**Design Document**

**By**

Erica Castro-Morrison

Jonathan Mahoney

Akeem Wilkins

Norfolk State University

April 7, 2019

**Table of Contents**

Table of Contents..............................................................................................1

Introduction..........................................................................................2

Architecture Design………………………………………………………..2

Database Design………………………………………………………..3-8

SQL Queries……………………………………………………..……..….9

Algorithms………………………………………………………………9-10

Graphical User Interface………………………………………...…..11-12

Class Diagram and Classes………………………………………….…12

Design Process…………………………………………………………..13

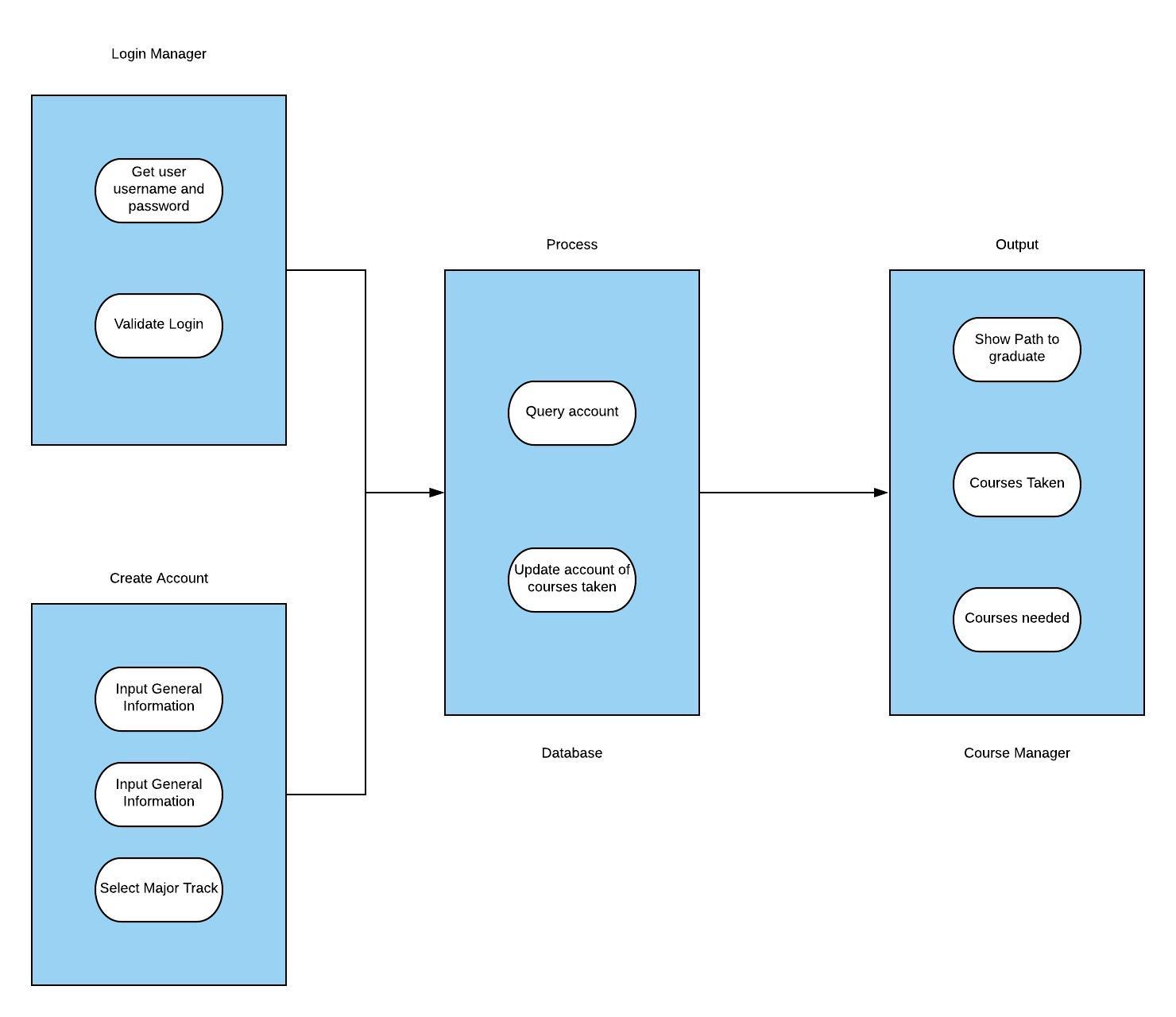
References………………………………………………………………..14

Design Document Template

**1. Introduction**

The purpose of this document is describe the implementation of the Electronic CSC Flowchart Application based on the software requirements previously documented. This document will go into detail with graphics for the architectural design and GUI. The database design will have tables to represent how the information is being stored. It will also map out the class diagrams and its relevant methods and parameters. The end goal of the Electronic CSC Flowchart Application is to allow students to plan, create and track their curriculum, update or modify an existing curriculum plan, and engage with their advisors in a convenient and transparent way.

