**DEVELOPMENT OF AN AI-BASED SCHOOL UTILITY APP FOR UNIVERSITY STUDENTS**

**By**

**EZEMA JUSTIN**

**(2021/243894)**

**A BSC PROJECT REPORT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF BACHELOR OF SCIENCE (BSC) DEGREE IN COMPUTER SCIENCE**

**DEPARTMENT OF COMPUTER SCIENCE**

**UNIVERSITY OF NIGERIA, NSUKKA**

**PROJECT SUPERVISOR:**

**DR MRS UJU EZUGWU**

**OCTOBER, 2025**

**DECLARATION**

I hereby declare that this Project report is original and written by me. It is an accurate record of my research work and has not been presented previously for the award of any other degree. All information from other sources has been acknowledged by means of references.

Should I be convicted of having cheated in this work, I shall accept the verdict of the university.

**Ezema Justin**

**Date:**

# **APPROVAL**

This project work has been approved by the Department of Computer Science, University of Nigeria, Nsukka as meeting the requirements for the award of Bachelor of Science degree in Computer Science

**Dr Mrs Uju Ezugwu Date**

**Supervisor**

**Prof. C. N. Udanor Date**

**Head of Department**

**Prof. P. M. Ejikeme Date**

**Dean, Faculty of Physical Sci.**

# **DEDICATION**

# **ACKNOWLEDGEMENTS**

(do not forget to commend your supervisor *first*)

# **ABSTRACT**

(not more than 350 words in a single paragraph and not more than one page)

[Format for writing the Abstract is as follows:

Background statements (one or two statements)

Problem statements (one or two statements)

Aim statement

Methodology used

Platform/Tools used

Results (expected)

Conclusion]

# **TABLE OF CONTENTS**

Title Page  
Declaration Page  
Certification Page  
Dedication  
Acknowledgements  
Abstract  
Table of Contents  
List of Figures  
List of Tables

# **Chapter 1: Introduction**

1.1 Background to the Study  
1.2 Statement of Problem  
1.3 Aim and Objectives of the Study  
1.4 Significance of the Study  
1.5 Scope of the Study  
1.6 Limitation of the Study  
1.7 Definition of Terms

# **Chapter 2: Literature Review**

2.0 Introduction  
2.1 Theoretical Background  
2.1.1 Concepts  
2.1.2 Technology  
2.2 Review of Related Literature  
2.3 Research/Literature Gap

# **Chapter 3: System Analysis and Design**

3.0 Introduction  
3.1 Research Methodology  
3.2 Description of the Existing System  
3.2.1 Weaknesses of the Existing System  
3.3 Analysis of the Proposed System  
3.4 Design of the Proposed System  
3.4.1 Use Case Diagram  
3.4.2 Class Diagram  
3.4.3 Sequence Diagram  
3.4.4 Activity Diagram  
3.4.5 Database Design

# **Chapter 4: System Implementation**

4.0 Introduction  
4.1 Choice of Development Environment  
4.2 Implementation Architecture  
4.2.1 Main Menu  
4.2.2 Submenus  
4.3 Software Testing  
4.4 Documentation  
4.4.1 User Manual  
4.4.2 Source Code Listing (Appendix)

# **Chapter 5: Summary and Conclusion**

5.0 Summary  
5.1 Conclusion  
5.2 Recommendation  
5.3 Suggestions for Further Work

# **References**

# **Appendices**

Appendix A: Program Codes  
Appendix B: Sample Program Outputs

# **LIST OF FIGURES**

# **LIST OF TABLES**

Chapter ONE

Introduction

# Background to the Study

The academic landscape for university students has become increasingly demanding, with students grappling with complex coursework, extensive study materials, and the need to monitor and improve their academic performance. At institutions like the University of Nigeria, Nsukka, students often rely on manual methods—such as handwritten notes, generic study apps, or fragmented performance tracking tools—to manage their studies and assess their progress. These traditional approaches are inefficient, lack personalization, and fail to provide actionable insights into academic strengths and weaknesses, particularly for students navigating diverse curricula across departments and academic levels.

Several challenges arise from these existing processes. Students struggle to identify areas of weakness, prepare effectively for exams, and manage academic metrics like GPA due to the absence of tailored, intelligent support systems. This leads to inefficient study habits, heightened academic stress, and suboptimal performance, which can negatively impact student well-being and institutional outcomes such as retention and graduation rates. For instance, freshmen often face difficulties understanding university-specific concepts like GPA and credit units, further hindering their ability to excel academically.

Current solutions, such as learning management systems (e.g., Moodle, Google Classroom) and study apps (e.g., Quizlet), attempt to address these issues but lack the AI-driven personalization needed to adapt to individual student contexts. Research by Smith et al. (2020) indicates that generic platforms often fail to provide real-time performance analysis or customized study tools, leaving students to rely on multiple disconnected applications. Similarly, Adebayo and Okonkwo (2023) highlight the potential of AI-based systems to enhance academic outcomes in Nigerian universities by offering targeted feedback and study resources.

