

MIS 6308

Project Report

INTELLIGENT RFID SYSTEM FOR MAINTENANCE MANAGEMENT



Group 8

Desai, Ekta Abhishek

Garcia, Victoria Elizabeth

Gondi, Dedeepya Sai

Jadhav, Sanika Ashok

Jamma, Aishwarya Atul

Nagraj Amrutha

Saw, Rupesh Umesh

1. Abstract.....	2
2. Objective of the Project	2
3. Problem Statement.....	3
4. Scope	3
4.1 Business Process Scope.....	3
4.2 Organization Scope	3
4.3 Location Scope	3
4.4 Application Scope	3
4.5 Data Scope	4
4.6 Technology Scope	4
5. Out of Scope	4
6. Functional Specifications for the proposed system	4
6.1 Components Used.....	4
6.1.1 RFID RC522.....	4
6.1.2 Raspberry Pi 3B+	6
6.1.3 NODEMCU.....	7
6.2 Software Used.....	9
6.2.1 Arduino IDE	9
7. Proposed Solution.....	11
8. Project Activities	12
9. Deliverables	13
10. Project Management	13
11. Project Organization	17
12. Roles and Responsibilities.....	17
13. Project Work Plan (WBS, Timeframes)	18
14. Business PROCESS MODEL Notation.....	19
15. Swimlane Diagram	21
16. System Context Diagram.....	21
17. USE CASE DIAGRAM	22
18. DATA DICTIONARY	23
19. Activity Diagram.....	23
20. Class Diagram.....	24
21. Project Timeline	24

1. Abstract

With the growth of industries and production in recent years, there has been a need to maintain and routinely monitor the machineries and heavy instruments in use. This is critical for the workers' safety as well as to avoid a large loss. Ignorance of the maintenance cycle may also result in license cancellation or a heavy fee, which will not only cause major trust issues but also result in the loss of employment for personnel in that field. To solve this challenge, we devised a novel technique that involves RFID tags. For this, we used a Raspberry Pi and an Android application to build two systems. It will begin by launching the program, which will allow only authorized employees to log in with their login Id and password. The Raspberry Pi unit/Android application will scan RFID tags attached to machines and equipment, and the information will be stored on a server.

Following that, the employee will be directed to a website with numerous actions to complete. The employee's modifications or updates will be reflected on the server, resulting in one service cycle. This service cycle is unique to each piece of equipment and can be tailored to the needs of the client. The final step is to log out of the software, after which the progress will be kept in a database and a report mailed to the designated incharge/manager. Only the appointed incharge has access to the system, which includes adding employees, deleting services, adding or upgrading services, and so on.

We not only make the data flow more efficient, but we also make it more secure and dependable by applying these methods. This will reduce the number of mistakes made when registering for equipment. All of these factors combine to make it a one-stop-shop for all industrial needs.

2. Objective of the Project

By 2050, the world's population is predicted to exceed ten billion people. This will put a great strain on manufacturing companies, increasing the risk to those who work there.

Furthermore, it is already challenging to locate skilled and motivated personnel, as well as to maintain manufacturing security. To fill this critical gap, we devised a system that tracks the present state of machines.

The old approach is unreliable and authentic, and many environmental conditions such as dampness, paper deterioration, record misplacement, and so on can put industry in jeopardy.

This error can result in severe fines, license cancellation, industrial distrust, and mishaps with persons working there, among other things.

Most appliances have a manual servicing record, which is prone to human errors such as untimed servicing, irregularly managing service records, and so on, which can lead to disaster.

So, a smart system is necessary to manage both timely servicing and retaining records to avoid these types of errors.

3. Problem Statement

Mismanagement and a lack of concern for the upkeep of industrial equipment and heavy gear could result in significant property damage and perhaps the death of workers in that industry.

The project must have key features such as data retrieval, data storage, reminders for forthcoming maintenance, missed maintenance, and authorized management.

4. Scope

4.1 Business Process Scope

- The business scope of the project is to eliminate human errors by developing an intelligent system that allows organizations to address customer services immediately upon request to avoid huge losses and ensure the safety of workers.
- The project aims in saving cost and time by implementing service cycles in the most unique, efficient, and reliable way.

4.2 Organization Scope

- Most organizations use a lot of guesswork to predict machinery failures due to inaccurate resources. Automation offers solutions to existing problems and provides the tools needed to track every machine. This helps organizations predict future outcomes and helps improve forecasting.
- The project also helps organizations to use real-time data with minimal human error to eliminate inefficiencies and repurpose resources.
- In any industry, Automation can initiate sales, attract new customers, new opportunities, and can further generate more profits.
- Greater visibility in organizations can lead stakeholders to make better decisions based on accurate data.

4.3 Location Scope

- This project can be applied to various industries, such as manufacturing, industrial, and medical organizations where the age and efficiency of machinery and its parts is imperative to the success of the business and its customers.

4.4 Application Scope

- Each data table will include its own unique name and ID to prevent misuse and protect data integrity.
- Users will have full administrative access to change the identifies and names of items (machinery/parts) to customize to business needs. Updates will automatically be represented in the SQL database.
- There will be cross-checks to ensure integrity within the data (no duplicates, etc.)

4.5 Data Scope

- Implementation of an SQL database to collect historical data for future alerts, processes, and analysis of specific machinery and parts within the organization.
- The main objective for this data will be to read and analyze whether servicing is required.

4.6 Technology Scope

- RFID Tags - ISO 14443A standard tags
- RFID Tag Reader - RFID RC522
- Raspberry Pi - 3b+
- Android Phone for providing the front-end of the application -- This will provide a UI for Adding an employee, deleting a service, adding a service, updating a service, etc.
- Database System -- MySQL will be the main DBMS utilized.
- Python programming -- Should be easily usable and updatable to the newest version of Python v3.
- Android App Development -- Basic interface style in order to test processes. It will utilize native functions and libraries from Android Studio.

5. Out of Scope

- Implementation of a front-end framework for a seamless and user-friendly interface for the Android application (React, etc.)
- Trainings, tutorials, etc. will not be taught nor implemented at this stage for users.
- Manipulation of data past send and receive. For example, the ability to utilize API calls to send and set appointments.
- Authentication for security purposes will be considered later based on budget and user/audience. A simple SSO or two-factor authentication will most likely be implemented.
- Cross-platform with iOS is out of scope.

6. Functional Specifications for the proposed system

6.1 Components Used

6.1.1 RFID RC522

One of the most affordable RFID alternatives is the NXP RC522 RFID module, which is based on the MFRC522 IC. The RC522 RFID Reader module is designed to communicate with RFID tags by producing a 13.56MHz electromagnetic field (ISO 14443A standard tags). A 4-pin Serial Peripheral Interface (SPI) with a maximum data rate of 10Mbps allows the reader to interface with a microcontroller. The I2C and UART protocols are also supported. An interrupt pin is provided with the module. Instead of repeatedly asking the RFID module, "Is there a card in view yet?" it will notify us when a tag approaches.

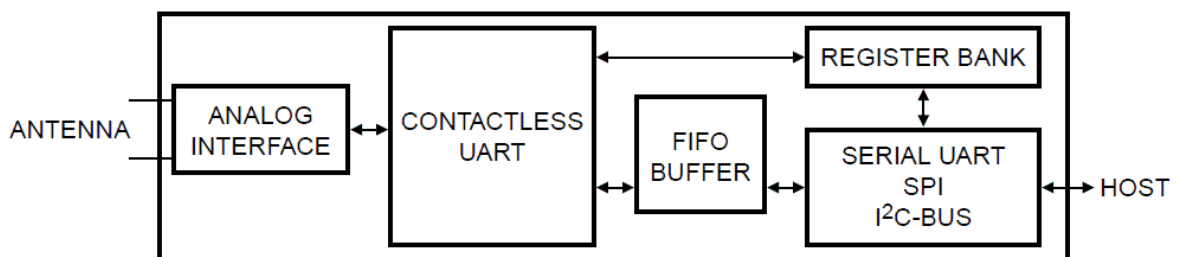
The module's operational voltage ranges from 2.5 to 3.3V, but the good news is that the logic pins are 5-volt tolerant, allowing us to connect it directly to an Arduino or other 5V logic microcontroller without the need for a logic level converter.

