RAJALAKSHMI ENGINEERING COLLEGE
RAJALAKSHMI NAGAR, THANDALAM – 602 105

RAJALAKSHMI
ENGINEERING COLLEGE
An AUTONOMOUS Institution
Affiliated to ANNA UNIVERSITY, Chennai

CP23211 ADVANCED SOFTWARE ENGINEERING LAB

Laboratory Record
Note Book

Name :…Sandeep Abinash A ……

Year / Branch / Section : 1st Year……. M.E-CSE ……..

University Register No. : 2116230711008……

College Roll No. : …08……

Semester : …2ND……SEM…..……

Academic Year : ……2023 - 2024……...

RAJALAKSHMI ENGINEERING COLLEGE

RAJALAKSHMI NAGAR, THANDALAM – 602 105

BONAFIDE CERTIFICATE

Name: Sandeep Abinash A

Academic Year: 2023-2024 Semester: 2 Branch: ME-CSE

Register No: 2116230711008

Certified that this is the bonafide record of work done by the above student in the CP23211- Advanced Software Engineering Laboratory during the year 2023- 2024

Signature of Faculty-in-charge

Submitted for the Practical Examination held on 22/06/2024

Internal Examiner External Examiner

INDEX

DEEP CLUSTERING TOPIC MODEL: AN APPROACH TO CYBER ANOMALY DETECTION AND PROFILING

OVERVIEW OF THE PROJECT:

This project presents an automated system for identifying and profiling emerging cyber threats based on natural language processing (NLP) techniques applied to Twitter data. With the proliferation of social media as a platform for discussing cybersecurity issues, leveraging Twitter as a valuable source of information is becoming increasingly pertinent. The proposed system follows a systematic approach, beginning with the collection of cybersecurity-related tweets, followed by preprocessing steps to clean and prepare the data. Using NLP techniques, the system identifies entities such as malware names, software vulnerabilities, hacker groups, and targeted organizations mentioned in the tweets. Sentiment analysis is then employed to gauge the sentiment associated with these entities, while topic modeling helps identify clusters of related discussions. Anomaly detection techniques are utilized to detect unusual patterns indicative of emerging threats. The system profiles these threats, aggregating relevant information for alert generation. Incorporating a feedback loop mechanism ensures continuous refinement of threat identification and profiling. By integrating with existing cybersecurity systems and addressing security and privacy concerns, this automated approach enhances organizations' ability to proactively identify and respond to emerging cyber threats in the dynamic landscape of cybersecurity.

SOFTWARE REQUIREMENTS SPECIFICATION (SRS)

EXP.NO: 1 **DATE: 05/03/2024**

CONTENTS

DEEP CLUSTERING TOPIC MODEL: AN APPROACH TO CYBER ANOMALY DETECTION AND PROFILING

1. Introduction

1.1 Purpose:

The purpose of the proposed system prioritizes security and privacy considerations to safeguard sensitive information and ensure regulatory compliance. Robust encryption protocols, access controls, and anonymization techniques are implemented to protect the confidentiality, integrity, and availability of the Twitter data being analyzed. By adhering to stringent security standards, the system instills confidence in users regarding the protection of their data and insights generated by the system.In summary, the proposed project presents an innovative approach to automated cyber threat identification and profiling, leveraging NLP techniques applied to Twitter data.

1.2 Scope:

By harnessing the power of social media analytics, organizations can gain timely insights into emerging cyber threats, enabling them to fortify their defenses and mitigate risks effectively.

2. Overall Description

2.1 Product Perspective:

The Deep clustered topic model for cyber anomaly and profiling is a machine learning model which can be deployed as a machine learning model capable of profiling anomaly and threats based on social media data

2.2 Features

2.2.1 Twitter Database Analysis:

- Users can view and manage twitter appended data and Cyber threat reports.
- Ability to perform data handling operation and data preprocessing operations.

2.2.2 Cyber Threat CVSS Prediction:

- Analyse the twitter database cyber attack reports.
- Perform entity recognition operation on reports.

- Perform CVSS score prediction based on report analyzed data.

2.2.3 Anomaly detection:

- Use the CVSS score to detect severity of each threat.
- Categorize each vulnerability and threat based on severity.
- Anomaly detection based on severity categorization.

2.2.4 Cyber threat profiling and alert generation:

- Start using the data categorized to start training the deep cluster model.
- Gives you a trained model with all threat clustering using NLP.
- Obtain output profiled graphs along with patch time for each vulnerability.

2.3 User Classes and Characteristics:

- Cyber Security Team: Primary users who will utilize the ML model for cyber threat profiling and alert generation with patch time
- Administrators: adds new emerging threat reports to the training model for profiling.

3. Specific Requirements

3.1 Functional Requirements:

3.1.1 Twitter Database Analysis:

- Users should be able to add, edit, and delete twitter database cyber reports.
- Set data to null variables in report data.
- Should be able to perform data handling and data preprocessing.

