**INDEX**

**TOPICS Page No’s**

**➢ Certificates**

**➢ Acknowledgement**

**➢ Abstract**

**➢ Figures/Tables**

**CHAPTER-1: INTRODUCTION**

**CHAPTER-2: LITERATURE SURVEY**

**CHAPTER-3: SYSTEM ANALYSIS**

3.1 Existing System

3.2 Proposed System

**CHAPTER-4: SYSTEM REQUIREMENTS**

4.1 Functional Requirement

4.2 Non-Functional Requirements

**CHAPTER-5: SYSTEM STUDY**

**5.1 Feasibility Study**

**5.2 Feasibility Analysis**

**CHAPTER-6: SYSTEM DESIGN**

6.1 SYSTEM ARCHITECTURE

6.2 UML DIAGRAMS

6.2.1 Use Case Diagram

6.2.2 Class Diagram

6.2.3 Sequence Diagram

6.2.4 Collaboration Diagram

6.2.5 Activity Diagram

6.2.6 Component Diagram

6.2.7 Deployment Diagram

6.2.8 Er Diagram

6.2.9 Data Dictionary

**CHAPTER-7: INPUT AND OUTPUT DESIGN**

7.1 Input Design

7.2 Output Design

**CHAPTER-8: IMPLEMENTATION**

8.1 Modules

8.1.1 Module Description

**CHAPTER-9: SOFTWARE ENVIRONMENT**

9.1 JAVA

9.2 Source Code

**CHAPTER-10: RESULTS/DISCUSSIONS**

10.1 System Testing

10.1.1 Test Cases

10.2 Output Screens

**CHAPTER-11: CONCLUSION**

11.1 Conclusion

11.2 Future Scope

**CHAPTER-12: REFERENCES**

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **TABLES/FIGURES** | **PAGE NO’S** |
| 1 | System Architecture |  |
| 2 | UML Diagrams |  |
|  | 2.1 Use Case Diagram |  |
|  | 2.2 Class Diagram |  |
|  | 2.3 Sequence Diagram |  |
|  | 2.4 Collaboration Diagram |  |
|  | 2.5 Activity Diagram |  |
|  | 2.6 Component Diagram |  |
|  | 2.7 Deployment Diagram |  |
|  | 2.8 ER Diagram |  |
|  | 2.9 Data Dictionary/Data Set |  |
| 3 | JAVA Source code with Compilation |  |
| 4 | Output Screens |  |

**FORCASTING AND ATICIPATORY ESTIMATION OF HUMAN MOVEMENT PATTERNS**

**ABSTRACT**

The study of human movement patterns is integral to numerous fields, from urban planning and transportation management to public safety and healthcare. In an increasingly interconnected world, the ability to forecast and anticipate human movement holds immense potential for optimizing resource allocation, enhancing infrastructure planning, and enabling timely interventions. This project delves into the development of an innovative framework for forecasting and anticipatory estimation of human movement patterns.

The proposed framework leverages advanced data analytics techniques and machine learning algorithms to analyze and interpret diverse sources of data, including historical movement data, demographic information, social events, and environmental factors. By synthesizing this multifaceted information, the framework aims to uncover hidden patterns and trends that influence human mobility.

A critical aspect of this project involves the creation of predictive models that can extrapolate future movement patterns based on past observations and contextual cues. These models empower decision-makers with the ability to foresee population flows, congestion hotspots, and travel trajectories. Furthermore, the project emphasizes the significance of anticipatory estimation, where the framework proactively identifies potential anomalies or disruptions in movement patterns, allowing for timely response strategies.

The potential applications of this research span a wide spectrum, including optimizing public transportation routes, mitigating traffic congestion, enhancing emergency response planning, and enabling personalized healthcare interventions. The results of this project contribute to the growing body of knowledge in predictive analytics, underscoring the power of data-driven insights in shaping a more efficient and responsive urban landscape.

In conclusion, this project advances the field of human movement pattern analysis by introducing a comprehensive framework for forecasting and anticipatory estimation. By harnessing the potential of data analytics and machine learning, this research opens avenues for informed decision-making and proactive intervention, ultimately fostering smarter, more resilient communities.

**CHAPTER-1**

**INTRODUCTION**

With the rapid development of the wireless and networking technology, mobile networks have imposed a profound impact on people’s daily life for their marvellous capability. These applications utilize the users’ current and historical location information records (LIR) to analyze their mobility patterns to enable numerous applications, such as targeted advertising, city planning and smart navigation.

Generally speaking, the LIR data collected from the mobile networks can be divided into two categories, i.e., data collected by Internet service provider (referred to as ISP-collected data) and data collected by applications (referred to as app-collected data). The ISP-collected data are passively and periodically collected regardless of behaviours of the users. This sort of data preserves the complete and consecutive trajectory of each user. Most of the existing studies are based on users’ ISP-collected location data. quantified the predictability in human mobility by studying the regularities shown in the trajectory. According to their studies, the potential predictability reaches 93% on a mobile phone record dataset. Wang et al. link the human mobility with the social network, by segregating the similar users using the information from social media, more general and universal mobility patterns on a certain group of people were extracted, suggesting the huge predictability of the individual’s movements. Moreover, applying the predictability into practice, many researches have also been conducted on the prediction of human mobility on various models, such as Markov Chain models, neural network, Bayesian network, finite state machine. On the other hand, however, few researches have focused on either predictability or prediction algorithm on app-collected location data. These aspects of researches remain to be explored.

In comparison to ISP-collected data, app-collected LIR data is actively triggered by users themselves in applications. This kind of location data will be collected when using the applications while the location information of the rest time remains unknown. It is exactly the characteristics of the app-collected data that arouses several difficulties to our study. First, the app-collected data contains the physical context of the location because the purpose of using the application certainly correlates with the location recorded, e.g., ordering a taxi, searching a restaurant. Such correlations provide valuable information to analyze the human mobility patterns. However, simple grid for the city apparently lose the information. Hence, it is essential to find a proper spatial division of the city to reserve the physical context of app-collected data. Second, the app-collected data are partially missing since usually the applications do not record users’ locations when they are not using the apps. Third, the app-collected data are heterogeneous in spatial and temporal domain since the time when people use the application is unevenly distributed. Under these circumstances, the methods aroused in the previous study apparently are not suitable for accurate predictions on the dataset. We need to propose new methods to adapt to these features of the app-collected data.

In this paper, we address the above three challenges to facilitate the analysis. Our work can be summarized as follows:

• In order to reserve the physical context of the locations, we contextually cluster the locations into multiple nonoverlapping districts of the city instead of using fixed coordinate grid that will lose the physical context. We also compare the predictability and the prediction accuracy between the two divisions to analyze the effect of context on prediction. Results reveal that the trajectories on context-based division are more predictable than those on division without context under the same spatial granularity.

• We design a Markov-based method using Gibbs sampling to solve the unevenly distribution and the high missing rate of the app-collected data. By restoring the trajectory, we estimate the transition matrix to make prediction of users’ movement. Results show that, based on app collected dataset, our method achieves the same accuracy of the previous studies on the ISP-collected dataset.

• In order to investigate the effect of heterogeneity, we carry out a thorough analysis of the predictability and prediction accuracy based on our designed method on the app-collected dataset. The varying factors include the spatial and temporal resolution, the orders of Markov models, the radius of gyration etc.

**CHAPTER-2**

**LITERATURE SURVEY**

**TITLE:** Limits of predictability in human mobility

**AUTHORS:** [Chaoming Song](https://www.researchgate.net/profile/Chaoming-Song?_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6InB1YmxpY2F0aW9uIiwicGFnZSI6InB1YmxpY2F0aW9uIn19), [Zehui Qu](https://www.researchgate.net/profile/Zehui-Qu-2?_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6InB1YmxpY2F0aW9uIiwicGFnZSI6InB1YmxpY2F0aW9uIn19), [Nicholas Blumm](https://www.researchgate.net/scientific-contributions/Nicholas-Blumm-15620873?_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6InB1YmxpY2F0aW9uIiwicGFnZSI6InB1YmxpY2F0aW9uIn19), [Albert-Laszlo Barabasi](https://www.researchgate.net/profile/Albert-Laszlo-Barabasi-2?_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6InB1YmxpY2F0aW9uIiwicGFnZSI6InB1YmxpY2F0aW9uIn19).

**ABSTRACT:**

A range of applications, from predicting the spread of human and electronic viruses to city planning and resource management in mobile communications, depend on our ability to foresee the whereabouts and mobility of individuals, raising a fundamental question: To what degree is human behaviour predictable? Here we explore the limits of predictability in human dynamics by studying the mobility patterns of anonymized mobile phone users. By measuring the entropy of each individual’s trajectory, we find a 93% potential predictability in user mobility across the whole user base. Despite the significant differences in the travel patterns, we find a remarkable lack of variability in predictability, which is largely independent of the distance users cover on a regular basis.

**TITLE:** Human mobility, social ties, and link prediction

**AUTHORS:** [Dashun Wang, Dino Pedreschi](about:blank), [Chaoming Song](about:blank). Fosca Giannotti, Albert-Laszlo Barabasi.

**ABSTRACT:**

Our understanding of how individual mobility patterns shape and impact the social network is limited, but is essential for a deeper understanding of network dynamics and evolution. This question is largely unexplored, partly due to the difficulty in obtaining large-scale society-wide data that simultaneously capture the dynamical information on individual movements and social interactions. Here we address this challenge for the first time by tracking the trajectories and communication records of 6 million mobile phone users. We find that the similarity between two individuals' movements strongly correlates with their proximity in the social network. We further investigate how the predictive power hidden in such correlations can be exploited to address a challenging problem: which new links will develop in a social network. We show that mobility measures alone yield surprising predictive power, comparable to traditional network-based measures. Furthermore, the prediction accuracy can be significantly improved by learning a supervised classifier based on combined mobility and network measures. We believe our findings on the interplay of mobility patterns and social ties offer new perspectives on not only link prediction but also network dynamics.

**TITLE:** Approaching the limit of predictability in human mobility.

**AUTHORS:** [Xin Lu](https://www.researchgate.net/profile/Xin-Lu-31?_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6InB1YmxpY2F0aW9uIiwicGFnZSI6InB1YmxpY2F0aW9uIn19), [Erik Wetter](https://www.researchgate.net/profile/Erik-Wetter?_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6InB1YmxpY2F0aW9uIiwicGFnZSI6InB1YmxpY2F0aW9uIn19), [Nita Bharti](https://www.researchgate.net/profile/Nita-Bharti?_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6InB1YmxpY2F0aW9uIiwicGFnZSI6InB1YmxpY2F0aW9uIn19), [Andrew Tatem](https://www.researchgate.net/profile/Andrew-Tatem?_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6InB1YmxpY2F0aW9uIiwicGFnZSI6InB1YmxpY2F0aW9uIn19).

**ABSTRACT:**

In this study we analyze the travel patterns of 500,000 individuals in Cote d'Ivoire using mobile phone call data records. By measuring the uncertainties of movements using entropy, considering both the frequencies and temporal correlations of individual trajectories, we find that the theoretical maximum predictability is as high as 88%. To verify whether such a theoretical limit can be approached, we implement a series of Markov chain (MC) based models to predict the actual locations visited by each user. Results show that MC models can produce a prediction accuracy of 87% for stationary trajectories and 95% for non-stationary trajectories. Our findings indicate that human mobility is highly dependent on historical behaviours, and that the maximum predictability is not only a fundamental theoretical limit for potential predictive power, but also an approachable target for actual prediction accuracy.

**TITLE:** A class of mobile motion prediction algorithms for wireless mobile computing and communications

**AUTHORS:** [George Liu](https://link.springer.com/article/10.1007/BF01193332" \l "auth-George-Liu-Aff1), [Gerald Maguire Jr.](https://link.springer.com/article/10.1007/BF01193332" \l "auth-Gerald-Maguire-Aff2)

**ABSTRACT:**

This paper describes a class of novel mobile motion prediction algorithms for supporting global mobile data accessing. Traditionally, mobility and routing management includes functions to passively keep track of the location of the users/terminals and to maintain connections to the terminals belonging to the system. To maintain uninterrupted high-quality service for distributed applications, it is important that a mobile system be more intelligent and can anticipate the change of the location of its user. We propose an aggressive mobility and routing management scheme, called predictive mobility management. A class of mobile motion prediction algorithms predicts the “future” location of a mobile user according to the user's movement history, i.e., previous movement patterns. By combining this scheme with mobility agent functions, the service and user routing data are actually pre-connected and pre-assigned at the locations to which the user is moving. Thus, the user can immediately receive service or data with virtually the same efficiency as at the previous location, i.e., without encountering a large “data structure handover” delay before service or data is available.

**TITLE:** Applied neural network for location prediction and resources reservation scheme in wireless networks

**AUTHORS:** [Shiang-Chun Liou](https://ieeexplore.ieee.org/author/37087926261), [Hsuan-Chia Lu](https://ieeexplore.ieee.org/author/37087927466).

**ABSTRACT:**

In this paper, NPS (neural network prediction scheme) is proposed to provide high accuracy location prediction of mobile host (MH) in target cell. Multimedia communications is urgently expected in wireless networks. One of the most important and complicated issues is quality of service guarantees in third-generation (3G) wireless networks. In other words, the problem to maintain the continuity of multimedia playing during the handoff is hard to solve. In order to avoid too early or over reservation resulting in a waste of resources, in this study, we present a method called TTRR (three times resource reservation scheme), to let reserved resource become really active upon MH entering into tire-2 (C.H. Choi, et al., Nov. 2000) area of cell. NSP and TTRR can efficiently improve the accuracy of MHs trajectory prediction, increase the success probability of resource reservation, and enhance bandwidth utilization.

