

**SMART CAMPUS MANAGEMENT SYSTEM PROPOSAL**

*Innovating Education Through Technology*

**Prepared by**: GILBERT KIPLANGAT

**Department**: Biological and Physical Sciences

**Admission No**: SMA/MC/01247/021

**Course**: CCS 403, Computer Science Project

**Instructor**: Dr Salmon Owidi

**Submission Date**: 2024 – 10 – 17

**Contact**: gilbertketer759@gmail.com, +254759104865

# **Abstract**

The following is a proposal for developing an integrated University Management System that will solve the problems experienced in contemporary universities. This includes student tracking for class attendance, facility management, safety features, and artificial intelligent advising, to mention but a few. With the help of Next.js, Django, and IBM Watson, which are incorporated in the proposed solution, it is possible to optimize everyday processes, engage students, and lay down valuable insights to help them make the right decisions. The innovative elements of the methodology involve user-oriented design, modularity of the solution and strict testing and review procedures. It is believed that this system will enhance the university experience for students, faculty and administrators and position the institution for the future of education.

# **Declaration**

I, Gilbert Kiplangat, hereby declare that this proposal is my original work and has not been submitted, either in part or in full, for any other degree or diploma at this or any other institution. All information derived from the work of others has been properly cited and referenced.

Signed: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: [2024 - 10 - 17]

# **1.0 Background**

Cultivating change in the dynamic environment puts universities under pressure regarding educational performance and activity effectiveness upgrades. Current approaches within the university's resources, control of attendance, and interaction between students, teachers, and other personnel are ineffective compared to using technologies.

The above challenges have been considered in proposing the University Management System that would incorporate the best technologies to develop an integrated system to support various aspects of university management. This system will incorporate features such as:

* Class attendance tracking
* Facility booking (classrooms, conference halls, labs)
* Safety monitoring
* Student dashboards
* AI-powered academic advising
* Automated notifications
* Task management
* Access control using RFID or QR codes
* User role management
* Student portfolios

Through this system, the university will experience a marked increase in efficiency, the quality of the learners' experience will be boosted, and the university will position itself to exploit future trends in the delivery of higher education.

# **2.0 Problem Statement**

Current challenges in universities are evident in their general operation and in delivering services to the learners, instructors, and other stakeholders. Some of the key issues include:

* **Inefficient attendance tracking**: Manual taking of attendance is tiresome and likely to cause many mistakes.
* **Complex resource management**: Booking classrooms, labs, and other facilities requires various procedures associated with many problems and misunderstandings.
* **Limited student engagement**: Such an approach to communication and academic advising students cannot be practical if it is based on traditional means.
* **Safety concerns**: The campus community is endangered due to the lack of monitoring and challenging emergency communication.
* **Administrative overhead**: The workload is significantly strained in handling various administrative tasks that may otherwise be automated.
* **Fragmented systems**: Fragmented and separate systems for individual purposes yield inefficiencies and disintegrate communication chains.
* **Limited insights**: The absence of an efficient central database and subsequent data processing means the university cannot decide based on data.
* **Outdated communication methods**: Using traditional means of passing information slows the information flow and decreases student interaction.
* **Inadequate student portfolio management**: Students have no place where they can host their performance at school and other accomplishments in student activities.
* **Inefficient task management**: The absence of central control of tasks implies challenging time-management scenarios and ineffective interaction of different departments.

The proposed university management system can solve the outlined problems. This practical, technologically advanced system will contribute to effective operation improvement and the overall university experience for everyone involved.

# **3.0 Justification**

The implementation of the proposed University Management System is justified by the following factors:

* **Technological advancement**: Given the pace of development and the fact that more and more current students have been born in the age of the internet, universities have to constantly evolve in order to remain attractive to learners and relevant to the modern world.
* **Operational efficiency**: Since most prior processes require manual intervention, the system will significantly decrease administrative burdens and enhance value.
* **Enhanced student experience**: Overall satisfaction and engagement are expected to rise due to the facility enabling students to access information, resources and or services from the platform.
* **Improved decision-making**: Centralized data collection and analysis will produce dependable information for the various tiers of the university administration system.
* **Cost-effectiveness**: Though one can think of the initial setup cost incurred while implementing it, the lengthy service period required to reduce manual labor and optimize resource usage will help reduce the cost.
* **Competitive advantage**: Modern technology solutions characterize universities most likely to attract and retain students in today’s competitive market.
* **Scalability**: They should be aware that the proposed system can easily be expanded to accommodate future growth and other features and functionality that may be required.
* **Compliance and security**: The system will also offer safety monitoring services to achieve compliance with Data Protection Regulations and the general safety of the campus, save paper, reduce negative environmental impacts and optimize resource use.
* **Environmental impact**: The system will help the university to save paper which will reduce the negative impacts on the environment and also the resource will be optimally used.
* **Preparation for future challenges**: The system's flexibility and adaptability are likely to ease the undertaking of future changes and challenges that may be expected to affect the university’s operations in the sector.

