database. Next, we removed the login function because we realized that it would limit access for people who did not liking making accounts. Finally, we removed the alert messages extension because all the fields that the user inputs are from drop down menus, therefore the user can’t possibly enter a false entry. Here are the two compared back to back with the updated on the right. Go to this [link](https://github.com/FearTheLion/IS436GoodFellas/blob/master/History/Deliverable3/New%20User%20Diagram%20Comparison.png) for a better view.



**Use Case Document Update:**

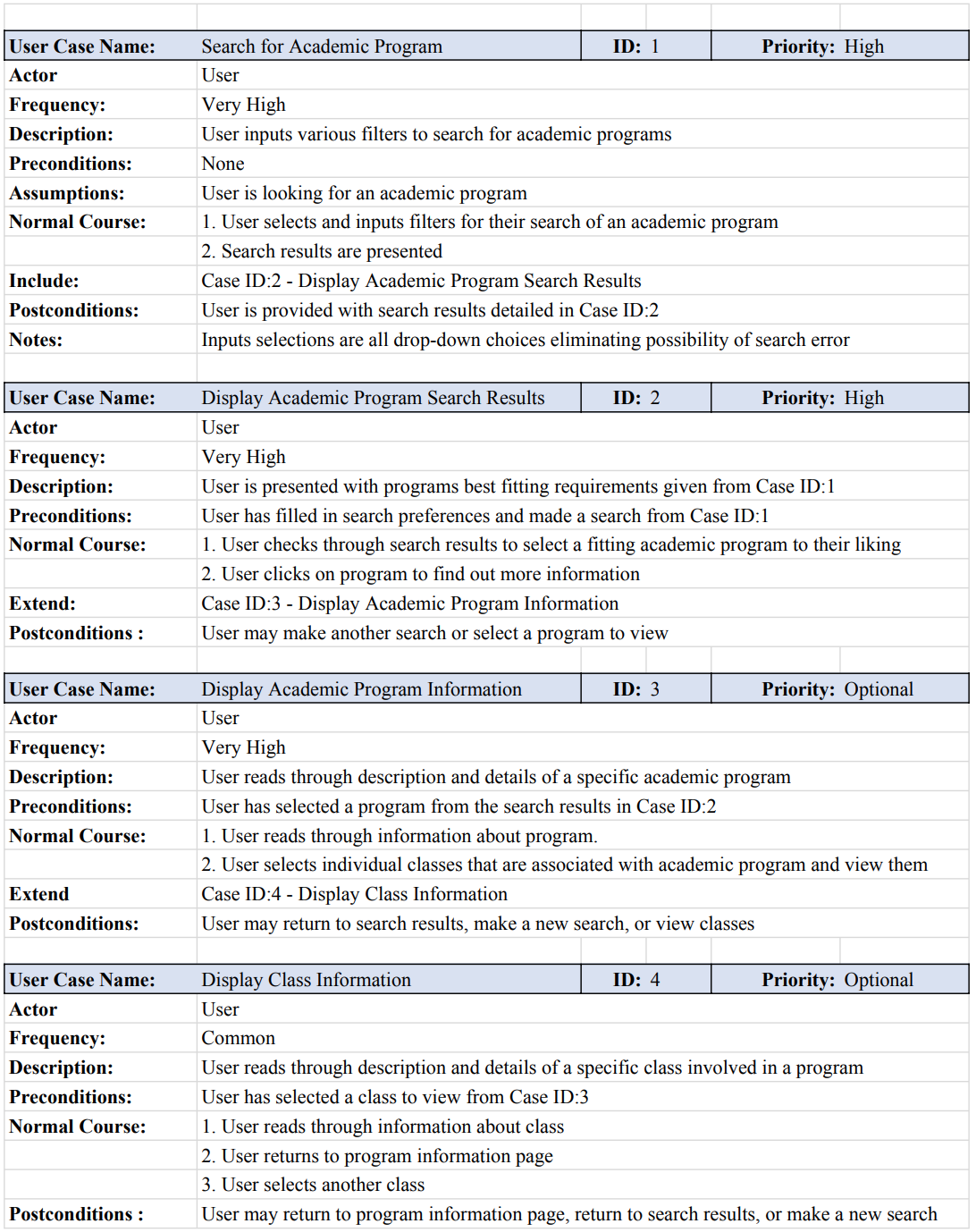
With a new use case diagram, we instated new use case documents to align with the updated system. The use case documents that we kept from deliverable 2 include:

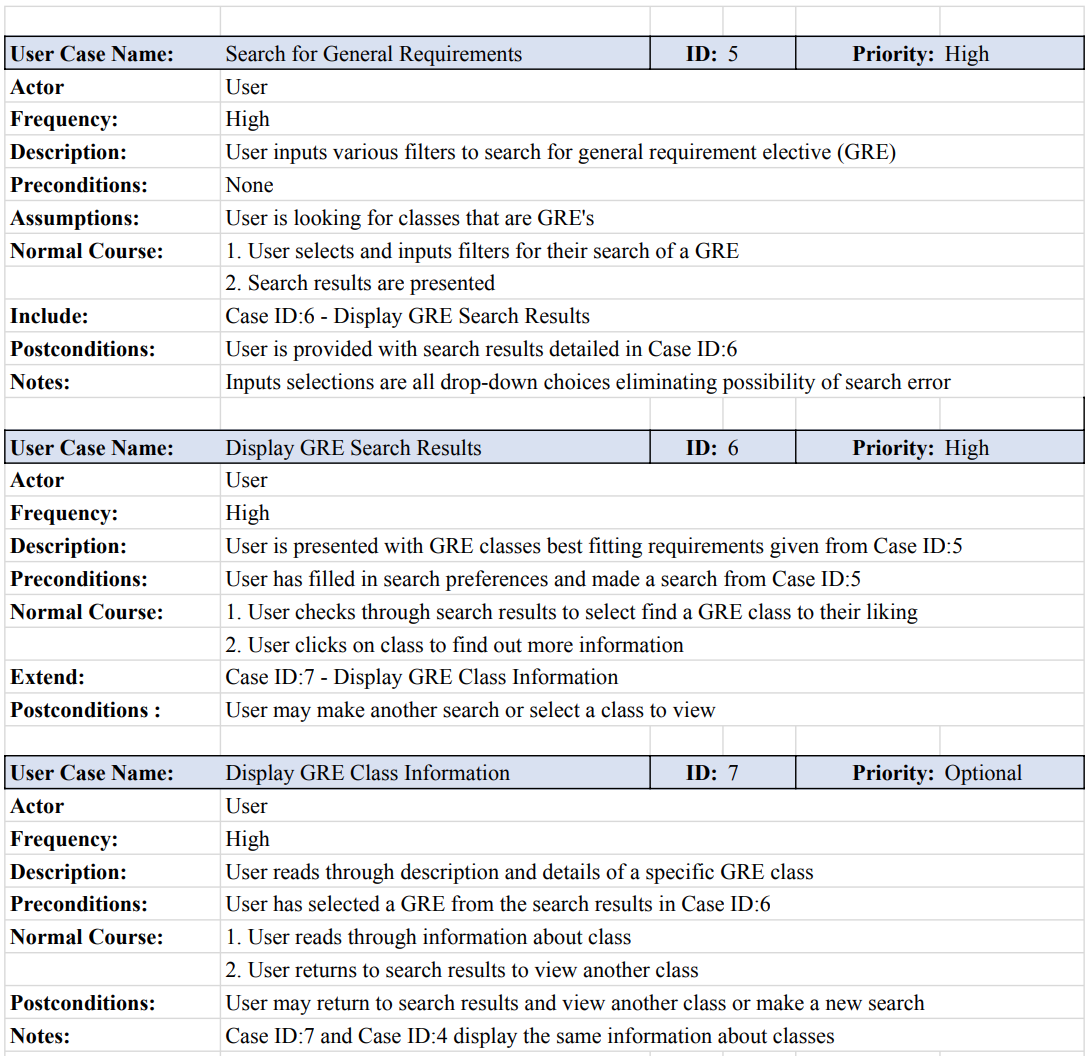
* Search for Academic Program
* Display Academic Program Search Results
* Display Academic Program Information
* Display Class Information
* Search for General Requirement
* Display GRE Search Results