**TITLE:** Movement prediction using Bayesian learning for neural networks

**AUTHORS:** [Sherif Akoush](https://ieeexplore.ieee.org/author/37714209000), [Ahmed Sameh](https://ieeexplore.ieee.org/author/37266494500).

**ABSTRACT:**

A technique for reducing the wireless cost of tracking mobile users with uncertain parameters is developed in this paper. Such uncertainty arises naturally in wireless networks, since an efficient user tracking is based on a prediction of its future call and mobility parameters. The conventional approach based on dynamic tracking is not reliable in the sense that inaccurate prediction of the user mobility parameters may significantly reduce the tracking efficiency. Unfortunately, such uncertainty is unavoidable for mobile users, especially for a burst mobility patterns. In this paper, we present a novel hybrid Bayesian neural network model for predicting locations on Cellular Networks (can also be extended to other wireless networks such as WI-FI and WiMAX). We investigate different parallel implementation techniques on mobile devices of the proposed approach and compare it to many standard neural network techniques such as: Back-propagation, Elman, Resilient, Levenberg-Marqudat, and One-Step Secant models. Bayesian learning for Neural Networks predicts location better than standard neural network techniques since it uses well founded probability model to represent uncertainty about the relationships being learned. The result of Bayesian training is a posterior distribution over network weights. We use Markov chain Monte Carlo methods (MCMC) to sample N values from the posterior weight’s distribution.

**CHAPTER-3**

**SYSTEM ANALYSIS**

**3.1 EXISTING SYSTEM**

Most of the existing prediction methods are based on the Markov model, which can only model one transition pattern, since it has only one transition kernel. For example, if a user visits his/her office in the morning, then he/she is likely to stay at the office in the following few hours. In contrast, if he/she visits his office in the afternoon, he/she probably will leave in the following time. If this example is modelled by a Markov chain, the prediction result will be the same, since the transition of the Markov chain only relies on its previous state, regardless of the time when the transition happens.

**Disadvantages**:

Plenty of previous work adopted the traditional Markov model, which suffers when the trajectory becomes sparse or it shows distinct mobility patterns in different time of day.

**3.2 PROPOSED SYSTEM**

We need to propose new methods to overcome the problem aroused by these features of the app-collected data. In this paper, we address the above three challenges to facilitate the analysis. Our work can be summarized as follows:

• In order to reserve the physical context of the locations, we contextually cluster the locations into multiple non-overlapping districts of the city instead of using fixed coordinate grid that will lose the physical context. We also compare the prediction accuracy between the two divisions to analyze the effect of context on prediction. Results reveal that the trajectories on context-based division are more predictable than those on division without context under the same spatial granularity.

• We design a time-variant Markov-based method using Gibbs sampling to model the distinct mobility patterns and solve data sparsity. By assuming multi-Markov chains, we sample the transition kernels of these chains and make predictions of users’ movements based on the kernels. Results show that, based on app-collected dataset, our method achieves higher prediction accuracy than the benchmark prediction method.

• In order to investigate the effect of heterogeneity, we carry out a thorough analysis of the predictability and prediction accuracy based on our designed method on the app-collected dataset. The varying factors include the spatial and temporal resolution, the orders of Markov models, the radius of gyration, etc.

**Advantages**:

1. We comprehensively analyze the mobility and predictability of each user.

2. We adopt Gibbs sampling method to simultaneously recover the missing part of trajectories and train the Markov chains, in order to solve the unevenly distribution and the high missing rate.

### **CHAPTER-4**

### **SYSTEM REQUIREMENTS**

**4.1 FUNCTIONAL REQUIREMENTS**

A functional requirement is a statement of how a system must behave. It defines what the system should do in order to meet the user’s needs or expectations. Functional requirements can be thought of as features that the user detects. They are different from non-functional requirements, which define how the system should work internally (e.g., performance, security, etc.).

Functional requirements are made up of two parts: function and behavior. The function is what the system does (e.g., “calculate sales tax”). The behavior is how the system does it (e.g., “The system shall calculate the sales tax by multiplying the purchase price by the tax rate.”).

Functional Requirements, as the name suggests, describe the functions of the system to be designed. It is a description of what the system will be and how it will function to satisfy user needs. They provide a clear description of how the system is supposed to respond to a particular command, the features, and what the users expect.

These are the modules of

* Admin

##### **4.2 NON – FUNCTIONAL REQUIREMENTS**

**HARDWARE REQUIREMENTS**

|  |  |  |
| --- | --- | --- |
| **MINIMUM (Required for Execution)** | | **MY SYSTEM (Development)** |
| **System** | Pentium IV 2.2 GHz | i3 Processor 5th Gen |
| **Hard Disk** | 20 Gb | 512 Gb |
| **Ram** | 1 Gb | 4 Gb |

**SOFTWARE REQUIREMENTS**

|  |  |
| --- | --- |
| **Operating System** | **Windows 10/11** |
| **Development Software** | JAVA 8.1 |
| **Programming Language** | Java |
| **Integrated Development Environment (IDE)** | NetBeans 8.1 |
| **Front End Technologies** | HTML5, CSS3, Java Script |
| **Database Language** | SQL |
| **Database (RDBMS)** | MySQL |
| **Database Software** | MySQL Server |
| **Web Server or Deployment Server** | Apache tomcat |
| **Design/Modelling** | Rational Rose |

**CHAPTER-5**

**SYSTEM STUDY**

**5.1 FEASIBILITY 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.

Three key considerations involved in the feasibility analysis are

* ECONOMICAL FEASIBILITY
* TECHNICAL FEASIBILITY
* SOCIAL FEASIBILITY

**ECONOMICAL 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.

### **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.

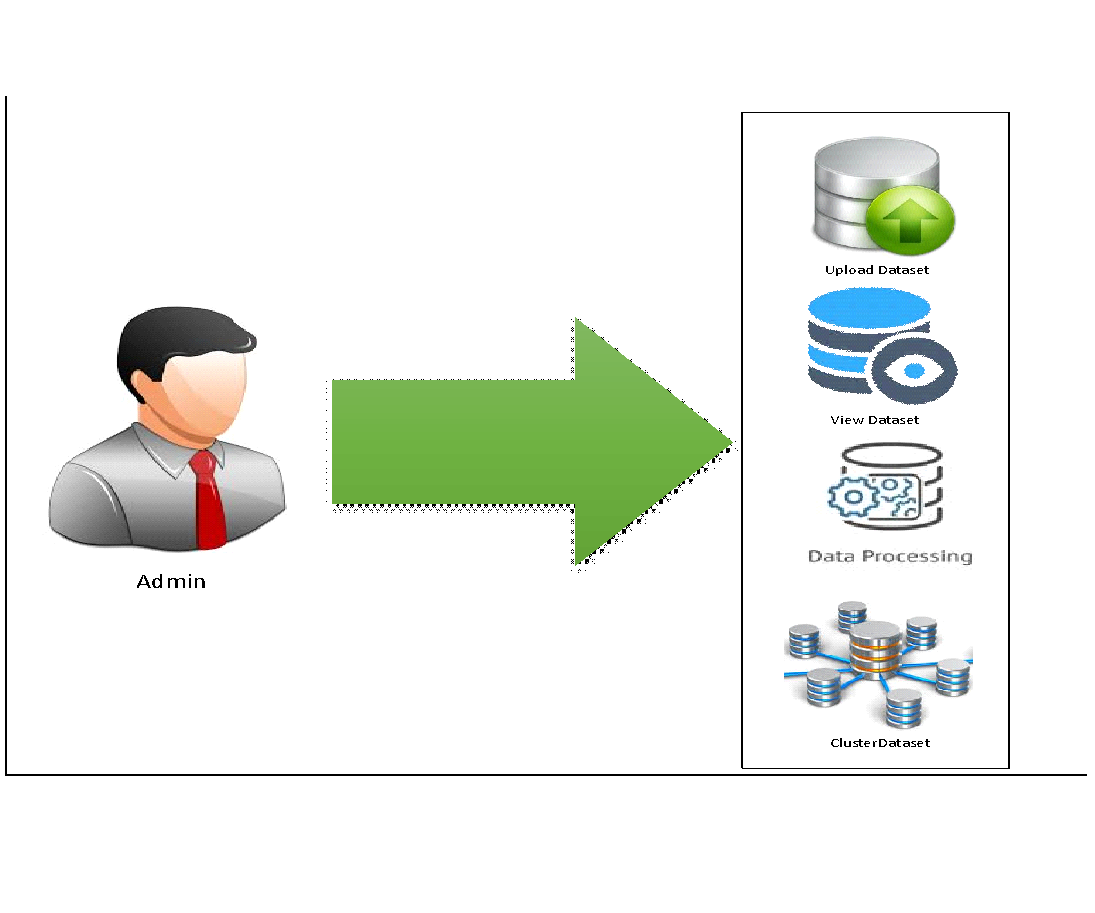
**SOCIAL FEASIBILITY**

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.

## **CHAPTER-6**

## **SYSTEM DESIGN**

**SYSTEM ARCHITECTURE**



**6.1 UML DIAGRAMS**

UML stands for Unified Modelling Language. UML is a standardized general-purpose modelling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object-oriented computer software. In its current form UML comprises two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML.

The Unified Modelling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software systems, as well as for business modelling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems.

The UML is a very important part of developing objects-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

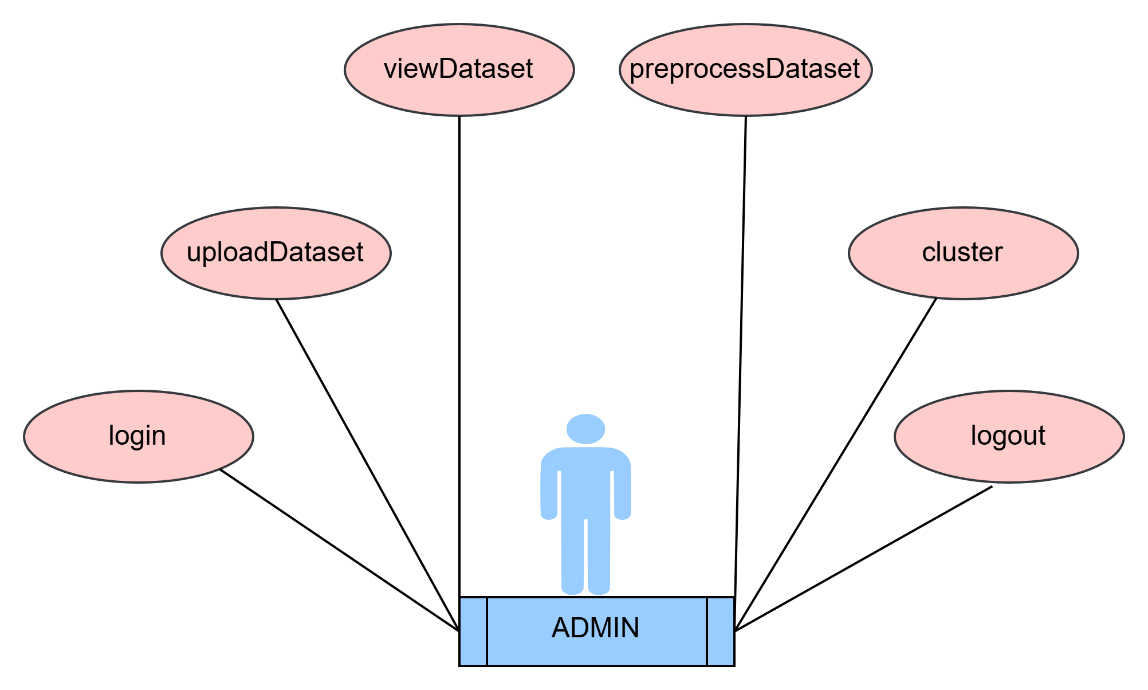
**GOALS:**

The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modelling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modelling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

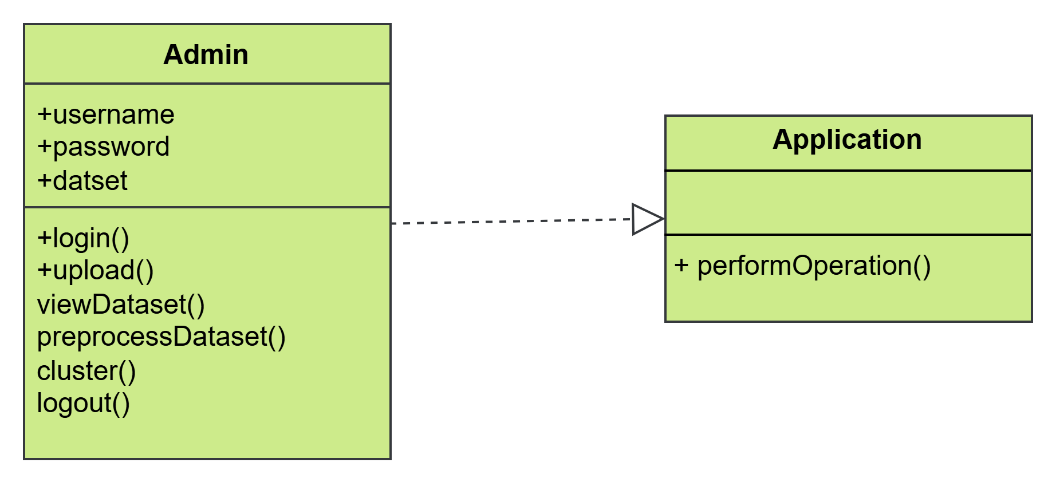
**6.1.1 USE CASE DIAGRAM:**

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.



