**ABSTRACT**

         This Android app serves as a pivotal tool in modernizing and optimizing the student experience within educational institutions. By leveraging the widespread usage of smartphones among students, this app provides a centralized platform for accessing essential campus services and information. Its user-friendly interface simplifies tasks such as registering for courses, accessing library resources, and making transactions at campus facilities like the canteen and stationary store. Additionally, the app fosters efficient communication between students and administrators through features like notifications for low balances, upcoming events, and transaction history. Moreover, the incorporation of QR code technology enhances security and streamlines processes, ensuring smooth and convenient interactions. Overall, this Android app represents a progressive approach to campus management, harnessing mobile technology to enhance accessibility, efficiency, and overall satisfaction for students and administrators alike.

INDEX

|  |  |  |
| --- | --- | --- |
| SR.NO | TITLE | PG.NO |
| 1) | INTRODUCTION | 1 |
| 2) | LITERATURE SURVEY | 5 |
| 3) | PROBLEM DEFINITION | 8 |
| 4) | REQUIREMENT ANALYSIS | 11 |
| 5) | PLANNING AND ESTIMATION | 13 |
| 6) | ALGORITHM | 15 |
| 7) | IMPLEMENTATION | 22 |
| 8) | ADVANTAGES & DISADVANTAGES | 27 |
| 9) | FUTURE MODIFICATIONS | 29 |
| 10) | APPLICATION | 31 |
| 11) | BIBLIOGRAPHY | 33 |
| 12) | SCREENSHOTS | 48 |
| 13) | SOURCE CODE |  |

**Chapter 1**

**INTRODUCTION**

**INTRODUCTION**

In today's fast-changing educational world, using technology is crucial for making the academic journey better. That's where creating an Android app comes in. It's a big step toward updating and improving different parts of campus life. Many students use smartphones, so having an app can really make things easier. This app becomes a central hub for students to find important campus services and info. It's designed with students' needs in mind, making it easy to use and helpful for getting things done. For example, students can use it to register for courses, manage library resources, and even make purchases at places like the canteen or bookstore. Plus, it's got cool features like QR codes to make transactions safer and faster. But it's not just about practical stuff. The app also helps students stay in the loop with notifications about their finances, upcoming events, and past transactions. By embracing technology, this app is all about making campus life smoother and more enjoyable for everyone involved.

Moreover, the integration of cutting-edge features, including QR code technology, not only enhances security measures but also expedites transactional processes, resulting in a more streamlined and convenient campus experience. Beyond its functional capabilities, the application fosters a culture of effective communication between students and administrative entities, enabling real-time notifications regarding financial balances, upcoming events, and transactional histories. Embracing mobile technology as a catalyst for progress, this Android application embodies a forward-thinking approach to campus management, poised to elevate accessibility, operational efficiency, and overall satisfaction levels for students and administrators alike.

**Aim of Project**

     The project aims to create an Android app to enhance campus life by leveraging smartphone ubiquity. It provides a centralized platform for accessing campus services, streamlining processes like course registration and library access using intuitive design and QR code technology. Additionally, the app facilitates transparent communication between students and administrators through real-time notifications. Overall, it seeks to boost accessibility, efficiency, and satisfaction for both students and administrative staff.

**Objectives of the Project**

The main objectives of this system are:

1. Simplify student life by offering a user-friendly mobile application.
2. Enhance accessibility to campus services like the canteen, library, and stationary store.
3. Ensure secure and transparent financial transactions using unique QR codes.
4. Enable effective communication between students and administrators.
5. Empower administrators with comprehensive oversight capabilities.
6. Promote student engagement through real-time notifications.
7. Utilize technology to optimize campus management processes.
8. Foster a sense of community and collaboration among students and administrators.

**Scope of the Project**

This project will have everything students need in one place, like signing up, logging in, and sections for important places on campus like the canteen, library, and store where you can buy school supplies. With this app, students can easily do things like checking their account balance or borrowing books. It'll also make paying for things safe and easy using special codes. For the people who run the school, there will be a special part of the app where they can keep track of student information and transactions. Students will also get updates on their phones about things like low balances or upcoming events. Overall, the project is about making campus life easier and more connected for students and staff.

