SOFTWARE REQUIREMENTS AND DESIGN SPECIFICATION

ADVISE

DreamTeam
Aubri Bowman, Brian Brown,
Taniya Cope, Darian Foster,
Morgan Hutcherson

Sept 2022- April 2023

Table of Contents

1. Planning

1.1 Project Details

Project Title: Automated Advising System
Team Name: Team Two / Dream Team
Time Frame: Sept 2022 – April 2023

Project Contacts

Aubri Bowman: abowman3@bulldogs.aamu.edu
Brian Brown: bbrown80@bulldogs.aamu.edu
Taniya Cope: tcope1@bulldogs.aamu.edu
Darian Foster: dfoste23@bulldogs.aamu.edu
Morgan Hutcherson: mhutceh2@bulldogs.aamu.edu

1.2 Project Summary

a) Background and Justification

Alabama A&M University's advising system is notoriously known for holding students back from graduating due to the lack of support, communication, and urgency. We aim to completely eliminate the current advising system and replace it with a completely automated software system.

b) Project Objectives

- To give students easier access as to which classes they should and should not take in the coming semesters
- To not have students depend on an advisor when everything is already presented for the student to follow
- To provide students with their graduation status and completion percentage
- To allow students to foreshadow what to take in the future semesters
- Showcase a variety of general education courses that could be pursued by that specific student
- Predicting that certain students expected graduation term
- Students will be able to know their expected graduation year
- Implement GitHub
- Cesar cipher for the login authentication
- Incorporating laschefishe for time overrides/time conflicts

1.3 Project Methodology

a) Project Approach

The overall approach this team would take would be to get a clear understanding of the result we all are looking for. We also would make a schedule to stay on track.

Each team member will have a weekly task that must be completed by a certain date. Each member will have multiple tasks to dedicate themselves to for this website to be as successful as possible. We will have a certified leader, with the other team members listening attentively and implementing the best ideas to help build this project.

Work Breakdown and Task Estimates – *TBD*
Project Deliverables – *TBD*

1.4 Project Risk Management

- Technology issues – Connect to a hotspot, remote [Medium]
- Connectivity issues - reload page, connecting SQL Server to JavaScript [Medium]
- Software malfunctions - check that all files are located in the same directory [Highly]

1.5 Milestones

Log-In → Analyze Work Progress → Make Changes Where Necessary → Monitoring and Controlling → Submit Work Changes → Close

2. Software Requirements List

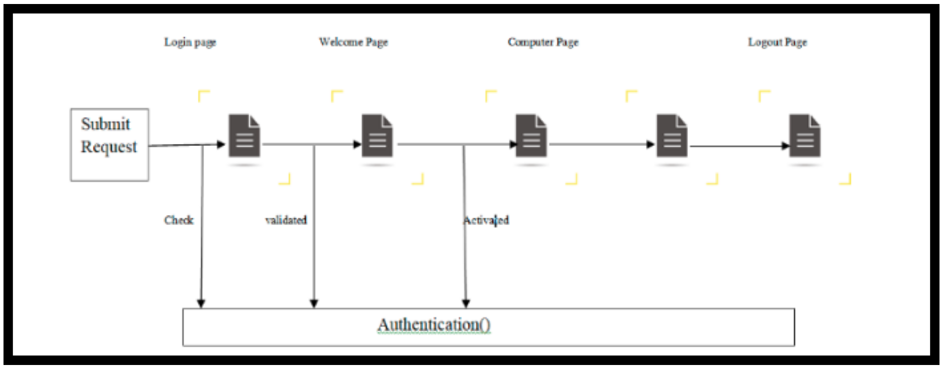
- Requires the data related to storing the students' info
- Requires data to be stored for each column related to the students' info for easy processing
- Requires storing specific data from the student that will be included in the projects database
- Requires critical information from the student that is needed in order for the model to function
- Requires the student to submit a request to the login page
- Requires the student to enter data into the username box
- Requires the student to enter data into the password box
- Requires the input data from the student to be checked and validated before

entering the welcome screen
- Requires the welcome screen to be activated once the data is entered, checked, and validated
- Requires user to submit a request by clicking on the logout button
- Requires the prototype to display a confirmation dialog asking if the user wants to logout
- Requires the prototype to ask for confirmation if the student wants to logout
- Requires the user to answer YES or NO to the logout confirmation. If YES, once the data is synced it will prompt the user's screen to return to the login screen. If NO, this will prompt the user's screen to return to the page they were previously on before trying to log out.
- Requires the data to be synced in whatever the users' choice is as to whether they want to log out or not