RC522 Features

- 13.56MHz RFID module
- Operating voltage: 2.5V to 3.3V
- Communication: SPI, I2C protocol, UART
- Maximum Data Rate: 10Mbps
- Read Range: 5cm
- Current Consumption: 13-26mA
- Power-down mode consumption: 10uA (min)

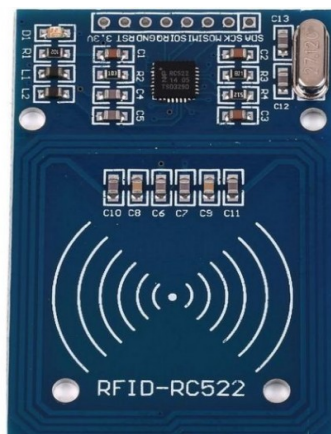
Block Diagram

- The Analog interface handles the modulation and demodulation of the analog signals.
- The contactless UART handles the protocol requirements for the communication schemes in co-operation with the host. The comfortable FIFO buffer allows a fast and convenient data transfer from the host to the contactless UART and vice versa.
- Various host interfaces are implemented to fulfil different customer requirements.



Block Diagram of RC522 Reader

Image of RFID RC522 Reader



6.1.2 Raspberry Pi 3B+

The Raspberry Pi 3 Model B is the Raspberry Pi's third iteration. This powerful credit-card-sized single-board computer succeeds the original Raspberry Pi Model B+ and Raspberry Pi 2 Model B and can be utilized for a variety of applications. The Raspberry Pi 3 Model B maintains the popular board format while providing a 10x faster processor than the original generation Raspberry Pi. It also includes wireless LAN and Bluetooth connectivity, making it an excellent choice for linked designs. The RASPBERRY PI 3 is a PI series development board. It can be thought of as a single-board computer that runs the LINUX operating system.

The board not only offers a lot of functionality but also has a fast processor, making it ideal for complex applications. The PI board was created with hobbyists and engineers in mind who are interested in LINUX systems and IoT. (Internet of Things).

The Raspberry Pi 3 Model B+ is the most recent addition to the Raspberry Pi 3 family, with a 64-bit quad-core processor running at 1.4GHz, dual-band 2.4GHz, and 5GHz wireless LAN, Bluetooth 4.2/BLE, faster Ethernet, and PoE functionality through a separate PoE HAT.

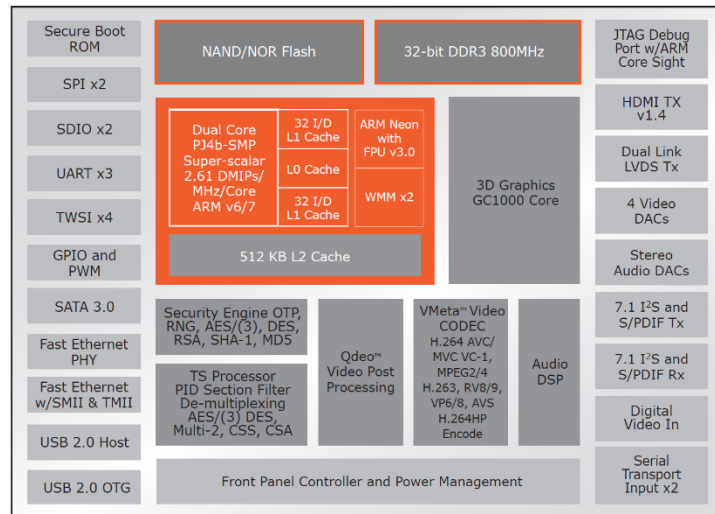
The dual-band wireless LAN has modular compliance certification, which allows the board to be incorporated into end devices with much reduced wireless LAN compliance testing, reducing both cost and time to market.

Both the Raspberry Pi 2 Model B and the Raspberry Pi 3 Model B have an identical mechanical footprint.

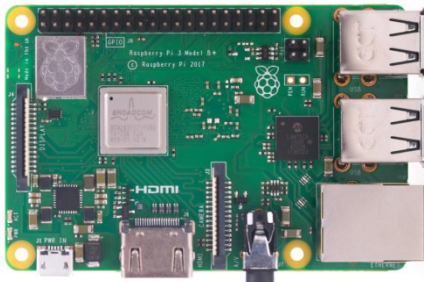
Features

- Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC @ 1.4GHz
- 1GB LPDDR2 SDRAM
- 2.4GHz and 5GHz IEEE 802.11.b/g/n/ac wireless LAN, Bluetooth 4.2, BLE
- Gigabit Ethernet over USB 2.0 (maximum throughput 300 Mbps)
- Extended 40-pin GPIO header
- Full-size HDMI
- 4 USB 2.0 ports
- CSI camera port for connecting a Raspberry Pi camera
- DSI display port for connecting a Raspberry Pi touchscreen display
- 4-pole stereo output and composite video port
- Micro SD port for loading your operating system and storing data
- 5V/2.5A DC power input
- Power-over-Ethernet (PoE) support (requires separate PoE HAT)

Block Diagram



Images of Raspberry Pi 3B+



6.1.3 NODEMCU

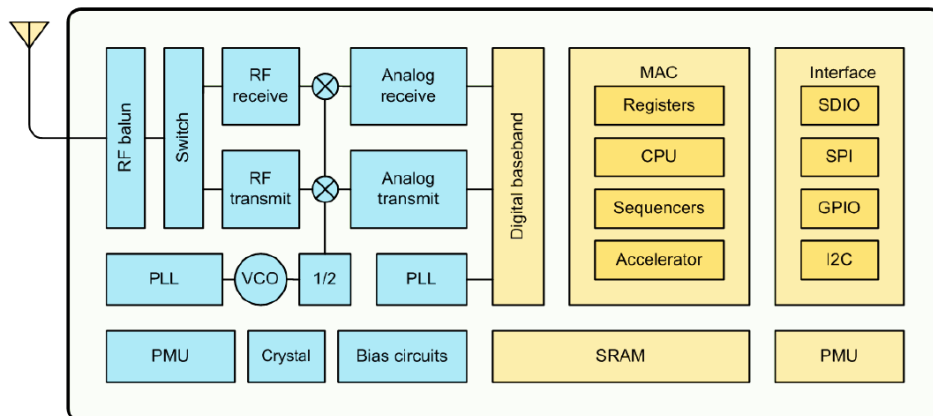
The ESP32 chip, which is scalable and adaptive, lies at the heart of this module. Individual control of two CPU cores is possible. The clock frequency can be adjusted from 80 to 240 MHz, and RTOS is supported. It's a Wi-Fi+BT+BLE MCU module for general use. ESP-WROOM-32s Traditional Bluetooth, Bluetooth low energy, and Wi-Fi are all included in the module. Various applications: Wi-Fi allows users to connect to a mobile phone or broadcast a BLE Beacon for signal detection; Bluetooth allows users to connect to a mobile phone or broadcast a BLE Beacon for signal detection.

For maximum wireless connectivity, the module offers data rates of up to 150 Mbps with a 20 dBm antenna output power. As a result, this module meets industry-leading criteria in terms of high integration, wireless transmission distance, power consumption, and network connectivity.