**Methodology**

The project will follow an iterative and collaborative methodology, beginning with thorough requirements gathering and analysis to ensure a clear understanding of user needs. This will be followed by the design phase, where user interfaces and system architecture will be planned. Development will proceed incrementally, with regular testing and feedback cycles to ensure functionality and usability meet expectations. The Android platform will be utilized for the mobile application, while Flask will serve as the backend framework, facilitating communication between the app and the MySQL database for data storage. Continuous integration and version control will be employed to manage code changes efficiently. Throughout the project, close communication between developers and stakeholders will be maintained to address any issues and incorporate necessary adjustments. Finally, thorough testing and deployment procedures will ensure a stable and reliable product ready for use by students and administrators alike.

**Chapter 2**

**LITERATURE SURVEY**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Title** | **Authors** | **Advantages** | **Disadvantages** | **Result** |
| **Mobile-Based Campus Services** | **Sayali Bhagat, Shamshoddin Mujawar, Pranati Salunke, Sayali Kokane** | **Convenience of accessing campus services through a mobile app** | **Dependency on smartphones for access** | **Improved student experience and efficiency in service access** |
| **QR Code Payment System** | **Rajesh Kannan Megalingam, Souraj Vishnu, Swathi Sekhar, Vishnu Sasikumar, Sreekumar S and Thejus R Nair** | **Quick and easy payment process using QR codes** | **Requires QR code scanning capability** | **Efficient and secure transaction processing** |
| **Automated Library Management** | **Adarsh Borkar, Madhura Ansingkar, Monali Khobragade, Pooja Nashikkar, Arti Raut** | **Simplified book issuance and return process** | **Potential technical issues with automated notifications** | **Improved library service delivery and book tracking** |
| **Micropayment Integration -** | **Alfred. P. F** | **- Real-time updates of student account balances** | **Possible security vulnerabilities in micropayment system** | **Streamlined financial transactions and balance monitoring** |
| **Centralized Admin Dashboard** | **Veronika LANG, Peter SITTLER** | **Efficient management of student data and transactions** | **Complexity in implementation and maintenance** | **Enhanced administrative control and oversight over campus services** |
| **Event Participation through QR** | **Veronika LANG, Peter SITTLER** | **Seamless event registration process using QR codes** | **Dependency on QR code technology for event entry** | **Enhanced participation and management of cultural events** |

**Chapter 3**

**PROBLEM**

**DEFINITION**

**Problem Statement**

In today's educational institutions, students often face challenges in accessing campus services efficiently and staying updated with important information. Additionally, administrators may encounter difficulties in managing student data and facilitating seamless transactions. To address these issues, there is a need for a user-friendly mobile application that can provide students with easy access to campus services such as canteen, library, and stationary shop, while also enabling administrators to efficiently oversee student activities and transactions. This project aims to develop such an application to enhance the overall student experience and streamline administrative tasks.

**Existing System**

The existing system for managing student activities and campus services typically relies on manual processes and traditional communication methods. In this system, students often have to physically visit different locations on campus to access services like the canteen, library, or stationary shop. For example, to buy food from the canteen or borrow a book from the library, students have to go there in person and interact with the staff. Similarly, administrators handle student data and transactions manually, which can be time-consuming and prone to errors. Communication between students and administrators also tends to be less efficient, relying on methods like email or in-person meetings. Moreover, there may be challenges in tracking student activities and managing resources effectively. For instance, students may forget to return library books on time, leading to late fees or loss of materials. Additionally, administrators may struggle to keep track of student balances for services like the canteen or stationary shop.

Overall, the existing system lacks efficiency, convenience, and real-time communication. Students face obstacles in accessing campus services promptly, while administrators may find it challenging to manage student activities and resources efficiently. As a result, there is a need for a more streamlined and technologically advanced system that can address these shortcomings and enhance the overall student experience.

**Disadvantages of Existing System:**

1. Manual processes are time-consuming and prone to errors.
2. Limited accessibility to campus services leads to inconvenience for students.
3. Communication between students and administrators is often delayed or inefficient.
4. Difficulty in tracking student activities such as library book returns or canteen transactions.
5. Security risks associated with handling data manually.
6. Lack of real-time updates.