**2. Architecture Design**



**3. Database Design**

# Tables schemas

|  |  |  |  |
| --- | --- | --- | --- |
| BS.CS |  |  |  |
| **Description** | This table describes the curriculum of the Norfolk State University BS in Computer Science degree plan. | | |
| **Attribute** | **Description** | Type | Example of Values |
| Course # & Name | This will display the course number with the title of the course | Text | CSC 295 Java App Prog |
| Credit Hour | This is the credit hour for the each course | Int | 1, 2, and 3 |
| Pre Req | These are courses that are needed before higher level courses | Text | ENG 101 is needed in order to take ENG 102 |
| Primary Key | no primary key is needed | | |
| Foreign Keys | None | | |
| SQL Code | CREATE TABLE `bscs` (  `Course Num & Name` text,  `Credit hour` int(11) DEFAULT NULL,  `Pre Req` text  ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci | | |
|  |  | | |

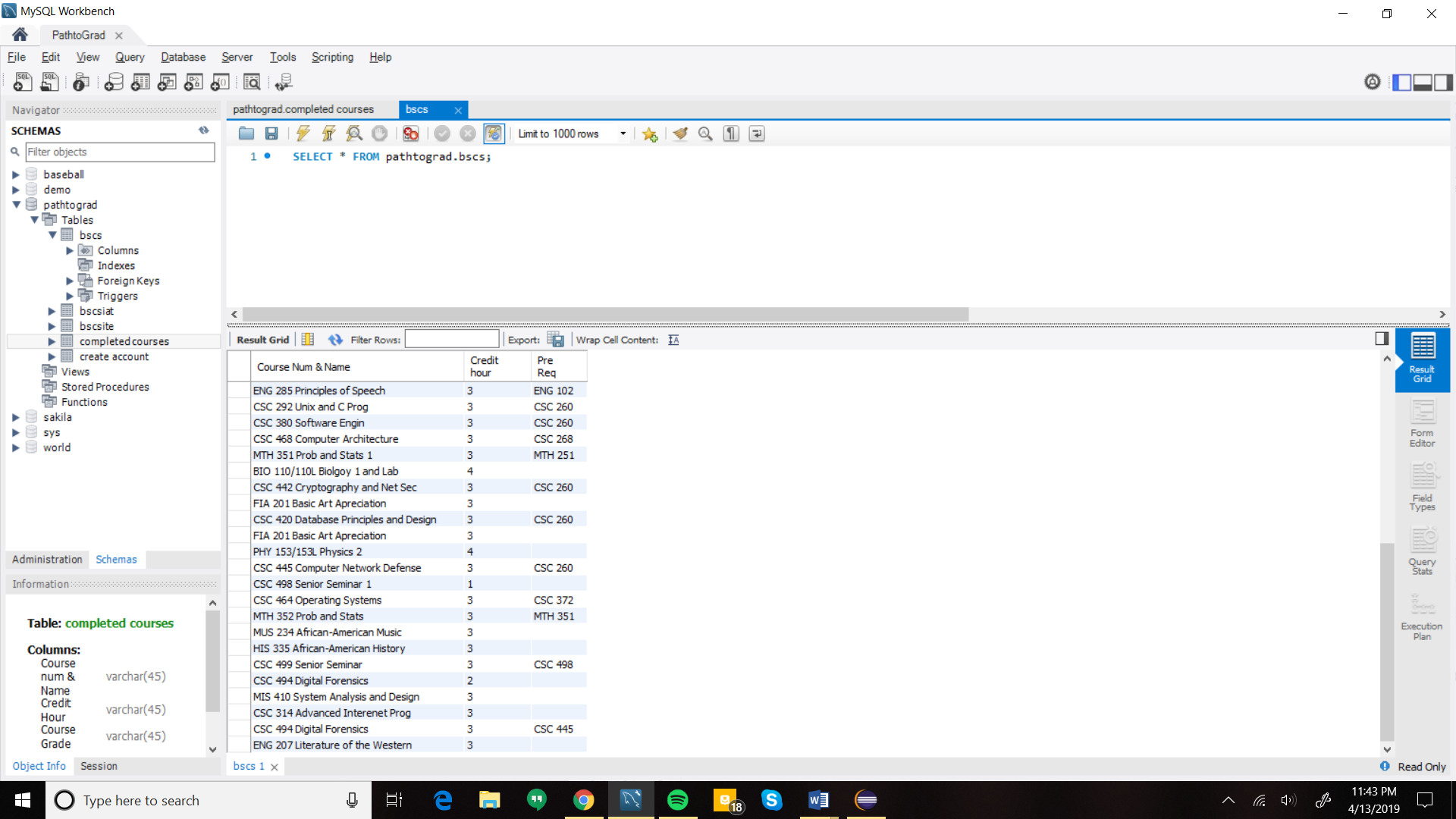
|  |  |  |  |
| --- | --- | --- | --- |
| BS.CS.IAT |  |  |  |
| **Description** | This table describes the curriculum of the Norfolk State University BS in Computer Science and Information Assurance track degree plan. | | |
| **Attribute** | **Description** | Type | Example of Values |
| Course # & Name | This will display the course number with the title of the course | Text | CSC 295 Java App Prog |
| Credit Hour | This is the credit hour for the each course | Int | 1, 2, and 3 |
| Pre Req | These are courses that are needed before higher level courses | Text | ENG 101 is needed in order to take ENG 102 |
| Primary Key | no primary key needed | | |
| Foreign Keys | none | | |
| SQL Code | CREATE TABLE `bscsiat` (  `Course # & Name` text,  `Credit Hour` int(11) DEFAULT NULL,  `Pre Req` text  ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci | | |
|  |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| BS.CS.ITE |  |  |  |
| **Description** | This table describes the curriculum of the Norfolk State University BS in Computer Science and Information Technology track degree plan. | | |
| **Attribute** | **Description** | Type | Example of Values |
| Course # & Name | This will display the course number with the title of the course | Text | CSC 295 Java App Prog |
| Credit Hour | This is the credit hour for the each course | Int | 1,2, and 3 |
| Pre Req | These are courses that are needed before higher level courses | Text | ENG 101 is needed in order to take ENG 102 |
| Primary Key | no primary key needed | | |
| Foreign Keys | none | | |
| SQL Code | CREATE TABLE `bscsite` (  `Course # & Name` text,  `Credit Hour` int(11) DEFAULT NULL,  `Pre Req` text  ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci | | |
|  |  | | |