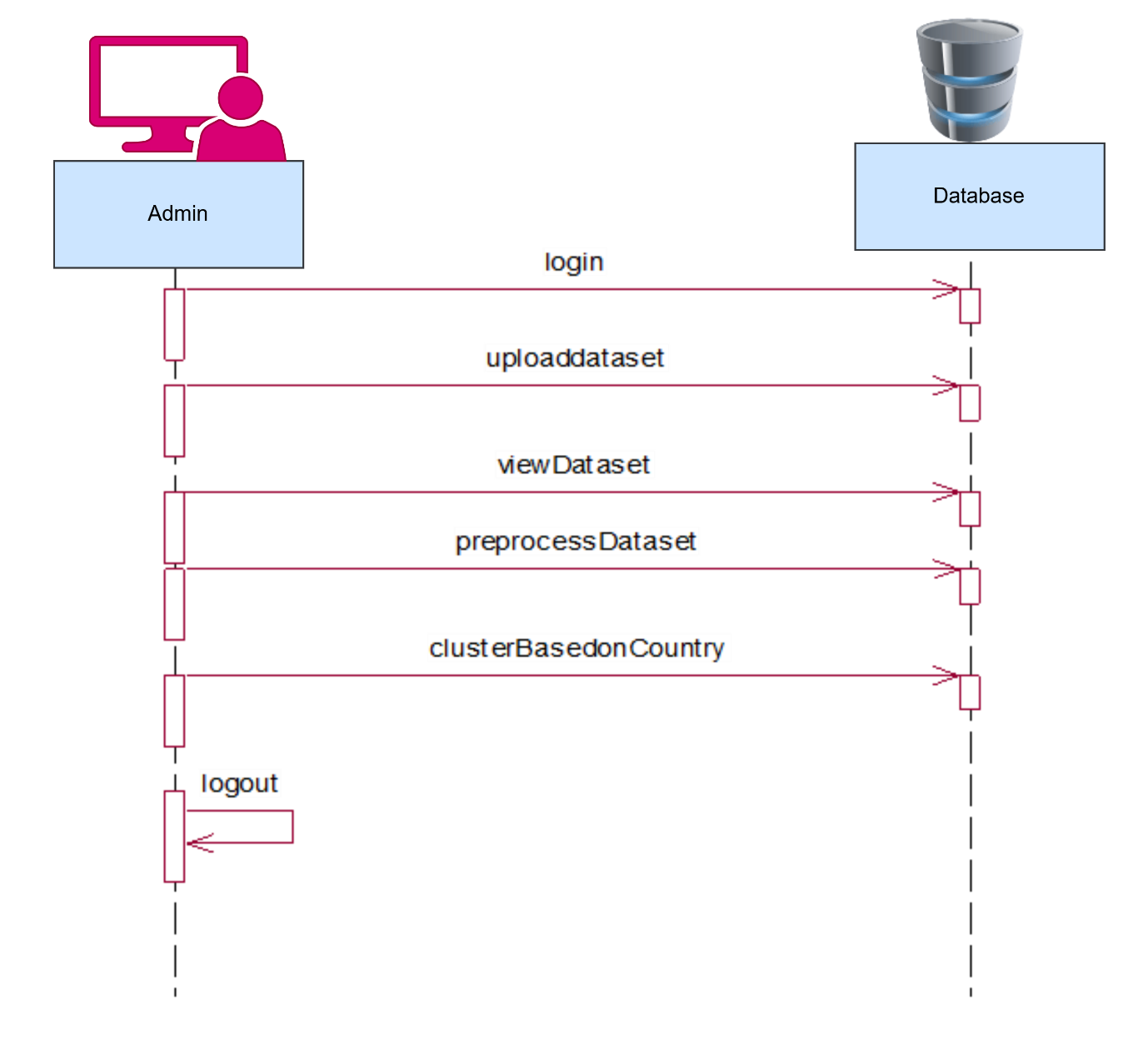
**6.1.2 CLASS DIAGRAM:**

In software engineering, a class diagram in the Unified Modelling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.



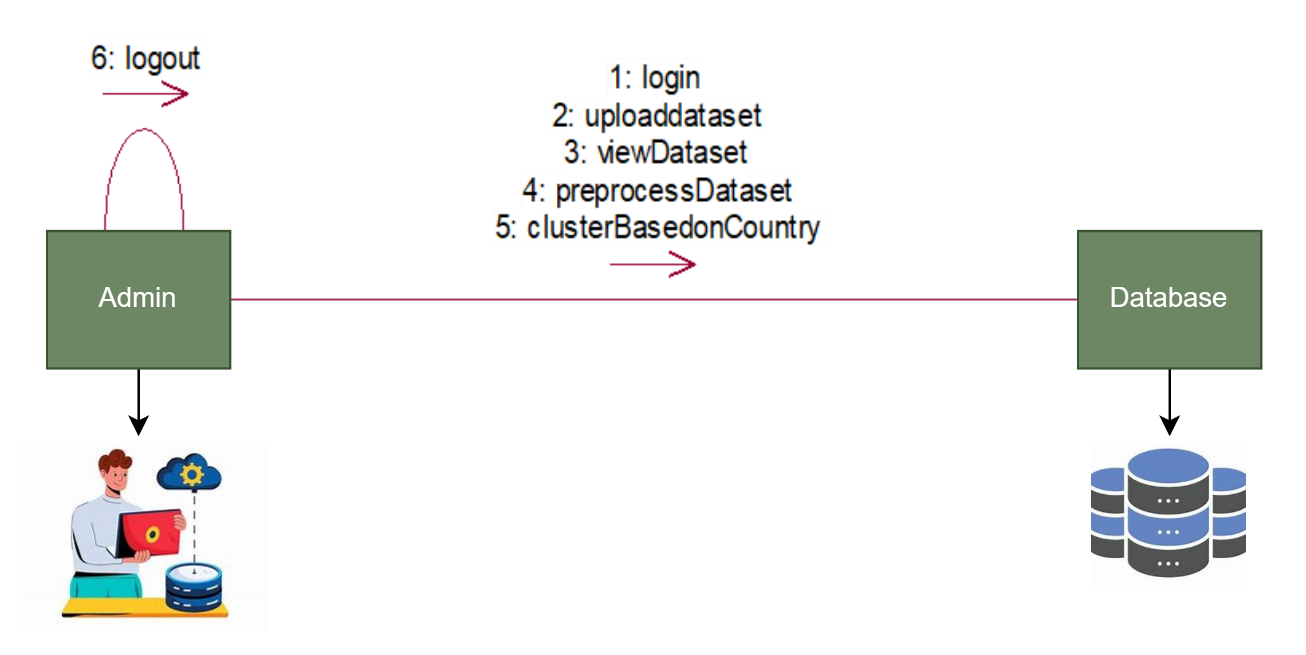
**6.1.3 SEQUENCE DIAGRAM:**

A sequence diagram in Unified Modelling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.



**6.1.4 COLLABORATION:**

## A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software objects in the Unified Modeling Language (UML). Developers can use these diagrams to portray the dynamic behavior of a particular use case and define the role of each object.



**6.1.5 ACTIVITY DIAGRAM :**

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

Cluster Dataset

Preprocess Data set

View Dataset

Upload Dataset

Admin

Login Again

LOGIN CHECK

START

NO

NO

YES

NO

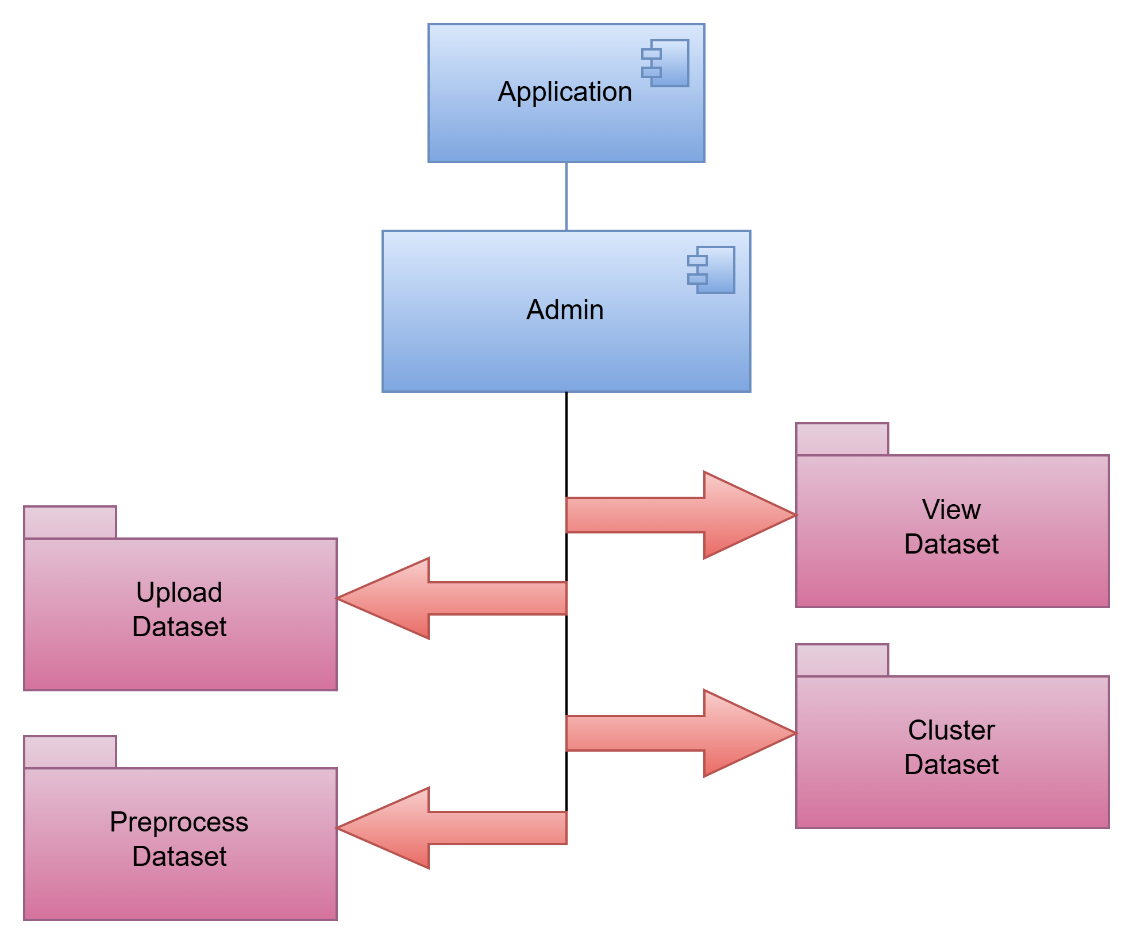
### 

END

NO

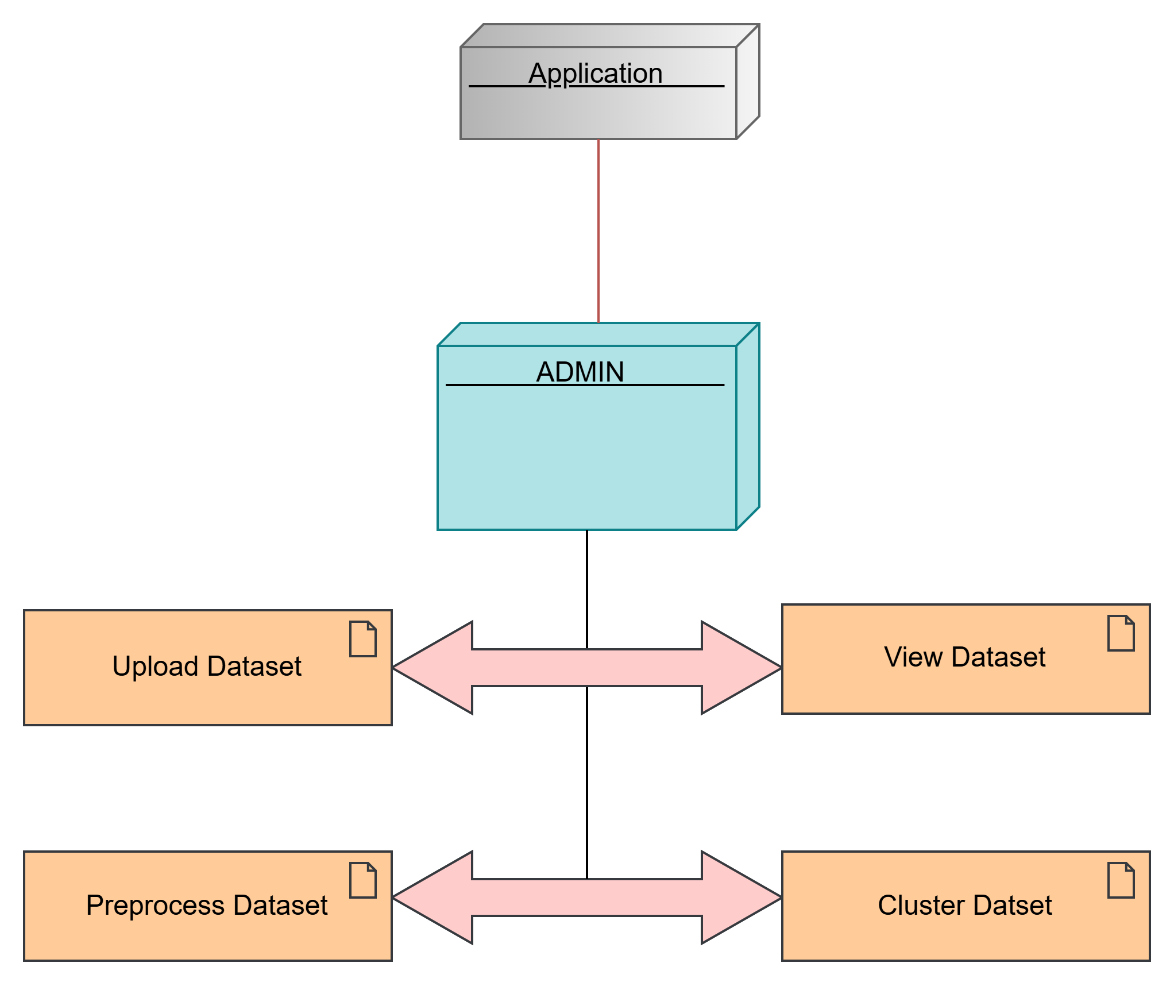
**6.2.6 COMPONENT DIAGRAM :**

A Component diagram is a type of Unified Modeling Language (UML) diagram used to visualize the organization and relationships between the components of a system or software application. Components represent modular and reusable parts of a system, such as classes, objects, libraries, modules, and subsystems. Component diagrams help in understanding the high-level structure of a system and how components interact with each other. They are particularly useful for managing complexity, promoting reusability, and ensuring modularity in software design.