**Proposed System**

The proposed system of the project aims to revolutionize the way students interact with campus services and administrators, leveraging modern technology to enhance convenience, efficiency, and communication. In contrast to the existing manual processes, the proposed system introduces a comprehensive mobile application with distinct modules for students and administrators.

For students, the system starts with a streamlined registration process where they provide essential details like their name, branch, year, and unique college ID. Once registered, students can log in securely using their college ID and password. The application offers a range of services tailored to student needs, including access to the canteen menu with prices and the ability to make purchases by scanning a unique QR code, ensuring seamless and cashless transactions. Similarly, the library module allows students to browse available books, borrow them, and receive notifications about due dates and late penalties if applicable. Additionally, a stationary module enables students to view and purchase college-related items through QR code scanning. The application also offers features like checking remaining balances, viewing transaction history, and tracking borrowed library books with due dates for returns. Furthermore, a contact form allows students to send queries directly to the admin email for prompt assistance.

On the administrative side, the system empowers administrators with comprehensive tools for managing student data and transactions efficiently. Administrators have access to student histories and details sorted by branch and year, with a convenient search bar for quick access to specific student information. They can update student accounts with micropayments received and assign unique QR codes to each student upon registration. Moreover, administrators receive notifications about students with low balances and can send timely reminders to students before college exams. The system records all transactions, including those related to the canteen, library, and stationary, providing administrators with valuable insights and oversight. Admins also have the capability to update information related to the canteen menu, books list, and stationary items list.

Additionally, the proposed system incorporates a cultural event section where administrators can update and add upcoming events, while students can participate by scanning QR codes for event entry. Notifications play a crucial role in keeping students informed about their transaction history and low balances, ensuring transparency and proactive management.

Overall, the proposed system promises to streamline campus services, facilitate communication, and enhance the overall student experience by leveraging Android technology, Flask for backend development, and MySQL for database management.

**Advantages of Proposed System:**

1. Streamlined registration and login process for students.
2. Convenient access to campus services like the canteen, library, and stationary shop.
3. Cashless transactions using QR code scanning, enhancing security and convenience.
4. Timely notifications for borrowed library books, late penalties, and low balances.
5. Easy tracking of remaining balances and transaction history for students.
6. Efficient management of student data and transactions for administrators.

.

**Chapter 4**

**HARDWARE & SOFTWARE REQUIREMENT**

**Hardware and Software requirements**

**Hardware:**

1. Processor: Intel Core i3 or more.

2. RAM: 4GB or more.

3. Hard disk: 250 GB or more.

**Software:**

1. Operating System : Windows 10, 7, 8.

2. Android studio

3. Python, flask.

**Technologies Used:**

**Android:**

Android provides a rich application framework that allows you to build innovative apps and games for mobile devices in a Java language environment. The documents listed in the left navigation provide details about how to build apps using Android's various APIs. Android apps are built as a combination of distinct components that can be invoked individually. For instance, an individual activity provides a single screen for a user interface, and a service independently performs work in the background.

**Python:**

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

Often, programmers fall in love with Python because of the increased productivity it provides. Since there is no compilation step, the edit-test-debug cycle is incredibly fast. Debugging Python programs is easy: a bug or bad input will never cause a segmentation fault. Instead, when the interpreter discovers an error, it raises an exception. When the program doesn't catch the exception, the interpreter prints a stack trace. A source level debugger allows inspection of local and global variables, evaluation of arbitrary expressions, setting breakpoints, stepping through the code a line at a time, and so on. The debugger is written in Python itself, testifying to Python's introspective power. On the other hand, often the quickest way to debug a program is to add a few print statements to the source: the fast edit-test-debug cycle makes this simple approach very effective.