Features

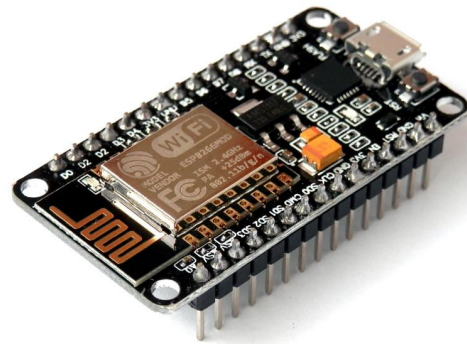
- 802.11b/g/n (802.11n, speed up to 150Mbps)
- WIFI Frequency Range 2.4GHz ~ 2.5GHz
- Clock frequency adjustment range from 80 MHz to 240 MHz, support for RTOS
- Built-in 2-channel 12-bit high-precision ADC with up to 18 channels
- Support UART/GPIO/ADC/DAC/SDIO/SD card/PWM/I2C/I2S interface
- Support multiple sleep modes, ESP32 chip sleep current is less than 5 μ A
- Embedded Lwip protocol stack
- Supports STA/AP/STA + AP operation mode
- Supports remote firmware upgrade (FOTA)
- General AT commands can be used quickly
- Support secondary development, integrated Windows, Linux development environment

Block Diagram



Block Diagram of NodeMCU

Images of NODEMCU



6.2 Software Used

6.2.1 Arduino IDE

Arduino is a free and open-source electronics platform with simple hardware and software. Arduino boards can detect inputs such as light on a sensor, a finger on a button, or a Twitter post and convert them to outputs such as turning on an LED, triggering a motor, or publishing anything online. By providing a set of instructions to the board's microcontroller, you may tell it what to do. The Arduino programming language (based on Wiring) and the Arduino Software (IDE) (based on Processing) are used to accomplishing this.

Thousands of projects have used Arduino throughout the years, ranging from simple household items to complicated scientific apparatus. This open-source platform has united a global community of makers - students, amateurs, artists, programmers, and professionals - whose contributions have added up to an enormous quantity of accessible knowledge that may be of tremendous benefit to novices and specialists alike. Arduino has been used in millions of projects and applications due to its simple and accessible user interface. Beginners will find the Arduino software simple to use, but advanced users will find it adaptable.

It is available for Mac, Windows, and Linux. It is used by teachers and students to build low-cost scientific equipment, to demonstrate chemistry and physics principles, and to begin learning programming and robotics. Designers and architects create interactive prototypes, while musicians and artists utilize it to create installations and try out new instruments. Makers, for example, utilize it to construct many of the items on display at the Maker Faire. Arduino is an essential tool for learning new skills. Anyone - youngsters, amateurs, artists, programmers - may get started tinkering by following the step-by-step instructions in a kit or sharing ideas with other Arduino members online.

For physical computing, there are numerous additional microcontrollers and microcontroller platforms. Similar capability is available in Parallax Basic Stamp, Netmedia's BX-24, Phidgets, MIT's Handyboard, and many other programs. All of these tools condense the complicated nuances of

microcontroller programming into a user-friendly package. Although Arduino simplifies the process of working with microcontrollers, it has one advantage over other systems for teachers, students, and curious amateurs:

- **Low-cost** - Compared to other microcontroller platforms, Arduino boards are affordable. The Arduino module's cheapest version can be assembled by hand, and even pre-made Arduino modules cost less than \$50.
- **Cross-platform** - The Arduino Software (IDE) is compatible with Windows, Macintosh OSX, and Linux. The majority of microcontroller systems are only compatible with Windows.
- **Easy-to-use programming environment** - The Arduino Software (IDE) is simple to use for novices while yet being flexible enough for advanced users. It's built on the Processing programming environment, which is useful for teachers because students learning to program in that environment will be familiar with the Arduino IDE.
- **Open source and extendable software** - The Arduino software is accessible as open source tools that experienced programmers can extend. C++ libraries can be used to extend the language, and those interested in learning more about the technical aspects can switch from Arduino to the AVR C programming language. If you desire, you can also include AVR-C code directly in your Arduino applications.
- **Open source and extendable hardware** - The Arduino boards' plans are made available under a Creative Commons license, allowing expert circuit designers to create their own version of the module, extending and upgrading it. The breadboard version of the module can be built by even inexperienced users to learn how it works and save money.

6.2.2 Android Studio

Built on JetBrains' IntelliJ IDEA software and designed exclusively for Android development, Android Studio is the official integrated development environment (IDE) for Google's Android operating system. It's available for Windows, macOS, and Linux users to download. It takes the position of the Eclipse Android Development Tools (ADT) as the primary IDE for developing native Android apps.

At the Google I/O conference on May 16, 2013, Android Studio was revealed. It was in an early access preview stage starting with version 0.1 in May 2013 and then moved into beta stage with version 0.8 in June 2014. Starting with version 1.0, the first stable build was released in December 2014.

Kotlin overtook Java as Google's recommended language for Android app development on May 7, 2019. Java and C++ are still supported.

Android Studio 3.0 or later supports Kotlin and "all Java 7 language features and a subset of Java 8 language features that vary by platform version," as well as "all Java 7 language features and a subset of Java 8 language features that vary by platform version." Some Java 9 features are backported by external projects. Although IntelliJ claims that Android Studio supports all available

Java versions, including Java 12, it is unclear to what extent Android Studio supports Java versions up to Java 12. (the documentation mentions partial Java 8 support).

Android supports at least some new language features up to Java 12.

Android Studio has vector graphics-enabled features that allow you to work with vector graphics, which results in better graphics and eliminates pixel distortion in the user interface. It also has the flutter plugin, which allows you to create apps for both Android and iOS with a single code using Google's Dart programming language.

Features

The following features are provided in the current stable version:

- Gradle-based build support
- Android-specific refactoring and quick fixes
- Lint tools to catch performance, usability, version compatibility and other problems
- ProGuard integration and app-signing capabilities
- Template-based wizards to create common Android designs and components
- A rich layout editor that allows users to drag-and-drop UI components, option to preview layouts on multiple screen configurations.
- Support for building Android Wear apps
- Built-in support for Google Cloud Platform, enabling integration with Firebase Cloud Messaging (Earlier 'Google Cloud Messaging') and Google App Engine[18]
- Android Virtual Device (Emulator) to run and debug apps in the Android studio.

7. Proposed Solution

For the delivery of the RFID System for Maintenance Management project, we discussed the various approaches discussed during our lectures and decided to use the Agile approach. The iterative nature of the Agile approach is the most suitable technique for the successful delivery of our project.

On a higher level, our entire application consists of 2 parts – a Raspberry PI system for reading RFID tokens and triggering notifications and an Android application for accessing the data. We plan to deliver the project in 4 iterations of Agile Sprints -

- i) In the first iteration we will be focusing on creating a system using raspberry pi which sends email to the authorized person if the service date is missed.
- ii) During the next iteration we will update this system to enable it to send the reminder of the upcoming machines to be serviced using email. This will mark the completion of our first part.
- iii) In the third iteration, we will implement the android application which will be used to access and manipulate the data in the database.

iv) In the last phase, we will update the application so that it also sends notification to the application user if the service date of the machine is missed.

