**A**

**PROJECT REPORT ON**

**UNSUPERVISED MACHINE LEARNING FOR MANAGING SAFETY ACCIDENTS IN RAILWAY**

**STATIONS**

**Submitted**

**To**

**Osmania University**

In partial fulfillment of the requirements for the award of the degree of

**MASTER OF COMPUTER APPLICATIONS BY**

**Mr. Jangala Sai**

 **(130422862109)**

**AURORA’S P.G. COLLEGE**

**(Affiliated to Osmania University)**

Peerzadiguda, Uppal, Hyderabad-50039

(2022-24)

**ACKNOWLEDGEMENT**

The satisfaction and euphoria that accompanies the successful completion of any task would be incomplete without the mention of the people who made it possible and whose encouragement and guidance have crowned out efforts with success.

I am extremely thankful to the principal, **AURORA'S PG COLLEGE(MCA), UPPAL**, for providing me with infrastructural facilities to work in, without which this work would not have been possible.

I would like to express my gratitude to **the Head, Department of Computer Application, Aurora's PG College (MCA), Uppal,** for his stimulating guidance, continuous encouragement, and supervision during the course of the present work.

I would like a place on record my deep sense of gratitude **to Faculty, Aurora's PG College(MCA), Uppal,** for their continuous guidance and valuable suggestions in completing the project right from its conception to conclusions.

Above all I express thanks to the Guide, my parents, and my friends without whose support, I could not have made this project.

**J.SAI**

**DECLARATION**

I hereby declare that the work presented in this project report titled at " is done by me in Master of Computer Applications, Aurora's PG College, Uppal. No part of the dissertation is copied from books/Journals/internet and whenever the portion is taken, that has been duly referred in the text. This is an original work done by me and has not been submitted to any other institution.

**Jangala sai**

**1304-22-862-109**

# ABSTRACT

For both passenger and freight transportation, railroad operations must bedependable, accessible, maintained, and safe (RAMS). In many urban areas, railway stations risk and safety accidents represent an essential safety concern for daily operations. Moreover, the accidents lead to damage to market reputation, including injuries and anxiety among the people and costs. This stations under pressure caused by higher demand which consuming infrastructure and raised the safety administration consideration. To analysing these accidents and utilising the technology such AI methods to enhance safety, it is suggested to use unsupervised topic modelling for better understand the contributors to these extreme accidents. It is conducted to optimise Latent Dirichlet Allocation (LDA) for fatality accidents in the railway stations from textual data gathered RSSB including 1000 accidents in the UK railway station. This research describes using the machine learning topic method for systematic spot accident characteristics to enhance safety and risk management in the stations and provides advanced analysing. The study evaluates the efficacy of text by mining from accident history, gaining information, lesson learned and deeply coherent of the risk caused by assessing fatalities accidents for large and enduring scale. This Intelligent Text Analysis presents predictive accuracy for valuable accident information such as root causes and the hot spots in the railway stations. Further, the big data analytics ’ improvement results in an understanding of the accidents’ nature in ways not possible if a considerable amount of safety history and not through narrow domain analysis of the accident reports. This technology renders stand with high accuracy and a beneficial and extensive new era of AI applications in railway industry safety and other fields for safety applications.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **INDEX** | | | | |
| **List of Contents** | | | | **Page No** |
|  | | | |  |
| 1.INTRODUCTION | | | | 1 |
| 1.1 | | Overview | | 1 |
| 1.2 | | Company Profile | | 4 |
| 2. LITERATURE OVERVIEW | | | | 5 |
| 3. ANALYSIS | | | | 46 |
| 3.1 System Requirement Specification | | | | 46 |
| 3.1.1 Introduction | | | | 46 |
| 3.1.2 Existing system | | | | 46 |
| 3.1.3 Proposed system | | | | 47 |
| 3.1.4 Scope | | | | 48 |
| 3.2 Software / Hardware Requirements | | | | 49 |
| 3.2.1 Hardware Requirements | | | | 49 |
| 3.2.2 Software Requirements | | | | 49 |
| 3.3 Modules | | | | 50 |
| 3.3.1 railway safety specialist | | | | 50 |
| 3.3.2 Remote operator | | | | 50 |
| 3.4 Functional Requirements | | | | 50 |
| 3.5 Feasibility Study | | | | 51 |
| 3.5.1 Economical Feasibility | | | | 51 |
| 3.5.2 Technical Feasibility | | | | 51 |
| 3.5.3 SOCIAL Feasibility | | | | 52 |
| 4. Design | | | | 53 |
| 4.1 Architecture | | | | 53 |
| 4.2 System Design | | | | 54 |
| 4.2.1 Class Diagram | | | | 54 |
| 4.2.2 Usecase Diagram | | | | 55 |
| 4.2.3 Sequence Diagram | | | | 56 |
| 4.2.4Activity Diagram | | | | 57 |
| 4.2.6 COMPONENT DIAGRAM | | | | 58 |
| 4.2.5 Data Dictionary | | | | 59 |
| 4.2.6 E.R. Diagram | | | | 59 |
| 5. System Implementation | | | | 62 |
| 5.1 | | Organization Impact | | 62 |
| 5.2 | | User Manger Assessment | | 65 |
| 5.3 | | Code | | 65 |
| 5.4 | | Screen Shots | | 76 |
| 6. Testing | | | | 79 |
| 6.1 | | Software Testing | | 79 |
| 6.2 | | Testing Approaches | | 79 |
| 6.3 | | Testing Types | | 79 |
|  | | 6.4 | | System Text | 80 |
|  | | 6.5 | | Test Cases | 82 |
| 7. | | Conclusion | |  | 85 |
| 8. | | Future Enhancement | | | 86 |
| 9. | | Bibliography | | | 87 |
| 10. Gantt Chart | | | | | 91 |
| 11. CD | | | | |  |

**1.INTRODUCTION**

* 1. **Overview:**

Trains as public transportation have been considered as safer than other means. However, passengers on trains stations sometimes face many risks because of many overlapping factors such as station operation, design, and passenger behaviors. Due to the gradually increasing demand and the heavily congested society and the state of some station’s layout and complexity in design, there are potential risks during the operation of the stations. Furthermore, Passenger, people and public safety is the main concern of the railway industry and one of the critical parts of the system. European Union put into practice Reliability, Availability, Maintainability and Safety (RAMS)as a standard in 1999 known as EN 50126. Aiming to prevent railway accidents and ensure a high level of safety in railway operations. The RAMS analyses concepts lead to minimizing the risks to acceptable levels and rise safety levels. However, that have been an urgent issue and still, the reports show several people are killed every year in the railway station, some accidents lead to injuries or fatalities. For example, In Japan in 2016, 420 accidents occurred that included being struck by a train, which resulted in 202 deaths. This including of those 420 accidents, 179 (resulting in 24 fatalities) included falling from a platform and following injury or death as a consequence of hitting with a train [1]. In the UK, 2019/20, it has been reported that Most passenger injuries occur from accidents in stations. Greatest Major injuries are the outcome of slips, trips and falls, of which there were approximately 200 [2] play significant impact in reducing injuries on station platforms and provide quality, reliable and safe travel environment for all passengers, worker and public. Even if some accident does not result in deaths or injuries, such accidents cause delay, cost, fear and anxiety among the people, interruption in the operations and damage the industry reputation. Also, to provide or invest any control safety measurements the stations it is crucial to considering the risks associated with the railway incidents and risks in the station and identification of many factors related to

the accident by a comprehensive knowledge of the root cause of accidents considering all the possible technology.

The objective of this research is to analysis a collection case of accidents between 01/01/2000 and 17/04/2020 data to introduce a smart method, which expected to develop the safety level future, the risk management process, and the way to collect data in the railway stations. This data been gathered by RSSBS and agreed to be used for the research purpose. Analyzing an extensive amount of data recorded in a different form are a challenging job. Nowadays, it is hard to obtain for specific information in such mix digitization big data in including Web, video, images and other sources, it is research of a needle in a haystack. Thus, a powerful tool for assistance manage, search and understand these vast amounts of information is needed indeed [3], [4]. Many pre-processing techniques and algorithms are required to obtain valuable characteristics from an enormous amount of safety data in the stations including textual. The study covers the topic modeling to identify useful characteristics such the root cause of the accidents and also exploring the factors which are multiple groups of words or phrases that explain and summarize the content covered by an accident’s reports reducing time with high accuracy of outcomes. Topic modeling techniques are robust smart methods that extensively applied in natural language processing to topic detection and semantic mining from unstructured documents. Consequently, It has been suggested in this work the LDA model which is one of the best-known probabilistic unsupervised learning methods that marks the topics implicit in collection of contexts [5]. Since increasing of applying new technologies and the revolution of data, the development of technology and utilizing AI in many fields it suggested in this paper a smart analysis utilizing the topic modeling techniques which can be very useful and effective to semantic mining and latent discovery context documents and datasets. The other source of data (Images-videos and numerical) been conducted utilizing AI approaches which cover supervised learning [6], [ so the unstructured textual data is targeted.

Hence, our motivation is to investigate the topic modeling approaches to risks and safety accident subjects in the stations. This work provides the method of topic modeling based on LDA with other models for advanced analytics, aiming to make contributions in the future of smart safety and risk management in the stations. Through applying the models, we investigate the safety accidents for fatality accident in the railway.

This paper establishes an innovative method in the area to studies how the textual source of data of railway station accident reports could be efficiently used to extract the root causes of accidents and establish an analysis between the textual and the possible cause. Where the full automated process that has ability to get the input of text and provide outputs not yet ready [8]. Applying this method expected to come overcome issues such as aid the decision-maker in real time and extract the key information to be understandable from non-experts, better identify the details of the accident in-depth, design expert smart safety system and effective usage of the safety history records. A Such results could support in the analysis of safety and risk management to be systematic and smarter. Our approach uses state-of-the-art LDA algorithm to capture the critical texts information of accidents and their causes. The rest of this paper is arranged as follows: In Section II, related work in both accident analysis and text classification with deep learning have been presented. Section III describes in detail the approach that has been used along with evaluation criteria. Section IV provides details of our implementations and section V reports the results. Finally, Section VI presents the conclusion

**1.2 Company Profile:**

**About: mentor technologies pvt.ltd.**

mentor technologies pvt.ltd.(MIPL) is a progressive and innovative IT company offering a

IIM alumnus company started in 1998 as a training partner of TCS iON. TataConsultancy Services Limited (TCS Ltd.) is a leading IT solutions company & one of the world's

leading spectrum of services including software development, staffing, recruitment and training. An Information Technology consulting, services and business process

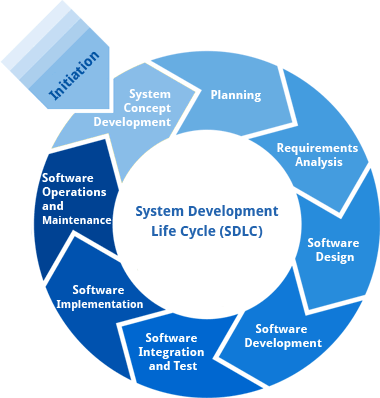
outsourcing organizations and a part of the Tata Group, India's best-known business conglomeratementor technologies has built a suite of services around its core offering of training.In the training arena mentor technologies has an unenviable track record of grooming several hundreds of students, building their skills and launching their careers. Alumni of mentor technologies are in senior IT positions across the globe mentor technologies also offers a host of training programs tailor made for IT companies.

The development arm of focuses on providing solutions to the Education & Training sector including assessments, analytics, online course delivery and ERP for training companies. Along the way mentor technologies has developed a strong expertise in JAVA, Android and open source technologies.Relations built with HR managers over the years have helped mentor technologies provide specialized services in the area of staffing and fresher’s recruitment for IT & ITES companies.Having started with a strong student focus, mentor technologies continues to provide other support services to B.Tech students including project guidance, internships and placement support.

**2.LITERATURE OVERVIEW**

**2.1The Systems Development Life Cycle (SDLC):**

The Systems Development Life Cycle (SDLC), or Software Development Life Cycle in systems engineering, information systems and software engineering, is the process of creating or altering systems, and the models and methodologies use to develop these systems.



**2.1.1Requirement Analysis and Design**

Analysis gathers the requirements for the system. This stage includes a detailed study of the business needs of the organization. Options for changing the business process may be considered. Design focuses on high level design like, what programs are needed and how are they going to interact, low-level design (how the individual programs are going to work), interface design (what are the interfaces going to look like) and data design (what data will be required). During these phases, the software's overall structure is defined. Analysis and Design are very crucial in the whole development cycle. Any glitch in the design phase could be very expensive to solve in the later stage of the software development. Much care is taken during this phase. The logical system of the product is developed in this phase.

**2.1.2Implementation**

In this phase the designs are translated into code. Computer programs are written using a conventional programming language or an application generator. Programming tools like Compilers, Interpreters, and Debuggers are used to generate the code. Different high level programming languages like PYTHON 3.6, Anaconda Cloud are used for coding. With respect to the type of application, the right programming language is chosen.

**2.1.3Testing**

In this phase the system is tested. Normally programs are written as a series of individual modules, this subject to separate and detailed test. The system is then tested as a whole. The separate modules are brought together and tested as a complete system. The system is tested to ensure that interfaces between modules work (integration testing), the system works on the intended platform and with the expected volume of data (volume testing) and that the system does what the user requires (acceptance/beta testing).

**2.1.4Maintenance**

Inevitably the system will need maintenance. Software will definitely undergo change once it is delivered to the customer. There are many reasons for the change. Change could happen because of some unexpected input values into the system. In addition, the changes in the system could directly affect the software operations. The software should be developed to accommodate changes that could happen during the post implementation period.

**2.2 Software Developing Process Model**

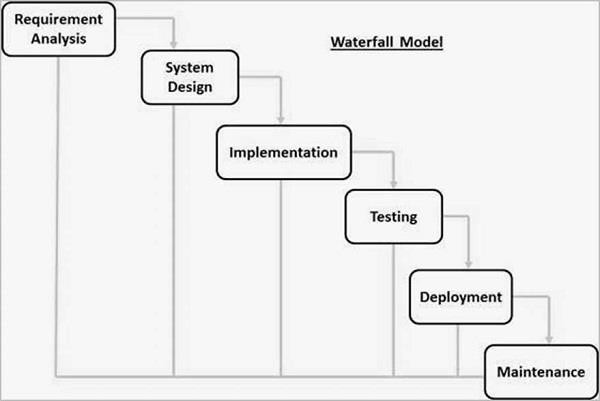
**Waterfall Model:**

The Waterfall Model was the first Process Model to be introduced. It is also referred to as a linear-sequential life cycle model. It is very simple to understand and use. In a waterfall model, each phase must be completed before the next phase can begin and there is no overlapping in the phases.

The Waterfall model is the earliest SDLC approach that was used for software development. The waterfall Model illustrates the software development process in a linear sequential flow. This means that any phase in the development process begins only if the previous phase is complete. In this waterfall model, the phases do not overlap.

