**Software Requirements Specification**

**Bison Advisor**

**Version: 1.0**

**Date: 9/20/2023**

**Author: MORGAN THORNTON**

Contents

[1. INTRODUCTION 3](#_Toc146125731)

[1.1. PURPOSE OF THIS DOCUMENT 3](#_Toc146125732)

[2. GENERAL DESCRIPTION 3](#_Toc146125733)

[2.1. PROBLEM STATEMENT 3](#_Toc146125734)

[2.2. PURPOSE OF THE PROPOSED SYSTEM 4](#_Toc146125735)

[2.3. USER CHARACTERISTCS 4](#_Toc146125736)

[2.4. USER OBJECTIVES 4](#_Toc146125737)

[3. FUNCTIONAL REQUIREMENTS 5](#_Toc146125738)

[4. NON-FUNCTIONAL REQUIREMENTS 6](#_Toc146125739)

[5. SYSTEM CONSTRAINTS 6](#_Toc146125740)

[5.1. PROGRAMMING LANGUAGES 6](#_Toc146125741)

[5.2. USER INTERFACE 6](#_Toc146125742)

[5.3. SOFTWARE FRAMEWORKS (APIs) 6](#_Toc146125743)

[5.4. DATA PERSISTENCE 7](#_Toc146125744)

[6. PRELIMINARY USE CASES 7](#_Toc146125745)

[7. APPENDICES 9](#_Toc146125746)

[7.1. DEFINITIONS, ACRONYMS, ABBREVIATIONS 9](#_Toc146125747)

[7.2. REFERENCES 9](#_Toc146125748)

# 1. INTRODUCTION

## PURPOSE OF THIS DOCUMENT

The purpose of this Software Requirements Documents is to propose the development of an automated advising system for students and advisors within Howard University. The current advising model in use has some major flaws for both the students and advisors of the institution. The proposed system aims to solve these problems and improve on the current quality of service for both students and advisors.

* 1. SCOPE OF THE PROJECT

The Bison Advisor project will automate the process of advising for both students and advisors. This will be achieved by creating a machine learning model that will use available data to give accurate advice to each student on a personal level. The project will also incorporate algorithms to calculate and evaluate a student’s performance as they continue with the learning process. Thus, the institution can evaluate accurately the student’s progress as they continue learning and identify possible issues that might be occurring in individuals with regards to their academics. The goal is to provide a system that is available to the students at all hours of the day, while also reducing the workload on advisors by having information on non-complex matters available to the students online.

# GENERAL DESCRIPTION

## PROBLEM STATEMENT

Howard university has an advising model for its students whereby students are free to seek for advice from advisors in the institution. This allows the students to seek for help when needed and helps new students acclimatize quickly to the new environment. This model has worked well for the period it has been in use. However, it has a few deficiencies. There many more students than there are advisors in the institution. Advisors can therefore be easily overwhelmed by the students, making it more probable that the advisors may not issue good advice to the students when required. Advisors are also only available during their working hours, usually 9.00am to 5.00pm, a similar period when most students are having classes. This further reduces their availability to a few minutes or hours between classes when the students can seek advice. The time constraint can make it difficult for the students to be helped sufficiently, especially if the matter is a complex one. The same can also be a contributing factor to students never seeking for help they need from the advisors and look for alternative sources that can be misleading.

The high ratio of advisors to students also makes it difficult for advisors to really personalize their sessions with the students in order to give them advice in the best way possible. Advice can therefore be very general and may not be of much help to the individual.

## PURPOSE OF THE PROPOSED SYSTEM

We propose to build Bison Advisor, a system that will aid in overcoming the shortcomings with the current system. Bison Advisor will be an advising assistant, powered by AI that will be able to:

* Personalize the advising experience for each student by using their academic history, their goals, and preferences.
* Offer 24/7 advice to all the students. Students will be able to access advice any time they wish.
* Provide the more routine tasks and advice to the students and refer only the most advanced and complex of cases to the advisors. This will ease the load on the advisors and allow students to receive the help they need more efficiently.
* Utilize data analysis techniques to analyze student academic data and identify areas of concern for each student, if any, and help the students deal with these areas.
* Offer advising services to a larger number of students more efficiently than the advisors would.
* Provide the students with resources and information that enable them to make informed decisions in their academic journey.
* Learn with time and new data in order to provide to the students the best advice possible.