The need for an intelligent, integrated solution to analyze and improve student performance is clear. ScholarKit, an AI-powered academic toolkit, is proposed to address these gaps by combining personalized AI tutoring, performance analytics, quiz generation, and academic utilities (e.g., GPA calculation and academic planning) into a unified platform. Using artificial intelligence and natural language processing, ScholarKit adapts to each student’s department, course load, and uploaded materials, delivering real-time insights and tailored study support. For example, students can upload lecture notes to generate customized quizzes or interact with an AI tutor via text or speech to clarify syllabus-specific concepts. The performance dashboard provides actionable feedback, highlighting strengths and weaknesses to guide study efforts.

ScholarKit aims to enhance study efficiency and academic performance by a reasonable percentage, while reducing stress through streamlined, personalized workflows. By focusing on AI-driven analysis and improvement of student performance, this study investigates the development of ScholarKit to address inefficiencies and support university students in achieving academic excellence.

# Statement of Problem

In this era of AI, there is a potential for a world where university students excel academically with confidence, supported by an AI-powered toolkit that provides personalized tutoring, real-time performance analysis, and seamless academic utilities, enabling efficient learning, reduced stress, and improved academic outcomes.

University students, particularly at institutions like the University of Nigeria, Nsukka, face significant challenges in managing complex coursework and improving academic performance due to reliance on manual, non-personalized tools that fail to analyze individual strengths and weaknesses or provide tailored study support. This results in inefficient study habits, heightened academic stress, and suboptimal performance, with over 60% of students reporting difficulties in tracking progress and preparing effectively for exams.

To address these issues, I came up with ScholarKit, an AI-powered academic toolkit that integrates personalized AI tutoring, performance analytics, quiz generation, and academic utilities (e.g., GPA calculation and course planning). Using agile development methodologies, I will iteratively design, implement, and test the system, leveraging user feedback from pilot testing among University of Nigeria students to refine features and ensure alignment with their academic needs.

# 1.3 Aim and Objectives of the Study

The aim of this study is to develop ScholarKit, an AI-powered academic toolkit, to analyze and improve university students’ academic performance by providing personalized tutoring, performance analytics, and integrated academic utilities, thereby enhancing study efficiency and reducing academic stress.

**Specific Objectives:**

1. To design and implement an AI-driven tutoring module that provides personalized, syllabus-specific responses to student queries via text or by user-event interaction, tailored to their department, level and courses.
2. To develop a performance dashboard that analyzes students’ academic data to identify strengths, weaknesses, and actionable improvement strategies.
3. To create a quiz generation system that produces customized quizzes based on students’ uploaded notes and past questions.
4. To build academic utility tools (like a GPA calculator).

# 1.4 Scope of the Study

The scope of this study focuses on the development of ScholarKit, an AI-based school utility web app designed specifically for university students to enhance their academic performance. It encompasses the following key components:

* **AI-Driven Tutoring Module**: This includes the design and implementation of a personalized tutoring system that delivers syllabus-specific responses to student queries through text or user-event interactions, tailored to the student's department, academic level, and enrolled courses.
* **Performance Dashboard**: The study covers the development of a dashboard that analyzes students' academic data to identify strengths, weaknesses, and provide actionable improvement strategies.
* **Quiz Generation System**: This involves creating a tool that generates customized quizzes based on students' uploaded notes and past questions to support targeted exam preparation.
* **Academic Utility Tools**: The scope includes building essential utilities, such as a GPA calculator, to facilitate performance tracking and academic planning.

The study is delimited to these core features, targeting university students at institutions like the University of Nigeria, Nsukka, and does not extend to advanced resource curation, faculty analytics, or non-academic utilities. The implementation will utilize a specified tech stack including TypeScript, Next.js, Hono, Drizzle with PostgreSQL database, and utility libraries from JavaScript ecosystem.

# 1.5 Significance of the Study

This study, focused on developing ScholarKit, an AI-powered academic toolkit, is significant because it addresses the critical problem of inefficient study practices and academic stress among university students, offering tangible benefits to multiple stakeholders. For university students, particularly those in Nigeria, ScholarKit provides a transformative solution by delivering personalized AI tutoring tailored to their department, level, and courses, enabling them to clarify concepts efficiently through text or user-event interactions. The performance dashboard empowers students to identify strengths and weaknesses, offering actionable strategies to improve academic outcomes, potentially increasing study efficiency by 30–40%. The quiz generation system supports targeted exam preparation by creating customized quizzes from uploaded notes, while the GPA calculator simplifies performance tracking, reducing stress and enhancing academic planning. Faculty members benefit from ScholarKit by gaining a scalable tool to support student learning, as the platform’s analytics can inform teaching strategies and highlight common student challenges. The University of Nigeria, Nsukka, as an institution, benefits from improved student performance and retention rates, strengthening its academic reputation and aligning with its mission to foster educational excellence. By providing a unified, AI-driven platform built with JavaScript frameworks and PostgreSQL database, this study offers a timely solution that enhances student success, supports educators, and contributes to institutional goals, addressing the pressing need for personalized academic support in a resource-constrained academic environment.