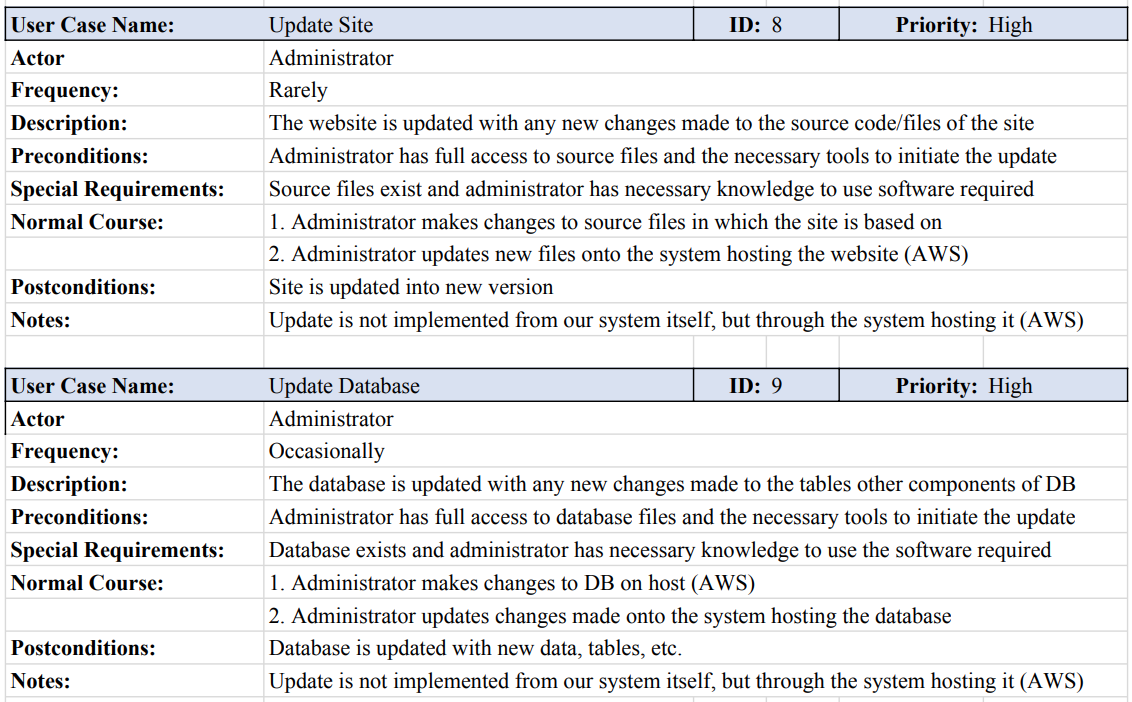
Display Class Information was divided into two use cases because even though they both display class information; they are accessed through different means. Since we no longer implement login, we decided instead to have an administrator who has direct access to the database and files outside of the user system. This added three use cases including:

* Display GRE Class Information
* Update Site
* Update Database

The new and retained use case documents are presented on the next page.