3. Requirement Workflows

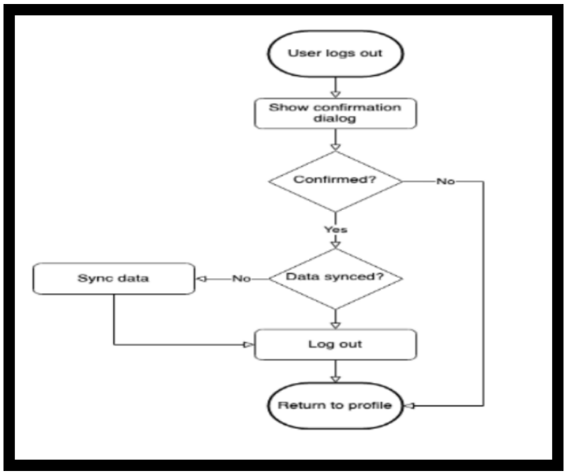
3.1 Workflow Name (Copy and repeat for each workflow)

- **Login Workflow**



The login workflow is when we login the page and submit the request validate the data after entering the details and validate the data and all the details show the welcome and then what will do on this page called the computer page and read the page the after can log out based on the requirement

- **Logout Workflow**



The logout workflow is when the student submits a request to log out after they are finished completing whatever objective they needed to when they first logged in. This will then take the user back out to the login screen, where they can repeat the login workflow to enter back to the welcome screen.

4. UI Design

4.1 UI Name (Copy and repeat for each UI)

- Login Window
- Sign-Up Window
- Homepage Window

4.1.1 UI Description and Related Requirements

- Description

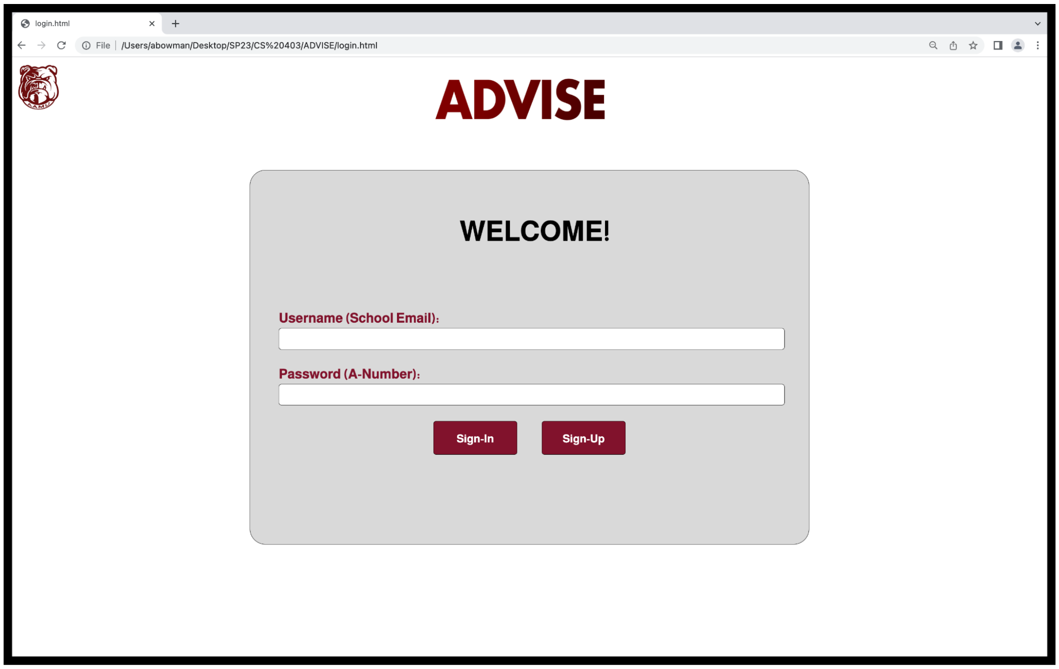
The login window introduces the users' progress throughout each school year (and semesters). Once the user opens this window, it will greet them with their current semester progress, their previous semester's classes, and the projected scheduling for the next semester. On the other hand, the start window also displays a dashboard with items such as their A#, concentration, major, GPA, etc. This window is to help and/or remind the student of what classes they are taking or have previously taken, which should help when it comes to registering for future classes.