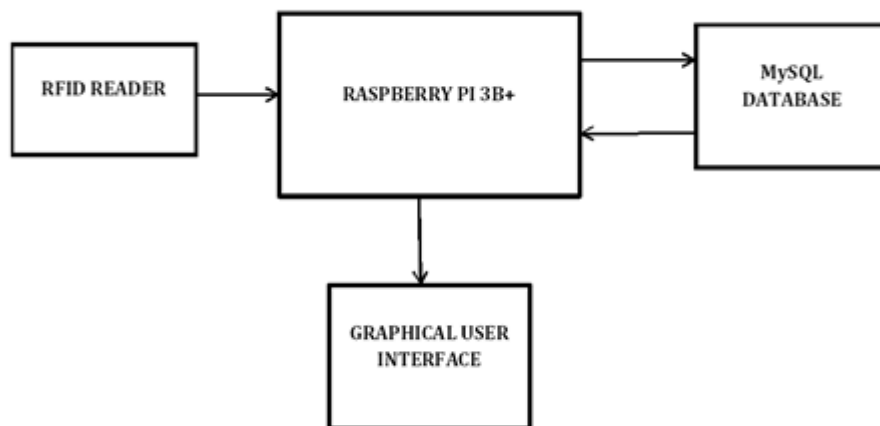
8. Project Activities

This project will cover the following activities as a part of successful project completion

8.1 Design

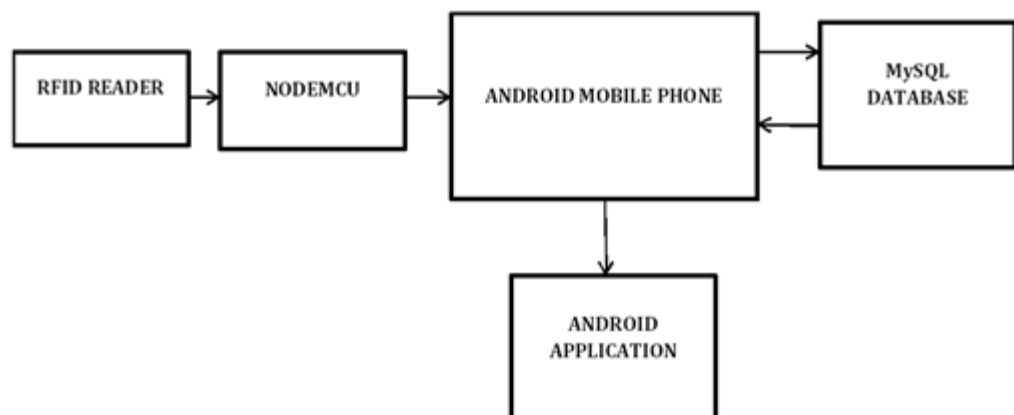
Since the system consists of two parts, our project design will consist of two separate blocks

Part I – Block Diagram for Raspberry Pi Model



The RFID reader will be connected to Raspberry Pi 3B+ using jumper wires. The GUI will be created using python programming language. MySQL Database will be stored on the server which can be accessed using the internet to store the data.

Part II – Block Diagram for Android App Model



The RFID reader will be connected to the NodeMCU which will contain the program to read the RFID tags using the RFID reader. NodeMCU will be connected to the android phone using USB OTG. The phone will be running the android application and will also be connected to MySQL database to store the data.

8.2 Build

The project will be developed using different IDEs

8.2.1 Arduino IDE

Arduino is an open-source electronics platform which is used for programming Arduino boards. These boards are microcontrollers which will be used to detect RFID tags and trigger events to update the database accordingly. Arduino is a good match for our project as it is inexpensive, cross-platform and open-source.

8.2.2 Android Studio IDE

Android Studio is the official integrated development environment (IDE) for Android development. It will be used for developing our android application, which will provide a GUI for the user to scan RFID and update the data in the database.

8.3 Deploy

All the components of this project will be deployed on localhost and on personal android devices

9. Deliverables

As a part of project completion, we will be able to deliver the following –

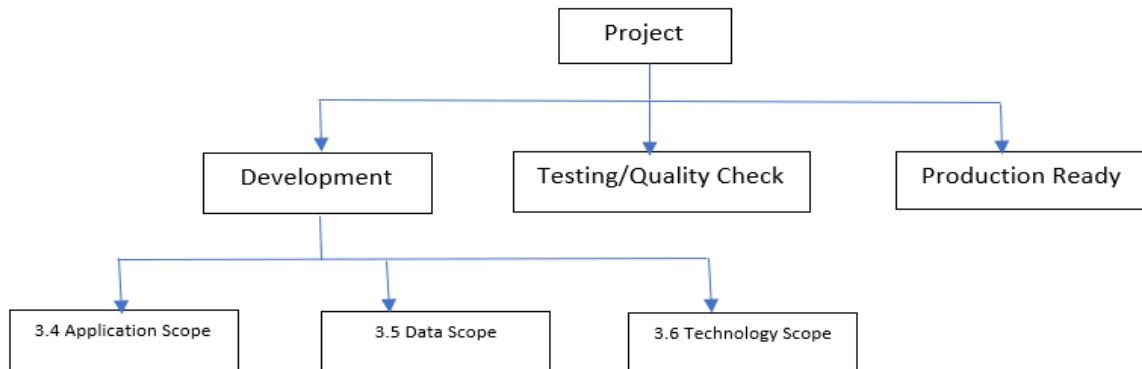
- i) Raspberry PI tool for RFID scanning – The Raspberry Pi RFID scanner will be able to read demo RFIDs from machineries. It will also be able to trigger maintenance reminder emails.
- ii) Maintenance Application in Android – This will be an android application that allows the user to update the maintenance information. It will also be used by Administrators to manage user information if required.
- iii) MySQL Database – This database will store all user information, machine information, and maintenance data. This information will be used by the application to trigger emails to users if a service was missed for a machine.
- iv) Project Report – Finally we will be submitting a project report which will contain all details about the project including different system diagrams, data flow diagrams, use cases, etc. This report will be useful for future maintenance of the application.

10. Project Management

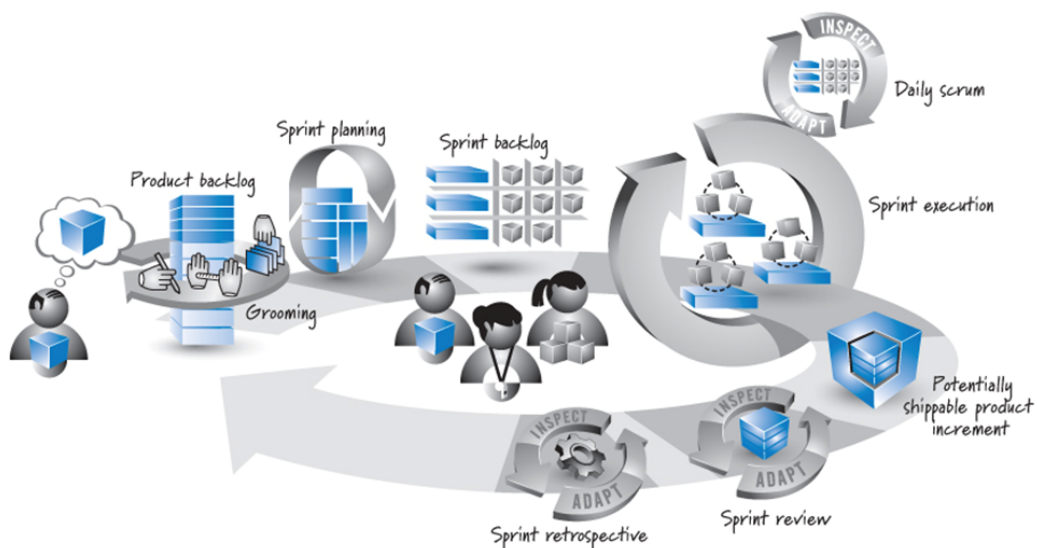
10.1 Management Approach

10.1.1 Agreed-upon Baseline

According to the agreed baseline, below is the Work Baseline Structure (WBS).



10.1.2 Process to Monitor Progress



Project Planning: Project is monitored using Project Management tool “JIRA”. All the tasks in the project are created as JIRA stories. All the user stories are sub divided into categories Testing, Development, DevOps to differentiate the stories.