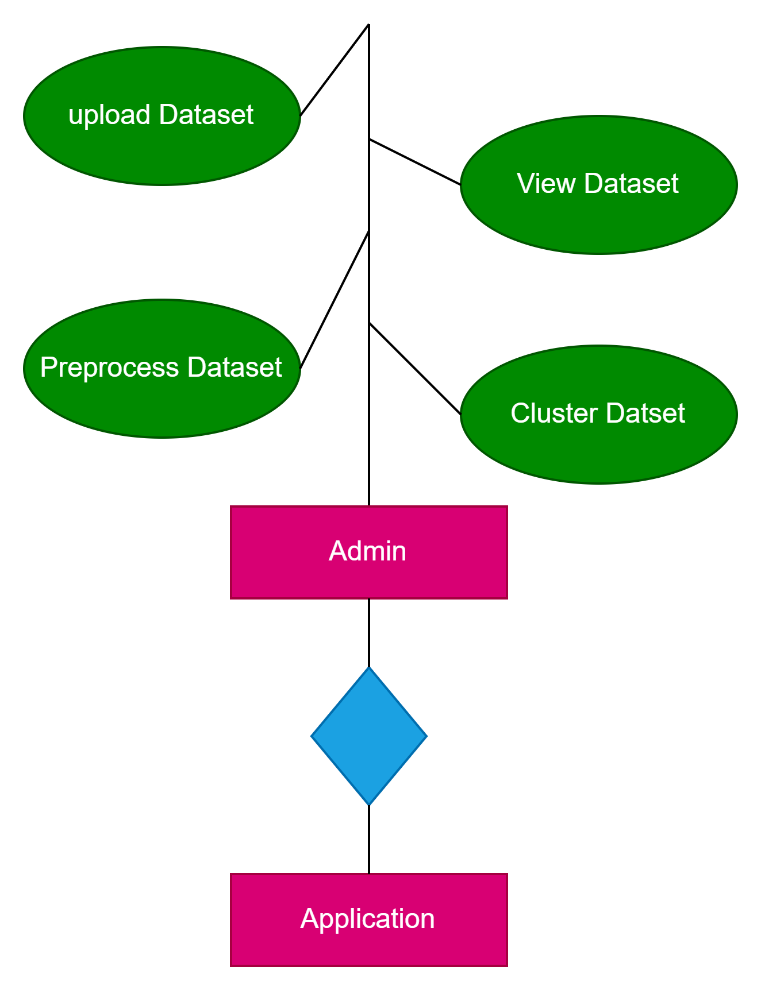
**6.2.7 DEPLOYMENT DIAGRAM:**

A Deployment Diagram in UML (Unified Modeling Language) is used to visualize the physical deployment of software components and hardware nodes within a system or application. It provides a high-level view of how software and hardware interact in a distributed computing environment.



**6.2.8 ENTITY-RELATIONSHIP DIAGRAM:**

An Entity-Relationship Diagram (ERD) is a visual representation used in database design and modeling to illustrate the relationships between entities (also known as tables) in a database. ERDs are a crucial tool for understanding the structure of a database system and how data is organized within it. Here's a description of the key components and concepts associated with an ERD.



**6.2.9 DATA DICTIONARY**

**ADMIN: -**

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUMN** | **DATATYPE** | **CONSTRAINTS** | **DESCRIPTION** |
| Username | Varchar (100) | Primary key (Unique) | To enter Username |
| Password | Number | Not Null | To enter Password |

### **CHAPTER-7**

### **INPUT & OUTPUT DESIGN**

**7.1 INPUT DESIGN**

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining privacy. Input Design considered the following things:

* What data should be given as input?
* How should the data be arranged or coded?
* The dialog to guide the operating personnel in providing input.
* Methods for preparing input validations and steps to follow when error occur.

**OBJECTIVES**

1. Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.

2. It is achieved by creating user-friendly screens for the data entry to handle large volumes of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data can be performed. It also provides record viewing facilities.

3. When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize instant. Thus, the objective of input design is to create an input layout that is easy to follow.

**7.2 OUTPUT DESIGN**

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other systems through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source of information to the user. Efficient and intelligent output design improves the system’s relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

2. Select methods for presenting information.

3. Create documents, reports, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

* Convey information about past activities, current status or projections of the
* Future.
* Signal important events, opportunities, problems, or warnings.
* Trigger an action.
* Confirm an action.

**CHAPTER-8**

**IMPLEMENTATION**

**8.1 MODULES**

1. ADMIN

**8.2 MODULE DESCRIPTION**

**ADMIN:**

In this module here admin can directly login with the application after admin successful login he can upload dataset and view dataset and also clean data set by preprocessing and we are clustering based on the country available in dataset.

**CHAPTER-9**

**SOFTWARE ENVIRONMENT**

#### **9.1 JAVA 8.1**

## Java Technology

Java technology is both a programming language and a platform.

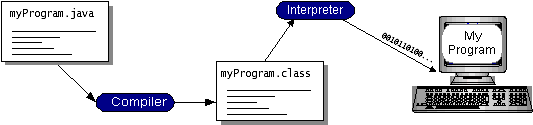
### The Java Programming Language

### The Java programming language is a high-level language that can be characterized by following buzzwords:

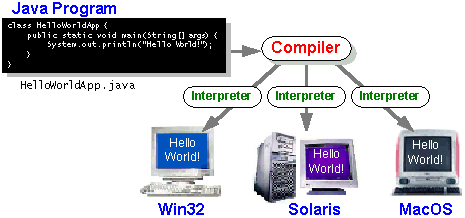
* + - Simple
    - Architecture neutral
    - Object oriented
    - Portable
    - Distributed
    - High performance
    - Interpreted
    - Multithreaded
    - Robust
    - Dynamic
    - Secure

With most programming languages, a program is either compiled or interpreted to run on a computer.

"The Java programming language is unique because it combines both compilation and interpretation. First, the compiler translates the program into an intermediate language called Java bytecode, which is platform-independent. Then, the interpreter processes and executes each bytecode instruction on the computer. Compilation occurs once, while interpretation happens every time the program runs."



Java bytecode acts as machine code instructions for the Java Virtual Machine (JVM). Every Java interpreter, whether a development tool or a web browser capable of running applets, is an implementation of the JVM. Java bytecode enables the 'write once, run anywhere' principle. You can compile your program into bytecode on any platform with a Java compiler, and it can then be executed on any JVM implementation. This means that as long as a computer has a JVM, the same Java program can run on Windows 2000, a Solaris workstation, or an iMac.



### The Java Platform

A platform refers to the hardware or software environment where a program runs. Some of the most popular platforms include Windows 2000, Linux, Solaris, and macOS. Typically, platforms consist of both an operating system and hardware. However, the Java platform is unique because it is a software-only platform that operates on top of other hardware-based platforms.

The Java platform has two components:

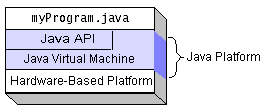
● The Java Virtual Machine (Java VM)

● The Java Application Programming Interface (Java API)

The JVM serves as the foundation of the Java platform and is ported onto various hardware-based platforms. The Java API is a comprehensive collection of pre-built software components that provide essential functionalities, including graphical user interface (GUI) widgets.

The Java API is grouped into libraries of related classes and interfaces; these libraries are known as packages. The next section, *What Can Java Technology Do?*, highlights what functionality some of the packages in the Java API provide.

The following figure illustrates a program running on the Java platform. As shown in the figure, the Java API and the virtual machine act as a layer of abstraction, shielding the program from the underlying hardware.



## Native code refers to compiled code that runs directly on a specific hardware platform. Because the Java platform is platform-independent, it may run slightly slower than native code. However, smart compilers, well-tuned interpreters, and just-in-time bytecode compilers can optimize performance to near-native levels without sacrificing portability.

## 

What can Java technology do?

The most common types of programs written in Java are applets and applications. If you’ve surfed the web, you’re probably familiar with applets—small programs that follow specific conventions to run within a Java-enabled browser.

However, Java is not just for creating simple web applets. As a general-purpose, high-level programming language, Java is also a powerful software platform. With Java's extensive API, you can develop various types of programs.

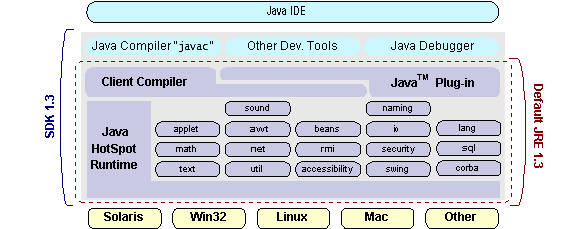
An application is a standalone program that runs directly on the Java platform. A special type of application, known as a server, serves and supports clients on a network. Examples include web servers, proxy servers, mail servers, and print servers. Another specialized program is a servlet, which functions similarly to an applet but runs on the server side. Java servlets are a popular choice for building interactive web applications, replacing CGI scripts.

.

How does the API support all these kinds of programs? It does so with packages of software components that provide a wide range of functionality. Every full implementation of the Java platform gives you the following features:

* **The essentials**: Objects, strings, threads, numbers, input and output, data structures, system properties, date and time, and so on.
* **Applets**: The set of conventions used by applets.
* **Networking**: URLs, TCP (Transmission Control Protocol), UDP (User Data gram Protocol) sockets, and IP (Internet Protocol) addresses.
* **Internationalization**: Help for writing programs that can be localized for users worldwide. Programs can automatically adapt to specific locales and be displayed in the appropriate language.
* **Security**: Both low level and high level, including electronic signatures, public and private key management, access control, and certificates.
* **Software components**: Known as JavaBeansTM, can plug into existing component architectures.
* **Object serialization**: Allows lightweight persistence and communication via Remote Method Invocation (RMI).
* **Java Database Connectivity (JDBCTM)**: Provides uniform access to a wide range of relational databases.

The Java platform also has APIs for 2D and 3D graphics, accessibility, servers, collaboration, telephony, speech, animation, and more. The following figure depicts what is included in the Java 2 SDK.



## How Will Java Technology Change My Life?

We can’t promise you fame, fortune, or even a job if you learn the Java programming language. Still, it is likely to make your programs better and requires less effort than other languages. We believe that Java technology will help you do the following:

* **Get started quickly**: Although the Java programming language is a powerful object-oriented language, it’s easy to learn, especially for programmers already familiar with C or C++.
* **Write less code**: Comparisons of program metrics (class counts, method counts, and so on) suggest that a program written in the Java programming language can be four times smaller than the same program in C++.
* **Write better code**: The Java programming language encourages good coding practices, and its garbage collection helps you avoid memory leaks. Its object orientation, its JavaBeans component architecture, and its wide-ranging, easily extensible API let you reuse other people’s tested code and introduce fewer bugs.
* **Develop programs more quickly**: Your development time may be as much as twice as fast versus writing the same program in C++. Why? You write fewer lines of code and it is a simpler programming language than C++.
* **Avoid platform dependencies with 100% Pure Java**: You can keep your program portable by avoiding the use of libraries written in other languages. The 100% Pure JavaTM Product Certification Program has a repository of historical process manuals, white papers, brochures, and similar materials online.
* **Write once, run anywhere**: Because 100% Pure Java programs are compiled into machine-independent byte codes, they run consistently on any Java platform.
* **Distribute software more easily**: You can upgrade applets easily from a central server. Applets take advantage of the feature of allowing new classes to be loaded “on the fly,” without recompiling the entire program.

### ODBC

Microsoft Open Database Connectivity (ODBC) is a standard programming interface for application developers and database systems providers. Before ODBC became a *de facto* standard for Windows programs to interface with database systems, programmers had to use proprietary languages for each database they wanted to connect to. Now, ODBC has made the choice of the database system almost irrelevant from a coding perspective, which is as it should be. Application developers have much more important things to worry about than the syntax that is needed to port their program from one database to another when business needs suddenly change.

Through the ODBC Administrator in the Control Panel, you can specify the particular database that is associated with a data source that an ODBC application program is written to use. Think of an ODBC data source as a door with a name on it. Each door will lead you to a particular database. For example, the data source named Sales Figures might be a SQL Server database, whereas the Accounts Payable data source could refer to an Access database. The physical database referred to by a data source can reside anywhere on the LAN.