**FLASK**:

A Flask is a Web Application Framework that is built with Flexibility and Speed In the Mind. Flask is Built in Python , which many data Scientists are familiar with . Flask takes care of the Environment and Project setup involved in web Applications Allowing the Developer to focus on their application rather than thinking about HTTP , routing , dataset etc. Flask allows Data Scientists to create simple Single page Applications and one should Help or look into if they want to create Products for Consumers Flask is a micro web framework written in Python. It is classified as a microframework because it doesn't require particular tools or libraries. its no database abstraction layer, form validation, or the other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions which will add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and a number of other common framework related tools Flask was created by Armin Ronacher of Pocoo, a world group of Python enthusiasts formed in 2004.According to Ronacher, the thought was originally an April Fools joke that was popular enough to form into significant application. When Ronacher and Georg Brandl created a bulletin board system written in Python, the Pocoo projects Werkzeug and Jinja were developed. Flask has become popular among Python enthusiasts. As of October 2020, its second most stars on GitHub among Python web-development frameworks, only slightly behind Django, and was voted the foremost popular web framework within the Python Developers Survey 2018.

These are some Important features of the Flask:

1. it is a Development Server

2. Debugger

3. RESTful request dispatching

4. Unicode Based

5. Flask have google app engine Compatibility

**Chapter 5**

**PLANNING AND ESTIMATION**

**Software development Life Cycle**

The entire project spanned for a duration of 6 months. In order to effectively design and develop a cost-effective model the Waterfall model was practiced.

**Requirement gathering and Analysis phase:**

This phase started at the beginning of our project, we had formed groups and modularized the project. Important points of consideration were

1. Define and visualize all the objectives clearly.

2.Gather requirements and evaluate them

Consider the technical requirements needed and then collect technical specifications of various peripheral components (Hardware) required.

3. Analyze the coding languages needed for the project.

4. Define coding strategies.

5. Analyze future risks / problems.

6. Define strategies to avoid these risks else define alternate solutions to these risks.

7. Check financial feasibility.

8. Define Gantt charts and assign time span for each phase.

By studying the project extensively we developed a Gantt chart to track and schedule the project. Below is the Gantt chart of our project.

**Timeline**

**Please make changes as per your requirement**

**Cost Estimation**

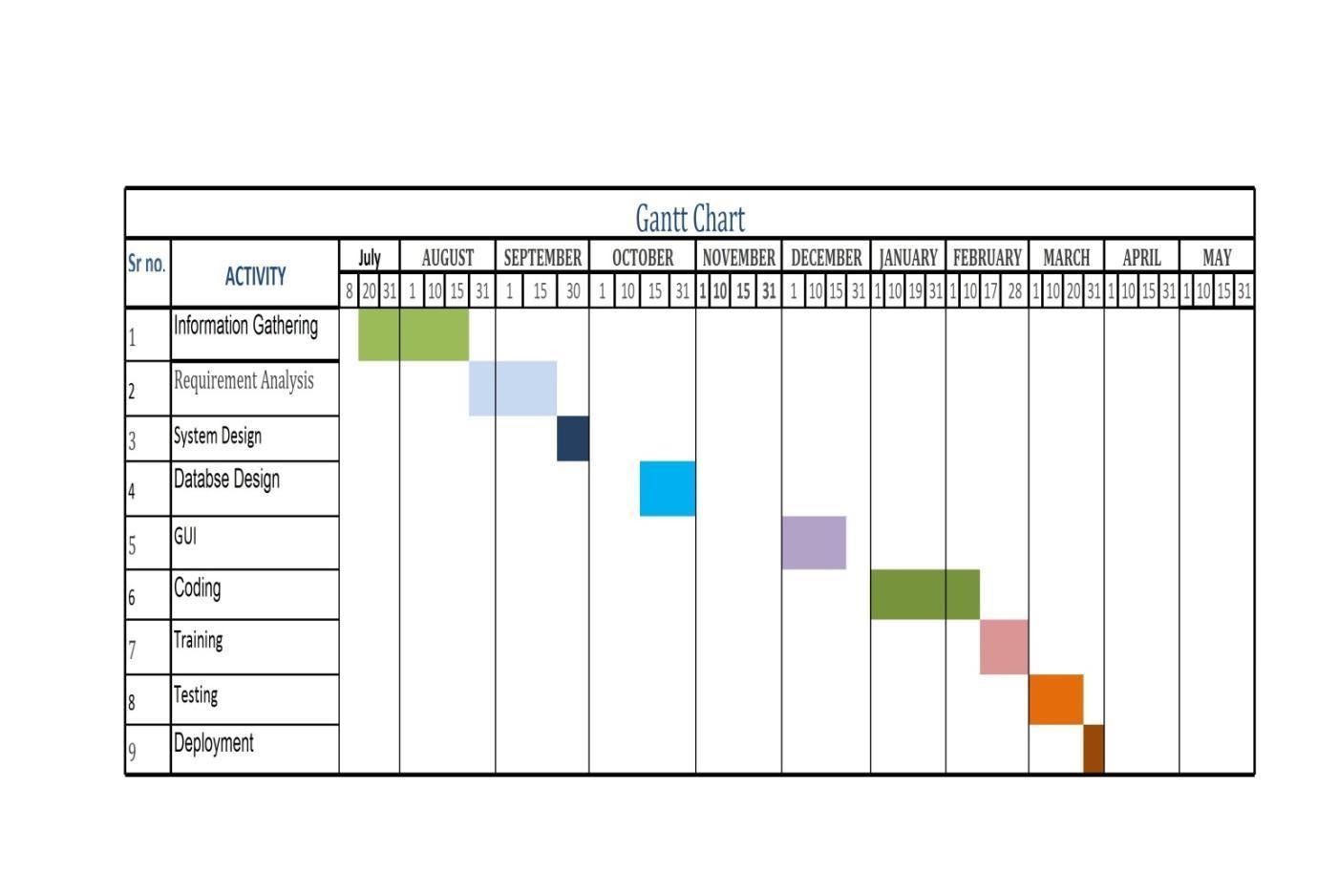
Cost estimation is done using cocomo model