# **4.0 Significance**

The effective establishment of the university management system is paramount for different stakeholders in the university environment:

**1. For Students:**

* A heightened level of access to information and resources.
* Increased dialogue between students and staff of the faculty and school
* Remote individualized academic counselling with the help of AI.
* Simplified processes related to attendance taking and the scheduling of the facilities
* A digital portfolio system for showcasing performance projects.

**2. For Faculty:**

* Minimized the burden of having to manage attendance through the use of automated systems.
* Positive changes included better ways of communicating with students
* Availability of teaching and learning material as well as facilities booking
* Increased capacity to follow up on students’ performance and supply the required help promptly.

**3. For Administration:**

* Increased efficiency as well as decreased reliance on manual tasks
* Optimized utilization of resources
* Promotion of decision-making skills through data collection and analysis
* Increased conformance to set standards
* Better handling of lone worker emergencies and security surveillance

**4. For the University as a Whole:**

* Higher perception regarding the technological competency in the institution.
* Increased student satisfaction level and enterprise retention rates
* Improved university efficiency and cost containment
* Higher preparedness in managing future challenges in higher learning institutions
* The advantages towards the negative ramifications to data security and privacy.

**5. For Parents:**

* Gaining more information about the academic performance of their child
* The changes brought out enhanced interaction with the university.
* Gained improved assurance and security of the campuses.

**6. For Potential Employers:**

* + Get quick access to detailed student portfolios
  + Improved understanding of the abilities and results of students

**7. For the Higher Education Sector:**

* + The following paper aims to enhance the general perception of university management systems by presenting better benchmarks.
  + Promoting innovativeness and advanced use of technology in learning.
  + Easing students’ behavior and educational performance research

**8. For Society:**

* + In terms of net benefit, many initiatives align with increasing the technical skills of the organization's workforce.
  + Encouraging the rational utilization of instructional materials
  + Promoting the quality of learning in higher education institutions

Regarding the needs of these different stakeholders, the University Management System will conclusively help transform the educational process of the institution and prepare it for possible future developments.

# **5.0 Methodology**

## **5.1 Design**

The University Management System will be modular, easily scaled up and designed to include several key components to form a single, coherent system solution. The design methodology will follow these fundamental principles:

**1. User-Centered Design:**

* + The performance of a series of field studies involving the target user groups including learners, instructors, and officials
  + Working up the user personas and the user journey maps that will inform the design activity.
  + Integration of fluid and adaptive navigational functional that change in form depending on the device type.

**2. Modular Architecture:**

* + The system design is a set of related modules, where every module may implement different functionality
  + To promote tight coupling only between components and modules so that exports and imports between them can be easily updated and other new ones added.
  + The usage of microservices to improve scalability and decrease maintainability.

**3. Data-Driven Approach:**

* + Both are easy to define, but in order to do so, we need to design a good data model that will contain all the necessary information.
  + Acquiring systems and solutions for data warehousing as well as analysis for improved decisions
  + Proper securing of the gathered data for the clients starting from encryption and setting up an appropriate access to it

**4. Integration and Interoperability:**

* + API integrations with other services already in place at the university
  + Compatibility with educational data standard formats and protocols
  + Using webhooks and event-based systems for updates in real-time

**5. Scalability and Performance:**

* + Using the proposed system, it is possible to process a large number of users at the same time
  + Making such changes such as using caching and optimizing databases for better performance
  + Providing design and development for horizontal-scalable applications to adapt for future scale or growth requirements.

**6. Security and Compliance:**

* + Assigning the authorizations of the possible user types to the role-based access control (RBAC)
  + Creating the privacy regulations (such as GDPR FERPA) as a consideration
  + Putting measures in adhering and recording activities

**7. Accessibility and Inclusivity:**

* + WCAG 2.1 guidelines as a basis for designing the user interface
  + Making sure that it is easily accessible with a keyboard

**8. Mobile-First Approach:**

* + The user interface is created with a mobile approach foremost in mind
  + Sticking to the responsive design issue so that it may be readily utilized across different devices is a proper solution.

**9. Extensibility:**

* + Proposal of the plugin structure to enable inclusion of new changes in the functionality of the application
  + Installation of robust API that will enable the third-party developers to develop the system infrastructure

## **5.2 Development Tools**

The development of the University Management System will incorporate the best technology stack, guaranteeing efficiency, extensibility, and sustainability. The following tools and technologies will be employed:

**1. Frontend Development:**

* + Next.js: Server-side rendering and static site generation utilizing React-based framework
  + TypeScript: Typed superset of JavaScript, leading to better quality code
  + Tailwind CSS: Useful and functional CSS kit for a lightning-fast UI creation
  + Redux: Library for complex application states of the state.
  + Jest and React Testing Library: For unit as well as integration testing

**2. Backend Development:**

* + Django: The mature Python web development framework for quick application.
  + Django REST Framework: Toolkit for building Web APIs
  + MySQL: Advanced relational database
  + Redis: A cache and real-time data structure stored in memory
  + Celery: To provide a solution for distributed, asynchronous background jobs, Celery was created.