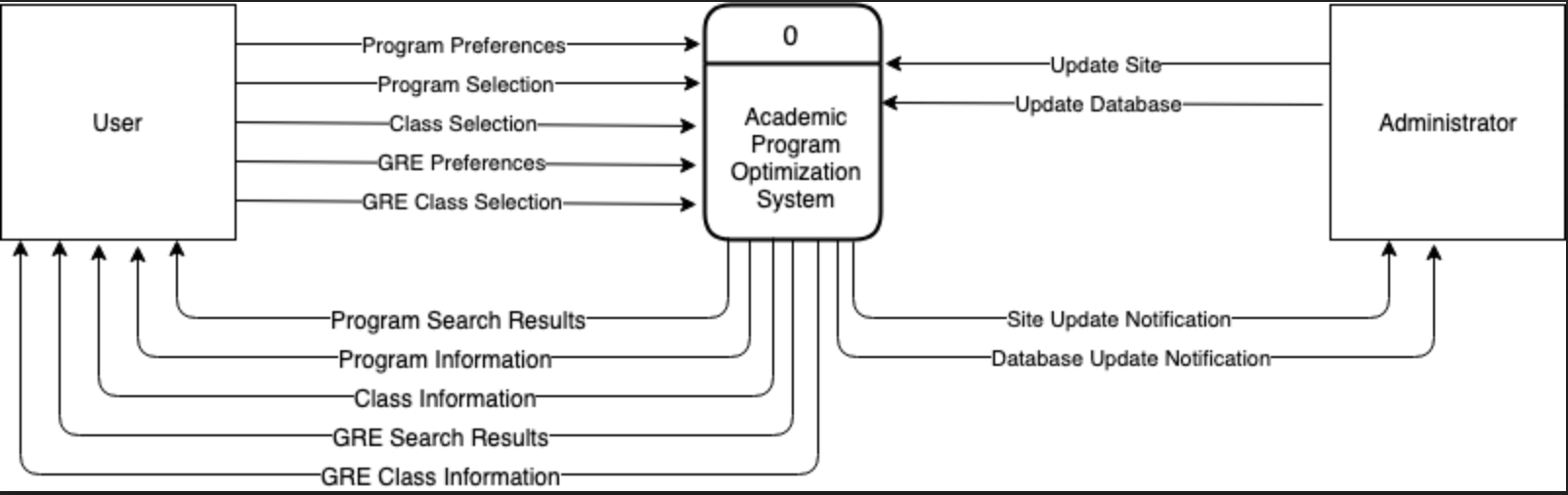


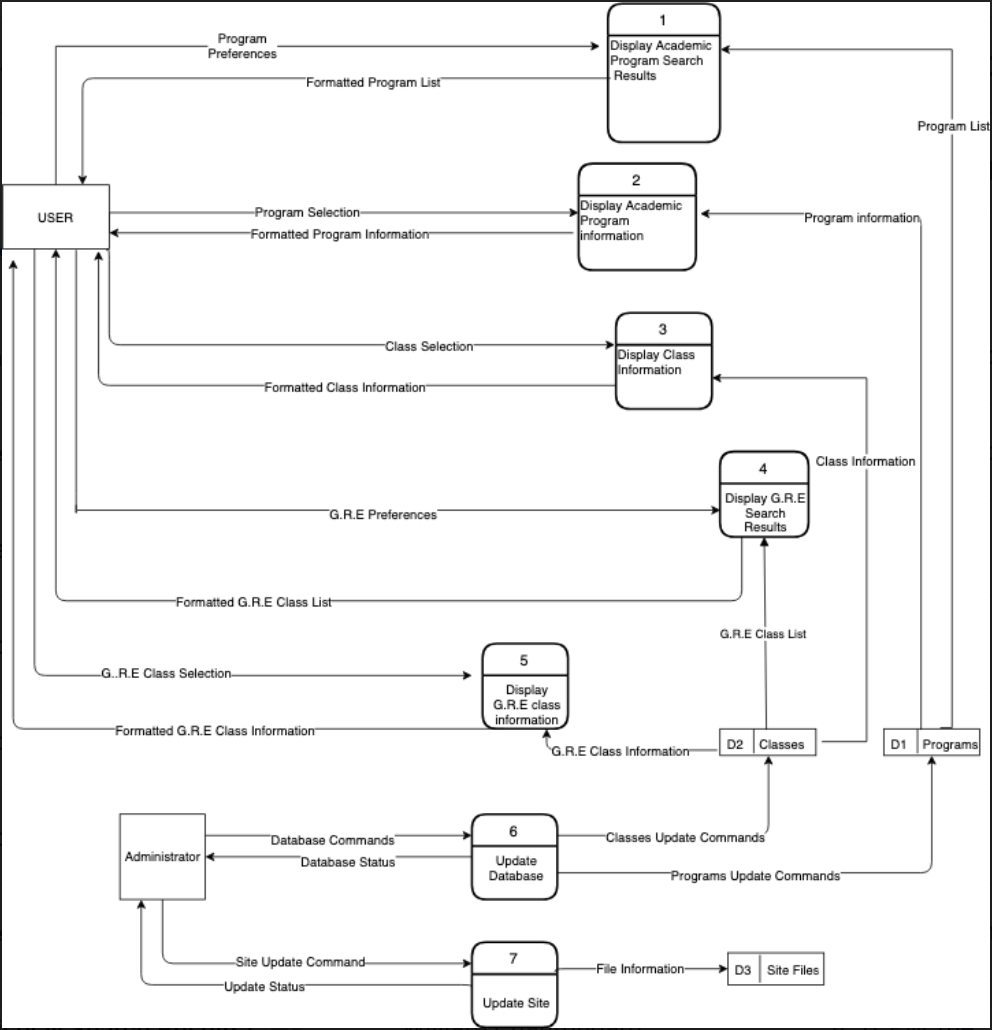




**Data Flow Diagram**

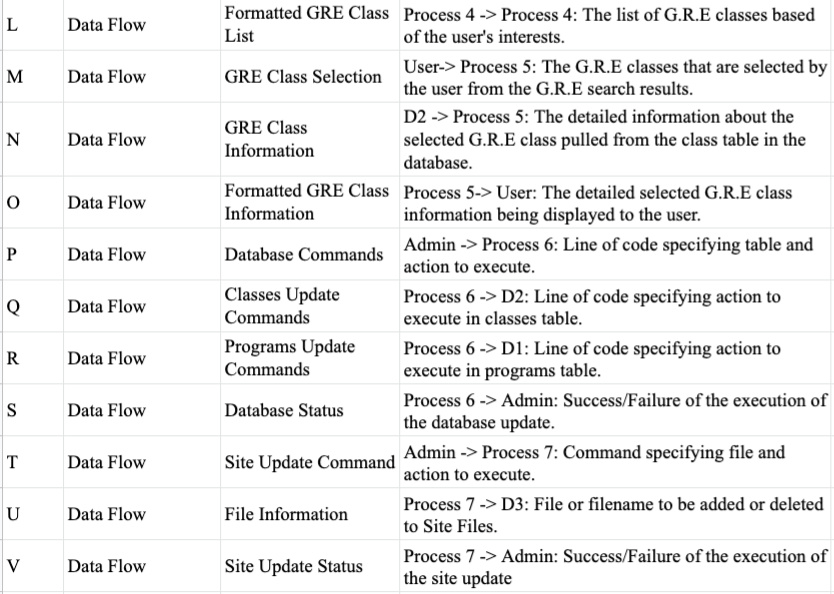
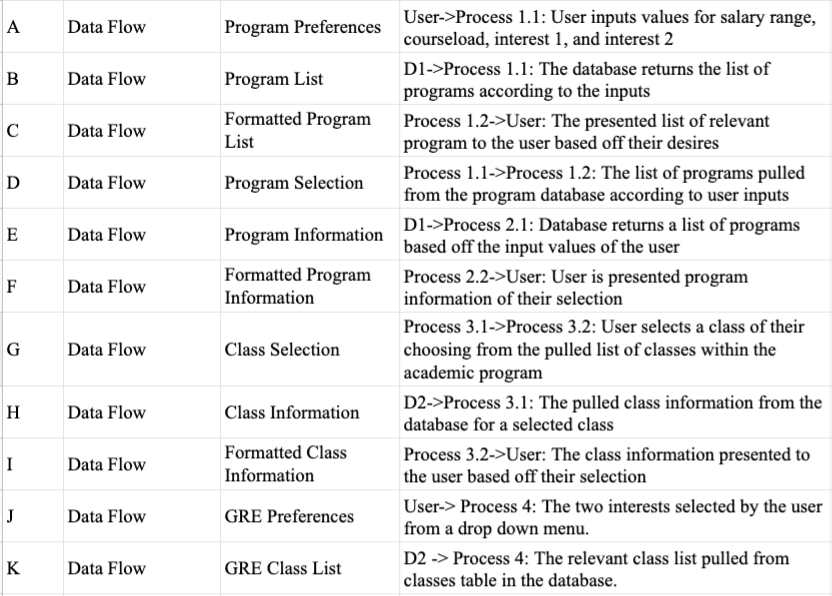
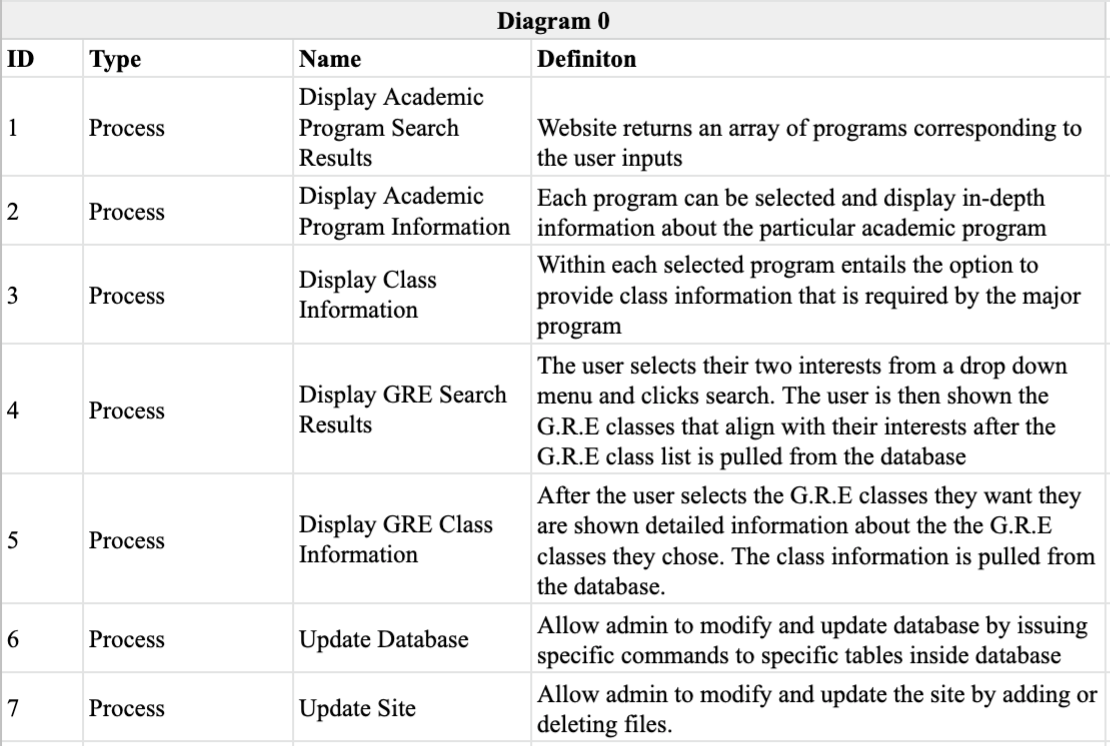
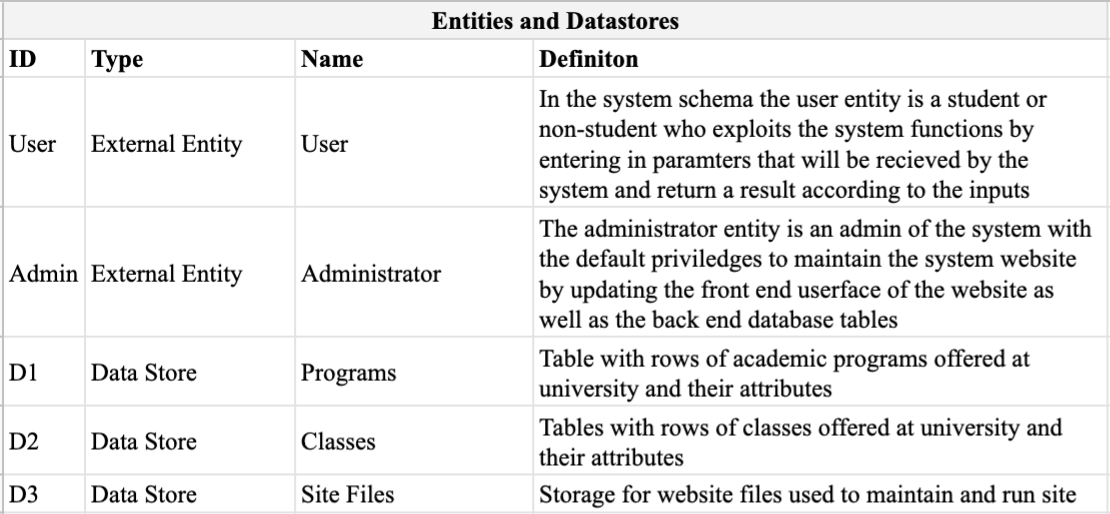
**Context Diagram:** Below is the basic context diagram in which we based our diagram 0 off of. The main entities include the user, administrator, and Academic Program Optimization System.



**Diagram 0:** Below is the diagram 0. It is important to note that since the same data that passes into the process “Search for Academic Program” also passes out, it is not necessary to place it in the diagram 0. Same goes for the process “Search for General Requirement”. 