## USER CHARACTERISTCS

The users in the system will primarily be students at Howard University. They will be able to ask the advisor questions in whatever area they seek advice, and the advisor will give them the best response to their queries, given information it has access to. Student will also have access to analysis of their academic progress and recommendations on what to do in order to improve academically.

The other user demographic will be advisors of the institution. They will have access to the student’s personalized sessions and view what was discussed between the student and advisor. This will enable the advisor to have a concrete view of the context of the problem and tackle the more complex issue more efficiently.

## USER OBJECTIVES

Students using the system will be able to:

* Create student accounts that will give them access to the system.
* Have real-time conversation with Bison Advisor concerning whatever advice the students want.
* Access comprehensive reports and analysis about their academic performance, trends, and any recommendations on how to improve their academic performance.
* Access previous advice sessions they had with either the assistant or a university advisor.
* Access generic information about university procedures from their portal

Advisors using the system will be able to:

* Create advisor accounts to give them advisor access to the system.
* Access limited student information with regards to sessions held with Bison Advisor that will enable them to give personalized advice services.
* Record sessions they have with the students on the portal for students to access at later times.
* Record recommendations for students to follow, after their sessions.

# FUNCTIONAL REQUIREMENTS

Users in the system should be able to:

1. Create an account as a user or advisor using information already held by the university. Key information will include names, email addresses and phone numbers. Registration attempts on data not already in the university’s systems will be rejected.
2. Login to their personal accounts using the information they provided. Users may be asked to verify their login using one-time passwords for added security.
3. Students interacting with the system will do so through an interactive dashboard that contains a number of sections. One will hold the Advising assistant, where they can ask questions and receive real-time answers. One will hold historical conversations held between the students and advising assistants and university advisors. The final section will hold academic information, evaluation, and analysis on the individual student.
4. The students will be able to update contact and profile information when they need to.
5. The system will be available as a web link, so that it can be available to as many students as possible. Progressive web apps may be used in cases of access using mobile devices.
6. Advisors will be able to view student information of students assigned to them, limited to academic and session details between the student and advising assistant.
7. Advisors will be able to record and save session information in the system, accessible by only them and the students.
8. Advisors can also mark which sessions between them, and students can be utilized by the advice assistant during the process of generative learning (Pappas, *Instructional design models and theories: The generative learning theory* 2021).
9. Advisors should be able to write recommendations to the students, which the students will follow from their accounts.

# NON-FUNCTIONAL REQUIREMENTS

1. The system should be able to fully load the web page in less than 4 seconds for every concurrent user regardless of the number of active users.
2. Requests made to the Bison Advisor should be handled in less than 3 seconds and results displayed to the user.
3. All database transactions should have ana average latency of 40ms, so that all transactions seem real-time.

# SYSTEM CONSTRAINTS

## PROGRAMMING LANGUAGES

The system will be primarily built with the python programming language. Flask Python will be used to create the website and backend API hosting the assistant. The generative model will be built with python and API calls used to interact with it.

The database model will be written in NoSQL, since data may be in an unstructured format due to the unstructured nature of human interaction.

## USER INTERFACE

The system will be built with material design concepts that keep the interface uniform throughout the system. Lazy loading will be used to keep the page as light as possible and user data and costs restricted to only what is required.

The primary colors of the system will be white, blue, and red, the primary colors of Howard University.

The website will have ability to convert to a progressive web application for easier access on mobile devices.

The website will be designed for all screen sizes to ensure that content fits on screen for all devices.

## SOFTWARE FRAMEWORKS (APIs)

The application will be built using Python’s Flask RESTful framework. Using the MVC design pattern, the API will be decoupled from the client side of the application.

The system’s API will be stateless, utilizing session tokens to maintain user state while they are active on the system.