Development: Before the Start of Sprint Planning, Product Owner/ Project Manager grooms the stories according to the iterations defined (refer 4.1). After the sprint Backlog, stories are assigned to the developers. Scrum Master is the process Authority here coaching all the team members in the sprint.

QA Testing: At the end of sprint all the stories are moved to testing environments and tested before moving to production environment. User Acceptance test reports are taken to analyze any discrepancies in the output.

Deployment: Once all the testing is done with no errors, everything will be moved to production and made it live to use. These deployments are taken care by DevOps Engineers to maintain isolation between environments.

Scrum Review & Retrospective: Development team, Project Owner, Project Manager and Scrum Master come together to discuss what is working and what is not working and how to improve sprints.

Scrum Daily Calls: Daily Standups are needed to give updates to everyone in the team. This brings transparency in the assigned work.

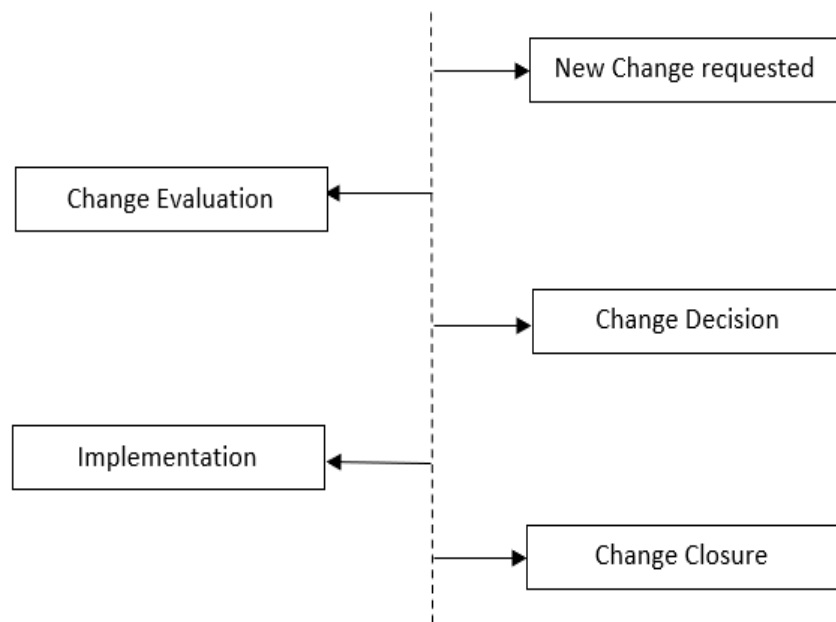
10.1.3 Means of Communication

- Regular Stand-up calls helps team to get updates on daily basis. Post standup calls, team maintains Minutes of Meeting (MoM) which tracks all the discussions.
- Communication is a key if any person is not able to make up for the meeting. An email is needed to convey the absence before the call of meeting.
- For individual dependency works one-to-one calls are sufficient but for team dependency works email communication should be preferred before scheduling any call.

10.1.4 Approach for Dealing with Issues

- Whenever there is a new issue, key aspects like the scenario & use case must be noted.
- After this, next phase will be finding the root cause of the issue.
- Documentation must be maintained as sometimes we can find answers to these issues.
- If the issue is a small correction in the current sprint, it is fine to be included, else it would be appropriate to move this issue into the next spring planning.

10.1.5 Change Control Procedure



A New Change timeline consists of stages ranging from Change request till the close of the change.

- **Change Request:** Whenever a new use case is encountered, the issue is noted and the scenario is also re-tested. If the issue still exists, then a new change is requested.
- **Change Evaluation:** Once the change is requested Product Owner, Project Manager, Scrum Master takes up a meeting and checks the use case and finalizes the request.
- **Change Decision:** A Decision is given whether the change is needed or its not required for the current MVP.
- **Implementation:** Once the change is included, it is moved to the Product Backlog with all the existing stories.
- **Closure:** As the stories are being completed in the sprints, all of them are closed after successful testing on production and test environments.

10.1.6 Risk Management

Risks can come from various sources including uncertainty in threats from equipment failures (at any phase in design, development, production, or sustaining of life-cycles), accidents, natural causes and disasters, deliberate attack from an adversary, or events of uncertain or unpredictable root-cause. So, for the maintenance of industrial equipment, risk management can be handled by following a 5-step strategy.

- Identify potential risks
- Measure frequency and severity
- Examine alternative solutions
- Decide which solution to use and implement
- Monitor results

10.1.7 Acceptance of Deliverables

The ultimate goal of deliverables acceptance management is to ensure successful maintenance of all the equipment in a specified time period, within budget and within scope. For achieving this goal, there are some key tasks that the project manager must accomplish.

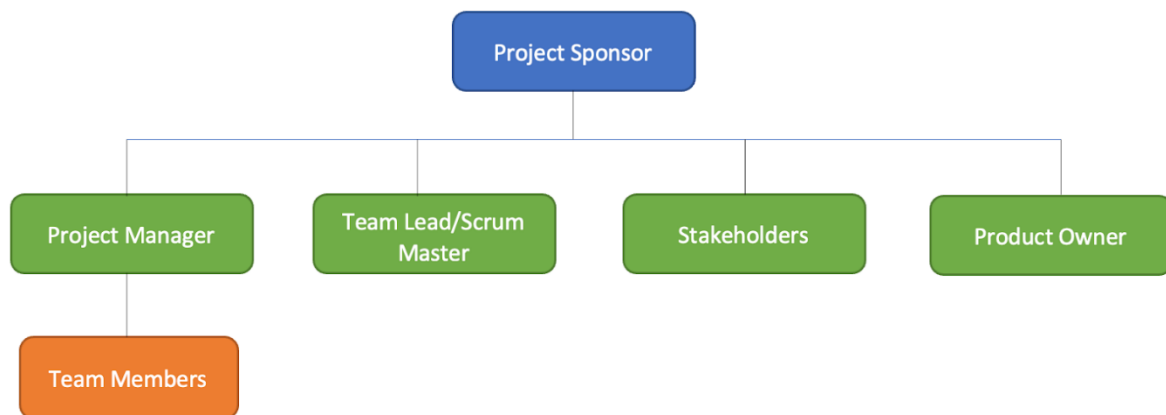
These tasks are:

- Determine and assign a person who will review the deliverables and assure their adherence to the criteria listed in the plan (as it's been said, normally the project manager accomplishes this task but there can be exceptions).
- Set up any time considerations that may be required for managing the acceptance of project deliverables and adherence to customer expectations.
- Make a deliverables acceptance document report that represents all the deliverables actually accepted and approved.

10.1.8 Project Completion Criteria

Completion criteria can be a simple checklist or a more comprehensive set of standards or protocols. Whatever will be used to determine if the task was successfully completed should be defined in advance, in the completion criteria for the task. Often, the task owner or the subject matter expert is the best person to create these completion criteria.