The homepage window is strictly to guide the users' when it comes to advising them as to what classes to take. It also provides a section for when they drop a class, it will generate other classes that they can replace for the dropped class(es). This is the advisory window that can give users better help when it comes to classes they still need in order to graduate

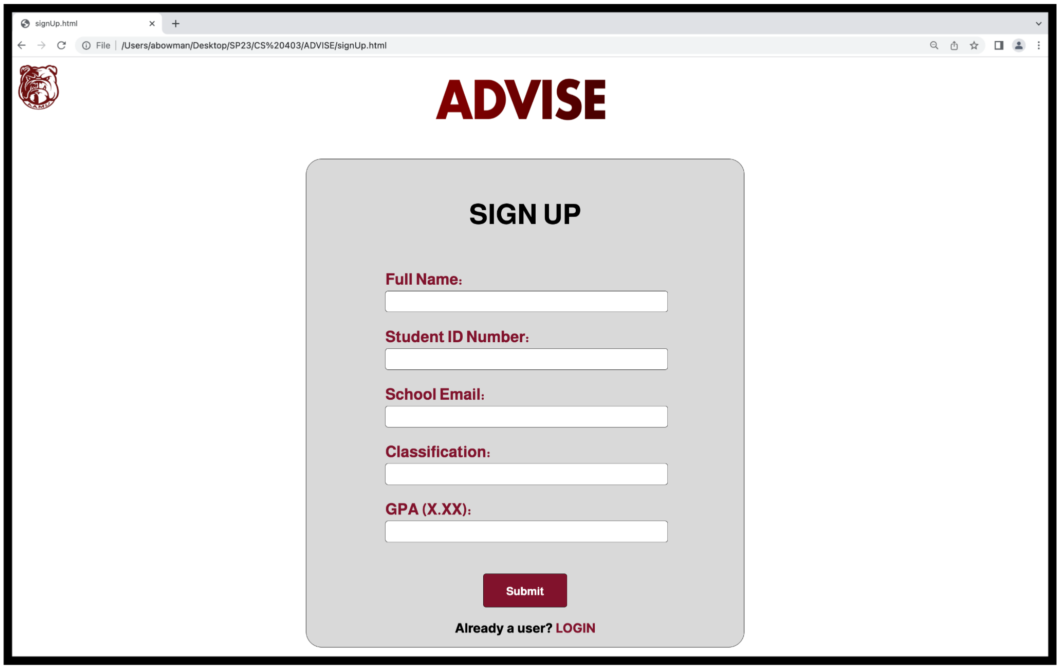
- ***Requirements*** - Login Window
    - **REQ #1: Requires the user to enter their username**
    - **REQ #2: Requires the user to enter their password**
- ***Requirements*** - Sign-Up Window
    - **REQ #1: Requires the user to enter their name**
    - **REQ #2: Requires the user to enter their A#**
    - **REQ #3: Requires the user to enter their school email address**
    - **REQ #4: Requires the user to enter their classification**
    - **REQ #5: Requires the user to enter their GPA**

4.1.2 UI Screenshot

- Login Window



- Sign-Up Window



ADVISE

SIGN UP

Full Name:

Student ID Number:

School Email:

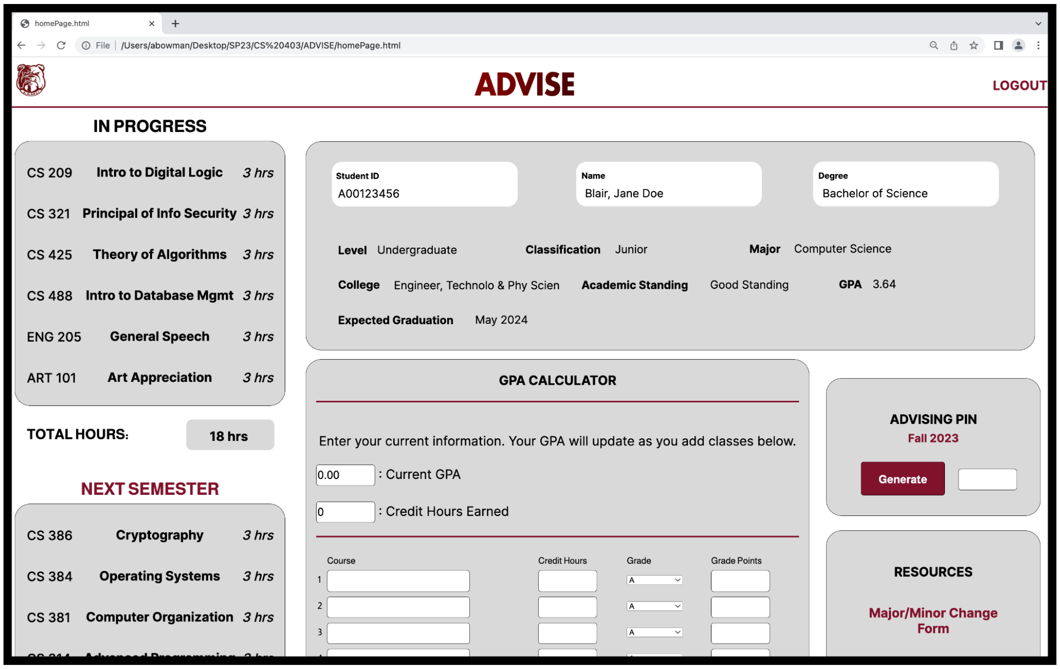
Classification:

GPA (X.XX):

Submit

Already a user? LOGIN

- Homepage Window



ADVISE

LOGOUT

IN PROGRESS

CS 209	**Intro to Digital Logic**	*3 hrs*
CS 321	**Principal of Info Security**	*3 hrs*
CS 425	**Theory of Algorithms**	*3 hrs*
CS 488	**Intro to Database Mgmt**	*3 hrs*
ENG 205	**General Speech**	*3 hrs*
ART 101	**Art Appreciation**	*3 hrs*

TOTAL HOURS: 18 hrs

NEXT SEMESTER

CS 386	**Cryptography**	*3 hrs*
CS 384	**Operating Systems**	*3 hrs*
CS 381	**Computer Organization**	*3 hrs*
CS 314	**Advanced Programming**	*3 hrs*

Student ID A00123456

Name Blair, Jane Doe

Degree Bachelor of Science

Level Undergraduate **Classification** Junior **Major** Computer Science

College Engineer, Technolo & Phy Scien **Academic Standing** Good Standing **GPA** 3.64

Expected Graduation May 2024

GPA CALCULATOR

Enter your current information. Your GPA will update as you add classes below.

0.00 : Current GPA

0 : Credit Hours Earned

	Course	Credit Hours	Grade	Grade Points
1			A	
2			A	
3			A	

ADVISING PIN

Fall 2023

Generate

RESOURCES

Major/Minor Change Form

4.1.3 UI Detailed Description

- Homepage Window

Field Name	Type Purpose and/or Action(s) to Perform
StudentID	text box to hold the characters for the user's A#
Name	text box to hold the characters for the user's name
Degree	text box to hold the characters for the user's degree type
Level	label displays whether the student is an undergraduate or a graduate student
Classification	label displays the students' classification
Major	label displays the students major of choice
College	label displays the college in which the students' major falls under
Academic standing	label displays whether the student is in a good or bad academic standing
GPA	label displays the GPA (grade point average) of the student
Expected Graduation	label displays the month and year of when the student can expect to graduate

- Advisory Window

Field Name	Type Purpose and/or Action(s) to Perform
Computer Science	Text box Stores info for classes needed relating to computer science
Math, Science	Text box Stores info for classes needed relating to math and science
General Education	Text box Stores info for classes needed relating to general education classes
Other Course	Text box Stores info for classes needed relating to other courses not related Computer science, math, science, and general education
Concentration	Text box Stores info for classes needed relating to concentration (if the student has one)
Dropped Classes	Text box Stores info for dropped classes

Class Replacements	Text box Stores info for classes that can be replaced for the students after they drop a class

5. Database Design

5.1 Table Name (Copy and repeat for each table)

- genEdCourses
- compSciCourses

5.1.1 Table Description and Related Requirements

● *genEdCourses TABLE -* The genEdCourses table stores data such as the course numbers, course titles, course category, and credit hours .

Column Name	**Type**	**PK\FK Description/Purpose and/or Action(s) to Perform**
courseNumber	varchar	
courseTitle	varchar	
courseCategory	varchar	
hours	int	

● *compSciCourses TABLE -* The compSciCourses table stores data such as the course numbers, course titles, course category, and credit hours .