**Waterfall Model – Design**

Waterfall approach was first SDLC Model to be used widely in Software Engineering to ensure success of the project. In "The Waterfall" approach, the whole process of software development is divided into separate phases. In this Waterfall model, typically, the outcome of one phase acts as the input for the next phase sequentially. The following illustration is a representation of the different phases of the Waterfall Model.



**Figure1.Different phases of waterfall model**

The sequential phases in Waterfall model are

* **Requirement Gathering and analysis** − All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document.
* **System Design** − The requirement specifications from first phase are studied in this phase and the system design is prepared. This system design helps in specifying hardware and system requirements and helps in defining the overall system architecture.
* **Implementation** − With inputs from the system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality, which is referred to as Unit Testing.
* **Integration and Testing** − All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.
* **Deployment of system** − Once the functional and non-functional testing is done; the product is deployed in the customer environment or released into the market.
* **Maintenance** − There are some issues which come up in the client environment. To fix those issues, patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

**Waterfall Model – Application**

Every software developed is different and requires a suitable SDLC approach to be followed based on the internal and external factors. Some situations where the use of Waterfall model is most appropriate are –

* + Requirements are very well documented, clear and fixed.
  + Product definition is stable.
  + Technology is understood and is not dynamic.
  + There are no ambiguous requirements.
  + Ample resources with required expertise are available to support the product.
  + The project is short.

**Waterfall Model – Advantages**

The advantages of waterfall development are that it allows for departmentalization and control. A schedule can be set with deadlines for each stage of development and a product can proceed through the development process model phases one by one.

Development moves from concept, through design, implementation, testing, installation, troubleshooting, and ends up at operation and maintenance. Each phase of development proceeds in strict order.

Some of the major advantages of the Waterfall Model are as follows –

* Simple and easy to understand anduse
* Easy to manage due to the rigidity of the model. Each phase has specific deliverables and a reviewprocess.
* Phases are processed and completed one at atime.
* Works well for smaller projects where requirements are very wellunderstood.
* Clearly definedstages.
* Well understoodmilestones.
* Easy to arrangetasks.
* Process and results are welldocumented

**Waterfall Model – Disadvantages**

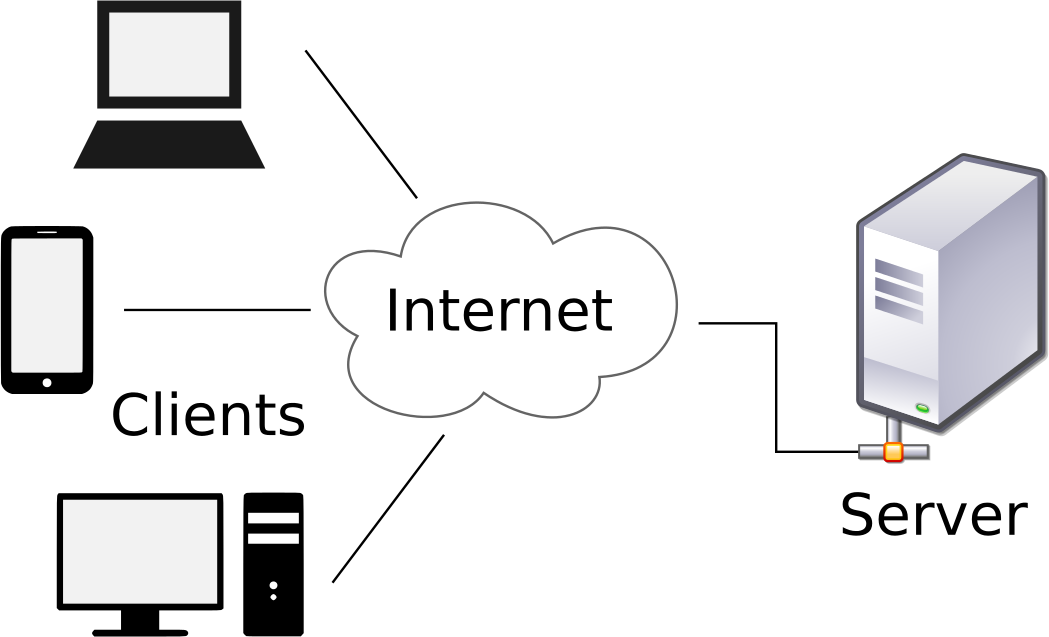
The disadvantage of waterfall development is that it does not allow much reflection or revision. Once an application is in the testing stage, it is very difficult to go back and change something that was not well-documented or thought upon in the concept stage.

The major disadvantages of the Waterfall Model are as follows –

* No working software is produced until late during the life cycle.
* Not suitable for the projects where requirements are at a moderate to high risk of changing. So, risk and uncertainty is high with this process model.
* Cannot accommodate changing requirements.
  1. **Architecture:**

**Client/server architecture:**Client/server architecture is a producer/consumer computing architecture where the server acts as the producer and the client as a consumer. The server houses and provides high-end, computing-intensive services to the client on demand. These services can include application access, storage, file sharing, printer access and/or direct access to the server’s raw computing power.

* Client/server architecture works when the client computer sends a resource or process request to the server over the network connection, which is then processed and delivered to the client. A server computer can manage several clients simultaneously, whereas one client can be connected to several servers at a time, each providing a different set of Services.
* In its simplest form, the internet is also based on client/server architecture where web servers serve many simultaneous users with website data.



**Figure 2: Client Server Architecture**

* 1. **Object Oriented System Development:**

The Unified Modeling Language (UML) is used to specify, visualize, modify, construct and document the artifacts of an object-oriented software intensive system under development. UML offers a standard way to visualize a system's architectural blueprints, including elements such as:

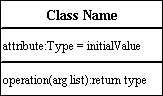
* Actors
* business processes
* components(logical)
* activities
* programming language statements
* database schemas, and
* Reusable software components.
  + 1. **Class Diagram:**

Class diagrams are the backbone of almost every object-oriented method including UML. They describe the static structure of a system.

**Basic Class Diagram Symbols and Notations:**

Classes represent an abstraction of entities with common characteristics. Associations represent the relationships between classes. Illustrate classes with rectangles divided into compartments. Place the name of the class in the first partition (centered, bolded, and capitalized), list the attributes in the second partition, and

Write operations into the third.



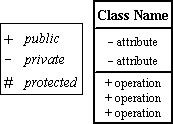
**Active Class**

Active classes initiate and control the flow of activity, while passive classes store data and serve other classes. Illustrate active classes with a thicker border.

Active Class

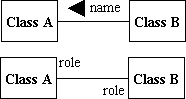
**Visibility**

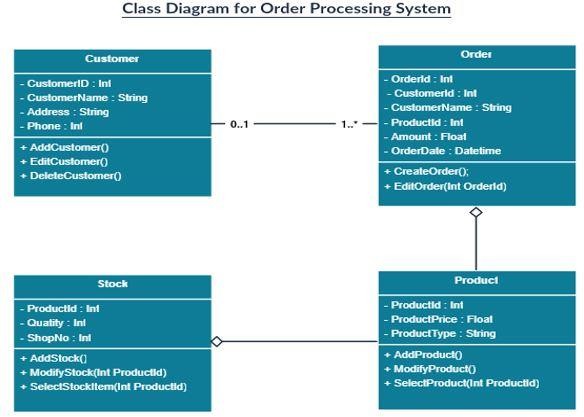
Use visibility markers to signify who can access the information contained within a class. Private visibility hides information from anything outside the class partition. Public visibility allows all other classes to view the marked information. Protected visibility allows child classes to access information they inherited from a parent class.



**Associations**

Associations represent static relationships between classes. Place association names above, on, or below the association line. Use a filled arrow to indicate the direction of the relationship. Place roles near the end of an association. Roles represent the way the two classes each other.





**Figure 4: Class Diagram**

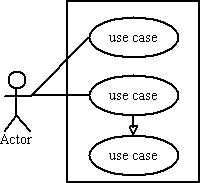
* + 1. **Use case Diagram**:

The purpose of a use case diagram in UML is to demonstrate the different ways that a user might interact with a system. Create a professional diagram for nearly any use case using our UML diagram tool.

**What is a use case diagram?**

In the Unified Modeling Language (UML), a use case diagram can summarize the details of your system's users (also known as actors) and their interactions with the system. To build one, you'll use a set of specialized symbols and connectors. Scenarios in which your system or application interacts with people, organizations, or external systems.

Goals that your system or application helps those entities (known as actors) achieve

The scope of your system.

**When to apply use case diagrams**

A use case diagram doesn't go into a lot of detail—for example; don't expect it to model the order in which steps are performed. Instead, a proper use case diagram depicts a high-level overview of the relationship between use cases, actors, and systems. Experts recommend that use case diagrams be used to supplement a more descriptive textual use case.UML is the modeling toolkit that you can use to build your diagrams. Use cases are represented with a labeled oval shape. Stick figures represent actors in the process, and the actor's participation in the system is modeled with a line between the actor and use case. To depict the system boundary, draw a box around the use case itself.

UML use case diagrams are ideal for:

Representing the goals of system-user interactions

Defining and organizing functional requirements in a system

Specifying the context and requirements of a system

Modeling the basic flow of events in a use case

Diagramming is quick and easy with Lucid chart. Start a free trial today to start creating and collaborating.

**Use case diagram components:**

Common components include:

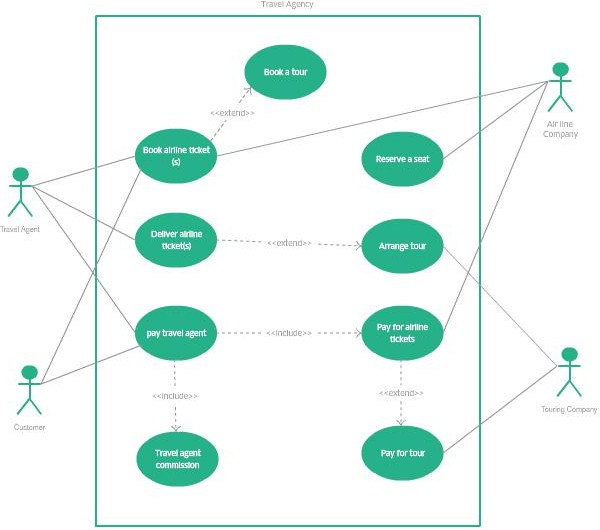
**Actors:** The users that interact with a system. An actor can be a person, an organization, or an outside system that interacts with your application or system. They must be external objects that produce or consume data.

**System:** A specific sequence of actions and interactions between actors and the system. A system may also be referred to as a scenario.

**Goals:** The end result of most use cases. A successful diagram should describe the activities and variants used to reach the goal.

**Use case Diagram Symbols and Notation:**

The notation for a use case diagram is pretty straightforward and doesn't involve as many types of symbols as other UML diagrams. Here are all the shapes you will be able to find in Lucid chart:

* **Use cases:** Horizontally shaped ovals that represent the different uses that a user might have.
* **Actors:** Stick figures that represent the people actually employing the use cases.
* **Primary Actors:** The Actor(s) using the system to achieve a goal. The Use Case documents the interactions between the system and the actors to achieve the goal of the primary actor.
* **Secondary Actors:** Actors that the system needs assistance from to achieve the primary actor’s goal. Secondary actors may or may not have goals that they expect to be satisfied by the use case, the primary actor always has a goal, and the use case exists to satisfy the primary actor.
* **Associations:** A line between actors and use cases. In complex diagrams, it is important to know which actors are associated with which use cases.
* **System boundary boxes:** A box that sets a system scope to use cases. All use cases outside the box would be considered outside the scope of that system. For example, Psycho Killer is outside the scope of occupations in the chain saw example found below.

**Figure 5: Use case Diagram**

* + 1. **Sequence Diagram:**

UML Sequence diagrams are interaction diagrams that detail how operations are carried out. As sequence diagrams can be used to capture the interaction between objects in the context of collaboration, one of the primary uses of sequence diagrams is in the transition from requirements expressed as use cases to the next and more formal level of refinement. Use cases are often refined into one or more sequence diagrams. Sequence diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent and when.

**Sequence Diagrams Captures Interaction In Different Level Of Granularity:**

* High-level interactions between user of the system and the system, between the system and other systems, or between subsystems (sometimes known as system sequence diagrams).
* The interaction that takes place in collaboration that either realizes a use case or an operation (instance diagrams or generic diagrams).
* Represent objects interact in (Model, View / Controller) MVC pattern of software framework.

**Sequence Diagram Notations:**

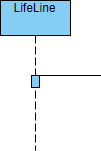
**Lifeline:**

A lifeline represents an individual participant in the Interaction.



**Activation:**

An activation is represented by a thin rectangle on a lifeline) represents the period during which an element is performing an operation. The top and the bottom of the of the rectangle are aligned with the initiation and the completion time respectively



**Messages:**

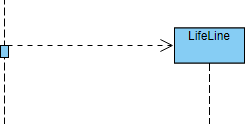
**Call Message:**

A call message defines a particular communication between lifelines of an interaction, which represents an invocation of operation of target lifeline.

UML Sequence Diagram: Call message example

**Create Message:**

A create message defines a particular communication between lifelines of an interaction, which represents the instantiation of (target) lifeline.

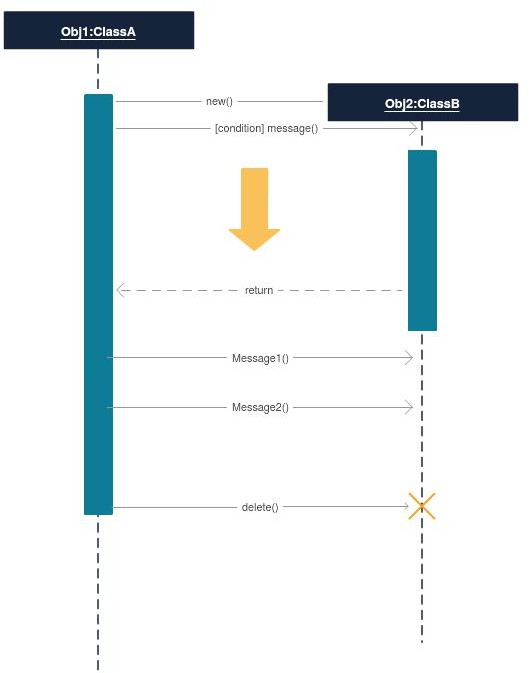


**When to draw sequence diagram:**

* Model high-level interaction between active objects in a system
* Model the interaction between object instances within a collaboration that realizes a use case
* Model the interaction between objects within a collaboration that realizes an operation
* Either model generic interactions (showing all possible paths through the interaction) or specific instances of a interaction (showing just one path through the interaction

**How to draw a sequence diagram:**

1. Identify a set of objects that will participate in the general collaboration (or use case scenario)
   1. If you derive the sequence diagram based on a scenario of a use case, select the normal scenarios first
   2. You should know the primary actor(s) who activates the use case
2. Consider what the system need to be done in order to response to the actor, when the actor send the message to the system
   1. What the system need to be handled before the return message response back from the system?
   2. E.g. A customer inserted an ATM card to the machine, the system will display "input pin number" in the normal scenario, right?
   3. Guess, what will to be handled inside the ADM by a set of objects at the "back" of the system? Something like, read and verify the ATM card (card reader), read the card information of the card holder (by the bank) and ask for the pin, or, return "invalid card type, insert another card", and etc.
   4. By this way, you will identify the candidate objects and operations of the target application for that particular scenario and you can also use this information as a basis to derive the class diagram incrementally.
3. Repeat each of the point of the scenario (or flow of event) and until you complete all the points in the scenario.