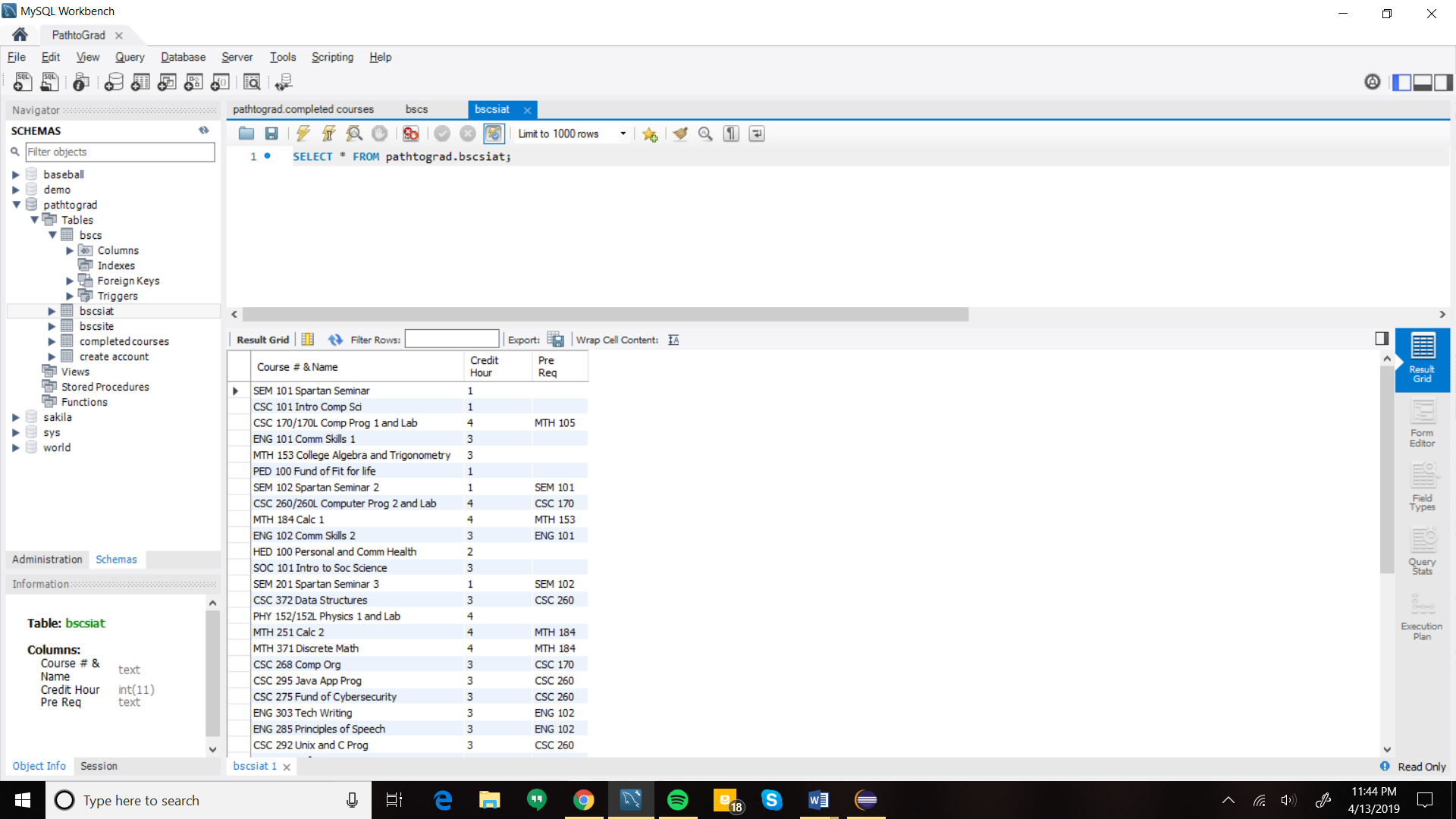
|  |  |  |  |
| --- | --- | --- | --- |
| Completed Courses |  |  |  |
| **Description** | This table describes the completed courses table that asks the user to check off which classes they have already completed. | | |
| **Attribute** | **Description** | Type | Examples of Values |
| Major Track | This is where the user has selected the major to follow in order to graduate. | VarChar | BS.CS, BS.CS.IAT, and BS.CS.ITE |
| Last Name | Last Name of a student | VarChar | Jones |
| Username | this is the users unique name created for their account | VarChar | j.jones1234 |
| Password | This is a security feature to protect 3rd parties from access the users information. | VarChar | \*\*\*\*\*\*\*\*\*\* |
| Primary Key | no primary key needed | | |
| Foreign Keys | none | | |
| SQL Code | CREATE TABLE `completed courses` (  `Course num & Name` varchar(45) NOT NULL,  `Credit Hour` varchar(45) DEFAULT NULL,  `Course Grade` varchar(45) DEFAULT NULL  ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci | | |
|  |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| Create Account |  |  |  |
| **Description** | This table describes the create account table that users register accounts to access the application to create a curriculum plan. | | |
| **Attribute** | **Description** | Type | Examples of Values |
| student number | Id of a student | Int | 0434354 |
| First Name | First Name of a student | VarChar | John |
| Last Name | Last Name of a student | VarChar | Jones |
| Major Track | This is the selected course the student will pursuing to graduate | VarChar | BS.CS, BS.CS.IAT, and BS.CS.ITE |
| Username | this is where the user will create their own unique username | VarChar | j.jones1234 |
| Password | This is a security feature to protect 3rd parties from access the users information. | VarChar | \*\*\*\*\*\*\*\*\* |
| Primary Key | the primary key is the student Number | | |
| Foreign Keys | none | | |
| SQL Code | CREATE TABLE `create account` (  `student num` int(11) NOT NULL,  `First Name` varchar(45) DEFAULT NULL,  `Last Name` varchar(45) DEFAULT NULL,  `Major Track` varchar(45) DEFAULT NULL,  `Username` varchar(45) DEFAULT NULL,  `Password` varchar(45) DEFAULT NULL,  PRIMARY KEY (`student num`)  ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci | | |
|  |  | | |