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Cost Drivers | **Ratings** | | | | | |
| Very Low | Low | Nominal | High | Very High | Extra High |
| **Product attributes** |  |  |  |  |  |  |
| Required software reliability | 0.75 | 0.88 | 1.00 | 1.15 | 1.40 |  |
| Size of application database |  | 0.94 | 1.00 | 1.08 | 1.16 |  |
| Complexity of the product | 0.70 | 0.85 | 1.00 | 1.15 | 1.30 | 1.65 |
| **Hardware attributes** |  |  |  |  |  |  |
| Run-time performance constraints |  |  | 1.00 | 1.11 | 1.30 | 1.66 |
| Memory constraints |  |  | 1.00 | 1.06 | 1.21 | 1.56 |
| Volatility of the virtual machine environment |  | 0.87 | 1.00 | 1.15 | 1.30 |  |
| Required turnabout time |  | 0.87 | 1.00 | 1.07 | 1.15 |  |
| **Personal attributes** |  |  |  |  |  |  |
| Analyst capability | 1.46 | 1.19 | 1.00 | 0.86 | 0.71 |  |
| Applications experience | 1.29 | 1.13 | 1.00 | 0.91 | 0.82 |  |
| Software engineer capability | 1.42 | 1.17 | 1.00 | 0.86 | 0.70 |  |
| Virtual machine experience | 1.21 | 1.10 | 1.00 | 0.90 |  |  |
| Programming language experience | 1.14 | 1.07 | 1.00 | 0.95 |  |  |
| **Project attributes** |  |  |  |  |  |  |
| **Use of software tools** | **1.24** | **1.10** | **1.00** | **0.91** | **0.82** |  |
| Application of software engineering methods | 1.24 | 1.10 | 1.00 | 0.91 | 0.83 |  |
| Required development schedule | 1.23 | 1.08 | 1.00 | 1.04 | 1.10 |  |

The Intermediate Cocomo formula now takes the form:

**E=*ai*(kloc)*(bi)*.EAF**

       Using above calculation we found that the total time period of the project is around 6 months, the per month cost comes out to be Rs.12, 000/- so the total comes to be Rs.72, 000/-



**Requirement gathering and Analysis phase:**

This phase started at the beginning of our project, we had formed groups and modularized the project. Important points of consideration were

1. Define and visualize all the objectives clearly.

2.Gather requirements and evaluate them

Consider the technical requirements needed and then collect technical specifications of various peripheral components (Hardware) required.