**Figure 6: Sequence Diagram**

* + 1. **Activity Diagram:**

Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc.

**Purpose of Activity Diagrams:**

The basic purpose of activity diagrams is similar to other four diagrams. It captures the dynamic behavior of the system. Other four diagrams are used to show the message flow from one object to another but activity diagram is used to show message flow from one activity to another.

Activity is a particular operation of the system. Activity diagrams are not only used for visualizing the dynamic nature of a system, but they are also used to construct the executable system by using forward and reverse engineering techniques. The only missing thing in the activity diagram is the message part.

It does not show any message flow from one activity to another. Activity diagram is sometimes considered as the flowchart. Although the diagrams look like a flowchart, they are not. It shows different flows such as parallel, branched, concurrent, and single.

**The purpose of an activity diagram can be described as**

* Draw the activity flow of a system.
* Describe the sequence from one activity to another.
* Describe the parallel, branched and concurrent flow of the system.

**How to draw an activity diagram:**

Activity diagrams are mainly used as a flowchart that consists of activities performed by the system. Activity diagrams are not exactly flowcharts as they have some additional capabilities. These additional capabilities include branching, parallel flow, swimlanes, etc. Before drawing an activity diagram, we must have a clear understanding about the elements used in activity diagram. The main element of an activity diagram is the activity itself. An activity is a function performed by the system. After identifying the activities, we need to understand how they are associated with constraints and conditions.

Before drawing an activity diagram, we should identify the following elements −

* Activities
* Association
* Conditions
* Constraints

Once the above-mentioned parameters are identified, we need to make a mental layout of the entire flow. This mental layout is then transformed into an activity diagram.

**Activity Diagram Notations:**

1. **Initial State –** The starting state before an activity takes place is depicted using the initial state.



A process can have only one initial state unless we are depicting nested activities. We use a black filled circle to depict the initial state of a system. For objects, this is the state when they are instantiated. The Initial State from the UML Activity Diagram marks the entry point and the initial Activity State.

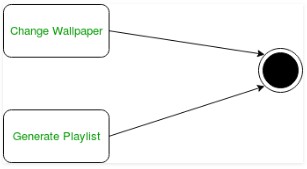
1. **Action or Activity State –** An activity represents execution of an action on objects or by objects. We represent an activity using a rectangle with rounded corners. Basically any action or event that takes place is represented using an activity.

UML-Activity-Diagram

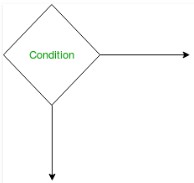
1. **Action Flow or Control flows –**Action flows or Control flows are also referred to as paths and edges. They are used to show the transition from one activity state to another.



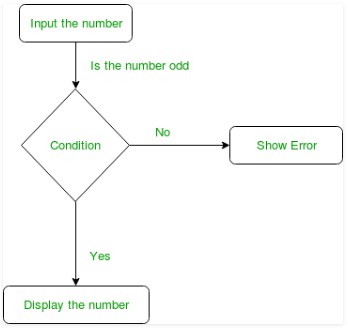
An activity state can have multiple incoming and outgoing action flows. We use a line with an arrow head to depict a Control Flow. If there is a constraint to be adhered to while making the transition it is mentioned on the arrow. Consider the example – Here both the states transit into one final state using action flow symbols i.e. arrows.



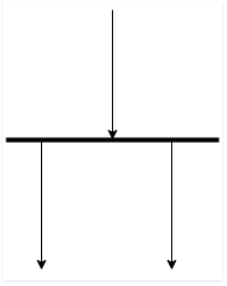
1. **Decision node and Branching –** When we need to make a decision before deciding the flow of control, we use the decision node.



The outgoing arrows from the decision node can be labeled with conditions or guard expressions. It always includes two or more output arrows.

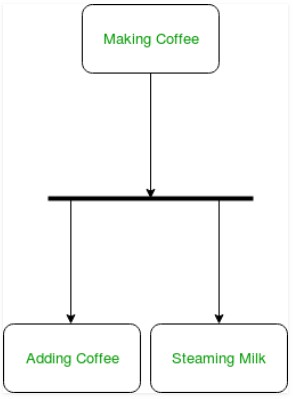


1. **Fork –** Fork nodes are used to support concurrent activities.

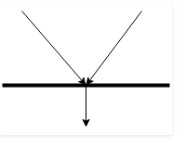


**Figure –** fork notation when we use a fork node when both the activities get executed concurrently i.e. no decision is made before splitting the activity into two parts. Both parts need to be executed in case of a fork statement. We use a rounded solid rectangular bar to represent a Fork notation with incoming arrow from the parent activity state and outgoing arrows towards then newly created activities.

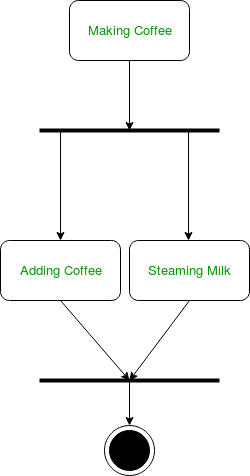
**For example:** In the example below, the activity of making coffee can be split into two concurrent activities and hence we use the fork notation.



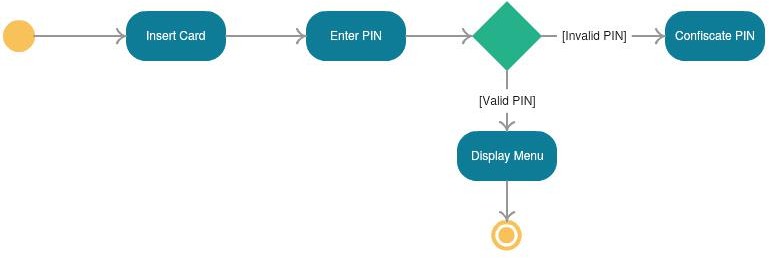
1. **Join –** Join nodes are used to support concurrent activities converging into one. For join notations we have two or more incoming edges and one outgoing edge.



**For example** – When both activities i.e. steaming the milk and adding coffee get completed, we converge them into one final activity.



1. **Final State or End State –** The state which the system reaches when a particular process or activity ends is known as a Final State or End State. We use a filled circle within a circle notation to represent the final state in a state machine diagram. A system or a process can have multiple final states.

**Figure 7: Activity Diagram**

* 1. **Data dictionary:**

Data Dictionary, also called a Data Definition Matrix, provides detailed information about the business data, such as standard definitions of data elements, their meanings, and allowable values. While a conceptual or logical Relationship Diagram will focus on the high-level business concepts, a Data Dictionary will provide more detail about each attribute of a business concept. Essentially, a data dictionary provides a tool that enables you to communicate business stakeholder requirements in such a way that your technical team can more easily design a relational database or data structure to meet those requirements. It helps avoid project mishaps such as requiring information in a field that a business stakeholder can’t reasonably be expected to provide, or expecting the wrong type of information in afield.

**Normalization:**

Normalization is a systematic approach of decomposing tables to eliminate data redundancy (repetition) and undesirable characteristics like Insertion, Update and Deletion Anomalies. It is a multi-step process that puts data into tabular form, removing duplicated data from the relation tables.

**First Normal Form (1NF):**

* A relation will be 1NF if it contains an atomic value.
* It states that an attribute of a table cannot hold multiple values. It must hold only single-valued attribute.
* First normal form disallows the multi-valued attribute, composite attribute, and their combinations.

**Example**: Relation EMPLOYEE is not in 1NF because of multi-valued attribute EMP\_PHONE.

**Second Normal Form (2NF)**

* In the 2NF, relational must be in 1NF.
* In the second normal form, all non-key attributes are fully functional dependent on the primary key

**Example:** Let's assume, a school can store the data of teachers and the subjects they teach. In a school, a teacher can teach more than one subject.

**Third Normal Form (3NF)**

* A relation will be in 3NF if it is in 2NF and not contain any transitive partial dependency.
* 3NF is used to reduce the data duplication. It is also used to achieve the data integrity.
* If there is no transitive dependency for non-prime attributes, then the relation must be in third normal form.

A relation is in third normal form if it holds at least one of the following conditions for every non-trivial function dependency X → Y.

* X is a super key.
* Y is a prime attribute, i.e., each element of Y is part of some candidate key.

**Example:** EMP\_STATE & EMP\_CITY dependent on EMP\_ZIP and EMP\_ZIP dependent on EMP\_ID. The non-prime attributes (EMP\_STATE, EMP\_CITY) transitively dependent on super key (EMP\_ID). It violates the rule of third normal form. That’s why we need to move the EMP\_CITY and EMP\_STATE to the new

<EMPLOYEE\_ZIP> table, with EMP\_ZIP as a Primary key.

**Boyce Codd normal form (BCNF)**

* BCNF is the advance version of 3NF. It is stricter than 3NF.
* A table is in BCNF if every functional dependency X → Y, X is the super key of the table.
* For BCNF, the table should be in 3NF, and for every FD, LHS is super key.

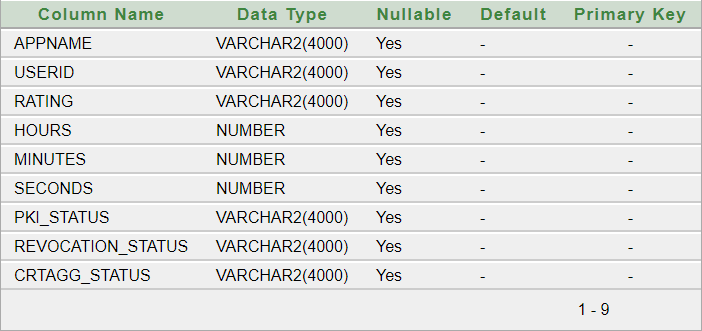
**Example:** Let's assume there is a company where employees work in more than one department.

**The Key Elements of a Data Dictionary:**

A Data Dictionary provides information about each attribute, also referred to as fields, of a data model. An attribute is a place in the database that holds information. For example, if we were to create a Data Dictionary representing the articles here on Bridging the Gap, we’d potentially have attributes for article title, article author, article category, and the article content itself. A Data Dictionary is typically organized in a spreadsheet format. Each attribute is listed as a row in the spreadsheet and each column labels an element of information that is useful to know about the attribute.

Let’s look at the most common elements included in a data dictionary.

* **Attribute Name** – A unique identifier typically expressed in business language that labels each attribute.
* **Optional/Required** – Indicates whether information is required in an attribute before a record can be saved.
* **Attribute Type** – Defines what type of data is allowable in a field. Common types include text, numeric, date/time, enumerated list,look- ups,Booleansandunique identifiers. While these are the core elements of a data dictionary, it’s not uncommon to document additional information about each element, which may include the source of the information, the table or concept in which the attribute is contained, the physical database

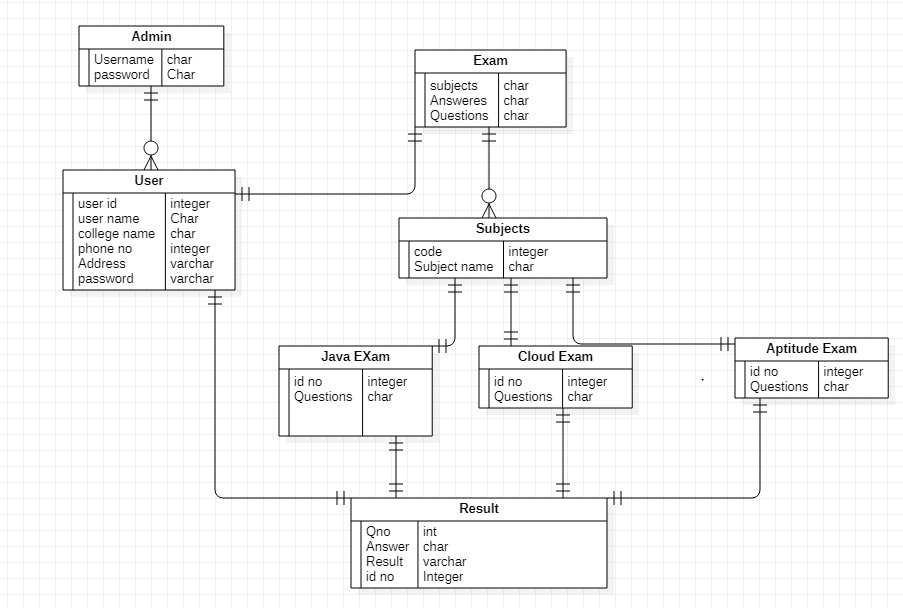


**Figure 8: Data Dictionary**

**2.6 ER-Diagram:**

An Entity Relationship (ER) Diagram is a type of flowchart that illustrates how “entities” such as people, objects or concepts relate to each other within a system.ER Diagrams are most often used to design or debug relational databases in the fields of software engineering, business information systems, education and research. Also Known as ERDs or ER models , they use a defined set of symbols such as rectangles , diamonds ovals and connecting lines to depict the interconnectedness of entities, relationships and their attributes. They mirror grammatical structure, with entities as nouns and relationships as verbs.

ER diagrams are related to data structure diagrams (DSDs) which focus on the relationships of elements within instead f relationships between entities themselves.ER diagrams also are often used in conjunction with data flow diagrams (DFDs), which map out the flow of information for processes or systems.

 **Figure 9: ER Diagram for Online Examination**

**Uses of entity relationship diagrams:**

1. **Database design:**

ER diagrams are used to model and design relational databases, in terms of logic and business rules (in a logical data model). In software engineering, an ER diagram is often an initial step in determining requirements for an information systems project .It’s also later used to model a particular database or databases. A relational database has an equivalent relational table and can potentially be expressed that way as needed.

1. **Database troubleshooting:**

ER diagrams are used to analyze existing databases to find and resolve problems in logic or deployment .Drawing the diagram should reveal where its going wrong.

1. **Business information systems:**

The diagrams are used to design or analyze relational databases used in business processes. Any business process that uses fielded data involving entities, actions and interplay can potentially benefit from a relational database. It can streamline processes, uncover information more easily and improve results.

1. **Business process re-engineering (BPR):**

ER diagrams help in analyzing databases used in business process re- engineering and in modeling in a new database setup.

1. **Education:**

Databases are today’s method of storing relational information for educational purposes and later retrieval, so ER Diagrams can be valuable in planning those data structures.

1. **Research:**

Since so much research focuses on structured data, ER diagram scan play a key role in setting up useful databases to analyze the data.