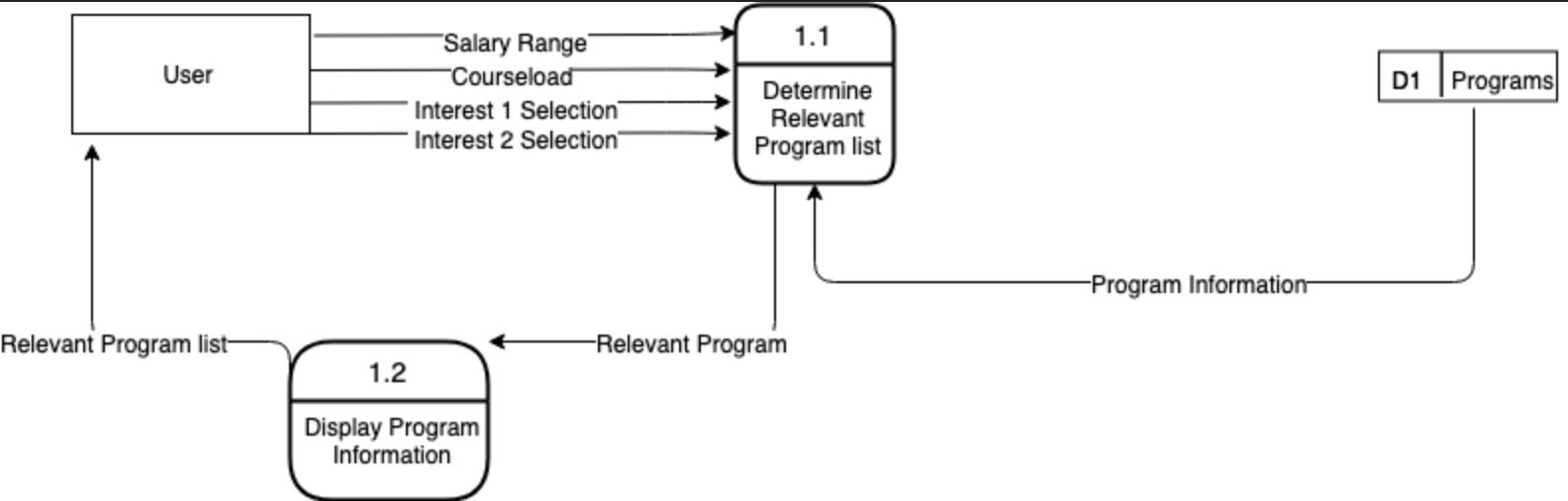
**Data Flow Diagram Definitions for Diagram 0:**

Below are images of the definitions for diagram 0. Due to the mass amount of data, please reference the “[DiagramDefinitions](https://github.com/FearTheLion/IS436GoodFellas/blob/master/History/Deliverable3/DiagramDefinitions.xlsx)” file on our GitHub repository for the Level 1 and 2 diagram definitions.

****

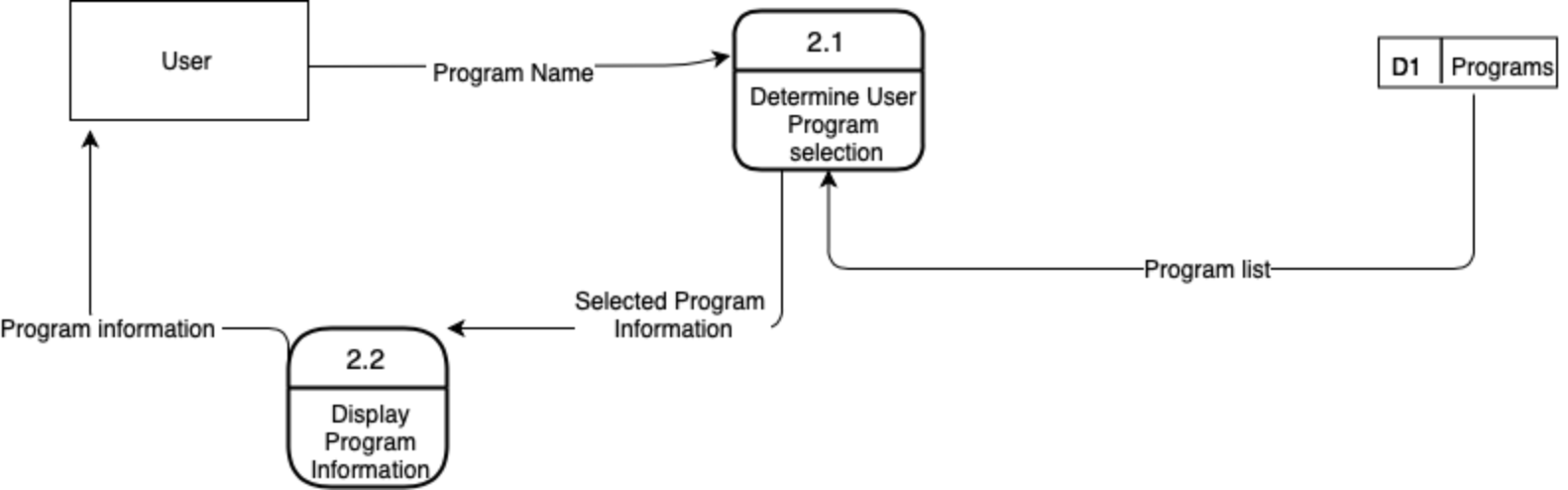
**Diagram Level 1: Display Academic Program Search Result**

The four inputs provided from the user are used as conditions to search for the closest matching academic programs from the programs table. This list of programs is then provided back to the user in a format the user can read.



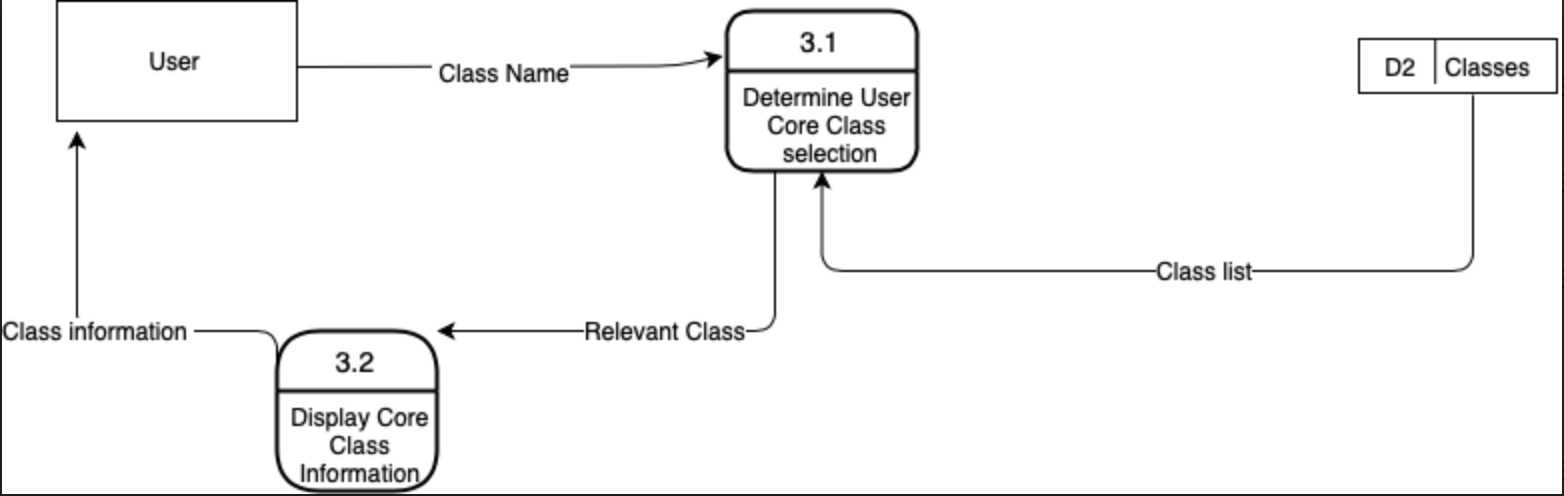
**Diagram Level 1: Display Program Information**