3. Analyze the coding languages needed for the project.

4. Define coding strategies.

5. Analyze future risks / problems.

6. Define strategies to avoid these risks else define alternate solutions to these risks.

7. Check financial feasibility.

8. Define Gantt charts and assign time span for each phase.

Testing:

|  |  |  |  |
| --- | --- | --- | --- |
| Precondition | Test | Expected Result | Result |
| Registration Page Loaded | Fill in valid registration details | User successfully registers an account | PASS |
| Login Page Loaded | Enter username and password | Successful login | PASS |
| Canteen Menu Loaded | Select food item and scan QR code | Payment deducted from student account, notification of deduction | PASS |
| Library Book List Loaded | Borrow a book | Notification of successful book issuance with due date displayed | PASS |
| Stationary List Loaded | Scan QR code to purchase item | Notification of successful purchase and deduction from account | PASS |
| Student Dashboard Loaded | View transaction history | List of transactions with date and time displayed | PASS |
| Contact Form Loaded | Submit query | Notification of query submission sent to admin email | PASS |
| Admin Dashboard Loaded | Search student details by branch and year | List of students meeting search criteria displayed | PASS |
| Cultural Event Section Loaded | Add new cultural event | Event added and displayed in the list of upcoming events | PASS |

**FEASIBILITY STUDY**

         This system is possible for all health care department like science lab hospital and clinic etc and this method can use while not specialists in this field anyone can use who have data concerning using online services which is able to facilitate to use this method any generation folks can use this method in laptop

**TECHNICAL FEASIBILITY**

The framework ought to be assessed from the specialized reason for read first the evaluation of this practicability ought to be upheld a rundown kind of the framework interest inside the provisions of info yield projects and techniques having known an outline framework the examination ought to keep up to suggest the kind of pack required approach building up the framework of running the framework whenever it has been planned

* Technical issues raised during the investigation are:
* Is the existing technology sufficient for the suggested one?
* Can the system expand if developed?

the undertaking should be created indicated the predetermined capacities and execution are accomplished among the limitations the task is created among most recent innovation through the innovation may become old once some measure of some time due to the specific undeniable truth that never form of same code upholds more seasoned variants the framework should in any case be utilized hence there are marginal imperatives included this task the framework has been created exploitation java the undertaking is in fact feasible for advancement

**ECONOMIC FEASIBILITY**

The creating framework ought to be even by worth and benefit. Measures to confirm that exertion is focused on a project, which may give best, come at the most punctual. one through and through the variables that affect the occasion of a new framework, is that the value it'd need. The following are assortment of the necessary cash questions asked all through starter examination:

* The costs conduct a full system investigation.
* The cost of the hardware and software.
* The benefits in the form of reduced costs or fewer costly errors.

     Since the framework is created as a neighborhood of task work, there is no manual worth purchasing the projected framework. Furthermore every one of the assets are as of now available, it offers an image of the framework is financially feasible for improvement.

**BEHAVIORAL FEASIBILITY**

This incorporates the following inquiries:

* Is there agreeable help for the clients?
* Will the arranged framework cause hurt?

The venture would be useful as an aftereffect of fulfilling the goals once created and introduced. All social perspectives are considered cautiously and presume that the undertaking is typically conceivable.

**RISK ANALYSIS PROCESS**

       Notwithstanding the obstacle strategies utilized potential perils is in a position to which can arise inside or outside the affiliation ought to be assessed regardless of the established truth that the exact arrangement of expected catastrophes or their after results district unit delayed to outlined its valuable to play out an intensive risk investigation of all threats which can sensibly happen to the relationship in spite of the kind of peril the goals of business recuperating emerging with locale unit to validate the security of buyers workers and particular representatives eventually of and following a breakdown the overall probability of a failure happening should be settled things to appear at in urgent the probability of a particular breakdown should be constrained to represent in any case not be confined to field characteristic study of the planet closeness to indispensable wellsprings of power streams and air terminals level of receptiveness to workplaces inside the affiliation history of local service organizations in giving persistent kinds of help history of the spaces condition to standard risks neighborhood to imperative turnpikes that vehicle bold waste and combustible item. Potential openings could even be delegated regular, specialized, or human dangers. Models include:

* **Characteristic** Threats: inner flooding, outer flooding, interior hearth, outside chimney, seismic movement, high breezes, snow and ice storms, emission, cyclone, typhoon, pandemic, torrent , hurricane.
* **Specialized Threats:** power disappointment/variance, warming, ventilation or air con disappointment, glitch or disappointment of hardware , disappointment of framework code, disappointment of use code, broadcast communications disappointment, gas spills, interchanges disappointment, atomic aftermath.
* **Human Threats:** robbery, bomb dangers, theft, blackmail, thievery, defacing, psychological warfare, common problem, synthetic spill, damage, blast, war, natural pollution, radiation tainting, perilous waste, vehicle crash, airdrome nearness, strike (Internal/External), PC wrongdoing.

All areas and offices should be encased inside the peril investigation maybe than attempting to sort out real prospects of every fiasco an overall relative game plan of high medium and low is utilized at first to distinguish the probability of the danger happening the possibility investigation also need to affirm the effect of such a likely danger on various capacities or offices inside the association a risk analysis type discovered here pdf format will work with the strategy the capacities or divisions can shift by kind of association the arranging strategy ought to set up and live the possibility of every single expected danger and in this way the effect on the association if that danger happened to attempt to this each division should be investigated severally in spite of the fact that the chief framework is furthermore the one most serious danger it isn't the solitary vital concern indeed even inside the first programmed associations a few offices will not be handled or programmed inside the smallest degree in totally programmed divisions essential records stay outside the framework as lawful records pc information programming bundle hang on diskettes or supporting documentation for data section the effect is evaluated as 0 no effect or break in tasks 1 noticeable effect break in activities for as long as eight hours 2 mischief to instrumentation and additionally offices break in tasks for eight 48 hours 3 major damage to the instrumentation or potentially offices break in tasks for very 48 hours all base camp or potentially pc focus capacities ought to be resettled bound suspicions is also important to consistently apply evaluations to every possible danger

Following are run of the mill suspicions which can be utilized all through the peril evaluation measure:

1. In spite of the fact that affect evaluations may fluctuate somewhere in the range of one and three for any office given a particular situation, appraisals applied should reflect expected, apparent or anticipated effect on each space.

2. each potential danger ought to be thought to be "confined" to the force being appraised.

3. Despite the fact that one potential danger could lead on to an uncommon likely danger (e.g., a typhoon may bring forth cyclones), no aftereffect ought to be expected.

4. On the off chance that the consequences of the danger wouldn't warrant development to Associate in Nursing substitute site(s), the effect ought to be appraised no over a "2."

5. The threat evaluation should be performed by the force . to gauge the likely dangers, a weighted reason rating framework is utilized .

**Functional requirement**

1. Users should be able to log in using their college ID and password.
2. Provide a section for users to send queries to admin email.
3. Allow admin access to student history and details sorted by branch and year.
4. Assign unique QR codes to students upon registration
5. Allow admin to update canteen menu, book list, and stationary items list.

**Non-functional requirement**

1. The system should be able to handle a growing number of users and transactions without significant degradation in performance.
2. Mobile apps should be connected with the internet.

**REQUIREMENT ANALYSIS:**

To start the gathering needs, it's far first important to discover every organization suffering from the challenge and recognize the wishes of everyone. With that facts in hand, an initial listing of required operational and non-operational necessities (see sections Requirements and non-functional Requirements) may be submitted to Product Backlog within side the form of user issues. Each time those needs are changed, it effects on this phase most effective defining the final requirements which might be a part of the Product Ratio after the project.

**Design part:**

The design of the project focuses on creating an intuitive and user-friendly interface for both students and administrators. The user interface prioritizes simplicity and clarity, with easy navigation and visually appealing layouts. For students, the design emphasizes quick access to essential features like canteen menus, library book lists, and account management options. Administrators have access to a comprehensive dashboard that provides a clear overview of student data and transaction records, with tools for efficient management and communication. Overall, the design aims to enhance the user experience by providing a seamless and visually pleasing platform for accessing campus services and managing administrative tasks.