**2.7 Python: -**

What do the alphabet and the programming language Python have in common? Right, both start with ABC. If we are talking about ABC in the Python context, it's clear that the programming language ABC is meant. ABC is a general-purpose programming language and programming environment, which had been developed in the Netherlands, Amsterdam, at the CWI (Centrum Wickenden &Informatica). The greatest achievement of ABC was to influence the design of Python. Python was conceptualized in the late 1980s. Guido van Rossum worked that time in a project at the CWI, called Amoeba, a distributed operating system. In an interview with Bill Venners1, Guido van Rossum said: "In the early 1980s, I worked as an implementer on a team building a language called ABC at Centrum voor Wickenden en Informatica (CWI). I don't know how well people know ABC's influence on Python. I try to mention ABC's influence because I'm indebted to everything I learned during that project and to the people who worked on it."Later on in the same Interview, Guido van Rossum continued: "I remembered all my experience and some of my frustration with ABC. I decided to try to design a simple scripting language that possessed some of ABC's better properties, but without its problems. So I started typing. I created a simple virtual machine, a simple parser, and a simple runtime. I made my own version of the various ABC parts that I liked. I created a basic syntax, used indentation for statement grouping instead of curly braces or begin-end blocks, and developed a small number of powerful data types: a hash table (or dictionary, as we call it), a list, strings, and numbers."

What is Machine Learning: -

Before we take a look at the details of various machine learning methods, let's start by looking at what machine learning is, and what it isn't. Machine learning is often categorized as a subfield of artificial intelligence, but I find that categorization can often be misleading at first brush. The study of machine learning certainly arose from research in this context, but in the data science application of machine learning methods, it's more helpful to think of machine learning as a means of building models of data.

Fundamentally, machine learning involves building mathematical models to help understand data. "Learning" enters the fray when we give these models tunable parameters that can be adapted to observed data; in this way the program can be considered to be "learning" from the data. Once these models have been fit to previously seen data, they can be used to predict and understand aspects of newly observed data. I'll leave to the reader the more philosophical digression regarding the extent to which this type of mathematical, model-based "learning" is similar to the "learning" exhibited by the human brain. Understanding the problem setting in machine learning is essential to using these tools effectively, and so we will start with some broad categorizations of the types of approaches we'll discuss here.

Categories Of Machine Learning: -

At the most fundamental level, machine learning can be categorized into two main types: supervised learning and unsupervised learning.

Supervised learning involves somehow modeling the relationship between measured features of data and some label associated with the data; once this model is determined, it can be used to apply labels to new, unknown data. This is further subdivided into classification tasks and regression tasks: in classification, the labels are discrete categories, while in regression, the labels are continuous quantities. We will see examples of both types of supervised learning in the following section.

Unsupervised learning involves modeling the features of a dataset without reference to any label, and is often described as "letting the dataset speak for itself." These models include tasks such as clustering and dimensionality reduction. Clustering algorithms identify distinct groups of data, while dimensionality reduction algorithms search for more succinct representations of the data. We will see examples of both types of unsupervised learning in the following section.

Need for Machine Learning

Human beings, at this moment, are the most intelligent and advanced species on earth because they can think, evaluate and solve complex problems. On the other side, AI is still in its initial stage and haven’t surpassed human intelligence in many aspects. Then the question is that what is the need to make machine learn? The most suitable reason for doing this is, “to make decisions, based on data, with efficiency and scale”.

Lately, organizations are investing heavily in newer technologies like Artificial Intelligence, Machine Learning and Deep Learning to get the key information from data to perform several real-world tasks and solve problems. We can call it data-driven decisions taken by machines, particularly to automate the process. These data-driven decisions can be used, instead of using programing logic, in the problems that cannot be programmed inherently. The fact is that we can’t do without human intelligence, but other aspect is that we all need to solve real-world problems with efficiency at a huge scale. That is why the need for machine learning arises.

Challenges in Machines Learning: -

While Machine Learning is rapidly evolving, making significant strides with cybersecurity and autonomous cars, this segment of AI as whole still has a long way to go. The reason behind is that ML has not been able to overcome number of challenges. The challenges that ML is facing currently are −

Quality of data − Having good-quality data for ML algorithms is one of the biggest challenges. Use of low-quality data leads to the problems related to data preprocessing and feature extraction.

Time-Consuming task − Another challenge faced by ML models is the consumption of time especially for data acquisition, feature extraction and retrieval.

Lack of specialist persons − As ML technology is still in its infancy stage, availability of expert resources is a tough job.

No clear objective for formulating business problems − Having no clear objective and well-defined goal for business problems is another key challenge for ML because this technology is not that mature yet.

Issue of overfitting & underfitting − If the model is overfitting or underfitting, it cannot be represented well for the problem.

Curse of dimensionality − Another challenge ML model faces is too many features of data points. This can be a real hindrance.

Difficulty in deployment − Complexity of the ML model makes it quite difficult to be deployed in real life.

Applications of Machines Learning: -

Machine Learning is the most rapidly growing technology and according to researchers we are in the golden year of AI and ML. It is used to solve many real-world complex problems which cannot be solved with traditional approach. Following are some real-world applications of ML −

• Emotion analysis

• Sentiment analysis

• Error detection and prevention

• Weather forecasting and prediction

• Stock market analysis and forecasting

• Speech synthesis

• Speech recognition

• Customer segmentation

• Object recognition

• Fraud detection

• Fraud prevention

• Recommendation of products to customer in online shopping

How to Start Learning Machine Learning?

Arthur Samuel coined the term “Machine Learning” in 1959 and defined it as a “Field of study that gives computers the capability to learn without being explicitly programmed”.

And that was the beginning of Machine Learning! In modern times, Machine Learning is one of the most popular (if not the most!) career choices. According to Indeed, Machine Learning Engineer Is The Best Job of 2019 with a 344% growth and an average base salary of $146,085 per year.

But there is still a lot of doubt about what exactly is Machine Learning and how to start learning it? So this article deals with the Basics of Machine Learning and also the path you can follow to eventually become a full-fledged Machine Learning Engineer. Now let’s get started!!!

How to start learning ML?

This is a rough roadmap you can follow on your way to becoming an insanely talented Machine Learning Engineer. Of course, you can always modify the steps according to your needs to reach your desired end-goal!

Step 1 – Understand the Prerequisites

In case you are a genius, you could start ML directly but normally, there are some prerequisites that you need to know which include Linear Algebra, Multivariate Calculus, Statistics, and Python. And if you don’t know these, never fear! You don’t need a Ph.D. degree in these topics to get started but you do need a basic understanding.

(a) Learn Linear Algebra and Multivariate Calculus

Both Linear Algebra and Multivariate Calculus are important in Machine Learning. However, the extent to which you need them depends on your role as a data scientist. If you are more focused on application heavy machine learning, then you will not be that heavily focused on maths as there are many common libraries available. But if you want to focus on R&D in Machine Learning, then mastery of Linear Algebra and Multivariate Calculus is very important as you will have to implement many ML algorithms from scratch.

(b) Learn Statistics

Data plays a huge role in Machine Learning. In fact, around 80% of your time as an ML expert will be spent collecting and cleaning data. And statistics is a field that handles the collection, analysis, and presentation of data. So it is no surprise that you need to learn it!!!

Some of the key concepts in statistics that are important are Statistical Significance, Probability Distributions, Hypothesis Testing, Regression, etc. Also, Bayesian Thinking is also a very important part of ML which deals with various concepts like Conditional Probability, Priors, and Posteriors, Maximum Likelihood, etc.

(c) Learn Python

Some people prefer to skip Linear Algebra, Multivariate Calculus and Statistics and learn them as they go along with trial and error. But the one thing that you absolutely cannot skip is Python! While there are other languages you can use for Machine Learning like R, Scala, etc. Python is currently the most popular language for ML. In fact, there are many Python libraries that are specifically useful for Artificial Intelligence and Machine Learning such as Keras, TensorFlow, Scikit-learn, etc.

So if you want to learn ML, it’s best if you learn Python! You can do that using various online resources and courses such as Fork Python available Free on GeeksforGeeks.

Step 2 – Learn Various ML Concepts

Now that you are done with the prerequisites, you can move on to actually learning ML (Which is the fun part!!!) It’s best to start with the basics and then move on to the more complicated stuff. Some of the basic concepts in ML are:

(a) Terminologies of Machine Learning

• Model – A model is a specific representation learned from data by applying some machine learning algorithm. A model is also called a hypothesis.

• Feature – A feature is an individual measurable property of the data. A set of numeric features can be conveniently described by a feature vector. Feature vectors are fed as input to the model. For example, in order to predict a fruit, there may be features like color, smell, taste, etc.

• Target (Label) – A target variable or label is the value to be predicted by our model. For the fruit example discussed in the feature section, the label with each set of input would be the name of the fruit like apple, orange, banana, etc.

• Training – The idea is to give a set of inputs(features) and it’s expected outputs(labels), so after training, we will have a model (hypothesis) that will then map new data to one of the categories trained on.

• Prediction – Once our model is ready, it can be fed a set of inputs to which it will provide a predicted output(label).

(b) Types of Machine Learning

• Supervised Learning – This involves learning from a training dataset with labeled data using classification and regression models. This learning process continues until the required level of performance is achieved.

• Unsupervised Learning – This involves using unlabelled data and then finding the underlying structure in the data in order to learn more and more about the data itself using factor and cluster analysis models.

• Semi-supervised Learning – This involves using unlabelled data like Unsupervised Learning with a small amount of labeled data. Using labeled data vastly increases the learning accuracy and is also more cost-effective than Supervised Learning.

• Reinforcement Learning – This involves learning optimal actions through trial and error. So the next action is decided by learning behaviors that are based on the current state and that will maximize the reward in the future.

Advantages of Machine learning :-

1. Easily identifies trends and patterns -

Machine Learning can review large volumes of data and discover specific trends and patterns that would not be apparent to humans. For instance, for an e-commerce website like Amazon, it serves to understand the browsing behaviors and purchase histories of its users to help cater to the right products, deals, and reminders relevant to them. It uses the results to reveal relevant advertisements to them.

2. No human intervention needed (automation)

With ML, you don’t need to babysit your project every step of the way. Since it means giving machines the ability to learn, it lets them make predictions and also improve the algorithms on their own. A common example of this is anti-virus softwares; they learn to filter new threats as they are recognized. ML is also good at recognizing spam.

3. Continuous Improvement

As ML algorithms gain experience, they keep improving in accuracy and efficiency. This lets them make better decisions. Say you need to make a weather forecast model. As the amount of data you have keeps growing, your algorithms learn to make more accurate predictions faster.

4. Handling multi-dimensional and multi-variety data

Machine Learning algorithms are good at handling data that are multi-dimensional and multi-variety, and they can do this in dynamic or uncertain environments.

5. Wide Applications

You could be an e-tailer or a healthcare provider and make ML work for you. Where it does apply, it holds the capability to help deliver a much more personal experience to customers while also targeting the right customers.

Disadvantages of Machine Learning :-

1. Data Acquisition

Machine Learning requires massive data sets to train on, and these should be inclusive/unbiased, and of good quality. There can also be times where they must wait for new data to be generated.

2. Time and Resources

ML needs enough time to let the algorithms learn and develop enough to fulfill their purpose with a considerable amount of accuracy and relevancy. It also needs massive resources to function. This can mean additional requirements of computer power for you.

3. Interpretation of Results

Another major challenge is the ability to accurately interpret results generated by the algorithms. You must also carefully choose the algorithms for your purpose.

4. High error-susceptibility

Machine Learning is autonomous but highly susceptible to errors. Suppose you train an algorithm with data sets small enough to not be inclusive. You end up with biased predictions coming from a biased training set. This leads to irrelevant advertisements being displayed to customers. In the case of ML, such blunders can set off a chain of errors that can go undetected for long periods of time. And when they do get noticed, it takes quite some time to recognize the source of the issue, and even longer to correct it.

**2.8 STUDY REALATED TO PROJECT**

**2.8.1: Unsupervised Machine Learning for Managing Safety Accidents in Railway Stations**[**Hamad**](https://ieeexplore.ieee.org/author/37087229343)[**ALawad;**](https://ieeexplore.ieee.org/author/37087229343)[**Sakdirat**](https://ieeexplore.ieee.org/author/37085874601)[**Kaewunruen**](https://ieeexplore.ieee.org/author/37085874601)

For both passenger and freight transportation, railroad operations must be dependable, accessible, maintained, and safe (RAMS). In many urban areas, railway stations risk and safety accidents represent an essential safety concern for daily operations. Moreover, the accidents lead to damage to market reputation, including injuries and anxiety among the people and costs. This stations under pressure caused by higher demand which consuming infrastructure and raised the safety administration consideration. To analysing these accidents and utilising the technology such AI methods to enhance safety, it is suggested to use unsupervised topic modelling for better understand the contributors to these extreme accidents. It is conducted to optimise Latent Dirichlet Allocation (LDA) for fatality accidents in the railway stations from textual data gathered RSSB including 1000 accidents in the UK railway station. This research describes using the machine learning topic method for systematic spot accident characteristics to enhance safety and risk management in the stations and provides advanced analysing. The study evaluates the efficacy of text by mining from accident history, gaining information, lesson learned and deeply coherent of the risk caused by assessing fatalities accidents for large and enduring scale. This Intelligent Text Analysis presents predictive accuracy for valuable accident information such as root causes and the hot spots in the railway stations. Further, the big data analytics ’ improvement results in an understanding of the accidents’ nature in ways not possible if a considerable amount of safety history and not through narrow domain analysis of the accident reports. This technology renders stand with high accuracy and a beneficial and extensive new era of AI applications in railway industry safety and other fields for safety applications.

**2.8.2 Learning From Accidents: Machine Learning for Safety at Railway Stations**

In railway systems, station safety is a critical aspect of the overall structure, and yet, accidents at stations still occur. It is time to learn from these errors and improve conventional methods by utilising the latest technology, such as machine learning (ML), to analyse accidents and enhance safety systems. ML has been employed in many fields, including engineering systems, and it interacts with us throughout our daily lives. Thus, we must consider the available technology in general and ML in particular in the context of safety in the railway industry. This paper explores the employment of the decision tree (DT) method in safety classification and the analysis of accidents at railway stations to predict the traits of passengers affected by accidents. The critical contribution of this study is the presentation of ML and an explanation of how this technique is applied for ensuring safety, utilizing automated processes, and gaining benefits from this powerful technology. To apply and explore this method, a case study has been selected that focuses on the fatalities caused by accidents at railway stations. An analysis of some of these fatal accidents as reported by the Rail Safety and Standards Board (RSSB) is performed and presented in this paper to provide a broader summary of the application of supervised ML for improving safety at railway stations. Finally, this research shows the vast potential of the innovative application of ML in safety analysis for the railway industry.