11. Project Organization



12. Roles and Responsibilities

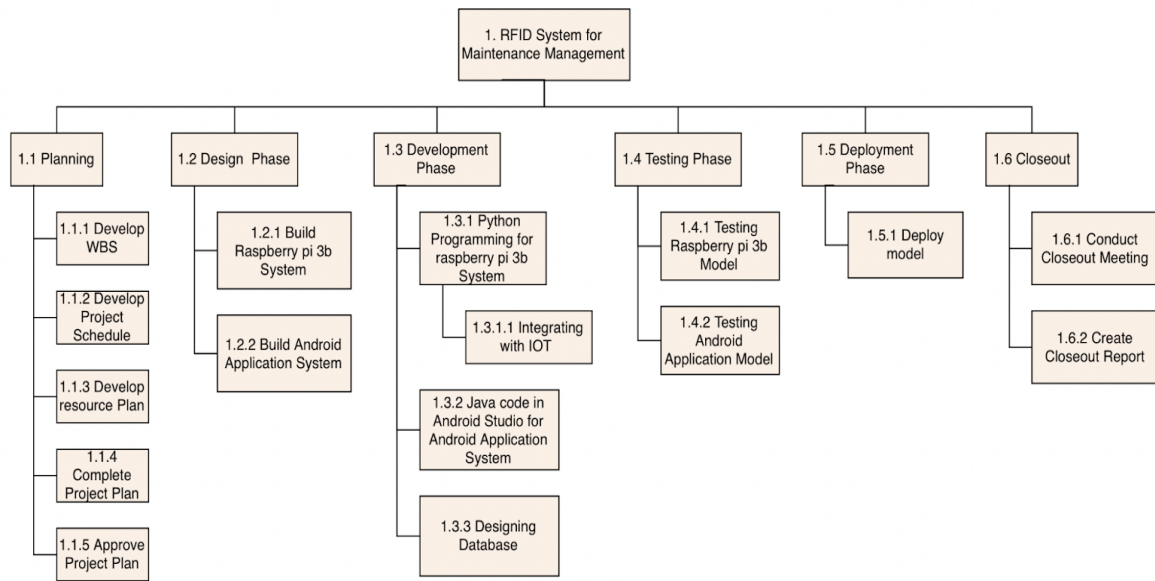
Sr.No	Name	Role	Responsibilities
1	Ekta Abhishek Desai	Database Administrator	<ul style="list-style-type: none"> • Find ways to store, organize, and manage your data using database software. • Ensuring that databases meet user requirements.

2	Victoria Garcia	Test Engineer	<ul style="list-style-type: none"> • Writing test plans and creating test cases for the product. • Troubleshooting any errors and streamlining the testing procedures.
3	Dedeepya Sai Gondi	IOT Engineer	<ul style="list-style-type: none"> • Designing, coding and testing features of IoT devices • Developing software that allows IoT devices to function and connect to other devices
4	Sanika Jadhav	Software Developer	<ul style="list-style-type: none"> • Collaborating with management, departments and customers to identify end-user requirements and specifications. • Producing efficient and elegant code based on Requirements.
5	Aishwarya Jamma	Software Developer	<ul style="list-style-type: none"> • Collaborating with management, departments and customers to identify end-user requirements and specifications. • Producing efficient and elegant code based on requirements.
6	Amrutha Nagaraj	Project Manager	<ul style="list-style-type: none"> • Plan,Develop the project Idea. • Monitor Project Progress and set deadlines.
7	Rupesh Saw	DevOps Engineer	<ul style="list-style-type: none"> • Planning the team structure, activities, and involvement in project management activities. • Implementing various development, testing, automation tools, and IT infrastructure.

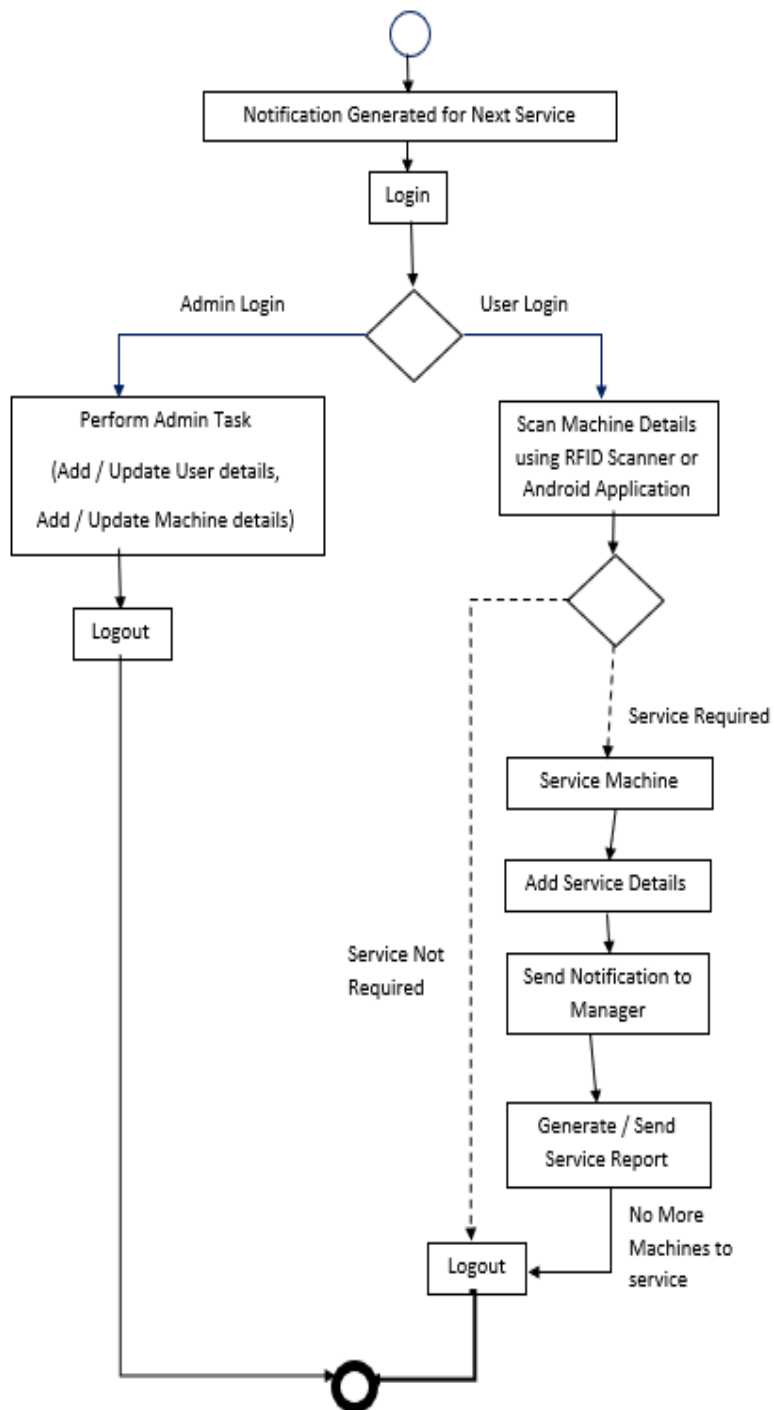
13. Project Work Plan (WBS, Timeframes)