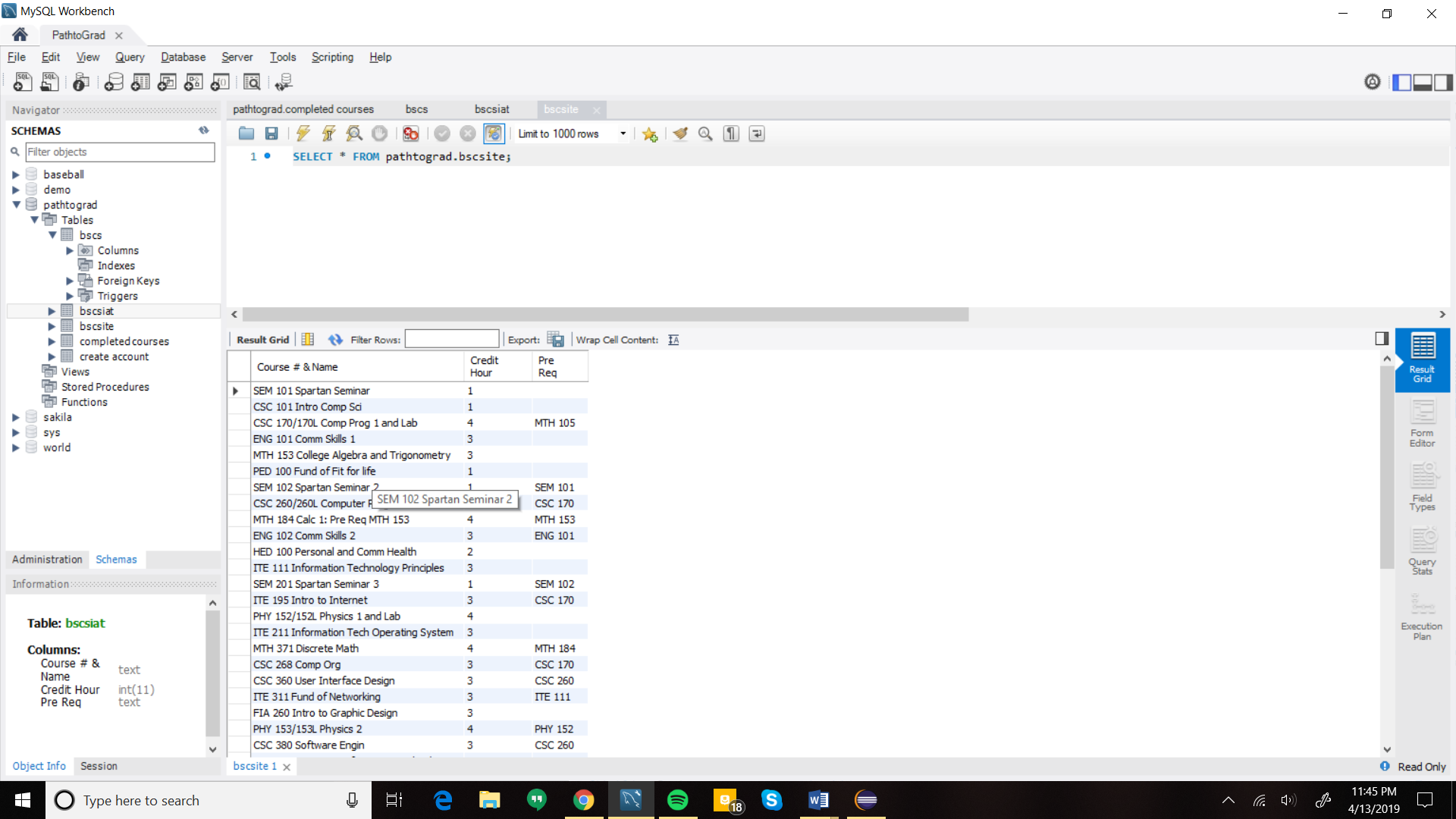
**Courses BS.CS**



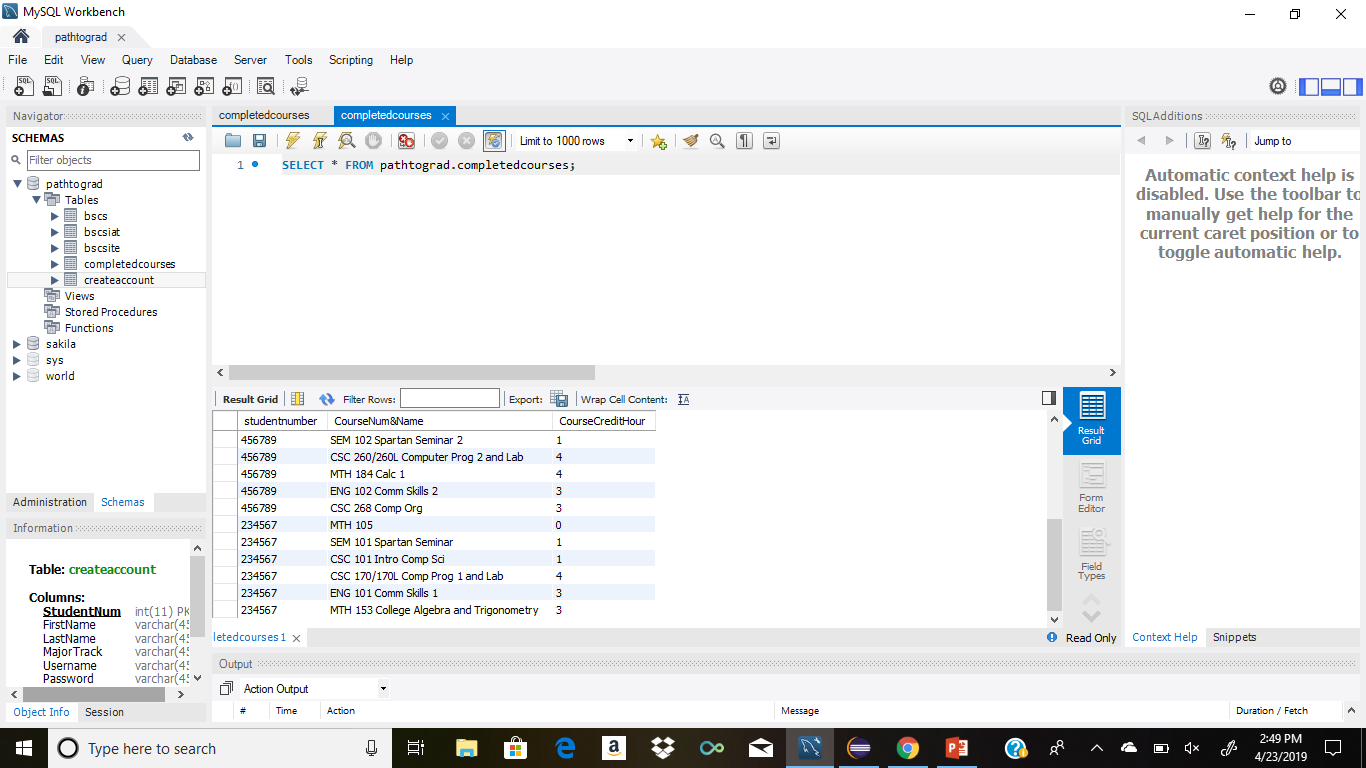
**Courses BS.CS.IAT**

****

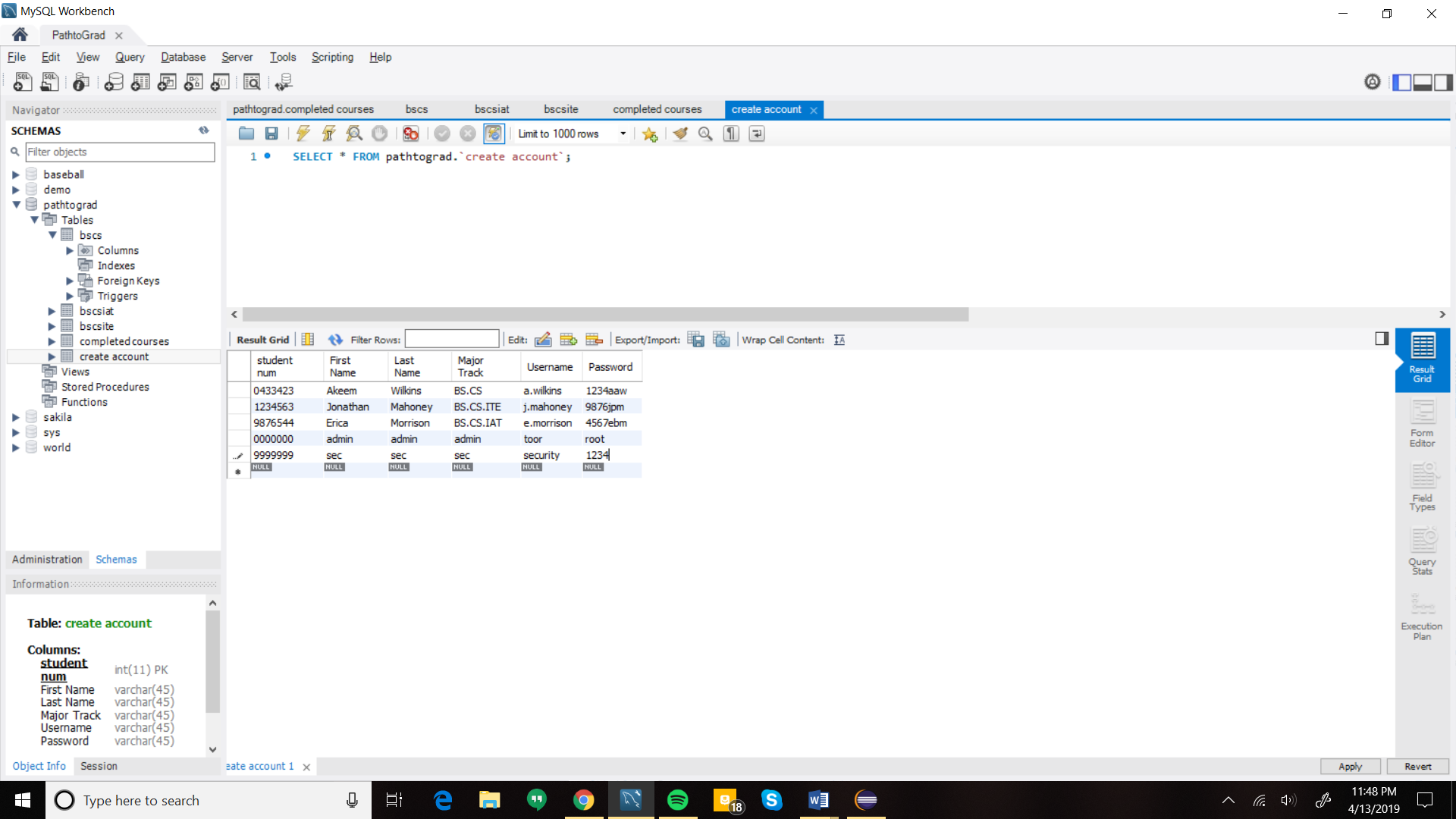
**Courses BS.CS.ITE**

****

**Completed Courses**



**Create Account**

****

**SQL queries:**

**BS.CS SQL queries:**

CREATE TABLE `bscs` (

`Course Num & Name` text,

`Credit hour` int(11) DEFAULT NULL,

`Pre Req` text

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci

**BS.CS.IAT SQL queries:**

CREATE TABLE `bscsiat` (

`Course # & Name` text,

`Credit Hour` int(11) DEFAULT NULL,

`Pre Req` text

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci

**BS.CS.ITE SQL queries:**

CREATE TABLE `bscsite` (

`Course # & Name` text,

`Credit Hour` int(11) DEFAULT NULL,

`Pre Req` text

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci

**Completed Course SQL queries:**

CREATE TABLE `completed courses` (

`Course num & Name` varchar(45) NOT NULL,

`Credit Hour` varchar(45) DEFAULT NULL,

`Course Grade` varchar(45) DEFAULT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci

**Create Account SQL queries:**

CREATE TABLE `create account` (

`student num` int(11) NOT NULL,

`First Name` varchar(45) DEFAULT NULL,

`Last Name` varchar(45) DEFAULT NULL,

`Major Track` varchar(45) DEFAULT NULL,

`Username` varchar(45) DEFAULT NULL,

`Password` varchar(45) DEFAULT NULL,

PRIMARY KEY (`student num`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci

**Algorithms**

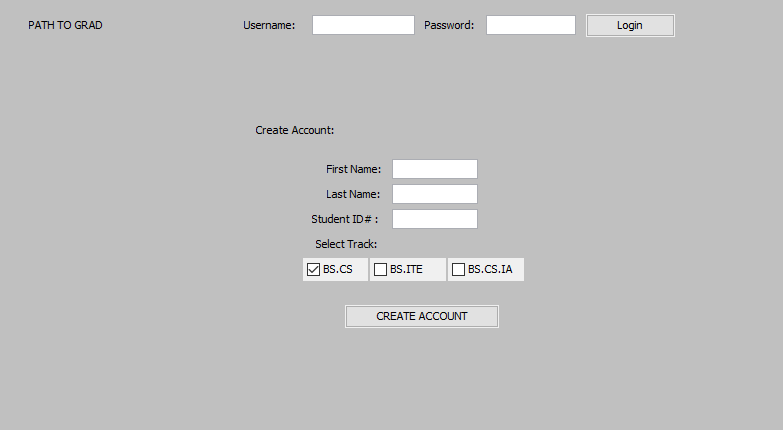
**Generate Track**

public void generateTrack(ResultSet dbTrack) throws SQLException  
53 {   
54   
55 System.out.println("inside generateTrack");  
56 while(dbTrack.next())  
57 {  
58 //System.out.println("past while");  
59 Course c = new Course();  
60 Course p1 = new Course();  
61   
62 c.setCourseName(dbTrack.getString("CourseNum&Name"));  
63 //System.out.println("past course name");  
64   
65 c.setCreditHours(dbTrack.getInt("CreditHour"));  
66 //System.out.println("past credithour");  
67   
68 p1.setCourseName(dbTrack.getString("PreReq"));  
69 c.getPrereqs().add(p1);  
70 //System.out.println("past prereq");  
71   
72 trackCourses.add(c);  
73 //System.out.println("added course");  
74 }  
75 }

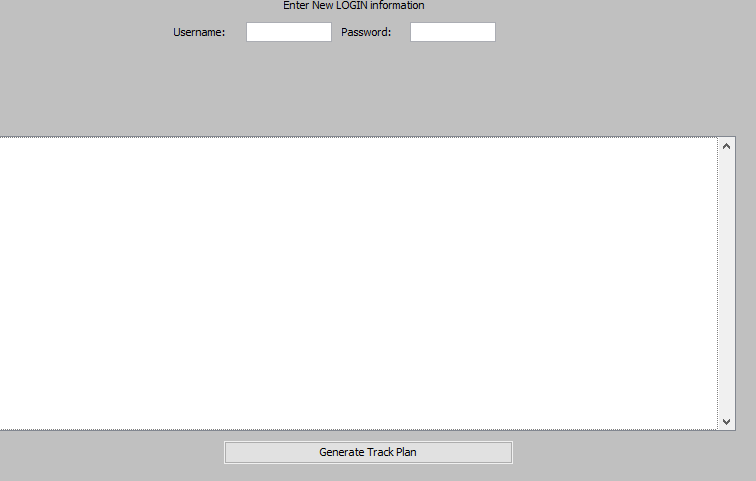
**Generate Courses**

46 public String generateCourse()  
47 {  
48 String s = "";  
49 for(int i = 0; i < prerequisite.size();i++)  
50 {  
51 if(prerequisite.get(i).getName().equals(""))  
52 {  
53 s = "none";  
54 }  
55 else  
56 {  
57 s = s + prerequisite.get(i).getName();  
58 }  
59   
60 }  
61 return nameCourse+" Credit Hours: "+creditHours+" Prerequisites: "+s;  
62 }

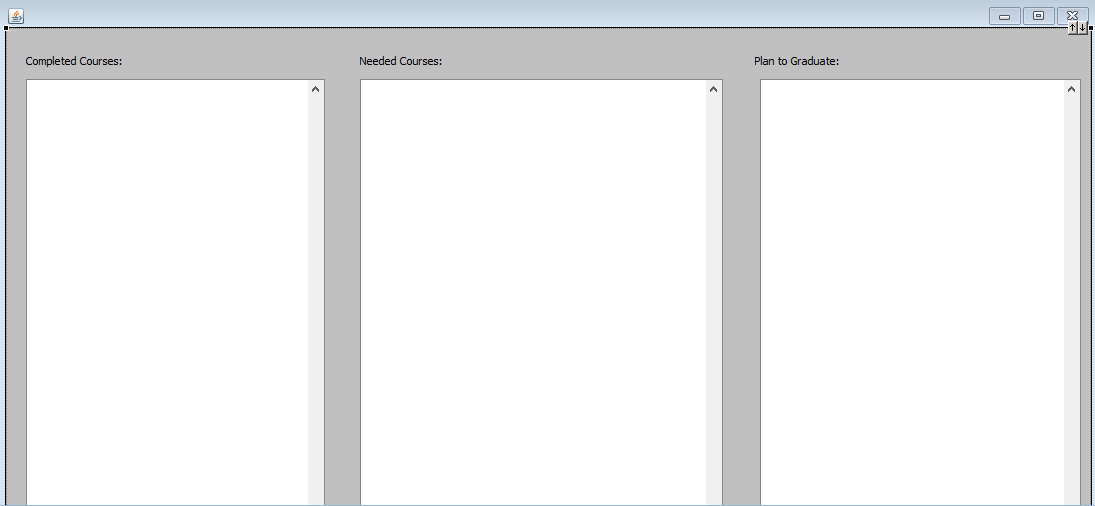
**4. Graphical User Interface**



For the Login screen, the user can either enter there information, or create a new account. Entering their info will to the login until the “Login” button is pressed this will cause action listener event to be triggered. This event then proceeds to create a connection to the database. Then based on the database collections the user will be granted access and generated their plan which is the 3rd page provided. If they fill in the account info that will generate a student object and begin to create one. Once they click create account the action event listener will be triggered. causing the next frame to be created which will generate a checklist of the track recorded.