**2.8.3 : A Deep Learning Approach Towards Railway Safety Risk Assessment**

Railway stations are essential aspects of railway systems, and they play a vital role in public daily life. Various types of AI technology have been utilised in many fields to ensure the safety of people and their assets. In this paper, we propose a novel framework that uses computer vision and pattern recognition to perform risk management in railway systems in which a convolutional neural network (CNN) is applied as a supervised machine learning model to identify risks. However, risk management in railway stations is challenging because stations feature dynamic and complex conditions. Despite extensive efforts by industry associations and researchers to reduce the number of accidents and injuries in this field, such incidents still occur. The proposed model offers a beneficial method for obtaining more accurate motion data, and it detects adverse conditions as soon as possible by capturing fall, slip and trip (FST) events in the stations that represent high-risk outcomes. The framework of the presented method is generalisable to a wide range of locations and to additional types of risks

**2.8.4 Analysis and prediction of railway accident risks using machine learning**

**2.8.5 Application of Classification in Machine Learning: An Analysis of Highway-Rail Grade Crossing Systems Safety**

and the destruction of the system and its environment, are at the basis of the implementation of a

"experience feedback" (REX) system considered as the essential means to promote the

improvement of safety. REX seeks to identify adverse events with a view to highlighting all the

causes that contributed to the occurrence of a particular accident and therefore to avoid at least

the reproduction of new accidents and similar incidents. Accident and incident investigation

reports provide a wealth of informative information for accident prevention. It would be

appropriate to exploit these reports in order to extract the relevant information and suggest ways

to avoid the reproduction of adverse events. In this context, knowledge of the causes of accidents

results mainly from the contribution of lessons learned and experiences gained, whether positive

or negative. However, the exploitation of information and the search for lessons from past events

is a crucial step in the REX process. This process of analyzing and using data from experience

can be facilitated if there are methods and tools available to technical investigators. It seems

advisable to consider the use of artificial intelligence (AI) techniques and in particular automatic

learning methods in order to understand the origins and circumstances of accidents and therefore

propose solutions to avoid the reproduction of similar insecurity events. The fact that the lessons

one can learn from a REX depends on the experiences of the situations experienced in the past,

constitutes in itself a key argument in favor of machine learning. Thus, the identification of

knowledge about rail accidents and incidents and share them among REX actors constitute a

process of learning sequences of undesirable events. The approach proposed in this manuscript

for the prevention of railway accidents is a hybrid method built around several algorithms and

uses several methods of automatic learning: Learning by classification of concepts, Rule-based

The harmful consequences of rail accidents, which sometimes lead to los Highway-rail grade crossing accidents are the second leading cause of fatalities within the railway transportation industry. Highway-rail grade crossing fatality statistics have failed to improve over the last decade, and in some years have increased. Because of these trends, research has found that considerable Federal funding initiatives exist to support and develop contemporary risk reduction methodologies involving machine learning and artificial intelligence. In accord with these initiatives, this thesis investigates the highway-rail grade crossing safety problem using machine learning to implement an ensemble decision treebased classification application. The resulting classifier has been validated to achieve a less than three percent false omission rate and less than fifteen percent false negative rate for fatal accident classification. Classifiers such as the type developed within this research effort have strong potential to serve as risk reduction tools for existing infrastructure throughout the United States and can be used to prioritize funding for the most at-risk crossings. The classifier would also be beneficial to new railway systems engineering efforts by using a risk-based approach for assessing potential designs.

**3. ANALYSIS**

**3.1 System Requirements Specification**

**3.1.1 Introduction:**

Trains as public transportation have been considered as safer than other means. However, passengers on trains stations sometimes face many risks because of many overlapping factors such as station operation, design, and passenger behaviors. Due to the gradually increasing demand and the heavily congested society and the state of some station’s layout and complexity in design, there are potential risks during the operation of the stations. Furthermore, Passenger, people and public safety is the main concern of the railway industry and one of the critical parts of the system. European Union put into practice Reliability, Availability, Maintainability and Safety (RAMS)as a standard in 1999 known as EN 50126. Aiming to prevent railway accidents and ensure a high level of safety in railway operations. The RAMS analyses concepts lead to minimizing the risks to acceptable levels and rise safety levels. However, that have been an urgent issue and still, the reports show several people are killed every year in the railway station, some accidents lead to injuries or fatalities

**3.1.2 existing system**

Despite the scatter of applying such method and the differences in terms been using in the literature, there is a shortage of such applications in the railway industry. Moreover, the NLP has been implemented to detect defects in the requirements documents of a railway signaling manufacturer [13].Also, for translating terms of the contract into technical specifications in the railway sector [14]. Additionally, identifying the significant factors contributing to railway accidents, the taxonomy framework was proposed using (Self-Organizing Maps – SOM), to classify human, technology, and organization factors in railway accidents [15].Likewise, association rules mining has been used to identify potential causal relationships between factorsin railway accidents [16].In the field of the machine learning and risk, safety accident, and occupational safety, there are many ML algorithms been used such as SVM, ANN, extreme learning machine (ELM), and decision tree (DT) [7], [17]. Scholars have been conducted the topic modeling in, where such method has been proved as one of themost powerful methods in data mining [18] many fields and applied in various areas such as software engineering [19], [4], [20], medical and health [21], [22], [23], [24]and linguistic science [25], [26], etc., Furthermore, from the literature It has been utilized this technique in for predictions some areas such as occupational accident [17], construction [8], [27],[28] and aviation [29], [30], [31]. For Understand occupational construction incidents in the construction and for construction injury prediction the method been conducted [32], [33], for analyzing the factors associated with occupational falls [34], for steel factory occupational incidents [35] and Cybersecurity and Data Science [36]. Moreover, From 156 construction safety accidents reports in urban rail transport in china risks information, relationships and factors been

**DISADVANTAGES;**

* The system never implemented ML algorithms been used such as SVM, ANN, extreme learning machine (ELM), and decision tree (DT) which are more accurate and efficient.
* The system didn’t implement Self-Organizing Maps–SOM model to classify human, technology, and organization factors in railway accidents

**3.1.3 Proposed System**

* This paper establishes an innovative method in the area to studies how the textual source of data of railway station accident reports could be efficiently used to extract the root causes of accidents and establish an analysis between the textual and the possible cause. where the full automated process that has ability to get the input of text and provide outputs not yet ready. Applying this method expected to come overcome issues such as aid the decision-maker in real time and extract the key information to be understandable from non-experts, better identify the details of the accident in-depth, design expert smart safety system and effective usage of the safety history records. A Such results could support in the analysis of safety and risk management to be

**ADVANTAGES**

* A DT is a determination support tool that applies a treelike pattern of decisions and their likely outcomes [40], [53]. There are many possible (ML) approaches towards safety analysis. More exactly, we train a DT to classify the accidents and the patterns that occurred in these accidents in the stations.
* The textual data have strong key information which can be used such as the time, description of the accidents, location and the range age of the victim. The time of accidents occurred been divided as the Parts of the Day for more mining to capture accurate times.

**3.1.4 Scope:**

This module explores the future horizons and potential advancements in leveraging unsupervised machine learning for managing safety accidents in railway stations. It delves into emerging trends, innovative technologies, and prospective developments that are poised to shape the landscape of railway safety in the years to come. Participants will gain insights into the latest technological innovations and advancements relevant to railway safety accident management. They will explore emerging trends such as Internet of Things (IoT), edge computing, artificial intelligence (AI), and robotics, and their potential applications in enhancing safety measures within railway stations

**3.2 Software/ Hardware Requirements**

**3.2.1 Software Requirement**

**HARDWARE REQUIREMENTS**

Processor -Pentium –IV

RAM - 4 GB (min)

Hard Disk - 20 GB

Key Board -Standard Windows Keyboard

Mouse -Two or Three Button Mouse

Monitor -SVGA

**SOFTWARE REQUIREMENTS**

* **Operating system :** Windows 7 Ultimate.
* **Coding Language :** Python.
* **Front-End :** Python.
* **Back-End :** Django-ORM
* **Designing :** Html, css, javascript.
* **Data Base :** MySQL (WAMP Server).

**3.3Modules:**

* + - Railway safety specialist
    - Remote operator

**3.3.1** **Railway safety specialist**

This module is designed to provide in-depth knowledge and skills required tobecomea proficient Railway Safety Specialist. Railway safety specialists play a critical role in ensuring the safety and security of railway operations, passengers, and employees. They are responsible for implementing and maintaining safety protocols, identifying potential risks, and developing strategies to mitigate accidents and incidents within

railway systems.

## 3.3.2 REMOTE OPERATOR

The Remote Operator Training module tailored for Unsupervised Machine Learning in Railway Safety Accident Management is crafted to empower participants with the essential competencies required to oversee and manage unmanned systems autonomously in railway safety accident prevention and response scenarios. In this specialized field, remote operators serve as pivotal figures in leveraging unsupervised machine learning technologies to enhance safety measures and mitigate accidents within railway stations.

**3.4 Functional Requirements**

**RAILWAY SAFETY SPECIALIST:**

This module is designed to provide in-depth knowledge and skills required to become a proficient Railway Safety Specialist. Railway safety specialists play a critical role in ensuring the safety and security of railway operations, passengers, and employees.

**REMOTE OPERATOR:**

The Remote Operator Training module tailored for Unsupervised Machine Learning in Railway Safety Accident Management is crafted to empower participants with the essential competencies required to oversee and manage unmanned systems autonomously in railway safety accident prevention and response scenarios

**3.5Feasibility Study**:

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

**3.5.1 Economic Feasibility:**

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

### 3.5.2 TECHNICAL FEASIBILITY

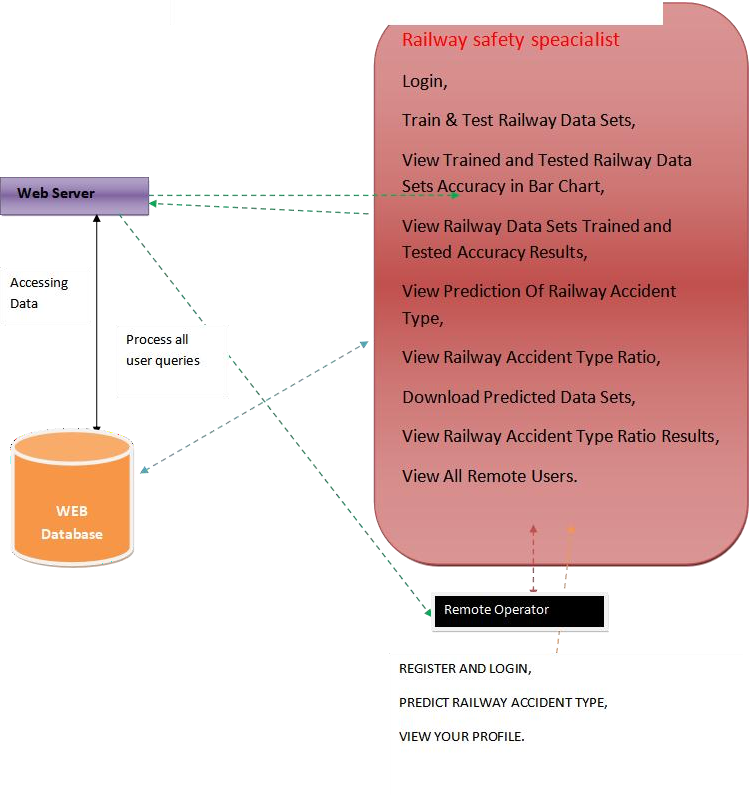
This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

### SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

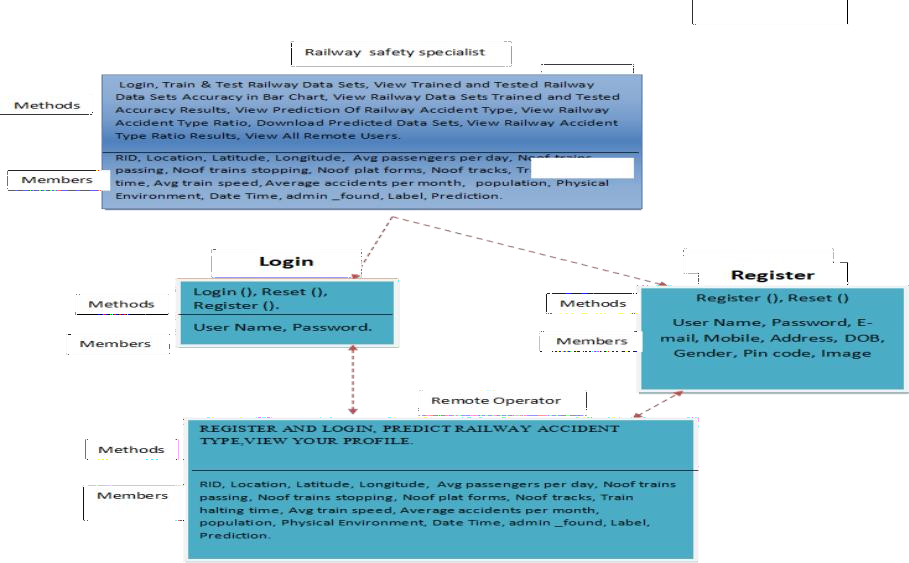
# 4. DESIGN

## 4.1 ARCHITECTURE

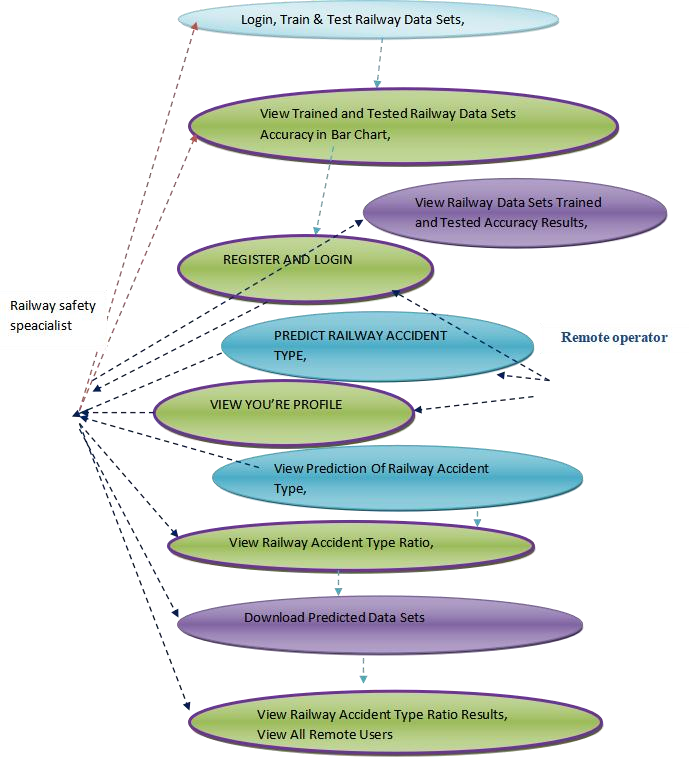


**4.2 System Design**

**4.2.1 class Diagram:**



## USECASE DIAGRAM



**4.2.3 sequence diagram**

Railway safety specialist

webserver

Remote operator

login

register and login

Train & Test railway data sets

predict railway accident type

view trained and tested railway data sets accuracy in a bar chart

view your profile

view railway and tested railway data setsvaccuracy results

view prediction of railway accident type

view railway accident type ratio

download predicted data sets

view railway accident type ratio results

view all remote users

## 4.2.4 COLLABRATION DIAGRAM

1: login

3: Train & Test railway data sets

Railway safety specialist

webserv er

5: view trained and tested railway data sets accuracy in a bar chart 7: view railway and tested railway data setsvaccuracy results