The user selects one of the programs returned from him in the previous diagram and the specific information about that program is fetched from the programs table and displayed to the user in a format they can read.



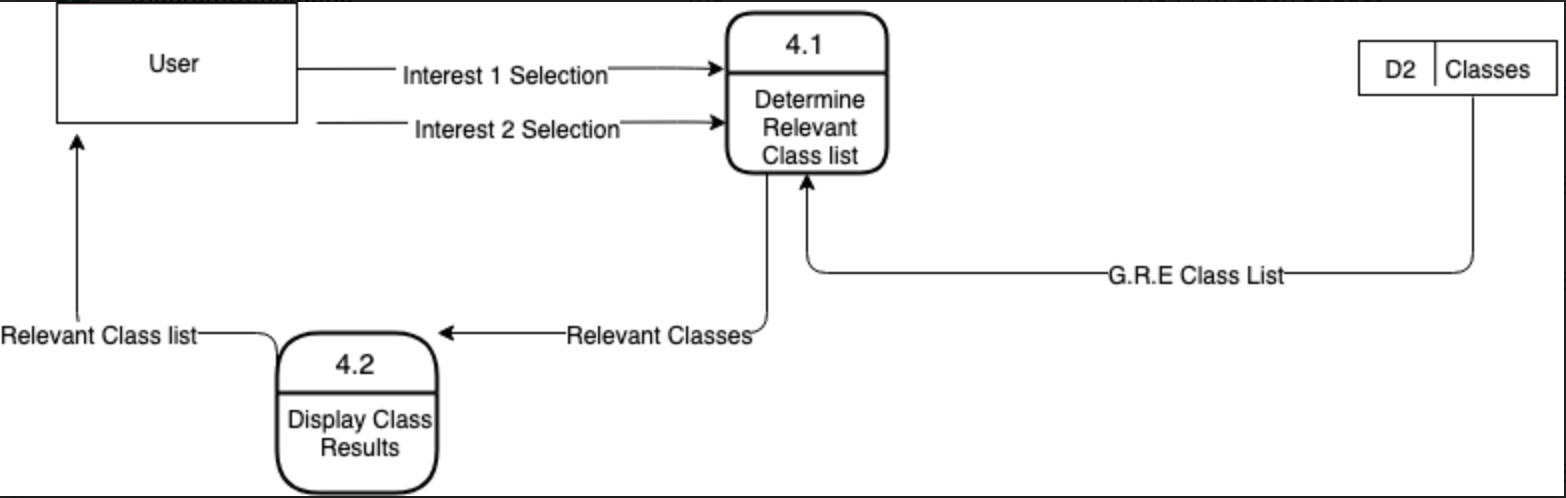
**Diagram Level 1: Display Class Information**

When the user is viewing the program information after going through Display Program Information, the user can select a class that is listed on the program to view it. Their selection is then requested and found by extracting it from the Classes table and displayed to the user.

****

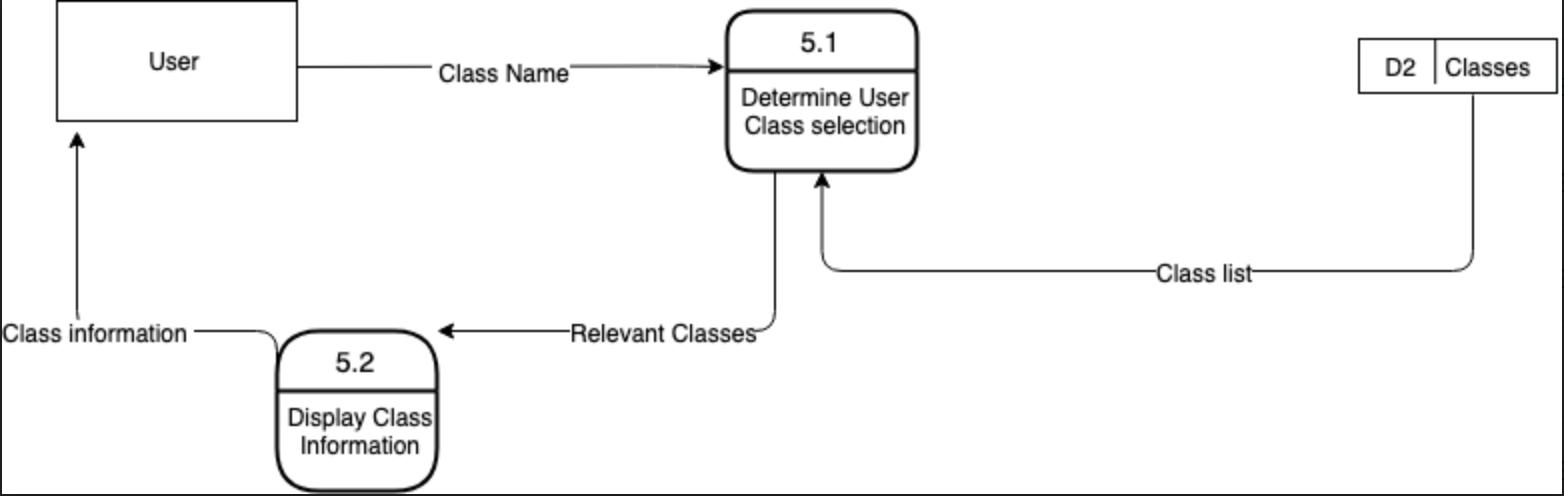
**Diagram Level 1: Display GRE Search Results**

The two inputs provided from the user are used as conditions to search for the closest matching General Requirement Electives (GRE) from the Classes table. This list of classes is then provided back to the user in a format the user can read.