The ODBC system files are not installed on your system by Windows 95. Rather, they are installed when you set up a separate database application, such as SQL Server Client or Visual Basic 4.0. When the ODBC icon is installed in the Control Panel, it uses a file called ODBCINST.DLL. It is also possible to administer your ODBC data sources through a stand-alone program called ODBCADM.EXE. There is a 16-bit and a 32-bit version of this program and each maintains a separate list of ODBC data sources.

From a programming perspective, the beauty of ODBC is that the application can be written to use the same set of function calls to interface with any data source, regardless of the database vendor. The source code of the application doesn’t change whether it talks to Oracle or SQL Server. We only mention these two as an example. There are ODBC drivers available for several dozen popular database systems. Even Excel spreadsheets and plain text files can be turned into data sources. The operating system uses the Registry information written by ODBC Administrator to determine which low-level ODBC drivers are needed to talk to the data source (such as the interface to Oracle or SQL Server). The loading of the ODBC drivers is transparent to the ODBC application program. In a client/server environment, the ODBC API even handles many of the network issues for the application programmer.

The advantages of this scheme are so numerous that you are probably thinking there must be some catch. The only disadvantage of ODBC is that it isn’t as efficient as talking directly to the native database interface. ODBC has had many detractors make the charge that it is too slow. Microsoft has always claimed that the critical factor in performance is the quality of the driver software that is used. In our humble opinion, this is true. The availability of good ODBC drivers has improved a great deal recently. And anyway, the criticism about performance is somewhat analogous to those who said that compilers would never match the speed of pure assembly language. Maybe not, but the compiler (or ODBC) gives you the opportunity to write cleaner programs, which means you finish sooner. Meanwhile, computers get faster every year.

**JDBC**

In an effort to set an independent database standard API for Java; Sun Microsystems developed Java Database Connectivity, or JDBC. JDBC offers a generic SQL database access mechanism that provides a consistent interface to a variety of RDBMSs. This consistent interface is achieved through the use of “plug-in” database connectivity modules, or *drivers*. If a database vendor wishes to have JDBC support, he or she must provide the driver for each platform that the database and Java run on.

To gain a wider acceptance of JDBC, Sun based JDBC’s framework on ODBC. As you discovered earlier in this chapter, ODBC has widespread support on a variety of platforms. Basing JDBC on ODBC will allow vendors to bring JDBC drivers to market much faster than developing a completely new connectivity solution.

JDBC was announced in March of 1996. It was released for a 90 day public review that ended June 8, 1996. Because of user input, the final JDBC v1.0 specification was released soon after.

The remainder of this section will cover enough information about JDBC for you to know what it is about and how to use it effectively. This is by no means a complete overview of JDBC. That would fill an entire book.

### JDBC Goals

Few software packages are designed without goals in mind. JDBC is one that, because of its many goals, drove the development of the API. These goals, in conjunction with early reviewer feedback, have finalized the JDBC class library into a solid framework for building database applications in Java.

The goals that were set for JDBC are important. They will give you some insight as to why certain classes and functionalities behave the way they do. The eight design goals for JDBC are as follows:

1. **SQLLevel *API***

The designers felt that their main goal was to define a SQL interface for Java. Although not the lowest database interface level possible, it is at a low enough level for higher-level tools and APIs to be created. Conversely, it is at a high enough level for application programmers to use it confidently. Attaining this goal allows for future tool vendors to “generate” JDBC code and to hide many of JDBC’s complexities from the end user.

1. **SQL Conformance**

SQL syntax varies as you move from database vendor to database vendor. In an effort to support a wide variety of vendors, JDBC will allow any query statement to be passed through it to the underlying database driver. This allows the connectivity module to handle non-standard functionality in a manner that is suitable for its users.

1. **JDBC must be implemental on top of common database interfaces**The JDBC SQL API must “sit” on top of other common SQL level APIs. This goal allows JDBC to use existing ODBC level drivers by the use of a software interface. This interface would translate JDBC calls to ODBC and vice versa.
2. **Provide a Java interface that is consistent with the rest of the Java system**

Because of Java’s acceptance in the user community thus far, the designers feel that they should not stray from the current design of the core Java system.

1. **Keep it simple**

This goal probably appears in all software design goal listings. JDBC is no exception. Sun felt that the design of JDBC should be very simple, allowing for only one method of completing a task per mechanism. Allowing duplicate functionality only serves to confuse the users of the API.

1. **Use strong, static typing wherever possible**

Strong typing allows for more error checking to be done at compile time; also, less error appears at runtime.

1. **Keep the common cases simple**

Because more often than not, the usual SQL calls used by the programmer are simple SELECT’s, INSERT’s, DELETE and UPDATE, these queries should be simple to perform with JDBC. However, more complex SQL statements should also be possible. Finally we decided to proceed with the implementation using Java Networking.And for dynamically updating the cache table we go for MS Access database.

Java has two things: a programming language and a platform.

Java is a high-level programming language that is all of the following

Simple Architecture-neutral

Object-oriented Portable

Distributed High-performance

Interpreted multithreaded

Robust Dynamic

Secure

Java is also unusual in that each Java program is both compiled and interpreted. With a compiler you translate a Java program into an intermediate language called Java byte codes the platform-independent code instruction is passed and run on the computer.

Compilation happens just once; interpretation occurs each time the program is executed. The figure illustrates how this works.

**Compilers**

**Interpreter**

**My Program**

**Java Program**

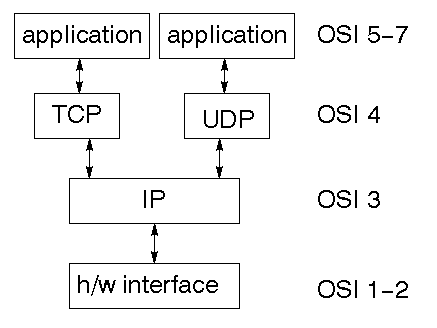
You can think of Java bytecodes as the machine code instructions for the Java Virtual Machine (Java VM). Every Java interpreter, whether it’s a Java development tool or a Web browser that can run Java applets, is an implementation of the Java VM. The Java VM can also be implemented in hardware.

Java byte codes help make “write once, run anywhere” possible. You can compile your Java program into byte codes on my platform that has a Java compiler. The byte codes can then be run on any implementation of the Java VM. For example, the same Java program can run Windows NT, Solaris, and Macintosh.

## **Networking**

### **TCP/IP stack**

The TCP/IP stack is shorter than the OSI one:



TCP is a connection-oriented protocol; UDP (User Datagram Protocol) is a connectionless protocol

### IP datagram

The IP layer provides a connectionless and unreliable delivery system. It considers each datagram independently of the others. Any association between datagrams must be supplied by the higher layers. The IP layer supplies a checksum that includes its own header. The header includes the source and destination addresses. The IP layer handles routing through the Internet. It is also responsible for breaking up large datagrams into smaller ones for transmission and reassembling them at the other end.

### UDP

UDP is also connectionless and unreliable. What it adds to IP is a checksum for the contents of the datagram and port numbers. These are used to give a client/server model - see later.

### TCP

TCP supplies logic to give a reliable connection-oriented protocol above IP. It provides a virtual circuit that two processes can use to communicate.

### Internet addresses

In order to use a service, you must be able to find it. The Internet uses an address scheme for machines so that they can be located. The address is a 32 bit integer which gives the IP address. This encodes a network ID and more addressing. The network ID falls into various classes according to the size of the network address.

### Network address

Class A uses 8 bits for the network address with 24 bits left over for other addresses. Class B uses 16 bit network addressing. Class C uses 24 bit network addressing and class D uses all 32.

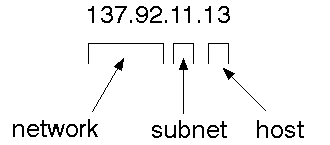
### Subnet address

Internally, the UNIX network is divided into sub networks. Building 11 is currently on one sub network and uses 10-bit addressing, allowing 1024 different hosts.

### Host address

8 bits are finally used for host addresses within our subnet. This places a limit of 256 machines that can be on the subnet.

### Total address



The 32 bit address is usually written as 4 integers separated by dots.

Port addresses

A service exists on a host, and is identified by its port. This is a 16 bit number. To send a message to a server, you send it to the port for that service of the host that it is running on. This is not location transparency! Certain of these ports are "well known".

### Sockets

A socket is a data structure maintained by the system to handle network connections. A socket is created using the call socket. It returns an integer that is like a file descriptor. In fact, under Windows, this handle can be used with Read File and Write File functions.

#include <sys/types.h>

#include <sys/socket.h>

int socket(int family, int type, int protocol);

Here "family" will be AF\_INET for IP communications, protocol will be zero, and type will depend on whether TCP or UDP is used. Two processes wishing to communicate over a network create a socket each. These are similar to two ends of a pipe - but the actual pipe does not yet exist.

**JFree Chart**

JFreeChart is a free 100% Java chart library that makes it easy for developers to display professional quality charts in their applications. JFreeChart's extensive feature set includes:

A consistent and well-documented API, supporting a wide range of chart types;

A flexible design that is easy to extend, and targets both server-side and client-side applications;

Support for many output types, including Swing components, image files (including PNG and JPEG), and vector graphics file formats (including PDF, EPS and SVG);

JFreeChart is "open source" or, more specifically, [free software](http://www.gnu.org/philosophy/free-sw.html). It is distributed under the terms of the [GNU Lesser General Public Licence](http://www.gnu.org/licenses/lgpl.html) (LGPL), which permits use in proprietary applications.

## 1. Map Visualizations

Charts showing values that relate to geographical areas. Some examples include: (a) population density in each state of the United States, (b) income per capita for each country in Europe, (c) life expectancy in each country of the world. The tasks in this project include:

Sourcing freely redistributable vector outlines for the countries of the world, states/provinces in particular countries (USA in particular, but also other areas);

Creating an appropriate dataset interface (plus default implementation), a rendered, and integrating this with the existing XYPlot class in JFreeChart;

Testing, documenting, testing some more, documenting some more.

## 2. Time Series Chart Interactivity

Implement a new (to JFreeChart) feature for interactive time series charts --- to display a separate control that shows a small version of ALL the time series data, with a sliding "view" rectangle that allows you to select the subset of the time series data to display in the main chart.

## 3. Dashboards

There is currently a lot of interest in dashboard displays. Create a flexible dashboard mechanism that supports a subset of JFreeChart chart types (dials, pies, thermometers, bars, and lines/time series) that can be delivered easily via both Java Web Start and an applet.

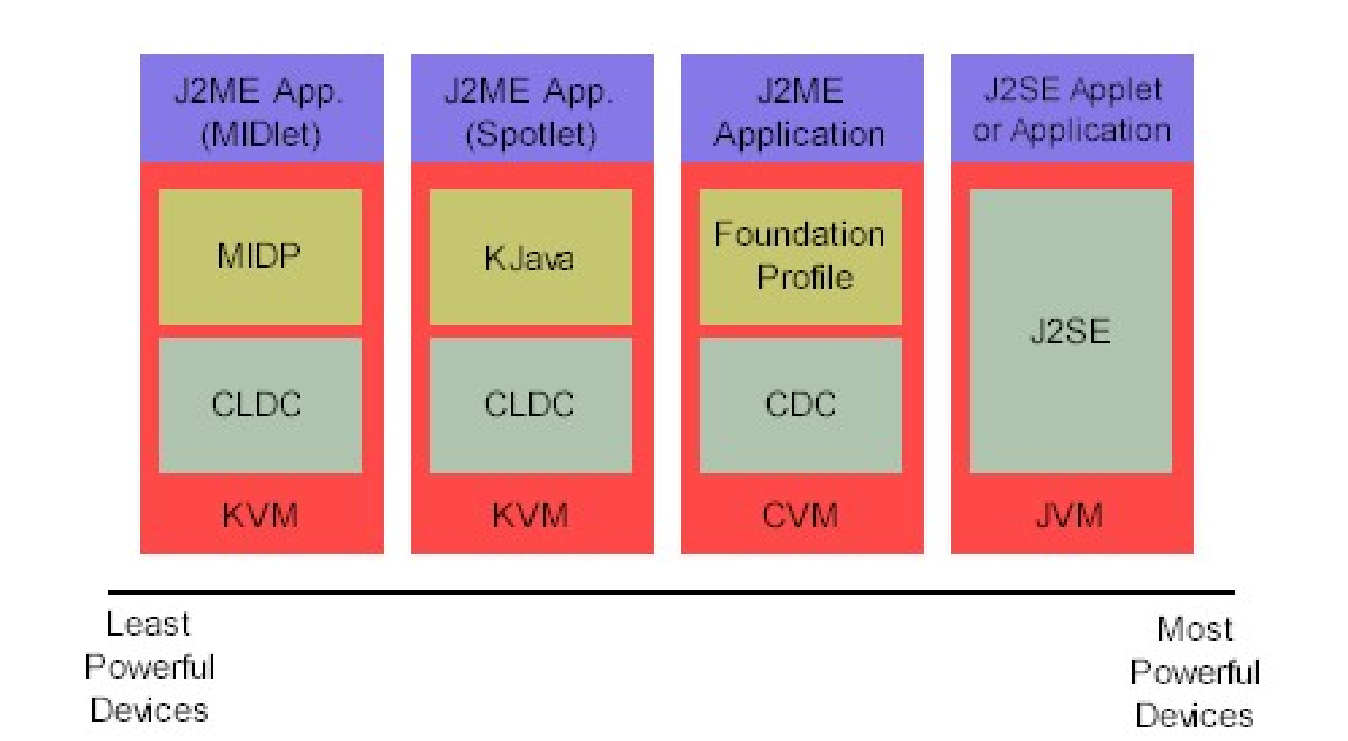
## 4. Property Editors

The property editor mechanism in JFreeChart only handles a small subset of the properties that can be set for charts. Extend (or reimplement) this mechanism to provide greater end-user control over the appearance of the charts.