8: vie2w: rpergeidsitcetrioandofloragiilnway accident type 4: pre9d: ivcitewrairlawialwyaaycaccidceidnet ntyt ptyepe ratio

10: download predicted data sets

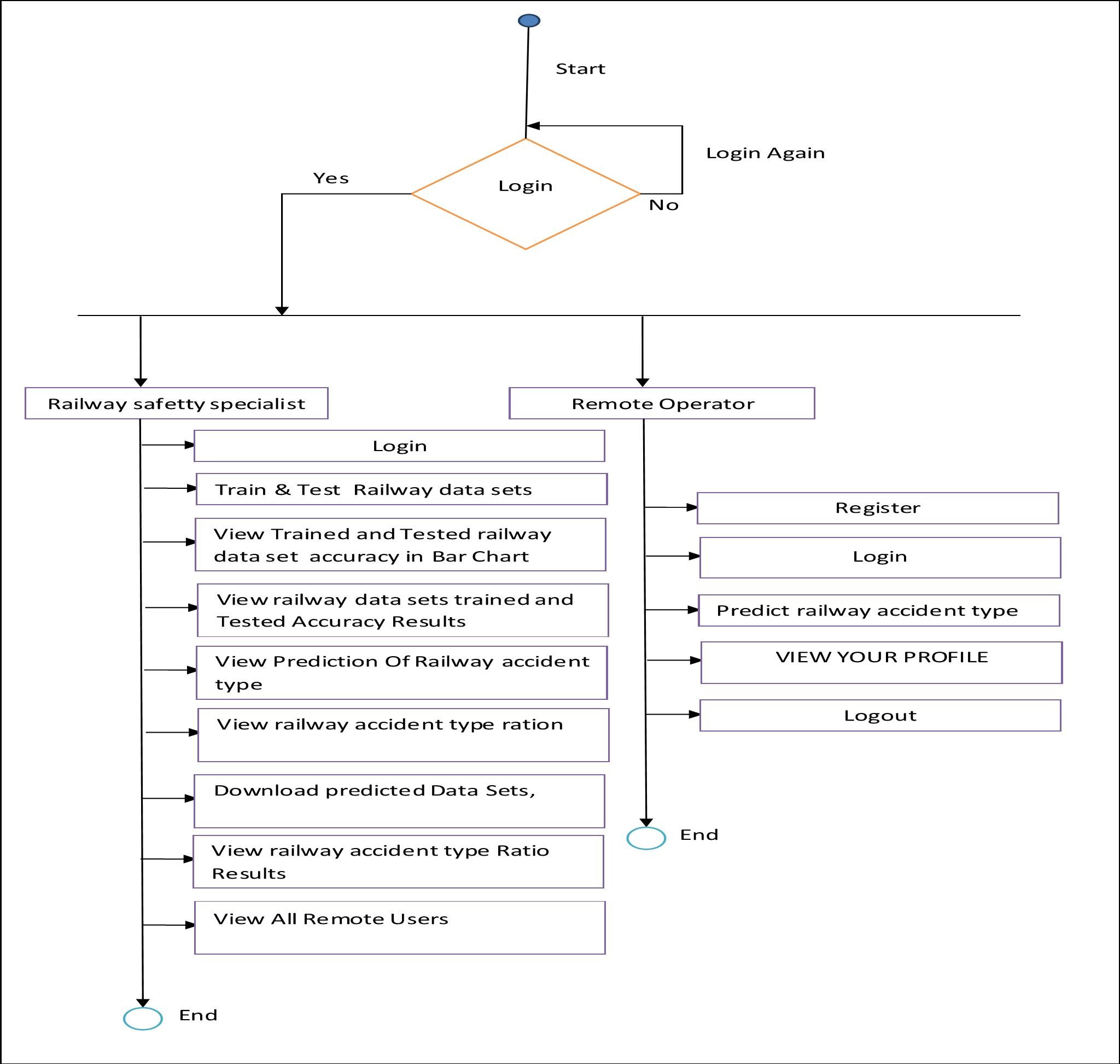
11: view railwa6y: avicecwidyeonut rtypproefilreatio results 12: view all remote users

Remote operator

## 

## 

## 4.2.5ACTIVITY DIAGRAM



login

**4..2.6 COMPONENT DIAGRAM**

Application

Railway safety specialist

Remote Opearator

predict railway acciddent type

register and login

view your profile

view railway accident type ratio results

view all remote users

downlaod predicted data sets

view prediction of railway accident

view railway accident type ratio

view railway data sets trained and tested accuracy results

view trained and tested railway data sets accuracy in bar chart

Train & tests railway data

**4.2.7DEPLOYMENT DIAGRAM**

login

view trained and tested railway data sets accuracy in bar chart

train & test railway data

view prediction of railway accident type

register and login

predicted railway accident type

view ypur profile

view railway data sets trained and tested accuracy results

view railway accident type ratio

download predicted data sets

view railway accident type ratio results

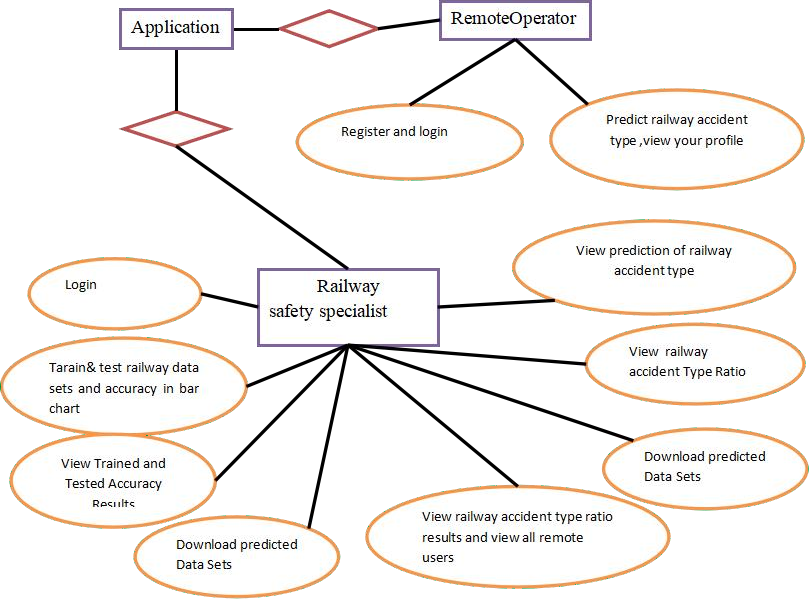
view all remote users

Railway safety specialist

application

remote operator

# 4.2.8 ER DIAGRAM



* + 1. **DATA DICTIONARY**

Database: Unsupervised machine learning for managing safety accidents in railway stations\_methods

Table name: auth\_group

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Data Type | Constraints | Description |
| id | Int(11) | Primary key | Unique Identifier |

|  |  |  |  |
| --- | --- | --- | --- |
| name | Varchar(1000) | Not null | name |

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Data Type | Constraints | Description |
| id | Int(11) | Primary key | Unique Identifier |
| Group id | Int(11) | Primary Key | Unique Identifier |
| Permission\_id | Int(11) | Primary Key | Unique Identifier |

Table Name: Auth\_group\_Permission Table Name: auth\_permission

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Data Type | Constraints | Description |
| id | Int(11) | Primary key | Unique Identifier |
| Name | Int(255) | Not Null | name |
| Content\_type\_id | Int(11) | Primary Key | Unique Identifier |
| Code name | Int(100) | Not Null | name |

Table Name: auth\_User

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Data Type | Constraints | Description |
| id | Int(11) | Primary key | Unique Identifier |
| Password | varchar(128) | Not Null | password |
| Last\_Login | Datetime(6) | Not Null | Last login |
| Is\_superuser | tinyint(1) | Not Null | Name |
| username | Varchar(150) | Not Null | Username |
| lastname | Varchar(30) | Not Null | Lastname |
| email | Varchar(150) | Not Null | Email id |

|  |  |  |  |
| --- | --- | --- | --- |
| Is\_staff | Tinyint(1) | Not Null | Staff |
| Is\_active | Tinyint(1) | Not Null | Active |
| Date\_joined | Datetime(6) | Not Null | Date and time |

Table Name: auth\_user\_groups

|  |  |  |  |
| --- | --- | --- | --- |
| Column | Data Type | Constraints | Description |
| id | Int(11) | Primary key | Unique Identifier |
| User\_id | Int(11) | Primary Key | Unique Identifier |
| Group\_id | Int(11) | Primary Key | Unique Identifier |

**5. SYSTEM IMPLEMENTATION**

**5.1 Organization Impact**

## RAILWAY SAFETY SPECIALIST

This module is designed to provide in-depth knowledge and skills required to become a proficient Railway Safety Specialist. Railway safety specialists play a critical role in ensuring the safety and security of railway operations, passengers, and employees. They are responsible for implementing and maintaining safety protocols, identifying potential risks, and developing strategies to mitigate accidents and incidents within railway systems.

## 5.2 REMOTE OPERATOR

The Remote Operator Training module tailored for Unsupervised Machine Learning in Railway Safety Accident Management is crafted to empower participants with the essential competencies required to oversee and manage unmanned systems autonomously in railway safety accident prevention and response scenarios. In this specialized field, remote operators serve as pivotal figures in leveraging unsupervised machine learning technologies to enhance safety measures and mitigate accidents within railway stations.

**5.3 coding :**

{% extends 'SProvider/Header.html' %}

{% block researchblock %}

<link rel="icon" href="images/icon.png" type="image/x-icon" />

<link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet">

<link href="https://fonts.googleapis.com/css?family=Righteous" rel="stylesheet">

<link href="https://fonts.googleapis.com/css?family=Fredoka+One" rel="stylesheet">

<style>

body {background-color:#000000;}

.container-fluid {padding:50px;}

.container{background-color:white;padding:50px; } #title{font-family: 'Fredoka One', cursive;

}

.text-uppercase{

font-family: 'Righteous', cursive;

}

input{

font-family:'Russo One', sans-serif; font-size:15px;

}

.style1 {color: #FF0000}

</style>

<body>

<div class="container-fluid">

<div class="contact”>

<div class="row">

<div class="col-md-5">

<form role="form" method="POST" >

{% csrf\_token %}

<fieldset>

<p class="text-uppercase pull-center style1">VIEW ALL REMOTE USERS !!! </p>

<div class="viewclients">

<table>

<tr>

<th bgcolor="#FF0000"style="color:yellow">USER NAME</th>

<th bgcolor="#FF0000"style="color:yellow">EMAIL</th>

<th bgcolor="#FF0000"style="color:yellow">Gender</th>

<! null >

</div>

</div>

</div>

</div>

{% endblock %}

HEADER.HTML

<!DOCTYPE html>

{% load static %}

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Service Provider</title>

<link href="https://fonts.googleapis.com/css?family=Russo+One" rel="stylesheet">

<style>

body{background: url("{% static 'bg.jpg' %}"); background-size:cover;

font-family: 'Russo One', sans-serif;

background-color: #000000;

}

h1{ color:white;

}

.topnav { overflow: hidden;

background-color: #812;

}

.topnav a { float: left;

color: #FFFFFF; text-align: center; padding: 14px 16px;

text-decoration: none; font-size: 12px;

}

.topnav a:hover { background-color: #ddd; color: black;

}

.topnav a.active { background-color: #8e4fd1; color: white;

}

.style1 {color: #FF0000}

</style>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8"></head>

<body>

<h1 align="center" class="style1">

Unsupervised Machine Learning for Managing Safety Accidents in Railway Stations</h1>

<div class="topnav">

<a href="{% url 'train\_model' %}">Train & Test Railway Data Sets </a>

<a href="{% url 'likeschart' 'bar' %}">View Trained and Tested Railway Data Sets Accuracy in Bar Chart </a>

<a href="{% url 'charts1' 'line' %}"> View Railway Data Sets Trained and Tested Accuracy Results </a>

<a href="{% url 'View\_Predicted\_Accident\_Type\_Details' %}">View Prediction Of Railway Accident Type</a>

<a href="{% url 'View\_Predicted\_Accident\_Type\_Ratio' %}">View Railway Accident Type Ratio </a>

<a href="{% url 'Download\_Predicted\_DataSets' %}">Download Predicted Data Sets</a>

<a href="{% url 'charts' 'line' %}">View Railway Accident Type Ratio Results </a>

<a href="{% url 'View\_Remote\_Users' %}">View All Remote Users</a>

<a href="{% url 'index' %}">Logout</a>

</div>

<div class="mainholder">

{% block researchblock %}

{% endblock %}

</div>

</body>

</html>

LOGIN.HTML

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML1.0Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

{% load static %}

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<title>Login</title>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link rel="stylesheet" type="text/css" href="{% static 'style.css'%} " />

<link rel="stylesheet" type="text/css" href="{% static 'coin-slider.css'%}" />

<script type="text/javascript" src="{% static 'cufon-yui.js'%} "></script>

<script type="text/javascript" src="{% static 'cufon-aller.js'%}"></script>

<script type="text/javascript" src="{% static 'jquery-1.4.2.min.js'%}"></script>

<script type="text/javascript" src="{% static 'script.js'%}"></script>

<script type="text/javascript" src="{% static 'coin-slider.min.js'%}"></script>

<style type="text/css">

<!--

.style5 {

font-size: 24px; color: #FF0000;

}

.style12 {font-weight: bold}

.style13 {font-size: 24px; color: #FF0000; font-weight: bold; }

.style16 {color: #FF0000}

.style20 {color: #FF0000; font-weight: bold; }

-->

</style>

</head>

<body>

<div class="main">

<div class="header">

<div class="header\_resize">

<div class="menu\_nav">

<p>&nbsp;</p>

</div>

<div class="mainbar">

<h1 align="center"><a href="index.html"><span class="content style5">

Unsupervised Machine Learning for Managing Safety Accidents in Railway Stations</span></a></h1>

<div class="tab-content tab-space">

<div class="tab-pane active" id="preview-alerts">

<link href="https://fonts.googleapis.com/css?family=Open+Sans:300,400,600,700" rel="stylesheet" />

<scriptsrc="https://kit.fontawesome.com/42d5adcbca.js"crossorigin="anonymous"></script>

<linkhref="https://unpkg.com/soft-ui-design-system@1.0.1/assets/css/soft-design-system.min.css"rel="stylesheet"/><div class="container py-5">

<div class="row">

<div class="alert alert-primary text-white font-weight-bold" role="alert">

<p align="center"><span class="active"><span class="style12"><a href="

{% url 'index' %}">Home| </a><a href="{% url 'login' %}">

Remote User </a>|<a href="{% url 'serviceproviderlogin' %}"> Service Provider </a></span></span></p>

</div>

<div>

</div>

<img src="{% static 'Banner.jpg'%}" width="1297" height="355" alt="" class="fl" />

</div>

<div class="clr"></div>

<div class="slider">

</div>

<div class="clr"></div>

</div>

</div>

<div class="content">

<div class="content\_resize">

<div class="mainbar">

<div class="article">

<h2 align="center" class="style13">Unsupervised machinelearning, topic model,accidents analysis, railway station, safety...</h2>