****

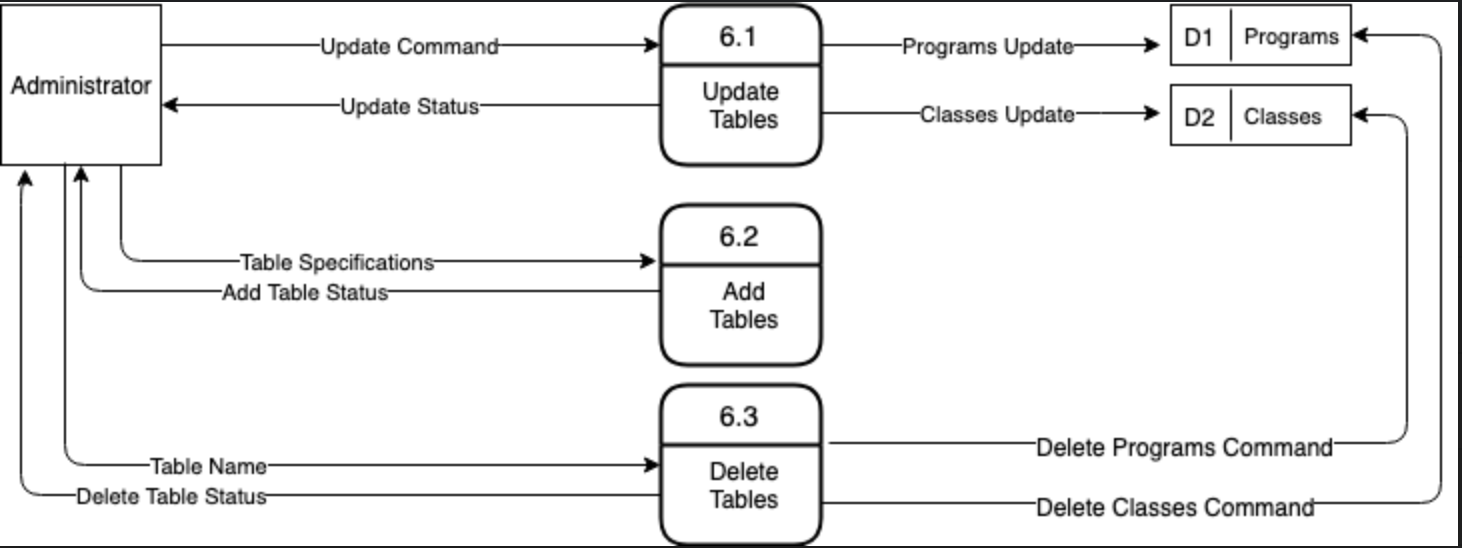
**Diagram Level 1: Display GRE Class Information**

After the user receives the results from Display Program Information, the user can select a GRE class that is listed in order to view it. Their selection is requested and found by extracting it from the Classes table and displayed to the user.

****

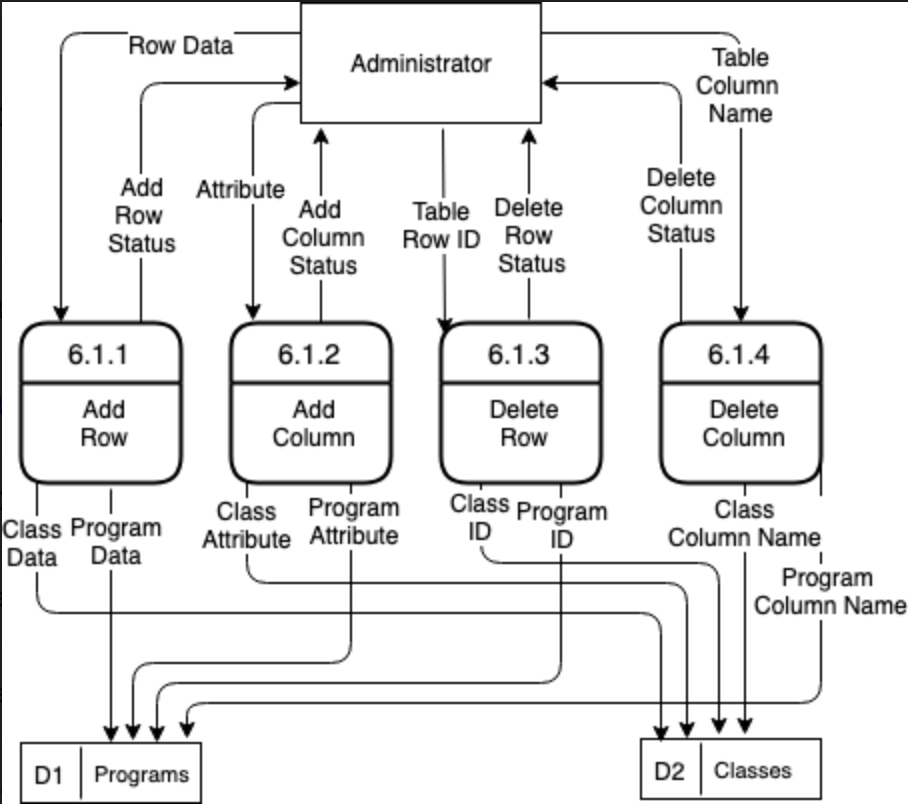
**Diagram Level 1: Update Database**

Administrator enters commands into the database system that allow it to update, add, and delete tables by providing it with and update command, table specifications, and table name respectively. The tables in the database, specifically Classes and Programs, can be manipulated through Update Tables and Delete tables. Administrator is given feedback about the success of each action through update, add table, and delete table status.