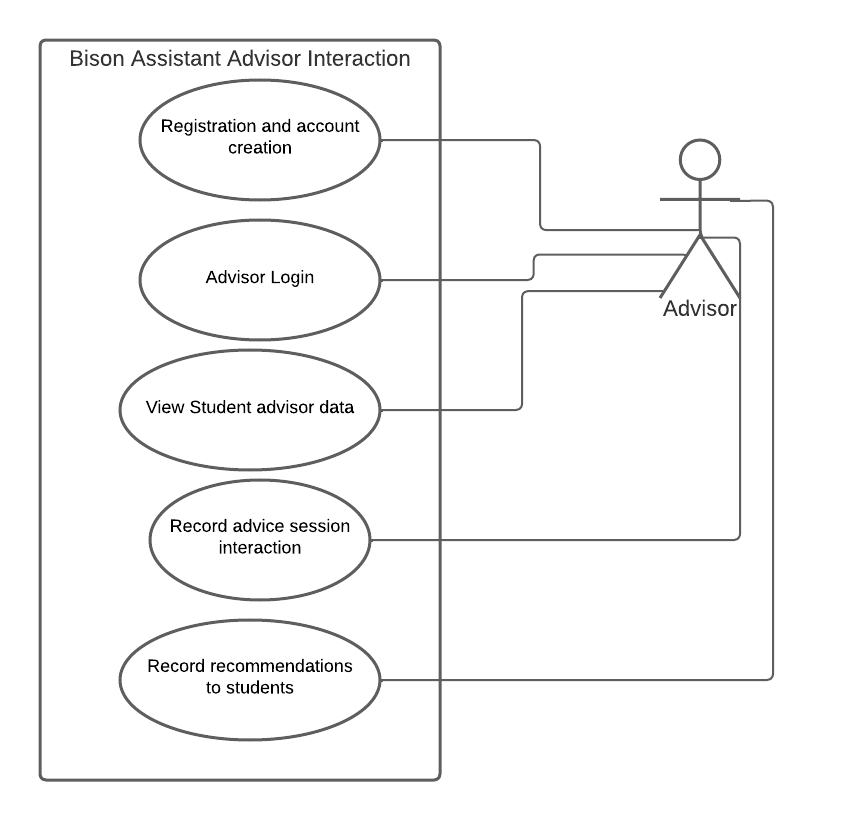
Data received from the API will be in JSON format.

## DATA PERSISTENCE

All data on the system will be stored in a NoSQL database. This database model was chosen since data stored will most likely be in an unstructured format. The database chosen for the system is MongoDB database.

# PRELIMINARY USE CASES

Bison Assistant Usecase diagram

# APPENDICES

## DEFINITIONS, ACRONYMS, ABBREVIATIONS

1. API – Application Programming Interface. These are protocols and definitions that are used when building and during integration of application software. (*What is a rest api?* 2020).
2. REST API – Representational State Transfer. An architecture that defines constraints to be used when implementing web services. Restful services should be independent of the client, stateless, cacheable, and layerable. (Johnson et al., *What is rest* 2022).
3. MVC – Model View Controller design pattern.
4. JSON – JavaScript Object notation.
5. SQL – Structured Query Language

## REFERENCES

1. Jafari, L. (2021, April 2). *What are functional requirements? types and examples.* WINaTALENT. https://winatalent.com/blog/2020/05/what-are-functional-requirements-types-and-examples/#:~:text=Functional%20requirements%20in%20an%20SRS,that%20focus%20on%20user%20needs.
2. Johnson, B. A., Gupta, L., Ravan, Admin, Jay, Person, Young, D., Packer, J., Choudhary, C., Luis, Tiamo, Alexander, Anto, Sunny, Gaurav, Talada, R., Andrea, Metkar, P., Iris, & Masne, S. (2022, April 7). *What is rest*. REST API Tutorial. https://restfulapi.net/
3. Pappas, C. (2021, May 12). *Instructional design models and theories: The generative learning theory*. eLearning Industry. https://elearningindustry.com/generative-learning-theory
4. *What is a rest api?* Red Hat - We make open-source technologies for the enterprise. (2020, May 8). https://www.redhat.com/en/topics/api/what-is-a-rest-api