<div class="img">

<div align="center"><form method="POST" role="form">

{% csrf\_token %}

<fieldset>

{% load static %}

<img src="{% static '/login.jpg' %}" alt="My image">

<p class="style5">Login Using Your Account: </p>

<div class="form-group">

<input type="text"name="username" placeholder="User Name" required>

<br />

<br />

</div>

<div class="form-group">

<input type="submit"name="submit1" class="btn btn-md" value="Login">

<br />

</p>

</div>

<div>

<button>

<span class="style16"><strong>Are You New User !!</strong></span><span class="style20">! </span><a href="{% url 'Register1' %}">REGISTER</a></button>

</div>

</fieldset>

</form></div>

</div>

<div class="mainbar">

<div class="tab-content tab-space">

<div class="tab-pane active" id="preview-alerts">

<link href="https://fonts.googleapis.com/css?family=Open+Sans:300,400,600,700" rel="stylesheet" />

<scriptsrc="https://kit.fontawesome.com/42d5adcbca.js"crossorigin="anonymous"></script>

<linkhref="https://unpkg.com/soft-ui-design-system@1.0.1/assets/css/soft-design-system.min.css"rel="stylesheet" /><div class="container py-5">

<div class="row">

<div class="alert alert-primary text-white font-weight-bold" role="alert">

<p align="center"><span class="active"><span class="style12"><a href="{% url 'login' %}">

Home| </a><a href="{% url 'login' %}">

Remote User </a>|<a href="{% url 'serviceproviderlogin' %}"> Service Provider </a></span></span></p>

</div>

<div>

</div>

<div class="post\_content"></div>

<div class="clr"></div>

</div>

</div>

<div class="sidebar">

<div class="searchform"></div>

<div class="clr"></div>

</div>

<div class="clr"></div>

</div>

</div>

<div class="fbg"></div>

<div class="footer"></div>

</div>

<div align=center></div>

</body>

</html>

INDEX.HTML

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

{% load static %}

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<title>Home Page</title>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link rel="stylesheet" type="text/css" href="{% static 'style.css'%} " />

<link rel="stylesheet" type="text/css" href="{% static 'coin-slider.css'%}" />

<script type="text/javascript" src="{% static 'cufon-yui.js'%} "></script>

<script type="text/javascript" src="{% static 'cufon-aller.js'%}"></script>

<script type="text/javascript" src="{% static 'jquery-1.4.2.min.js'%}"></script>

<script type="text/javascript" src="{% static 'script.js'%}"></script>

<script type="text/javascript" src="{% static 'coin-slider.min.js'%}"></script>

<style type="text/css">

<!--

.style5 {

font-size: 24px;

color: #FF0000;

}

.style12 {font-weight: bold}

.style13 {font-size: 24px; color: #FF0000; font-weight: bold; }

.style16 {

color: #FF0000; font-weight: bold;

}

-->

</style>

</head>

<body>

<div class="main">

<div class="header">

<div class="header\_resize">

<div class="menu\_nav">

<p>&nbsp;</p>

</div>

<div class="mainbar">

<h1 align="center"><a href="index.html"><span class="content style5">

Unsupervised Machine Learning for Managing Safety Accidents in Railway Stations</span></a></h1>

<div class="tab-content tab-space">

<div class="tab-pane active" id="preview-alerts">

<link href="https://fonts.googleapis.com/css?family=Open+Sans:300,400,600,700" rel="stylesheet" />

<scriptsrc="https://kit.fontawesome.com/42d5adcbca.js"crossorigin="anonymous"></script>

<linkhref="https://unpkg.com/soft-ui-design-system@1.0.1/assets/css/soft-design-system.min.css"rel="stylesheet" /><div class="container py-5">

<div class="row">

<div class="alert alert-primary text-white font-weight-bold" role="alert">

<p align="center"><span class="active"><span class="style12">

<a href="{% url 'index' %}">Home| </a>

<a href="{% url 'login' %}">Remote User </a>|

<a href="{% url 'serviceproviderlogin' %}"> Service Provider </a>

</span>

</span>

</p>

</div>

<div>

</div>

<img src="{% static 'Banner.jpg'%}" width="1297" height="421" alt="" class="fl" />

</div>

<div class="clr"></div>

<div class="slider">

</div>

<div class="clr"></div>

</div>

</div>

<div class="content">

<div class="content\_resize">

<div class="mainbar">

<div class="article">

<h2 align="center" class="style13">Unsupervised machine learning,topic model,accidents analysis, railway station, safety..</h2>

<div class="img">

<div align="center"><img src="{% static 'img1.jpg'%}" width="630" height="221" alt="" class="fl" /> </div>

</div>

<div class="post\_content">

<p align="center" class="style16">For both passenger and freight transportation, railroad operations must be dependable, accessible, maintained, and safe (RAMS). In many urban areas, railway stations risk and safety accidents<br />

represent an essential safety concern for daily operations. Moreover, the accidents lead to damage to market reputation, including injuries and anxiety among the people and costs. This stations under pressure caused by higher demand which consuming infrastructure and raised the safety administration consideration. To analysing these accidents and utilising the technology such AI methods to enhance safety, it is suggested to use unsupervised topic modelling for better understand the contributors to these extreme accidents. It is conducted to optimise Latent

Dirichlet Allocation (LDA) for fatality accidents in the railway stations from textual data gathered RSSB including 1000 accidents in the UK railway station. This research describes using the machine learning topic method for systematic spot accident characteristics to enhance safety and risk management in the stations and provides advanced analysing. The study evaluates the efficacy of text by mining from accident history, gaining information, lesson learned and deeply coherent of the risk caused by assessing fatalities accidents for large and enduring scale. This Intelligent Text Analysis presents predictive accuracy for valuable accident information such as root causes and the hot spots in the railway stations. Further, the big data analytics ’ improvement results in an understanding of the accidents’ nature in ways not possible if a considerable amount of safety history and not through narrow domain analysis of the accident reports. This technology renders stand with high accuracy and a beneficial and extensive new era of AI applications in railway industry safety and other fields for safety applicatons...</p>

</div>

<div class="tab-content tab-space">

<div class="tab-pane active" id="preview-alerts">

<link href="https://fonts.googleapis.com/css?family=Open+Sans:300,400,600,700" rel="stylesheet" />

<scriptsrc="https://kit.fontawesome.com/42d5adcbca.js"crossorigin="anonymous"></script><linkhref="https://unpkg.com/soft-ui-design-system@1.0.1/assets/css/soft-design-system.min.css" rel="stylesheet" /><div class="container py-5">

<div class="row">

<div class="alert alert-primary text-white font-weight-bold" role="alert">

<p align="center"><span class="active"><span class="style12"><a href="{% url 'login' %}">Home| </a><a href="{% url 'login' %}">Remote User </a>|<a href="{% url 'serviceproviderlogin' %}"> Service Provider </a></span></span></p>

</div>

<div>

</div>

<div class="clr"></div>

</div>

</div>

<div class="sidebar">

<div class="searchform"></div>

<div class="clr"></div>

</div>

<div class="clr"></div>

</div>

</div>

<div class="fbg"></div>

<div class="footer"></div>

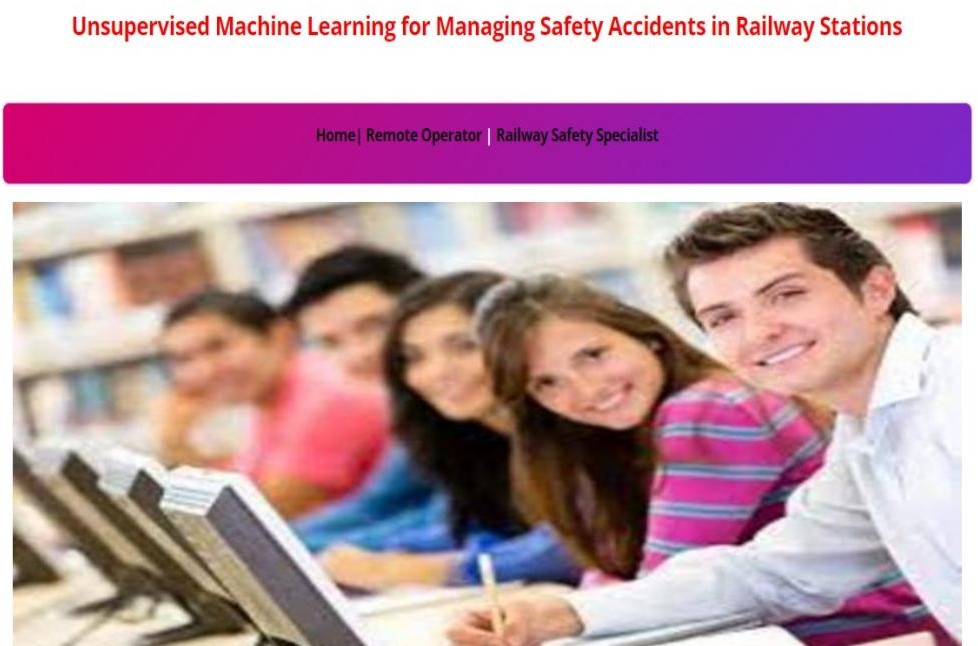
</div>

<div align=center></div>

</body>

</html>

## SCREENSHOTS



**HOME PAGE:** It is the home screen of the project, if we need to login the user the process

starts in this screen only.

**FIG-1: Home page**



**Login Page:** It is the client login page by giving username and password the user will be

logged in to the user page. And this page also have option to register the new user.

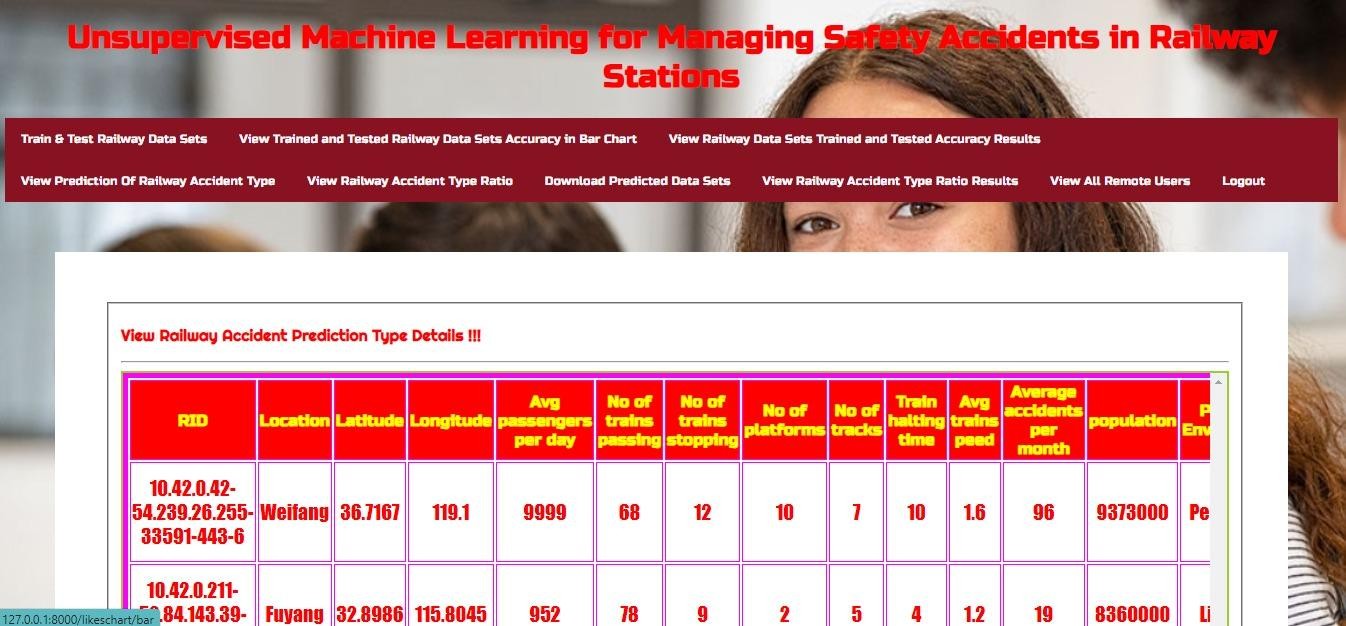
**FIG-2** Login railway officer



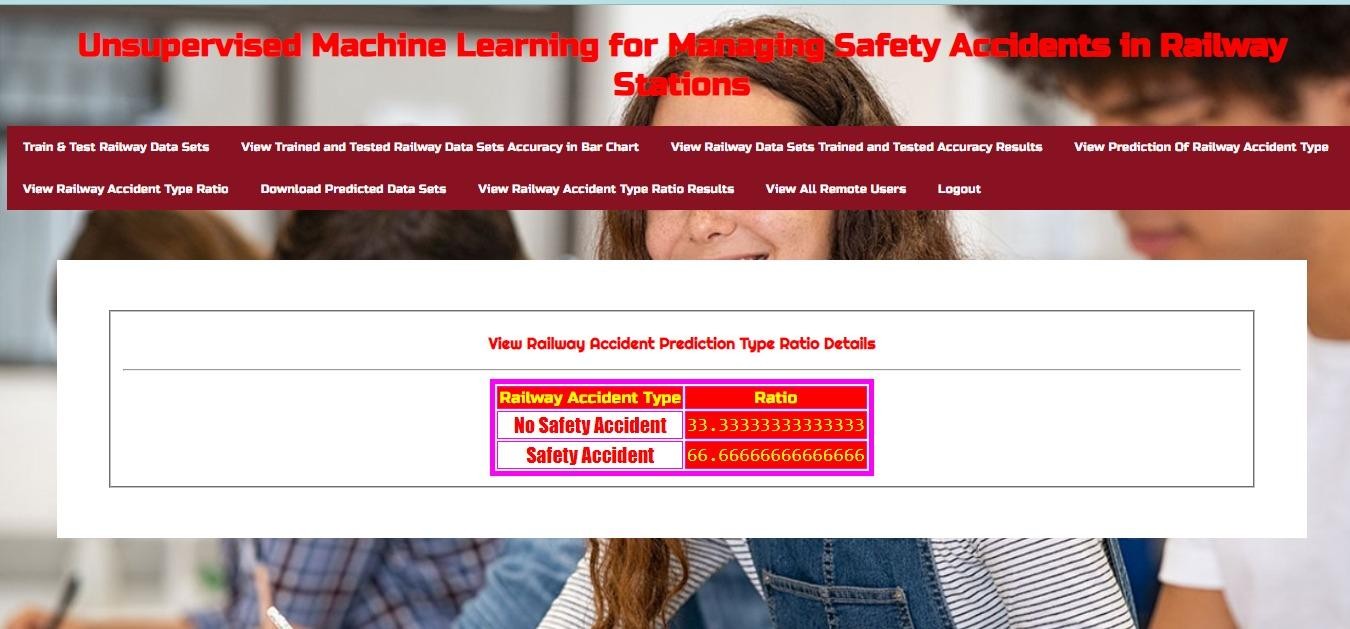
**Service operator login** : It is the user registration page where the new user will be register

by giving the details like usename, password, mail, mobile no etc.Then new user will added to the database.