3.1.2 Cyber Threat CVSS Prediction:

- Users can get organizing data from twitter database reports and tweets.
- Get entity names for each threat and its report.
- Get predicted CVSS score for each report of cyber threat.

3.1.3 Anomaly detection:

- Use CVSS score to evaluate each cyber threat.
- Predict severity threat levels and impact analysis.
- Detect anomaly in threat behaviour patters.

3.1.4 Cyber threat profiling and alert generation:

- Using threat behaviour and report data train the deep clustering model.
- Generate profiling clusters based on predictions from cluster model.
- Set graph and cluster maps along with patch time of each cyber threat.

3.2 Non-Functional Requirements:

3.2.1 Usability

- The Model must be usable very easily.
- It should be easily deploy-able in any environment.

3.2.2 Performance

- Quick response time for loading data and prediction.
- Scalability to accommodate increasing threat database.

3.2.3 Security

- Implement secure user authentication and authorization.
- Should work efficiently in isolation zones as well.

4. External Interface Requirements

4.1 Data appending Interfaces:

- Should be able to append and convert data types.
- Normalization of data format and null value prediction must be efficient.

4.2 Profiling Interfaces:

- Should be able to provide profiling clusters more clear and understandable.

4.3 Patching time graph Interfaces:

- Classification clusters must be easily understandable.
- Patch graphs must be detailed with better understandably and analysis report.

5. Conclusion

In general, the automated system is a huge step forward in cybersecurity capabilities when it comes to detecting and categorizing new cyber threats. The technology enables companies to proactively detect and react to emerging risks by using natural language processing methods and social media data. This strengthens their resilience and protects them from possible cyber assaults in our interconnected world. This SRS document serves as a guideline for the development team to create a robust and user-friendly Cyber Threat profiling and alert system.

SCRUM METHODOLOGY

EXP.NO: 2 **DATE: 14/03/2024**

1. Introduction

The deep clustering topic model: an approach to cyber anomaly detection and profiling introduces a system that uses natural language processing (NLP) methods applied to Twitter data to automatically detect and profile new cyber risks using Twitter as a resource for cybersecurity knowledge is becoming more important as more and more people use social media to talk about cybersecurity.

2. Objectives

- Develop a deep clustering topic model that can effectively identify anomalous cyber activities.
- Analyze the effectiveness of the model in profiling normal vs. anomalous network behaviors based on learned topics
- Ensure seamless collaboration between Cyber Security Team and Cyber Threat Profiling system .

3. Product Backlog Introduction

The product backlog is a dynamic list of features, enhancements, and fixes prioritizedby the product owner. It serves as a roadmap for the development team.

4. Product Backlog

4.1 For Administrators

- **Twitter Database Analysis:**
  - Data appending operation from twitter Database.
  - Data Handling and Data Preprocessing.

- **Cyber Threat CVSS Prediction:**
  - Entity Recognition System and tokenization.
  - Data analysis for each threat for CVSS score prediction .

- **Anomaly detection:**
  - Severity analysis and anomaly detection.

- **Cyber threat profiling and alert generation:**
  - Custer map, Patch time graph and alert features.

4.2 For Cyber Security Team

- **Cyber threat profiling and alert generation:**
  - Create and manage user accounts.
  - View and analyse profiled alerts along with patch time.

5. User Stories

- **As an Administrator:**
  - I want to perform data appending, handling and preprocessing.
  - I want to get CVSS, Severity and grouping matrix.
  - I want a train and deploy NLP model in any environment.

- **As a Cyber Security Team:**
  - I want to be able to manage user accounts and permissions.
  - I want to customize system to get custom patch graphs and profiling reports of new threats.

6. Sprint

- A time-boxed iteration during which a set of user stories is implemented and tested.

7. Sprint Backlog

The sprint backlog is a list of tasks selected from the product backlog for a specific sprint. In other terms Sprint Backlog could also be defined as the subset of Product backlog whichis chosen for a specific sprint. In general a sprint backlog allows the development team to work on the tasks necessary to implement the User Stories with in the selected sprint.

8. Sprint Review

A meeting held at the end of each sprint to review and demonstrate the completed work.

Sprint 1 Review:

- The sprint review is a meeting that includes the demonstration of implemented features at that particular sprint that is conducted at the end of each sprint.
- In the deep clustering topic model: an approach to cyber anomaly detection and profiling the sprint backlog for the first week is the Use case Diagram and the sprint backlog for the second use case is Software Requirement Specification document.

9. Software Used

- **Development Platform:** Jupiter IDE, Python R Studio

10. Conclusion

The Agile Scrum framework for the deep clustering topic model: an approach to cyber anomaly detection and profiling ensures a systematic approach to development, focusing on user needs and iterative improvements. By breaking down the project into manageable sprints, the framework allows for continuous feedback, resulting in a robust and user-friendly NLP Model for Cyber Security Team and administrators alike.