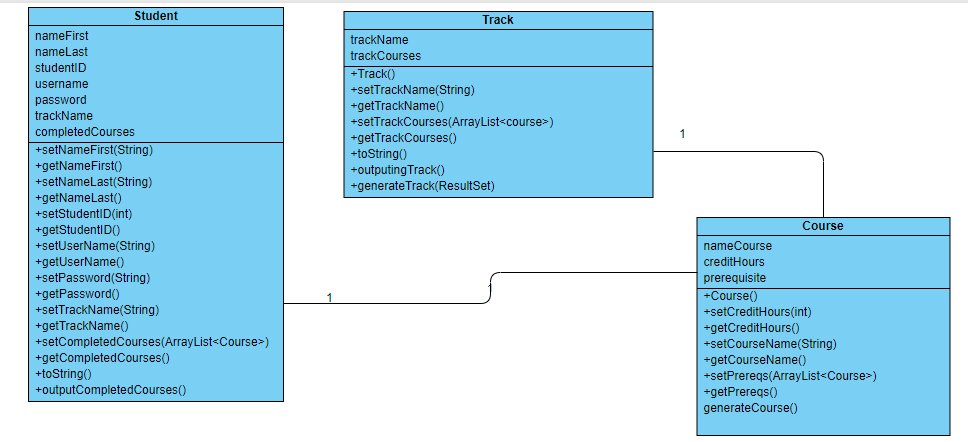


Based on the track clicked a checkList will be generated in the scroll field centered. They will also be required to enter new login information before they can generate their track plan. When the “Generate Track Plan” button is clicked the action event listener will read in all of the info collected and save to the database the information then generate the plan on page 3.



page 3 shows courses taken, courses needed then the plan to graduate. The exit button is the only clickable button from this page for now. Another small textfield showing how many credits required and left will also be added.

**5. Class Diagram and Classes**



**6. Design process**

The methodologies for the application used were the reuse-oriented software model and prototyping model. This application is modelled after the existing Norfolk State University curriculum planning and registration process but modifying it in way to aid the student’s course planning to be more efficient and help keep them on track. The design process was split into several parts: (1) Taking the information learned from the faculty and student surveys and questionnaires to map out needed functions. (2) Creating a general layout of the interface and proposed ways each method would function. (3) Research and choose the best programming languages and the IDE. (4) Research and choose the best database that works with the chosen programming language. (5) Research and learn the programming language if needed and how to implement the programming language with the database. (6) Create the application.

As stated, aspects of the prototyping method were used due to this not being a complete application that can be utilized for the registration and planning processes right away. This application is a prototype that is connected locally to a database. If the application were to continue it would be connected to a server where users would be able to use on the go to plan and track their courses, interact with faculty, and save, email, or print their curriculum plans.

**7. References**

Besinaiz, B., Lamarre, I., Babel, A. and Jennings, K. (2019). *DESIGN DOCUMENT*. [online] Available at: https://nsu.blackboard.com/bbcswebdav/pid-2450694-dt-content-rid-39217466\_1/courses/CSC-380-01-182/Design\_Document.pdf [Accessed 14 Apr. 2019].

Butler, C., Ing, J. and Lauk, E. (2019). *Design Document for Movies2you*. [online] Available at: https://nsu.blackboard.com/bbcswebdav/pid-2450694-dt-content-rid-39217467\_1/courses/CSC-380-01-182/SEDesignDocument.pdf [Accessed 14 Apr. 2019].

Butler, C., Ing, J. and Lauk, E. (2019). [online] Available at: https://nsu.blackboard.com/bbcswebdav/pid-2448758-dt-content-rid-39172596\_1/courses/CSC-380-01-182/CSC380RequirementsDocument.pdf [Accessed 30 Mar. 2019].

Morsi, R. (2019). [online] www.Nsu.edu. Available at: https://www.nsu.edu/getattachment/Academics/Faculty-and-Academic-Divisions/Schools-and-Colleges/College-of-Science-Engineering-and-Technology/Departments/Engineering/Programs/B-S-Electrical-Electronics-Engineering/EEE\_curriculum-CalReady\_2019.pdf.aspx?lang=en-US [Accessed 30 Mar. 2019].

Smith, C., McFall, C. and Dodson, A. (2019). [online] Available at: https://nsu.blackboard.com/bbcswebdav/pid-2448758-dt-content-rid-39172595\_1/courses/CSC-380-01-182/CSC380ProjectRequirementsDocument.pdf [Accessed 30 Mar. 2019].