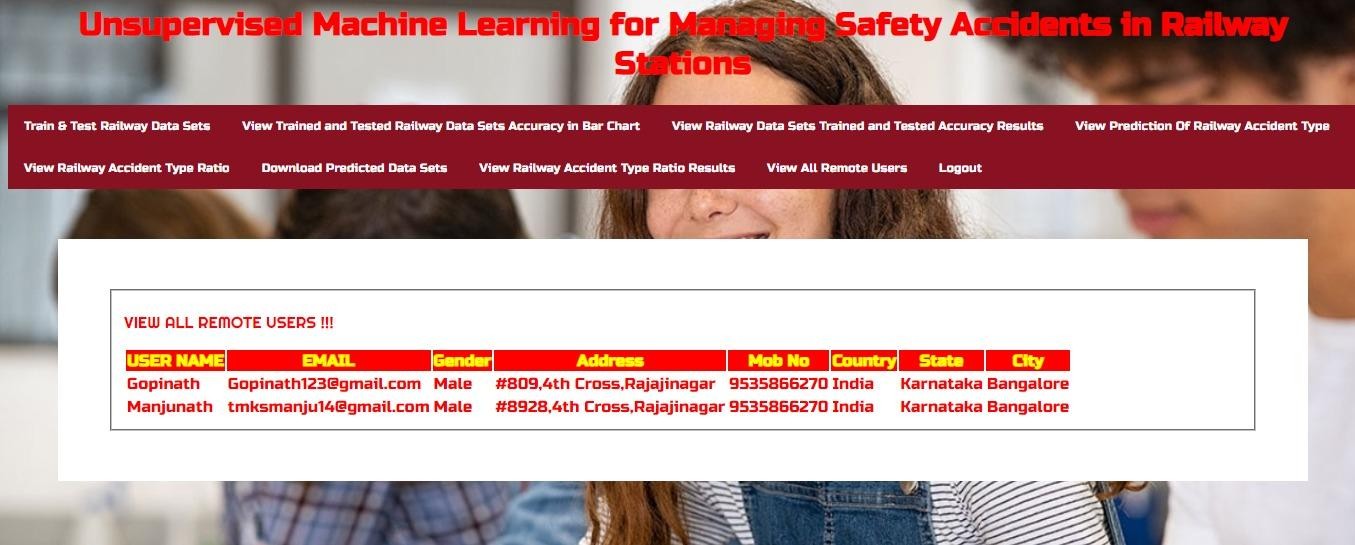
**FIG-3** Service operator login



**FIG-4** View railway prediction type detail



**FIG-5** View railway accident type ratio details



In this page the use admin can view all the registered remote

**FIG-7** View all remote users

# 6. TESTING

## Software Testing

## 6.2 SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

## 6.3 TYPES OF TESTS UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

## INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the

combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

## FUNCTIONAL TEST

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

### Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted. Invalid Input : identified classes of invalid input must be rejected. Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised. Systems/Procedures : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

## 6.4 SYSTEM TEST

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

## WHITE BOX TESTING

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas t be reached from a black box level.

## BLACK BOX TESTING

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

## UNIT TESTING

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

## TEST STRATEGY AND APPROACH

Field testing will be performed manually and functional tests will be written in

detail.

### Test objectives

* All field entries must work properly.
* Pages must be activated from the identified link.
* The entry screen, messages and responses must not be delayed.

### Features to be tested

* Verify that the entries are of the correct format
* No duplicate entries should be allowed
* All links should take the user to the correct page.

### Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

## ACCEPTANCE TESTING

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

**Test Results:** All the test cases mentioned above passed successfully. No defects encountered.

### TEST CASES: Test case1:

Test case for Login form:

|  |  |
| --- | --- |
| FUNCTION: | LOGIN |
| EXPECTED RESULTS: | Should Validate the user and check his  existence in database |
| ACTUAL RESULTS: | Validate the user and checking the user  against the database |
| LOW PRIORITY | No |
| HIGH PRIORITY | Yes |

### Test case2:

Test case for User Registration form:

|  |  |
| --- | --- |
| FUNCTION | REMOTE **10.2 SCREENSHOTS**  OPERATOR REGISTRATION |
| EXPECTED RESULTS: | Should check if all the fields are filled by the  user and saving the user to database. |
| ACTUAL RESULTS: | Checking whether all the fields are filled by  user or not through validations and saving user. |
| LOW PRIORITY | No |
| HIGH PRIORITY | Yes |

### Test case3:

Test case for Change Password:

When the old password does not match with the new password , then this results in displaying an error message as “ OLD PASSWORD DOES NOT MATCH WITH THE NEW PASSWORD”.

### Test case 4:

Test case for Forget Password:

When a user forgets his password he is asked to enter Login name, ZIP code, Mobile number. If these are matched with the already stored ones then user will get his Original password.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Modu le** | **Functio nality** | **Test Case** | **Expected Results** | **Actual Results** | **Res ult** | **Priori ty** |
| User | Login Usecase | 1. Navigate To Www.Sample.Co m 2. Click On Submit Button Without Entering Username and Password | A Validation Should Be As Below “Please Enter Valid Username & Password” | A  Validation Has Been Populated As Expected | Pass | High |
|  |  | 1. aNavigate To Www.Sample.Co m 2. Click On Submit Button With Out Filling Password And With Valid Username | A Validation Should Be As Below “Please Enter Valid Password Or Password Field Can Not Be Empty “ | A  Validation Is Shown As Expected | Pass | High |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | 1. NNavigate To Www.Sample.Co m 2. Enter Both Username And Password Wrong   And Hit Enter | A Validation Shown As Below “The Username Entered Is Wrong” | A  Validation Is Shown As Expected | Pass | High |
|  |  | 1. Navigate To Www.Sample.Co m 2. Enter Validate Username And Password And   Click On Submit | Validate Username And Password In DataBase And Once If They Correct Then Show The Main Page | Main Page/ Home Page Has Been Displayed | Pass | High |

## 7.CONCLUSION

**CONCLUSION**

In this paper, since the phenotypic numerical features contained in agricultural texts have practical significance, combined with the experimental tasks of this study, we proposed a classification method of agricultural texts based on dynamic fusion of multiple features. Firstly, three convolution kernels with different sizes are used to obtain more abundant local semantic features at different levels of wheat cold resistance text, and Bi-LSTM with attention-fusion mechanism is used to obtain important semantic expressions between contexts in the global scope. Then independent numerical features are constructed by using the text of wheat traits with numerical expression. Attention mechanism is introduced again in the process of multi-feature dynamic fusion to dynamically adjust the weights of different semantic expressions in the process of feature fusion and capture the key information beneficial to text classification in different levels of features. The performance of agricultural text classification shows that the proposed multi-feature dynamic fusion method fully excavates the potential semantic features in the text data, which is effective for the agricultural text classification task with phenotypic values. The next work will expand the data set and sample equalization to improve the performance of wheat cold resistance text classification and recognition.

## 8.FUTURE ENHANCEMENT

This module explores the future horizons and potential advancements in leveraging unsupervised machine learning for managing safety accidents in railway stations. It delves into emerging trends, innovative technologies, and prospective developments that are poised to shape the landscape of railway safety in the years to come. Participants will gain insights into the latest technological innovations and advancements relevant to railway safety accident management. They will explore emerging trends such as Internet of Things (IoT), edge computing, artificial intelligence (AI), and robotics, and their potential applications in enhancing safety measures within railway stations

# 9.BIBLIOGRAPHY

1. S. Terabe, T. Kato, H. Yaginuma, N. Kang, and K. Tanaka, ‘‘Risk assessment model for railway passengers on a crowded platform,’’ *Transp.Res. Rec., J. Transp. Res. Board*, vol. 2673, no. 1, pp. 524–531, Jan. 2019, doi: 10.1177/0361198118821925.
2. *Annual Health and Safety Report 19/2020*, RSSB, London, U.K., 2020.
3. D. M. Blei, ‘‘Probabilistic topic models,’’ *Commun. ACM*, vol. 55, no. 4, pp. 77–84, Apr. 2012, doi: 10.1145/2133806.2133826.
4. M. Gethers and D. Poshyvanyk, ‘‘Using relational topic models to capture coupling among classes in object-oriented software systems,’’ in *Proc. IEEE Int. Conf. Softw. Maintenance*, Sep. 2010, pp. 1–10, doi: 10.1109/ICSM.2010.5609687.
5. D. M. Blei, A. Y. Ng, and M. I. Jordan, ‘‘Latent Dirichlet allocation,’’

*J. Mach. Learn. Res.*, vol. 3, nos. 4–5, pp. 993–1022, Mar. 2003, doi:10.1016/B978-0-12-

411519-4.00006-9.

1. H. Alawad, S. Kaewunruen, and M. An, ‘‘A deep learning approach

towards railway safety risk assessment,’’ *IEEE Access*, vol. 8,pp. 102811–102832, 2020, doi: 10.1109/ACCESS.2020.2997946.

1. H. Alawad, S. Kaewunruen, and M. An, ‘‘Learning from accidents:

Machine learning for safety at railway stations,’’ *IEEE Access*, vol. 8,pp. 633–648, 2020, doi: 10.1109/ACCESS.2019.2962072.

1. A. J.-P. Tixier, M. R. Hallowell, B. Rajagopalan, and D. Bowman, ‘‘Automated content analysis for construction safety: A natural language

processing system to extract precursors and outcomes from unstructured

injury reports,’’ *Autom. Construct.*, vol. 62, pp. 45–56, Feb. 2016, doi:10.1016/j.autcon.2015.11.001.

1. J. Sido and M. Konopik, ‘‘Deep learning for text data on mobile devices,’’ in *Proc. Int. Conf. Appl. Electron.*, Sep. 2019, pp. 1–4, doi: 10.23919/AE.2019.8867025.
2. A. Serna and S. Gasparovic, ‘‘Transport analysis approach based on big

data and text mining analysis from social media,’’ *Transp. Res. Proc.*,vol. 33, pp. 291–298, Jan. 2018, doi: 10.1016/j.trpro.2018.10.105.

1. P. Hughes, D. Shipp, M. Figueres-Esteban, and C. van Gulijk, ‘‘From free-text to structured safety management: Introduction of a semiautomated

classification method of railway hazard reports to elementson a bow-tie diagram,’’ *Saf. Sci.*, vol. 110, pp. 11–19, Dec. 2018, doi:10.1016/j.ssci.2018.03.011.

1. A. Chanen, ‘‘Deep learning for extracting word-level meaning from safety

report narratives,’’ in *Proc. Integr. Commun. Navigat. Surveill. (ICNS)*,Apr. 2016, pp. 5D2-1– 5D2-15, doi: 10.1109/ICNSURV.2016.7486358.

1. A. Ferrari, G. Gori, B. Rosadini, I. Trotta, S. Bacherini, A. Fantechi,

and S. Gnesi, ‘‘Detecting requirements defects with NLP patterns: Anindustrial experience in the railway domain,’’ *Empirical Softw. Eng.*, vol. 23, no. 6, pp. 3684–3733, Dec. 2018, doi: 10.1007/s10664-018-9596-7.

1. G. Fantoni, E. Coli, F. Chiarello, R. Apreda, F. Dell’Orletta, and

G. Pratelli, ‘‘Text mining tool for translating terms of contract into technical specifications: Development and application in the railway sector,’’ *Comput. Ind.*, vol. 124, Jan. 2021, Art. no. 103357, doi: 10.1016/j.compind.2020.103357.

1. G. Yu, W. Zheng, L. Wang, and Z. Zhang, ‘‘Identification of significant factors contributing to multi-attribute railway accidents dataset (MARA-D) using SOM data mining,’’ in *Proc. 21st Int. Conf. Intell. Transp. Syst.(ITSC)*, Nov. 2018, pp. 170–175, doi: 10.1109/ITSC.2018.8569336.
2. Y. Wang, W. Zheng, H. Dong, and P. Gao, ‘‘Factors correlation mining on railway accidents using association rule learning algorithm,’’ in *Proc. IEEE 23rd Int. Conf. Intell. Transp. Syst.*, vol. 22, Sep. 2020, pp. 1–6, doi: 10.1109/ITSC45102.2020.9294317.
3. S. Sarkar, S. Vinay, R. Raj, J. Maiti, and P. Mitra, ‘‘Application of optimized machine learning techniques for prediction of occupational accidents,’’ *Comput. Oper. Res.*, vol. 106, pp. 210–224, Jun. 2019, doi: 10.1016/j.cor.2018.02.021.
4. H. Jelodar,Y.Wang, C.Yuan, X. Feng, X. Jiang,Y. Li, and L. Zhao, ‘‘Latent Dirichlet allocation (LDA) and topic modeling: Models, applications,

a survey,’’ *Multimedia Tools Appl.*, vol. 78, no. 11, pp. 15169–15211, Jun. 2019, doi: 10.1007/s11042-018-6894-4.

1. S. W. Thomas, ‘‘Mining software repositories using topic models,’’ in *Proc. 33rd Int. Conf. Softw. Eng. (ICSE)*, May 2011, pp. 1138–1139, doi: 10.1145/1985793.1986020.
2. H. U. Asuncion, A. U. Asuncion, and R. N. Taylor, ‘‘Software traceability with topic modeling,’’ in *Proc. ACM/IEEE 32nd Int. Conf. Softw. Eng.*,vol

. 1, May 2010, pp. 95–104, doi: 10.1145/1806799.1806817.

1. Z. Jiang, X. Zhou, X. Zhang, and S. Chen, ‘‘Using link topic model to analyze traditional Chinese medicine clinical symptom-herb regularities,’’

in *Proc. IEEE 14th Int. Conf. e-Health Netw., Appl. Services (Healthcom)*, Oct. 2012, pp. 15–18, doi: 10.1109/HealthCom.2012.6380057.

1. M. J. Paul and M. Dredze, ‘‘You are what you tweet: Analyzing Twitter for public health,’’ in *Proc. Int. AAAI Conf. Weblogs Social Media (ICWSM)*,

2011, pp. 265–272.

1. W. Zhao, W. Zou, and J. J. Chen, ‘‘Topic modeling for cluster analysis of large biological and medical datasets,’’ *BMC Bioinf.*, vol. 15, no. 11,pp. 1–11, Oct. 2014, doi: 10.1186/1471-2105-15-S11-S11.
2. H.-M. Lu, C.-P. Wei, and F.-Y. Hsiao, ‘‘Modeling healthcare data using multiple-channel latent Dirichlet allocation,’’ *J. Biomed. Informat.*, vol. 60,pp. 210–223, Apr. 2016, doi: 10.1016/j.jbi.2016.02.003.
3. S. Bauer, A. Noulas, D. Ó. Séaghdha, S. Clark, and C. Mascolo, ‘‘Talking places: Modelling and analysing linguistic content in foursquare,’’ in

*Proc. ASE/IEEE Int. Conf. Privacy, Secur., Risk Trust, ASE/IEEE Int. Conf. Social Comput. (SocialCom/PASSAT)*, Sep. 2012, pp. 348–357, doi: 10.1109/SocialCom-PASSAT.2012.107.

**10 GANTT CHART**