**J2ME (Java 2 Micro edition):-**

Sun Microsystems defines J2ME as "a highly optimized Java run-time environment targeting a wide range of consumer products, including pagers, cellular phones, screen-phones, digital set-top boxes and car navigation systems." Announced in June 1999 at the JavaOne Developer Conference, J2ME brings the cross-platform functionality of the Java language to smaller devices, allowing mobile wireless devices to share applications. With J2ME, Sun has adapted the Java platform for consumer products that incorporate or are based on small computing devices.

**1. General J2ME architecture**



J2ME uses configurations and profiles to customize the Java Runtime Environment (JRE). As a complete JRE, J2ME consists of a configuration, which determines the JVM used, and a profile, which defines the application by adding domain-specific classes. The configuration defines the basic run-time environment as a set of core classes and a specific JVM that run on specific types of devices. We'll discuss configurations in detail in the The profile defines the application; specifically, it adds domain-specific classes to the J2ME configuration to define certain uses for devices. We'll cover profiles in depth in the The following graphic depicts the relationship between the different virtual machines, configurations, and profiles. It also draws a parallel with the J2SE API and its Java virtual machine. While the J2SE virtual machine is generally referred to as a JVM, the J2ME virtual machines, KVM and CVM, are subsets of JVM. Both KVM and CVM can be thought of as a kind of Java virtual machine -- it's just that they are shrunken versions of the J2SE JVM and are specific to J2ME.

**2.Developing J2ME applications**

Introduction In this section, we will go over some considerations you need to keep in mind when developing applications for smaller devices. We'll take a look at the way the compiler is invoked when using J2SE to compile J2ME applications. Finally, we'll explore packaging and deployment and the role preverification plays in this process.

**3.Design considerations for small devices**

Developing applications for small devices requires you to keep certain strategies in mind during the design phase. It is best to strategically design an application for a small device before you begin coding. Correcting the code because you failed to consider all of the "gotchas" before developing the application can be a painful process. Here are some design strategies to consider:

\* Keep it simple. Remove unnecessary features, possibly making those features a separate, secondary application.

\* Smaller is better. This consideration should be a "no brainer" for all developers. Smaller applications use less memory on the device and require shorter installation times. Consider packaging your Java applications as compressed Java Archive (jar) files.

\* Minimize run-time memory use. To minimize the amount of memory used at run time, use scalar types in place of object types. Also, do not depend on the garbage collector. You should manage the memory efficiently yourself by setting object references to null when you are finished with them. Another way to reduce run-time memory is to use lazy instantiation, only allocating objects on an as-needed basis. Other ways of reducing overall and peak memory use on small devices are to release resources quickly, reuse objects, and avoid exceptions.

**4.Configurations overview**

The configuration defines the basic run-time environment as a set of core classes and a specific JVM that run on specific types of devices. Currently, two configurations exist for J2ME, though others may be defined in the future:

\* **Connected Limited Device Configuration (CLDC)** is used specifically with the KVM for 16-bit or 32-bit devices with limited amounts of memory. This is the configuration (and the virtual machine) used for developing small J2ME applications. Its size limitations make CLDC more interesting and challenging (from a development point of view) than CDC. CLDC is also the configuration that we will use for developing our drawing tool application. An example of a small wireless device running small applications is a Palm hand-held computer.

\* **Connected Device Configuration (CDC)** is used with the C virtual machine (CVM) and is used for 32-bit architectures requiring more than 2 MB of memory. An example of such a device is a Net TV box.

**5.J2ME profiles**

**What is a J2ME profile?**

As we mentioned earlier in this tutorial, a profile defines the type of device supported. The Mobile Information Device Profile (MIDP), for example, defines classes for cellular phones. It adds domain-specific classes to the J2ME configuration to define uses for similar devices. Two profiles have been defined for J2ME and are built upon CLDC: KJava and MIDP. Both KJava and MIDP are associated with CLDC and smaller devices. Profiles are built on top of configurations. Because profiles are specific to the size of the device (amount of memory) on which an application runs, certain profiles are associated with certain configurations.

A skeleton profile upon which you can create your own profile, the Foundation Profile, is available for CDC.

**Profile 1: KJava**

KJava is Sun's proprietary profile and contains the KJava API. The KJava profile is built on top of the CLDC configuration. The KJava virtual machine, KVM, accepts the same byte codes and class file format as the classic J2SE virtual machine. KJava contains a Sun-specific API that runs on the Palm OS. The KJava API has a great deal in common with the J2SE Abstract Windowing Toolkit (AWT). However, because it is not a standard J2ME package, its main package is com.sun.java. We'll learn more about the KJava API later in this tutorial when we develop some sample applications.

**Profile 2: MIDP**

MIDP is geared toward mobile devices such as cellular phones and pagers. The MIDP, like KJava, is built upon CLDC and provides a standard run-time environment that allows new applications and services to be deployed dynamically on end user devices. MIDP is a common, industry-standard profile for mobile devices that is not dependent on a specific vendor. It is a complete and supported foundation for mobile application

development. MIDP contains the following packages, the first three of which are core CLDC packages, plus three MIDP-specific packages.

\* java.lang

\* java.io

\* java.util

\* javax.microedition.io

\* javax.microedition.lcdui

\* javax.microedition.midlet

\* javax.microedition.rms

**9.7 SOURCE CODE**

**Home.html**

<!DOCTYPE html>

<html>

<head>

<title>FORECASTING AND ANTICIPATORY ESTIMATION OF HUMAN MOVEMENT PATTERNS</title>

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

<link rel="stylesheet" href="layout/styles/layout.css" type="text/css" />

</head>

<body id="top">

<div class="wrapper col1">

<div id="header">

<div id="logo" style="width:auto;">

<h1><a href="#">FORECASTING AND ANTICIPATORY ESTIMATION OF HUMAN MOVEMENT PATTERNS</a></h1>

<p><strong>Based on Application-Collected Location Data</strong></p>

</div>

<br class="clear" />

</div>

</div>

<div class="wrapper col2">

<div id="topbar">

<div id="topnav">

<ul>

<li class="active"><a href="index.html">Home</a></li>

<li><a href="Admin.jsp">Admin</a></li>

<li class="last"><a href="Cloud.jsp"></a></li>

</ul>

</div>

<br class="clear" />

</div>

</div>

<div class="wrapper col3">

<div id="intro">

<div class="fl\_left">

<p>

In order to preserve the physical context of the locations, we contextually cluster the locations into multiple nonoverlapping districts of the city instead of using fixed coordinate grid that will lose the physical context. We also compare the predictability and the prediction accuracy between the two divisions to analyze the effect of context on prediction. Results reveal that the trajectories on context-based division are more predictable than those on division without context under the same spatial granularity

</p>

</div>

<div class="fl\_right"></div>

<br class="clear" />

</div>

</div>

<div class="wrapper col5">

<div id="container">

<div id="content">

<h2>About This Project</h2>

<p align="justify">

In the modern information society, accurate prediction of human mobility becomes increasingly essential in various areas such as city planning and resource management. With users’ historical trajectories, the inherent patterns of their movements can be extracted and utilized to accurately predict the future movements. In this paper, based on a dataset of 100,000 individuals’ actively uploaded location information collected by apps, we discover the average theoretical limits of the predictability to be as high as 93%. Since the app-collected data contains the physical context of the location, we implement a clustering method based on the contextual information that cluster the locations into three divisions, street, district and region. In order to solve the unevenly distribution and the high missing rate of the application collected location data, we firstly use the Gibbs sampling algorithm to complete the missing data of the trajectory and then employ a high-order Markov chain model to predict the most likely locations visited by each user. Result shows that our prediction algorithm can achieve accuracy as high as 67%, 78%, 87% for the three context-based divisions respectively, which are 10% higher on average than the divisions without context. In addition, the correlation coefficient between prediction accuracy and predictability reaches as high as 0.86. Finally, we investigate various factors including spatial and temporal resolution, orders of Markov models, radius of gyration, in order to explore the predictability under different circumstances.

</p>

</div>

<div id="column">

</div>

<br class="clear" />

</div>

</div>

<div class="wrapper col6">

<div id="footer">

<br class="clear" />

</div>

</div>

<div class="wrapper col7">

<div id="copyright">

<p class="fl\_left">FORECASTING AND ANTICIPATORY ESTIMATION OF HUMAN MOVEMENT PATTERNS</p>

<br class="clear" />

</div>

</div>

</body>

</html>

**Admin.jsp**

<%--

Document : Admin

Created on : 9 Jul, 2021, 6:10:38 PM

Author : KishanVenky

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<title>FORECASTING AND ANTICIPATORY ESTIMATION OF HUMAN MOVEMENT PATTERNS</title>

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

<link rel="stylesheet" href="layout/styles/layout.css" type="text/css" />

</head>

<body id="top">

<div class="wrapper col1">

<div id="header">

<div id="logo" style="width:auto;">

<h1><a href="#">FORECASTING AND ANTICIPATORY ESTIMATION OF HUMAN MOVEMENT PATTERNS</a></h1>

<p><strong>Based on Application-Collected Location Data</strong></p>

</div>

<br class="clear" />

</div>

</div>

<div class="wrapper col2">

<div id="topbar">

<div id="topnav">

<ul>

<li><a href="index.html">Home</a></li>

<li class="active"><a href="Admin.jsp">Admin</a></li>

<li class="last"><a href="Cloud.jsp"></a></li>

</ul>

</div>

<br class="clear" />

</div>

</div>

<div class="wrapper col3">

<div id="intro">

<div class="fl\_left">

<h3>Admin Login Here</h3>

<center>

<form action="AdminAction.jsp" method="post">

<table style="border:1px solid black;height:150px">

<tr><th>UserName</th><td><input type="text" name="username" required=""/></td></tr>

<tr><th>Password</th><td><input type="password" name="password" required=""/></td></tr>

<tr><th></th><td><input type="submit" value="Login" />

<input type="reset" value="Reset" /></td></tr>

</table>

</form>

</center>

</div>

<div class="fl\_right"></div>

<br class="clear" />

</div>

</div>

<div class="wrapper col5">

<div id="container">

<div id="content">

</div>

<div id="column">

</div>

<br class="clear" />

</div>

</div>

<div class="wrapper col6">

<div id="footer">

<br class="clear" />

</div>

</div>

<div class="wrapper col7">

<div id="copyright">

<p class="fl\_left">FORECASTING AND ANTICIPATORY ESTIMATION OF HUMAN MOVEMENT PATTERNS.</p>

<br class="clear" />

</div>

</div>

</body>

</html>

**AdminHome.jsp**

<%--

Document : AdminHome

Created on : 15 Jul, 2021, 3:41:06 PM

Author : KishanVenky

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<title>FORECASTING AND ANTICIPATORY ESTIMATION OF HUMAN MOVEMENT PATTERNS</title>

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

<link rel="stylesheet" href="layout/styles/layout.css" type="text/css" />

</head>

<body id="top">

<div class="wrapper col1">

<div id="header">

<div id="logo" style="width:auto;">

<h1><a href="#">FORECASTING AND ANTICIPATORY ESTIMATION OF HUMAN MOVEMENT PATTERNS</a></h1>

<p><strong>Based on Application-Collected Location Data</strong></p>

</div>

<br class="clear" />

</div>

</div>

<div class="wrapper col2">

<div id="topbar">

<div id="topnav">

<ul>

<li class="active"><a href="AdminHome.jsp">Home</a></li>

<li ><a href="Upload.jsp">Upload</a></li>

<li><a href="ViewRecords.jsp">View No.of Records</a></li>

<li><a href="ProcessDataset.jsp">Process Dataset</a></li>

<li><a href="Cluster.jsp">Cluster</a></li>

<li class="last"><a href="Admin.jsp">Logout</a></li>

</ul>

</div>

<br class="clear" />

</div>

</div>

<div class="wrapper col3">

<div id="intro">

<div class="fl\_left">

<marquee><h3>Welcome to Admin Home Page</h3></marquee>

<center>

<p align="justify">

We also compare the predictability and the prediction accuracy between the two divisions to analyze the effect of context on prediction. Results reveal that the trajectories on context-based division are more predictable than those on division without context under the same spatial granularity

</p>

</center>

</div>

<div class="fl\_right">

<image src="images/owner.png" width="350" height="300"/>

</div>

<br class="clear" />

</div>

</div>

<div class="wrapper col5">

<div id="container">

<div id="content">

</div>

<div id="column">

</div>

<br class="clear" />

</div>

</div>

<div class="wrapper col6">

<div id="footer">

<br class="clear" />

</div>

</div>

<div class="wrapper col7">

<div id="copyright">

<p class="fl\_left">FORECASTING AND ANTICIPATORY ESTIMATION OF HUMAN MOVEMENT PATTERNS</p>

<br class="clear" />

</div>

</div>

</body>

</html>

**Uplode.jsp**

<%--

Document : Admin

Created on : 9 Jul, 2021, 6:10:38 PM

Author : KishanVenky

--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<title>FORECASTING AND ANTICIPATORY ESTIMATION OF HUMAN MOVEMENT PATTERNS</title>