**DFD**

A data flow diagram (DFD) is a graphical representation of the flow of data through an information system. A data flow diagram can also be used for the visualization of data processing (structured design). It is common practice for a designer to draw a context-level DFD first which shows the interaction between the system and outside entities. This context-level DFD is then exploded to show more detail of the system being modeled.

**Symbols:**

**The four components of a data flow diagram (DFD) are:**

* External Entities/Terminators are outside of the system being modeled. Terminators represent where information comes from and where it goes. In designing a system, we have no idea about what these terminators do or how they do it.
* Processes modify the inputs in the process of generating the outputs
* Data Stores represent a place in the process where data comes to rest. A DFD does not say anything about the relative timing of the processes, so a data store might be a place to accumulate data over a year for the annual accounting process.

**Chapter 6**

**Design & Implementation**

**E-R Diagram:**

**Class diagram:**

**DFD-1:**

**Activity Diagram :**

**Flow chart :-**

**Use Case Diagram :-**

**Chapter 7**

**Advantages**

**Advantages:**

1. The app allows Users to access campus services and manage their accounts conveniently
2. Streamlined processes such as canteen payments and library book borrowing save time and effort.
3. Users can easily track their transactions, balances, and borrowed items, enhancing transparency.
4. Users receive notifications for various activities like book issuances, due dates, and low balances, ensuring timely action.
5. The app facilitates communication between students and administrators, allowing queries to be sent directly to admin email.
6. Administrators can efficiently manage student data, transactions, and resources, enhancing administrative oversight.

**Chapter 08**

**FUTURE MODIFICATIONS**

**&**

**CONCLUSION**

**Future Modification**

In the future, the project could be enhanced by incorporating features such as integration with digital payment platforms for broader payment options, implementation of artificial intelligence for personalized notifications and recommendations, and expansion to include additional campus services like transportation or health facilities. Moreover, enhancing the app's user interface and accessibility features would cater to a wider range of users. Additionally, exploring the integration of emerging technologies like blockchain for enhanced security and transparency could be considered for future modifications, ensuring the app remains relevant and effective in meeting the evolving needs of students and administrators.

**Conclusion**

In conclusion, the project offers a comprehensive solution to streamline campus services and improve communication between students and administrators. By providing convenient access to essential services like the canteen, library, and stationary shop, the app enhances efficiency and transparency in managing student activities. With features such as timely notifications and effective account management, users can stay informed and engaged in their academic endeavors. The project lays a strong foundation for future enhancements and demonstrates the potential of technology to enhance the overall student experience in educational institutions.

**Chapter 9**

**BIBLIOGRAPHY**

**References**

1. K. V. Thangam, T. S. Kumar, V. Yogesh, and S. Prabhu, “Android Application for College Management System (M-Insproplus),” Int. J. Mod. Trends Eng. Res., vol. 4, no. 2, pp. 41–44, 2017.
2. M. N. Dedhia and D. V. C. Kotak, “ANDROID BASED CAMPUS SOLUTION FOR COLLEGE,” Int. J. Comput. Sci. Mob. Comput., vol. 6, no. 11, pp. 12–17,2017.
3. C. Science, C. Science, C. Science, and C. Science, “ANDROID APPLICATION ON COLLEGE,” Int. J. Emerg. Technol. Comput. Sci. Electron., vol. 14, no. 2, pp. 811–812, 2015.
4. K. Datarkar, N. Hajare, N. Fulzele, S. Kawle, V. Suryavanshi, and D. Radke, “Online College Management System,” Int. J. Comput. Sci. Mob. Comput., vol. 5, no. 4, pp. 118–122, 2016.
5. P. G. Scholar and E. Engineering, “DEVELOPING AN ANDROID APPLICATION FOR COLLEGE MANAGEMENT SYSTEM”.

**Chapter 10**

**SCREENSHOTS**

**Chapter 11**

**SOURCE CODE**