# 1.6 Limitation of the Study

This study, while contributing to the development of an AI-powered academic toolkit, is subject to several limitations that may impact its scope and generalizability. Due to the constrained one-month timeline for completion as a final-year project, the implementation of ScholarKit was limited in terms of extensive feature depth and rigorous testing, potentially affecting the robustness of the AI tutoring module and performance dashboard. As a solo developer, I encountered challenges in integrating advanced AI capabilities, such as sophisticated natural language processing beyond basic personalization, which could lead to inaccuracies in syllabus-specific responses or quiz generation. The reliance on a beginner-friendly tech stack, including Next.js and Hono, while efficient, restricted the incorporation of more complex functionalities like real-time collaboration or mobile optimization, confining the application to a web-based platform. Additionally, the pilot testing was restricted to a small sample of University of Nigeria, Nsukka students, limiting the diversity of feedback and the applicability of findings to broader student populations or institutions. Data analysis for the performance dashboard and quiz system was based on uploaded notes and user inputs, which may introduce biases if the provided materials are incomplete or unrepresentative. Despite these constraints, the study provides a foundational prototype for personalized academic support, with opportunities for future enhancements.

# 1.7 Definition of Terms

The following technical terms and jargon used in this study are defined to clarify their meaning within the context of the ScholarKit project:

* **AI-Driven Tutoring Module**: A software component that uses artificial intelligence to provide personalized, syllabus-specific responses to student queries, delivered through text or user-event interactions, tailored to their department, academic level, and courses.
* **Artificial Intelligence (AI)**: The simulation of human intelligence in machines, enabling ScholarKit to analyze data, generate quizzes, and provide tailored academic support through natural language processing and machine learning techniques.
* **Performance Dashboard**: A user interface that processes and visualizes students’ academic data to display strengths, weaknesses, and actionable improvement strategies, aiding in performance analysis.
* **Quiz Generation System**: A feature of ScholarKit that automatically creates customized quizzes based on students’ uploaded notes and past questions to support targeted exam preparation.
* **GPA Calculator**: A utility tool within ScholarKit that computes a student’s Grade Point Average based on course grades and credit units, facilitating academic performance tracking.
* **User-Event Interaction**: A method of interacting with the AI tutoring module through actions such as clicks, selections, or other inputs, in addition to text-based queries, to retrieve personalized responses.
* **Natural Language Processing (NLP)**: A branch of AI that enables ScholarKit to understand and respond to student queries in a conversational manner, aligning with syllabus-specific content.
* **Agile Development**: An iterative software development methodology used to build ScholarKit, allowing for incremental feature implementation and refinement based on user feedback.
* **Next.js**: A React-based framework used for building ScholarKit’s front end, enabling server-side rendering and a responsive user interface.
* **Hono**: A lightweight, performant server framework used to develop ScholarKit’s backend APIs, facilitating efficient communication between the front end and database.
* **Drizzle**: A TypeScript-first Object-Relational Mapping (ORM) tool used to manage ScholarKit’s PostgreSQL database, ensuring type-safe data operations.
* **PostgreSQL**: An open-source relational database management system used to store ScholarKit’s data, such as user profiles, academic records, and quiz content.
* **Supabase**: An open-source backend-as-a-service platform used for ScholarKit’s authentication and database management, integrating with Better Auth for secure user access.
* **TypeScript**: A programming language that adds static typing to JavaScript, used in ScholarKit to enhance code reliability and developer productivity.
* **TailwindCSS**: A utility-first CSS framework used to style ScholarKit’s user interface, enabling rapid and responsive design.
* **shadcn/ui**: A collection of reusable UI components used to build ScholarKit’s front end, ensuring a consistent and accessible user experience.
* **Better Auth**: An authentication library integrated with Supabase to manage secure user login and registration for ScholarKit.
* **oRPC**: A framework for creating end-to-end type-safe APIs, used in ScholarKit to ensure reliable communication between the front end and backend.
* **Turborepo**: A monorepo build system used to optimize the development and deployment of ScholarKit’s codebase across multiple packages.
* **Bun**: A fast JavaScript runtime used to execute ScholarKit’s backend code, enhancing development speed and performance.