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

<link rel="stylesheet" href="layout/styles/layout.css" type="text/css" />

</head>

<body id="top">

<div class="wrapper col1">

<div id="header">

<div id="logo" style="width:auto;">

<h1><a href="#">FORECASTING AND ANTICIPATORY ESTIMATION OF HUMAN MOVEMENT PATTERNS</a></h1>

<p><strong>Based on Application-Collected Location Data</strong></p>

</div>

<br class="clear" />

</div>

</div>

<div class="wrapper col2">

<div id="topbar">

<div id="topnav">

<ul>

<li><a href="AdminHome.jsp">Home</a></li>

<li class="active"><a href="Upload.jsp">Upload</a></li>

<li><a href="ViewRecords.jsp">View No.of Records</a></li>

<li><a href="ProcessDataset.jsp">Process Dataset</a></li>

<li><a href="Cluster.jsp">Cluster</a></li>

<li class="last"><a href="Admin.jsp">Logout</a></li>

</ul>

</div>

<br class="clear" />

</div>

</div>

<div class="wrapper col3">

<div id="intro">

<div class="fl\_left">

<h3>Upload Dataset Here</h3>

<center>

<form action="ExcelReader" method="post" enctype="multipart/form-data">

<table style="border:1px solid black;height:150px">

<tr><th>Choose DataSet(xlsx)</th><td><input type="file" name="excel" required=""/></td></tr>

<tr><th></th><td><input type="submit" value="upload" />

<input type="reset" value="Reset" /></td></tr>

</table>

</form>

</center>

</div>

<div class="fl\_right"></div>

<br class="clear" />

</div>

</div>

<div class="wrapper col5">

<div id="container">

<div id="content">

</div>

<div id="column">

</div>

<br class="clear" />

</div>

</div>

<div class="wrapper col6">

<div id="footer">

<br class="clear" />

</div>

</div>

<div class="wrapper col7">

<div id="copyright">

<p class="fl\_left">FORECASTING AND ANTICIPATORY ESTIMATION OF HUMAN MOVEMENT PATTERNS</p>

<br class="clear" />

</div>

</div>

</body>

</html>

**ViewRecords.jsp**

<%--

Document : Admin

Created on : 9 Jul, 2021, 6:10:38 PM

Author : KishanVenky

--%>

<%@page import="com.database.Queries"%>

<%@page import="java.sql.ResultSet"%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<title>FORECASTING AND ANTICIPATORY ESTIMATION OF HUMAN MOVEMENT PATTERNS</title>

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

<link rel="stylesheet" href="layout/styles/layout.css" type="text/css" />

</head>

<body id="top">

<div class="wrapper col1">

<div id="header">

<div id="logo" style="width:auto;">

<h1><a href="#">FORECASTING AND ANTICIPATORY ESTIMATION OF HUMAN MOVEMENT PATTERNS</a></h1>

<p><strong>Based on Application-Collected Location Data</strong></p>

</div>

<br class="clear" />

</div>

</div>

<div class="wrapper col2">

<div id="topbar">

<div id="topnav">

<ul>

<li><a href="AdminHome.jsp">Home</a></li>

<li><a href="Upload.jsp">Upload</a></li>

<li class="active"><a href="ViewRecords.jsp">View No.of Records</a></li>

<li><a href="ProcessDataset.jsp">Process Dataset</a></li>

<li><a href="Cluster.jsp">Cluster</a></li>

<li class="last"><a href="Admin.jsp">Logout</a></li>

</ul>

</div>

<br class="clear" />

</div>

</div>

<div class="wrapper col5">

<div id="container">

<div id="content">

<%

try{

ResultSet r=Queries.getExecuteQuery("select count(\*) from dataset");

int i=0;

while(r.next()){

i=r.getInt(1);

}

%>

<H3> TOTAL NO.OF RECORDS : <%=i%> </H3>

<%}catch(Exception e){

out.println(e);

}%>

<center>

<iframe src="ViewData.jsp" width=1000" height="500"></iframe>

</center>

</div>

<div id="column">

</div>

<br class="clear" />

</div>

</div>

<div class="wrapper col6">

<div id="footer">

<br class="clear" />

</div>

</div>

<div class="wrapper col7">

<div id="copyright">

<p class="fl\_left">FORECASTING AND ANTICIPATORY ESTIMATION OF HUMAN MOVEMENT PATTERNS</p>

<br class="clear" />

</div>

</div>

</body>

</html>

**ProcessDataset.jsp:**

<%--

Document : Admin

Created on : 9 Jul, 2021, 6:10:38 PM

Author : KishanVenky

--%>

<%@page import="com.database.Queries"%>

<%@page import="java.sql.ResultSet"%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<title>FORECASTING AND ANTICIPATORY ESTIMATION OF HUMAN MOVEMENT PATTERNS</title>

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

<link rel="stylesheet" href="layout/styles/layout.css" type="text/css" />

</head>

<body id="top">

<div class="wrapper col1">

<div id="header">

<div id="logo" style="width:auto;">

<h1><a href="#">FORECASTING AND ANTICIPATORY ESTIMATION OF HUMAN MOVEMENT PATTERNS</a></h1>

<p><strong>Based on Application-Collected Location Data</strong></p>

</div>

<br class="clear" />

</div>

</div>

<div class="wrapper col2">

<div id="topbar">

<div id="topnav">

<ul>

<li><a href="AdminHome.jsp">Home</a></li>

<li><a href="Upload.jsp">Upload</a></li>

<li><a href="ViewRecords.jsp">View No.of Records</a></li>

<li class="active"><a href="ProcessDataset.jsp">Process Dataset</a></li>

<li><a href="Cluster.jsp">Cluster</a></li>

<li class="last"><a href="Admin.jsp">Logout</a></li>

</ul>

</div>

<br class="clear" />

</div>

</div>

<div class="wrapper col5">

<div id="container">

<div id="content">

<%try{

Queries.getExecuteUpdate("delete from dataset where Residential='null'");

ResultSet r=Queries.getExecuteQuery("select count(\*) from dataset");

int i=0;

while(r.next()){

i=r.getInt(1);

}

%>

<H3> TOTAL NO.OF RECORDS : <%=i%> </H3>

<%}catch(Exception e){

out.println(e);

}%>

<center>

<iframe src="ViewData.jsp" width=1000" height="500"></iframe>

</center>

</div>

<div id="column">

</div>

<br class="clear" />

</div>

</div>

<div class="wrapper col6">

<div id="footer">

<br class="clear" />

</div>

</div>

<div class="wrapper col7">

<div id="copyright">

<p class="fl\_left">FORECASTING AND ANTICIPATORY ESTIMATION OF HUMAN MOVEMENT PATTERNS</p>

<br class="clear" />

</div>

</div>

</body>

</html>

**Cluster.jsp:**

<%--

Document : Admin

Created on : 9 Jul, 2021, 6:10:38 PM

Author : KishanVenky

--%>

<%@page import="java.sql.Statement"%>

<%@page import="java.sql.Connection"%>

<%@page import="com.database.Dbconnection"%>

<%@page import="com.database.Queries"%>

<%@page import="java.sql.ResultSet"%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<title>FORECASTING AND ANTICIPATORY ESTIMATION OF HUMAN MOVEMENT PATTERNS</title>

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

<link rel="stylesheet" href="layout/styles/layout.css" type="text/css" />

</head>

<body id="top">

<div class="wrapper col1">

<div id="header">

<div id="logo" style="width:auto;">

<h1><a href="#">FORECASTING AND ANTICIPATORY ESTIMATION OF HUMAN MOVEMENT PATTERNS</a></h1>

<p><strong>Based on Application-Collected Location Data</strong></p>

</div>

<br class="clear" />

</div>

</div>

<div class="wrapper col2">

<div id="topbar">

<div id="topnav">

<ul>

<li><a href="AdminHome.jsp">Home</a></li>

<li><a href="Upload.jsp">Upload</a></li>

<li><a href="ViewRecords.jsp">View No.of Records</a></li>

<li><a href="ProcessDataset.jsp">Process Dataset</a></li>

<li class="active"><a href="Cluster.jsp">Cluster</a></li>

<li class="last"><a href="Admin.jsp">Logout</a></li>

</ul>

</div>

<br class="clear" />

</div>

</div>

<h3>Country wise Cluster</h3>

<%

try{

Connection con=Dbconnection.getcon();

Statement st=con.createStatement();

ResultSet rr=Queries.getExecuteQuery("select distinct country from dataset");

while(rr.next()){

String cname=rr.getString(1);

%>

<table border="1">

<tr><th>Country Name</th><th colspan="7"> <font color="red"><h4><%=cname%></h4></font></th></tr>

<tr>

<th>Date</th>

<th>Retail\_Recreation</th>

<th>Grocery\_Pharmacy</th>

<th>Parks</th>

<th>Transit\_stations</th>

<th>Workplaces</th>

<th>Residential</th>

<th>Not\_enough\_Data</th>

</tr>

<%

ResultSet r=st.executeQuery("select \* from dataset where country='"+cname+"'");

while(r.next()){

%>

<tr>

<td><%=r.getString(2)%></td>

<td><%=r.getString(3)%></td>

<td><%=r.getString(4)%></td>

<td><%=r.getString(5)%></td>

<td><%=r.getString(6)%></td>

<td><%=r.getString(7)%></td>

<td><%=r.getString(8)%></td>

<td><%=r.getString(9)%></td>

</tr>

</tr>

<%

}

}

}catch(Exception e){

out.println(e);

}

%>

</table>

<div class="wrapper col5">

<div id="container">

<div id="content">

</div>

<div id="column">

</div>

<br class="clear" />

</div>

</div>

<div class="wrapper col6">

<div id="footer">

<br class="clear" />

</div>

</div>

<div class="wrapper col7">

<div id="copyright">

<p class="fl\_left">FORECASTING AND ANTICIPATORY ESTIMATION OF HUMAN MOVEMENT PATTERNS</p>

<br class="clear" />

</div>

</div>

</body>

</html>

**CHAPTER-10**

**RESULT/DISCUSSION**

**10.1 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 tests. Each test type addresses a specific testing requirement.

**TYPES OF TESTING**

**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 centred 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.

**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 that cannot 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: | USER 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 field 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.

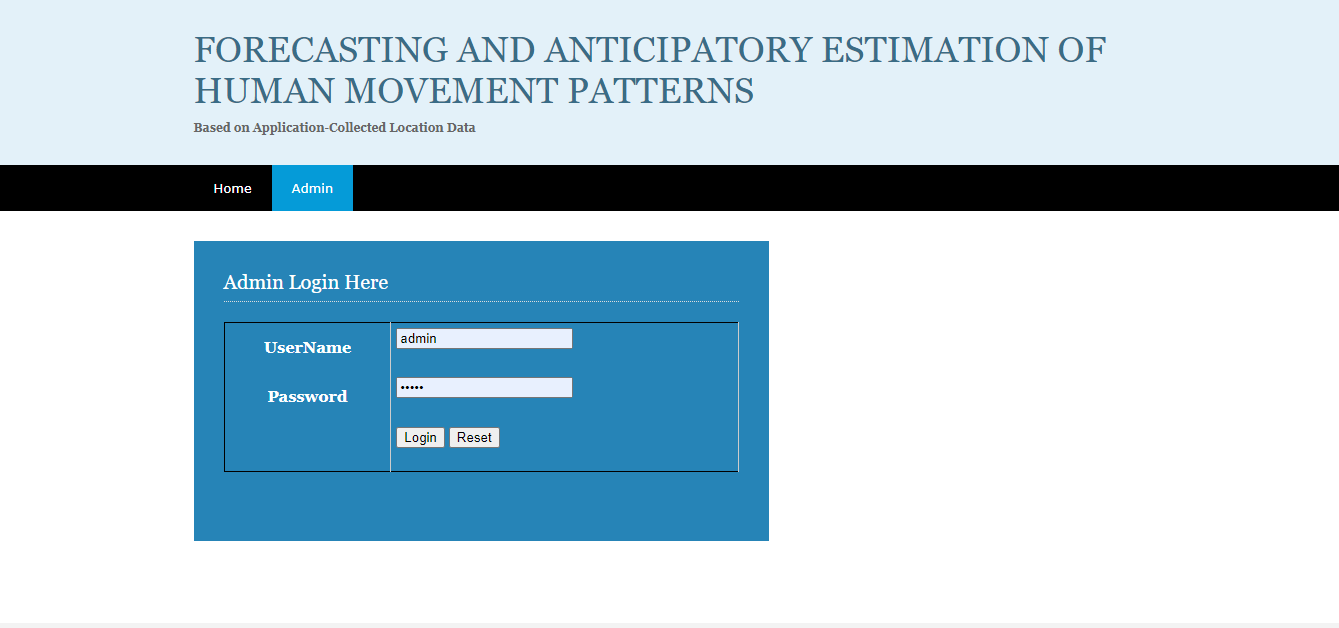
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Modu le** | **Functio nality** | **Test Case** | **Expected Results** | **Actual Results** | **Res ult** | **Priori ty** |
| User | Login Usecase | 1. Navigate To Www.Sample.Co m 2. Click On Submit Button Without Entering Username   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 |