Column Name	**Type**	**PK\FK Description/Purpose and/or Action(s) to Perform**
courseNumber	varchar	
courseTitle	varchar	
courseCategory	varchar	
hours	int	

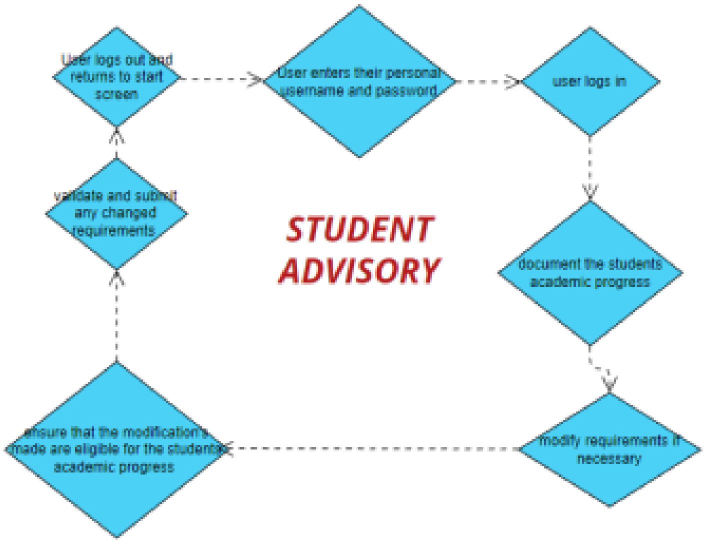
6. Process Design

6.1 Process Name (Copy and repeat for each process)

- *StudentAdvisory*

6.1.1 Process Description and Related Requirements

- This process design displays the sequence of how the prototype is supposed to work out. This is the design model that will help guide us (and the outsiders) on the best way to get past each problem in order to reach the primary goal. Our process design allows us to better understand the approach we will take as far as building our prototype.


User logs out and returns to start screen
User enters their personal username and password
user logs in
validate and submit any changed requirements
STUDENT ADVISORY
document the students academic progress
ensure that the modification's made are eligible for the students academic progress
modify requirements if necessary

Features:

Feature	Description
Calculating GPA	Once the classes are graded, the system will take the GPA points of every class and divide the point amount by the classes taken during that said semester
Calculating a 'What-If' GPA	The system will display the student's GPA after calculating the averages of each of their GPAs from each semester
Predicting Graduation Term	The users' term of graduation will be determined by the system and displayed on the screen

Predicting Future Semester Courses	As a user chooses what class(es) they plan to take, the system will generate the next classes that are in line for the user to take.
List Backup Courses	If a user plans to drop a course, the system will generate a pool of classes that the user can choose to replace the dropped class.
List All Available GE Courses	The user will be able to access a list of classes that will be available for them to take based on their major(s)
Display Graduation Completion Status	The system will calculate the number of credits received and the number of credits needed to be left to show the length of time the user needs to graduate.
Link Laserfiche form to change/declare major.	The user will have access to a Laserfiche form link to change/declare their major for the school year
Generate an advising pin each semester	The system will provide an advising pin for the user for them to register for classes each semester.
Secure login page	The user will be greeted with a secure login page. Their username will be their bulldog email, and their password will be their A#. Once they log in, their information will be encrypted and kept in a secure database.

Status of Completion:

Feature	**Description**	**% Finished**	**Comments**
Calculating GPA	The user's overall cumulative GPA will be calculated and displayed	100%	Take the overall grade and the credits applied and calculate the average
Calculating a 'What-If' GPA	The system will display the student's GPA after calculating the averages of each of their GPAs from each semester	100%	displays the student's GPA after calculating the averages

Predicting Graduation Term	The user's term of graduation will be determined by the system and displayed on the screen	85%	Determine the grade of the user and display the number of semesters needed to be left to graduate
List Backup Courses	The user will have access to a list of backup courses in case their main courses are filled.	90%	Using SQL/ MS Access to make a database of the courses
List All Available GE Courses	The user will be able to access a list of classes that will be available for them to take based on their major(s)	90%	Using SQL/ MS Access to make a database of the courses
DIsplay Graduation Completion Status	The system will calculate the number of credits received and the number of credits needed to be left to show the length of time the user needs to graduate.	0.0 %	Determine the classification of the user and display the number of semesters needed to graduate
Link the user to the Laserfiche form to change/declare a major	The user will have access to a Laserfiche form link to change/declare their major for the school year	100 %	Using SQL/ MS Access to make a database of the Laserfiche Forms
Generate advising pin each semester	The system will provide an advising pin for the user for them to register for classes each semester.	95%	Using SQL/ MS Access to make a database of advising pins
Secure login page	The user will be greeted with a secure login page. Their username will be their bulldog email, and their password will be their A#. Once they log in, their information will be encrypted and kept in a secure database.	90%	Working to make the textboxes accept input; make the buttons clickable

Github Link: https://github.com/aubrib/Automated-Advising-System

7. Maintainability

In order to ensure maintainability, we would have go under contract with the university in order to gain 100% access to the university's database. Until that can happen, the application will function more as a theoretical application. The GPA function of the website is the only portion that functions 100% by itself.