Work Breakdown Structure

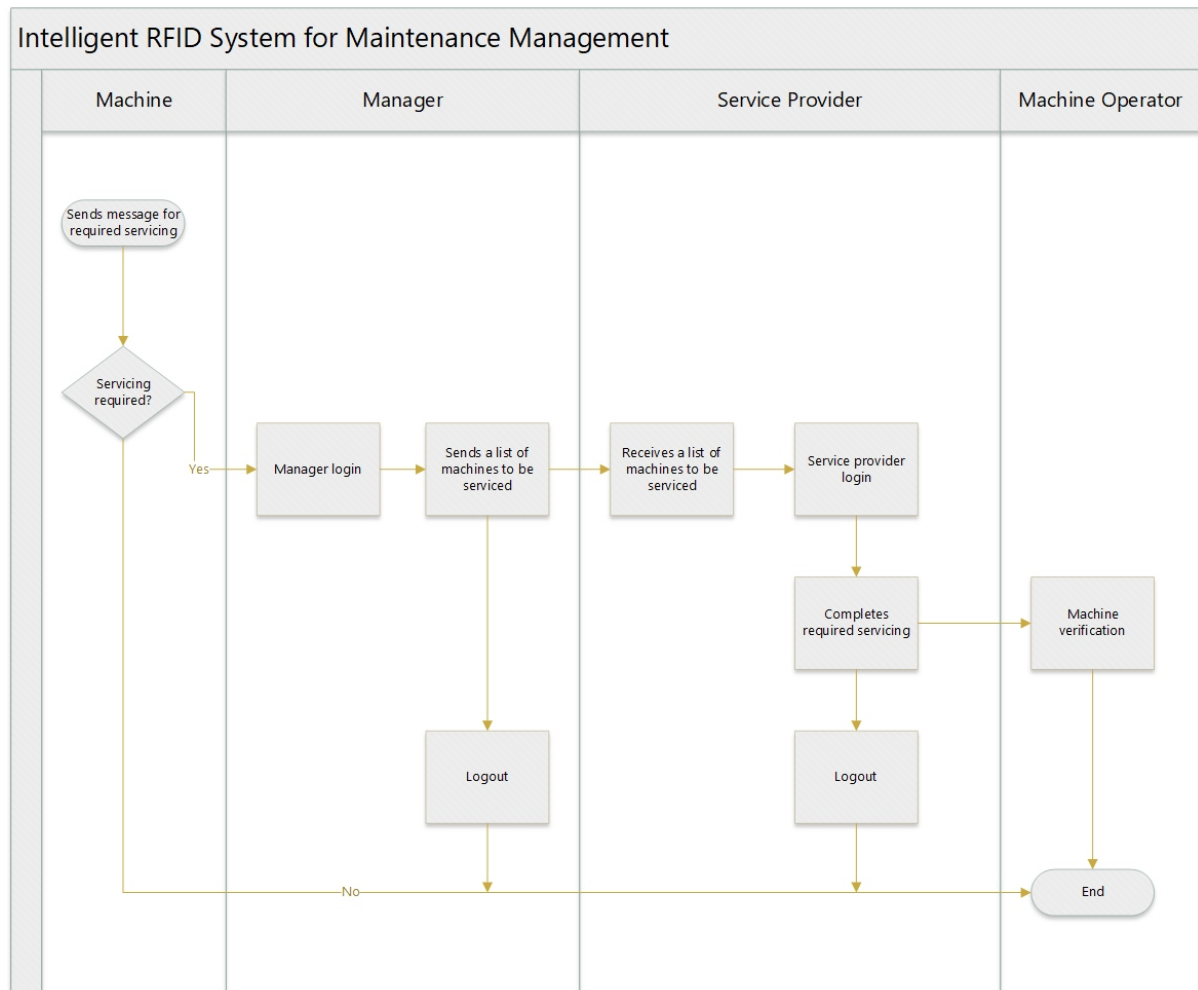
INTELLIGENT RFID SYSTEM FOR MAINTENANCE MANAGEMENT



14. Business PROCESS MODEL Notation

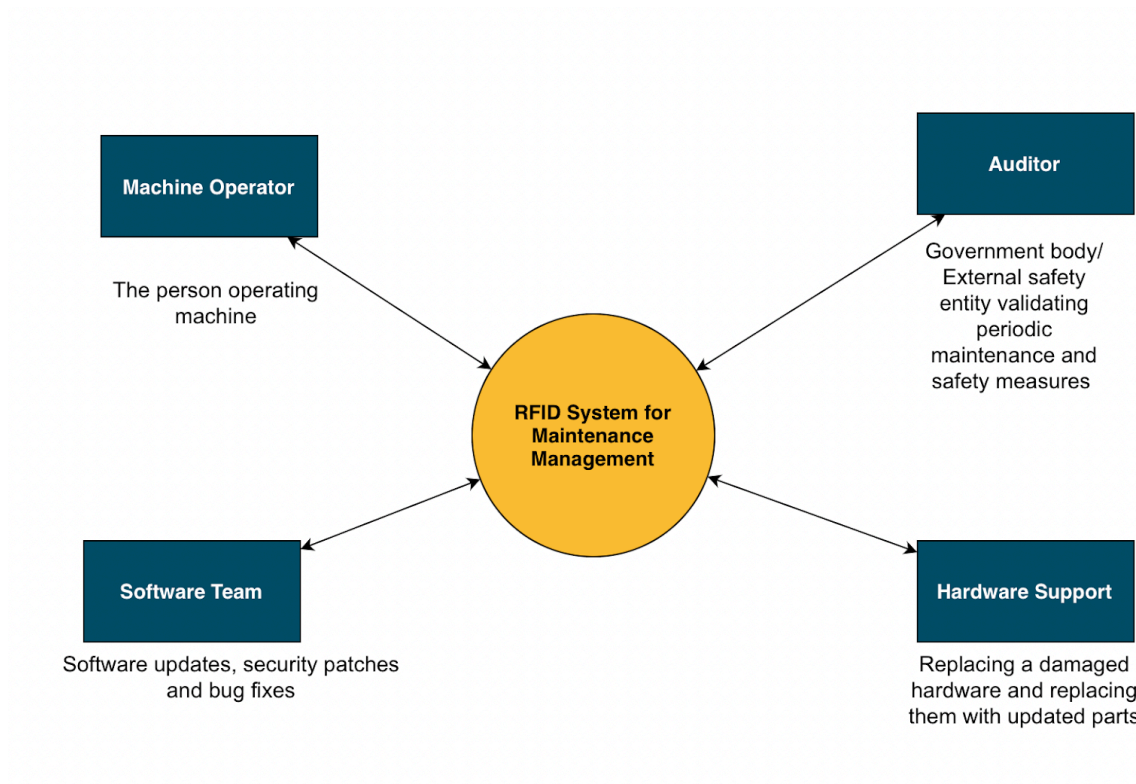


15. Swimlane Diagram

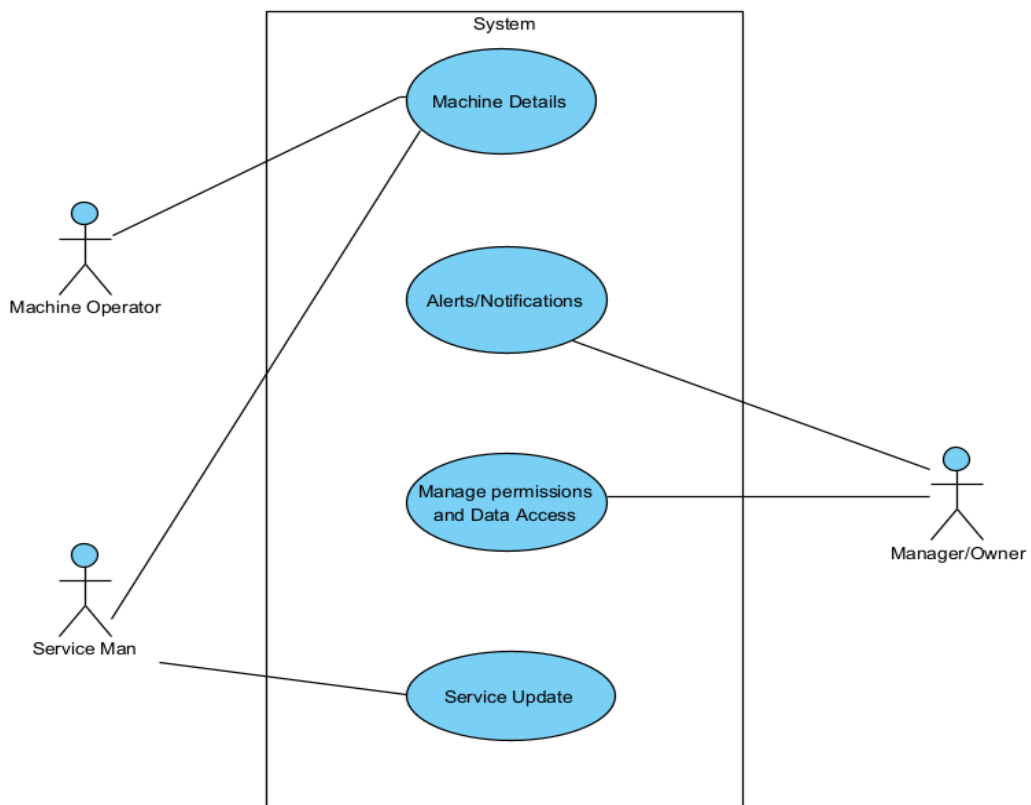


16. System Context Diagram

The below diagram illustrates the context diagram which gives an overall functionality of the proposed system:



17. USE CASE DIAGRAM



18. DATA DICTIONARY

RFIDMaintenanceSystem = AdminLogin + UserLogin

AdminLogin = AddUserDetails + UpdateUserDetails + AddMachineDetails + UpdateMachineDetails

UserLogin = ScanMachineDetails [RFIDScanner | AndroidApp] + ServiceRequired [Yes | No]

ServiceRequired = AddServiceDetails + SendNotifications + SendServiceReport

MachineOperator = OperatorID + OperatorName + MachineID

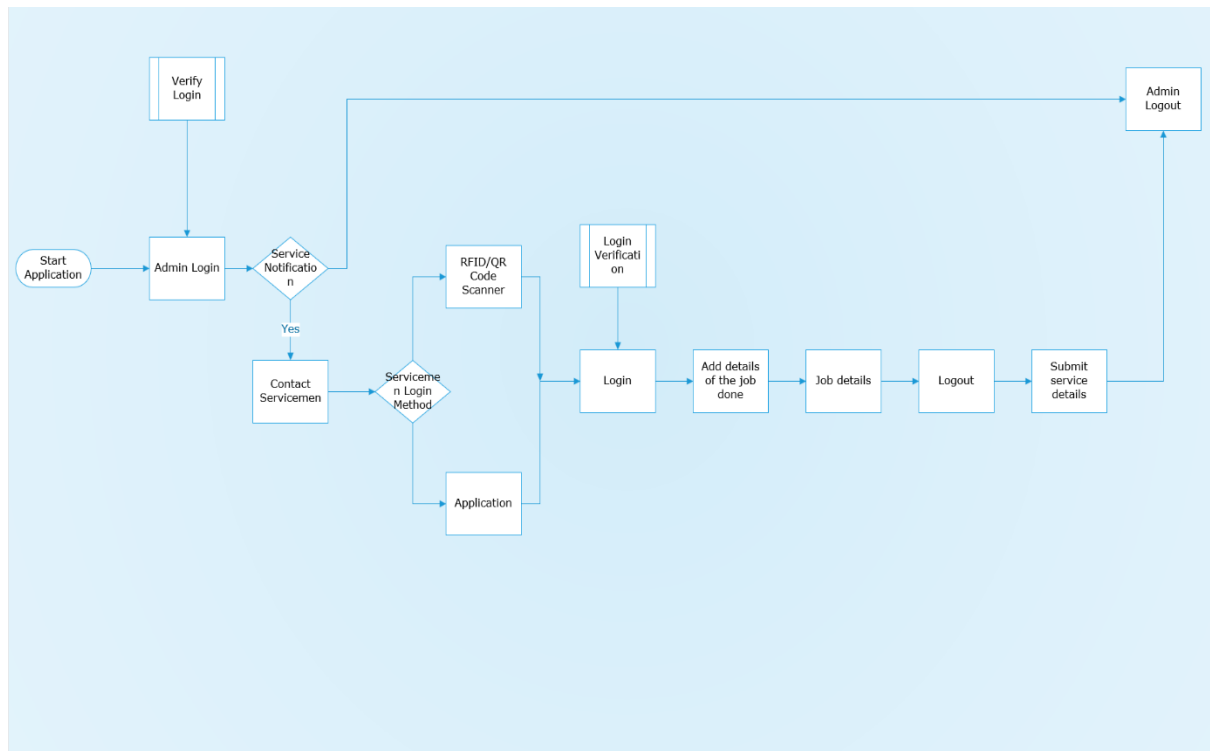
Machine = MachineID + MachineType + Weight + Price + NoInStock + MachineStatus + (ServiceID) + ManagerID

Service = ServiceID + ServiceName + ServiceType + ServiceTime + isServiceRequired + NoOfServiceRequired

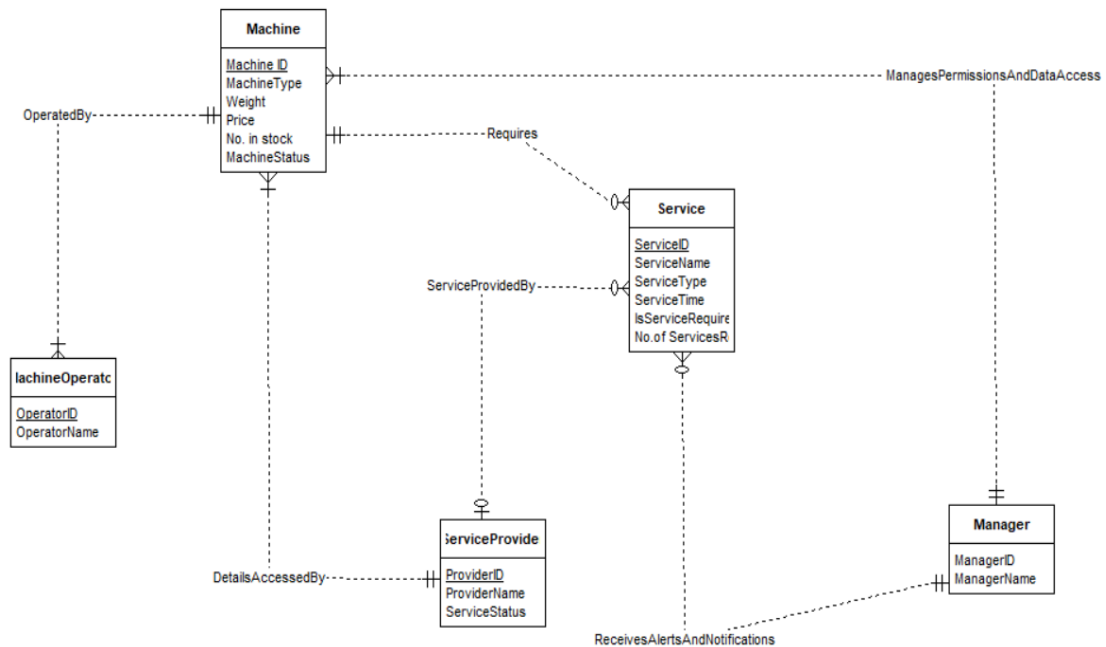
ServiceProvider = ProviderID + ProviderName + ServiceStatus + ServiceID

Manager = ManagerID + ManagerName + ServiceID

19. Activity Diagram



20. Class Diagram



21. Project Timeline

Task Name	Assigned To	Start Date	End Date
Research and Topic Finalization	Ekta Abhishek Desai	Jan 19,2022	Feb 02,2022
	Victoria Garcia		
	Dedeepya Sai Gondi		
	Sanika Jadhav		
	Aishwarya Jamma		
	Amrutha Nagaraj		

INTELLIGENT RFID SYSTEM FOR MAINTENANCE MANAGEMENT

	Rupesh Saw		
Project Planning	Amrutha Nagaraj	Feb 09,2022	Feb 27,2022
Development- Software	Sanika Jadhav Aishwarya Jamma	Mar 2,2022	Mar 23,2022
Hardware Integration	Rupesh Saw Dedeepya Sai Gondi	Mar 30,2022	Apr 6,2022
Testing	Ekta Abhishek Desai Victoria Garcia	Apr 13,2022	Apr 20,2022
Deployment	Ekta Abhishek Desai Victoria Garcia Dedeepya Sai Gondi Sanika Jadhav Aishwarya Jamma Amrutha Nagaraj Rupesh Saw	Apr 27,2022	May 4,2022