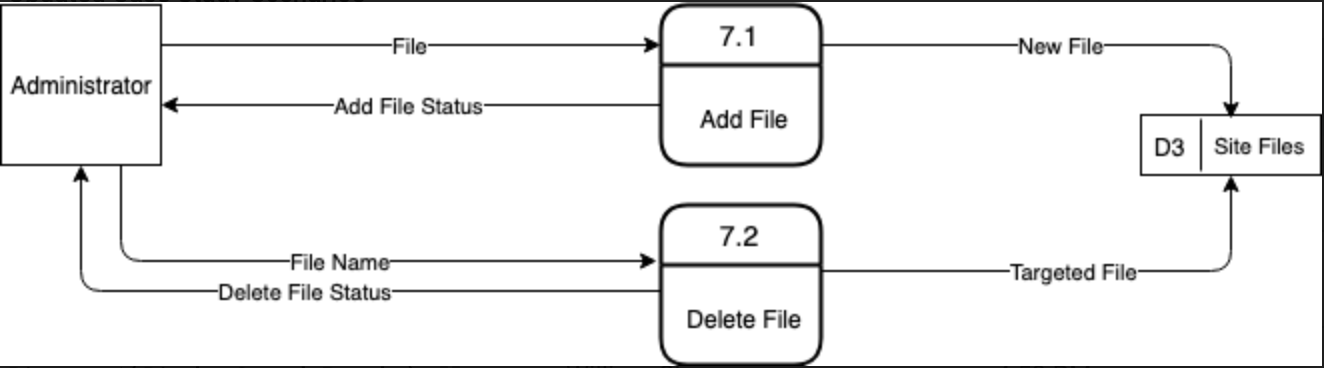
**Diagram Level 2: Update Tables**

A level two diagram based off of the update tables process from the Update Database Diagram. Unfortunately, we could not find a way to avoid having the lines cross at the bottom, but they were each oriented to one side to make the separations easier to identify. When the Administrator updates the database, he can add rows, add columns, delete rows, and delete columns to manipulate individual tables. There are two tables in which these processes can be applied which are the Programs and Classes tables. Administrator receives feedback through “Status” if the specific process worked or not.

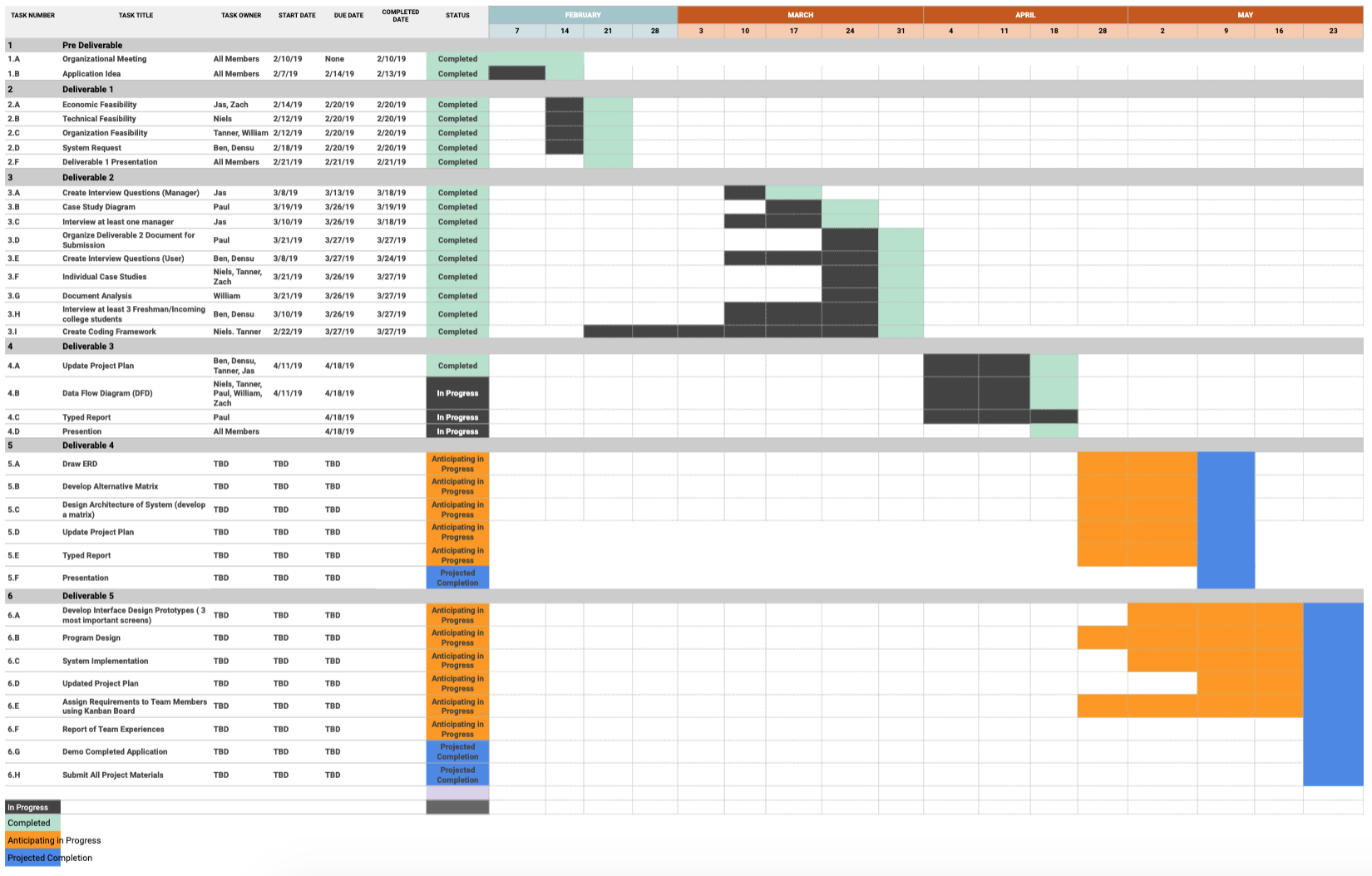
****

**Diagram Level 1: Update Site**

Administrator adds or deletes a file to the system that maintains the site. If a file with the same name is added to Site Files, it overwrites it. Administrator enters the file name that it wants to delete. Administrator receives feedback through “Status” whether the process was successful.

****

**Updated Project Plan**

****

Note: Due to the size of the image, it is recommended that this be viewed on the “[Gantt Chart](https://github.com/FearTheLion/IS436GoodFellas/blob/master/History/Deliverable3/GanttChart.png)” file on our GitHub repository.

**Gantt Chart Summary:**

In our updated project plan we have a Gantt chart listing all tasks for each deliverable, to whom has participated in completing them, the start dates, due dates, and completion dates as well. This is so we can effectively negotiate and manage reasonable and achievable deadlines across the team. We color coded our chart with four different colors. Ideally you have two main colors representing “in progress” dates and a “completed” date. Since we still have two deliverables left (four & five) we decided to create two more colors that help show the estimated timelines on our Gantt chart. We created a color for when we “anticipate progress” and also an estimated “project completion” date. Regarding the timeline/dates, the dates that class will be held in the future or already has been held we used. For our Kanban Flow, use this [link](https://github.com/FearTheLion/IS436GoodFellas/blob/master/History/Deliverable3/KanbanDeliverable3.png) to view the board and this one to view an [example description](https://github.com/FearTheLion/IS436GoodFellas/blob/master/History/Deliverable3/KanbanDescriptionExample.png).