**10.2 OUTPUT SCREENS**

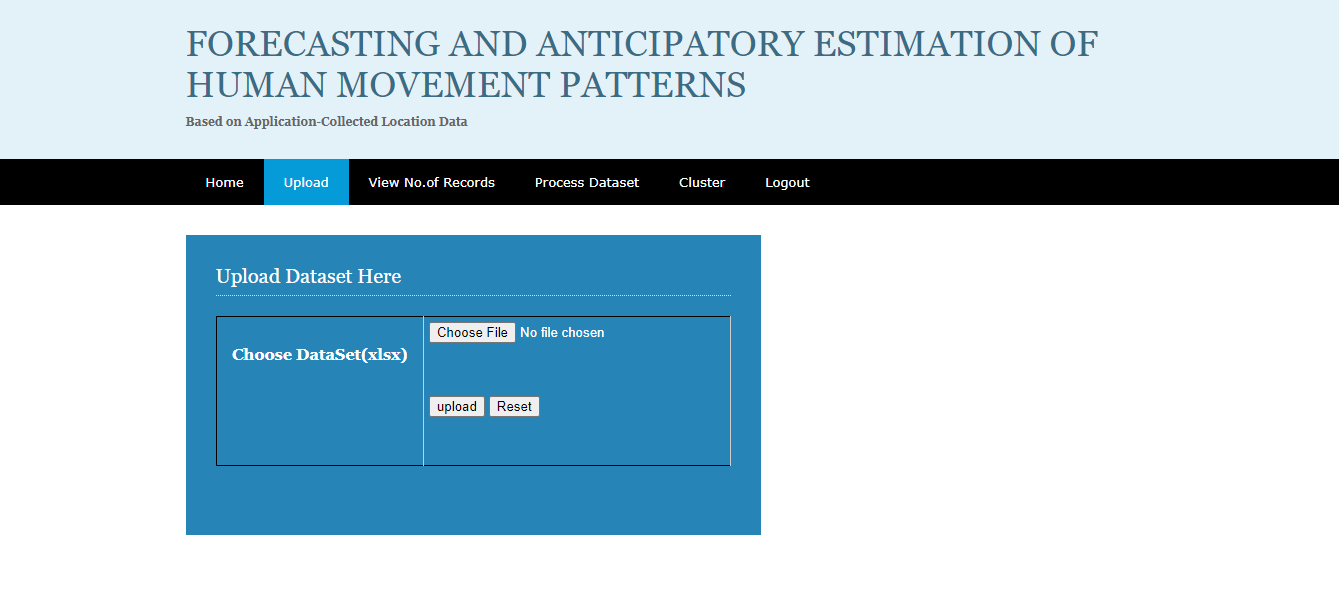
1. HOME PAGE



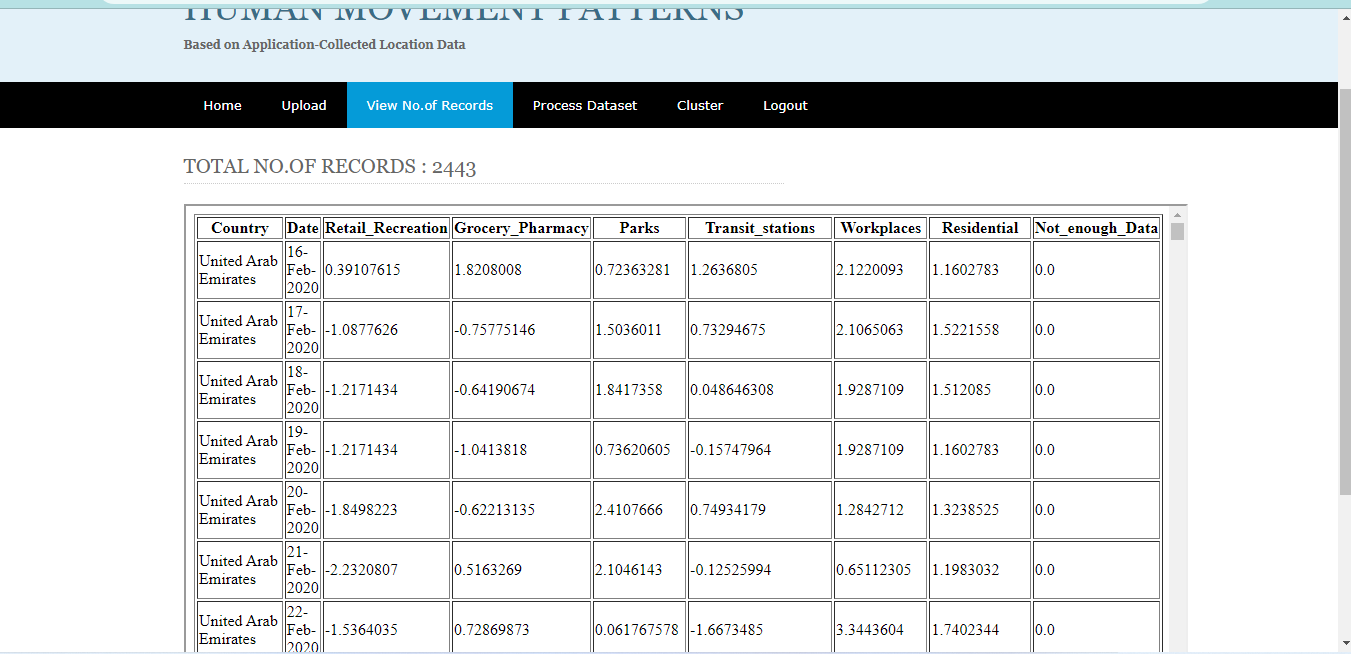
1. ADMIN LOGIN



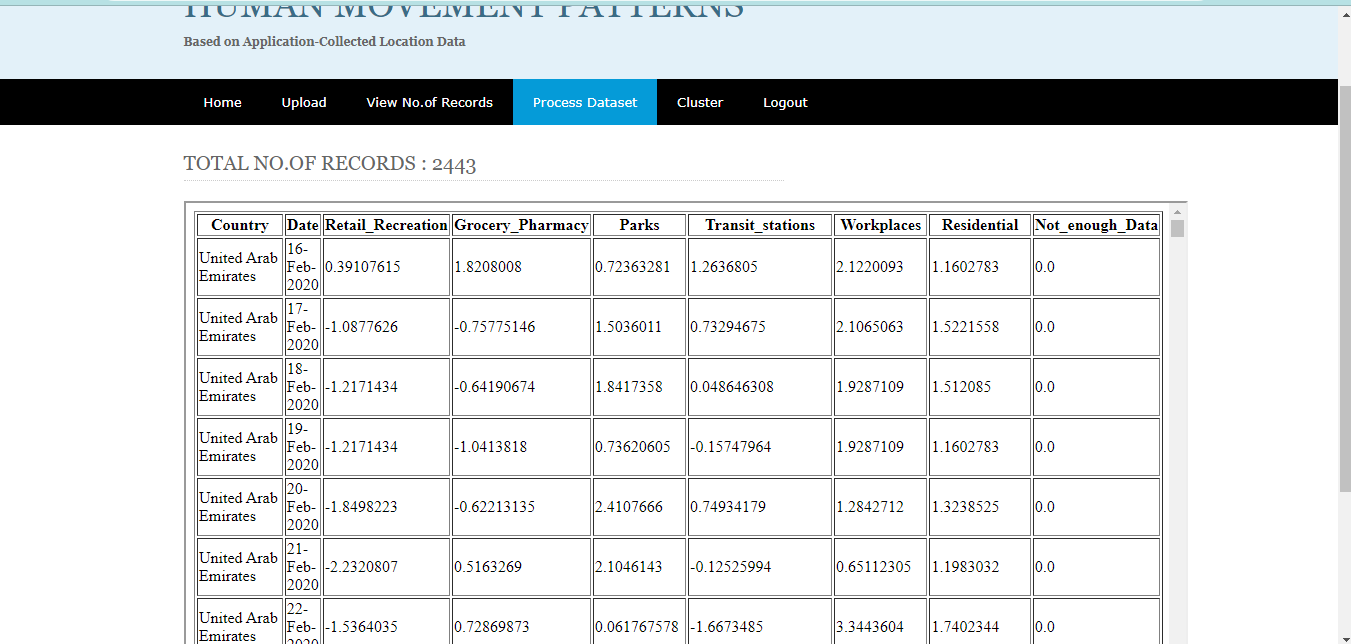
1. UPLODE DATA



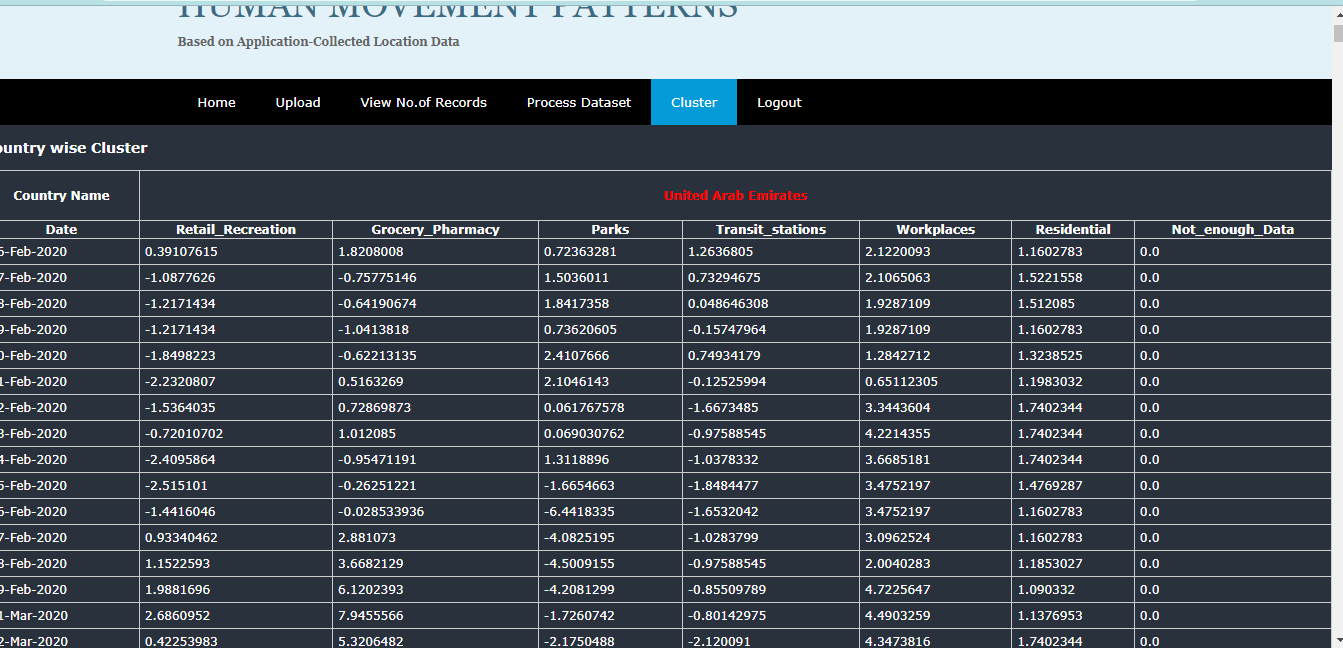
1. VIEW NO.OF RECORDS



1. PROCESS DATA



1. CLUSTER



**CHAPTER-11**

**CONCLUSION**

**11.1 CONCLUSION**

Predicting users' historical trajectories is crucial for enhancing location-based services. In this paper, we address three key challenges in app-collected datasets by conducting a comprehensive analysis of predictability and prediction methods on large-scale location data.

First, we apply a context-based segmentation method to preserve the physical context of app-collected data. This approach improves trajectory prediction accuracy by 10% compared to methods without context.

Second, we develop a Markov chain model using Gibbs sampling to address uneven data distribution and high missing rates. Our method achieves a strong correlation coefficient of 0.86 between prediction accuracy and predictability.

Third, we analyze various factors influencing prediction accuracy, including spatial and temporal resolution, Markov model orders, and radius of gyration.

Overall, our analysis provides a systematic and comprehensive understanding of users' mobility patterns in app-collected datasets.

**CHAPTER-12**

**REFERENCES**

[1] C. Song, Z. Qu, N. Blumm, and A.-L. Barabasi, “Limits of predictability ´ in human mobility,” Science, vol. 327, no. 5968, pp. 1018–1021, 2010.

[2] D. Wang, D. Pedreschi, C. Song, F. Giannotti, and A. L. Barabasi, “Human mobility, social ties, and link prediction,” in ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, Ca, Usa, August, pp. 1100–1108, 2011.

[3] X. Lu, E. Wetter, N. Bharti, A. J. Tatem, and L. Bengtsson, “Approaching the limit of predictability in human mobility,” Scientific reports, vol. 3, 2013.

[4] G. Liu and G. M. Jr, “A class of mobile motion prediction algorithms for wireless mobile computing and communications,” Mobile Networks and Applications, vol. 1, no. 2, pp. 113–121, 1996.

[5] S. C. Liou and H. C. Lu, “Applied neural network for location prediction and resources reservation scheme in wireless networks,” in International Conference on Communication Technology Proceedings, pp. 958–961 vol.2, 2003.

[6] S. Akoush and A. Sameh, “Movement prediction using bayesian learning for neural networks,” in International Conference on Systems and Networks Communications, p. 6, 2007.

[7] J. Petzold, F. Bagci, W. Trumler, and T. Ungerer, “Global and local state context prediction,” 2003.

[8] F. Xu, P. Zhang, and Y. Li, “Context-aware real-time population estimation for metropolis,” in ACM International Joint Conference on Pervasive and Ubiquitous Computing, pp. 1064–1075, 2016.

[9] I. Kontoyiannis, P. H. Algoet, Y. M. Suhov, and A. J. Wyner, “Nonparametric entropy estimation for stationary processes and random fields, with applications to english text,” IEEE Transactions on Information Theory, vol. 44, no. 3, pp. 1319–1327, 1998.

[10] RobertM, Transmission of information : a statistical theory of communications. M.I.T. Press, 1961.