**3. AI and Machine Learning:**

* + IBM Watson: For the creation of the AI advisor component
  + IBM Cloud storage: This is used to host the Watson instances in this project.
  + NLTK: Natural Language Toolkit for student query processing

**4. DevOps and Deployment:**

* + Docker: Platform for creating and deploying in containers with ease
  + GitLab CI/CD: For CI/CD and DevOps processes that include integration and deployment or any complex, real-world pipeline.

**5. Cloud Infrastructure:**

* + IBM Cloud, Linode or Vercel: Thus, for safe and cost-effective cloud hosting

**6. API Development and Documentation:**

* + Swagger/Open API: For documentation of the API and testing
  + Postman: This is for API usage by different service developers. For API development and testing

**7. Notification Services:**

* + Twilio or Vonage: When the applicant subscribes to receive an SMS or to be updated via the WhatsApp application
  + Resend: For email notifications

**8. Version Control:**

* + Git: Distribution management for versions of the code base
  + GitHub or GitLab: For source code management and cooperation

**9. Database Management:**

* + MySQL: widely used free-of-cost and open-source relational database management system (RDBMS).
  + Redis is a free-to-use, open-source data sharing operating on memory-based database technology.

**10. Code Quality and Linting:**

* + ESLint: For linting in JavaScript as well as in TypeScript
  + Black: For Python code formatting

Through the modern development tools and technologies mentioned above, I can create a structural and sustainable University Management System that embraces high quality, optimum performance, and ease of maintenance.

# **6.0 Review**

The procedures for reviewing the University Management System will also be elaborate and continuous to guarantee optimal functionality throughout the development and installation process. The review methodology will include the following components:

**1. Requirements Review:**

* + Scheduling stakeholders’ meetings to revise and update the characteristics of systems accordingly
  + Applying the matrices of requirements traceability to guarantee that all features are fulfilled
  + Carrying out the activities of gap analysis to determine any gaps within the requirement models for the system.

**2. Design Review:**

* + Engaging with the development team as well as other stakeholders to undertake formal design meetings
  + To capture and discuss specific design decisions in computer systems, an approach to follow specific architecture decision records (ADRs)
  + Considering the opportunities of carrying out security design reviews before the beginning of the primary construction phase

**3. Code Review:**

* + When using code reviews, adopting peer review with the help of pull requests
  + Levin applying the automated code review tools to insist on them following specific coding standards and practices

**4. Testing Review:**

* + By reviewing test plans and test cases, I often ensure complete coverage in the test phase.
  + Often, the result can fail; hold routine test result review meetings to discuss the cause of failure or discrepancy.
  + That is why all potential vulnerabilities must be detected and addressed during security testing reviews.

**5. Performance Review:**

* + Having performance testing performed regularly and going through the results
  + Monitoring system metrics and logs to determine the root cause of either an overloaded or under-performing system
  + Database queries and indexing strategies: modifications and analysis

**6. User Experience Review:**

* + Organizing and administering usability testing sessions among users of different categories
  + Assessing the obtained users' comments and their activity metrics to find weak links
  + Auditing of the DDA compliance and make necessary changes

**7. Security Review:**

* + Security audits and penetration testing should be done occasionally.
  + Checking on who can access what and modify the user roles
  + It requires conducting data privacy impact assessments

**8. Documentation Review:**

* + Regularly revisiting and modifying documentation of the system
  + We want to guarantee that the documentation in an API project is correct and current.

**9. Scalability and Performance Review:**

* + Monitoring the control of the system under different loading conditions
  + Examining the scalability factors and the signs of application constraint
  + The second strategy is reviewing and optimizing the utilization of available resources.

**10. Integration Review:**

* + Check integration interfaces and external systems and API
  + This is achieved because integrated systems always have to contain consistent data.
  + Looking at the looseness of error handling and reliability of integration

**11. Change Management Review:**

* + Examining and endorsing the suggested amendments to the system
  + Measuring the effects of changes regarding feature and result availability
  + The following competency is understanding how changes affect one another and ensuring the team adequately documents them.

**12. User Acceptance Review:**

* + Performing organized checking by the end-users as they are important stakeholders of the system
  + Compare what has been done and what users have said during the UAT
  + This means that the system must also meet all the acceptance criteria

**15. Post-Implementation Review:**

* + It is recommended that the assessment be carried out after the system implementation is complete.
  + How metrics regarding the usages and the feedback from the users and maintainers of the system reveal a complex picture of the institutional reality.
  + Review the opportunities for growth and define further improvement and upgrade strategies

Although this is a general overview of the approach proposed in the dissertation, I am confident that by applying this comprehensive review methodology, the University Management System will fulfil all the identified requirements, remain of high quality, and enhance the quality of the services it provides to the members of the University community.