USER STORIES

EXP.NO: 3 **DATE: 26/03/2024**

1.1 As an Administrator:

1. Twitter Database Analysis

User Story: I want to perform data appending, handling and preprocessing.

Acceptance Criteria:

- The Administrator can perform data appending.
- The Administrator can handle null data prediction.
- The Administrator can perform data preprocessing operations.

2. Cyber Threat CVSS Prediction

User Story: I want to get CVSS, Severity and grouping matrix.

Acceptance Criteria:

- The Administrator can get CVSS scores of threats.
- The Administrator can identify Severity .
- The Administrator can get grouping matrices of data.

3. Anomaly Detection

User Story: I want to perform Severity analysis and anomaly detection.

Acceptance Criteria:

- The Administrator can perform Severity analysis .
- The Administrator can perform anomaly detection .

4. Cyber threat profiling and alert generation

User Story: I want to a train and deploy NLP model in any environment.

Acceptance Criteria:

- The Administrator can train and deploy NLP model in any environment.
- Get Profiled data at the set times.

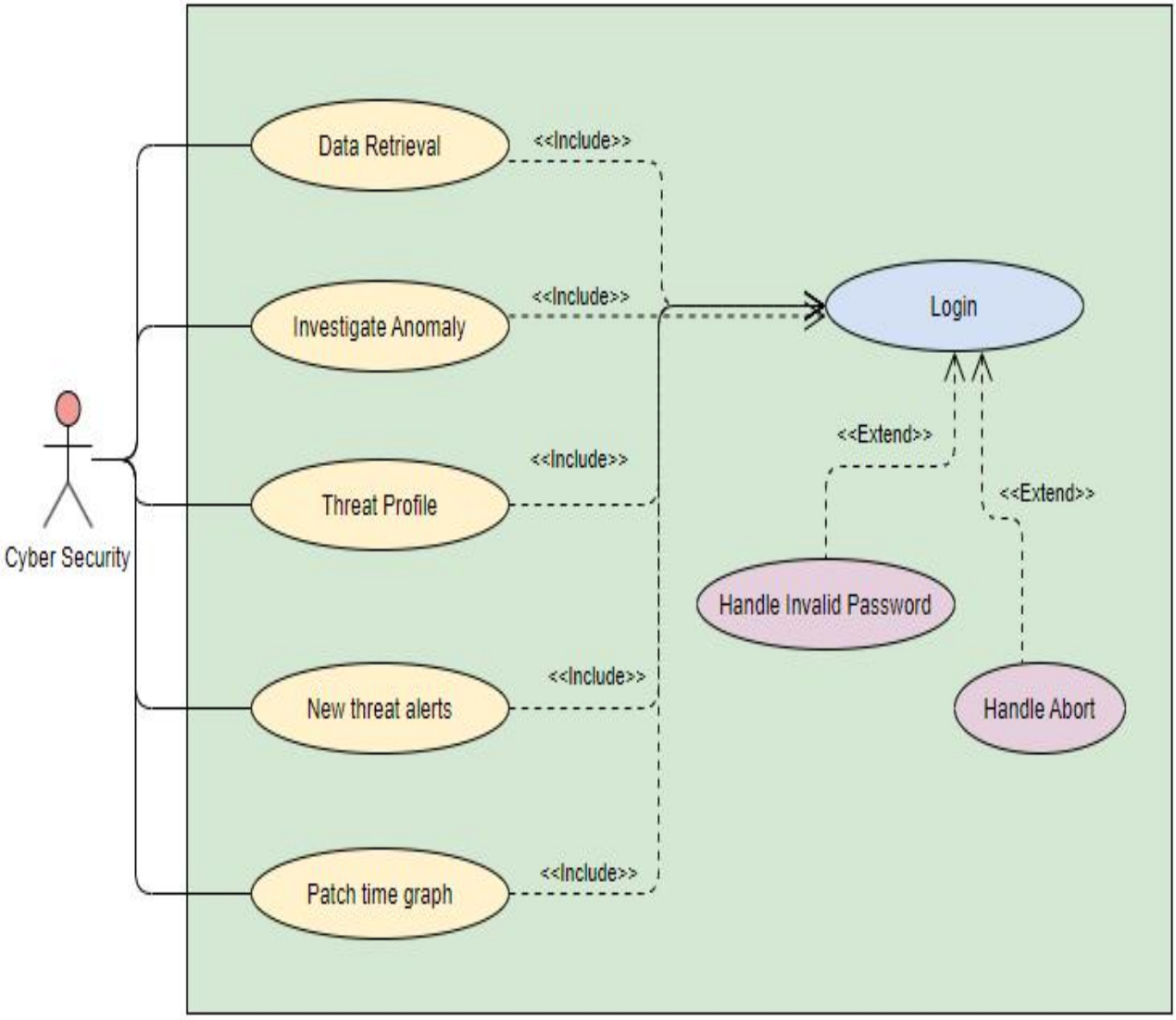
As a Cyber Security Team:

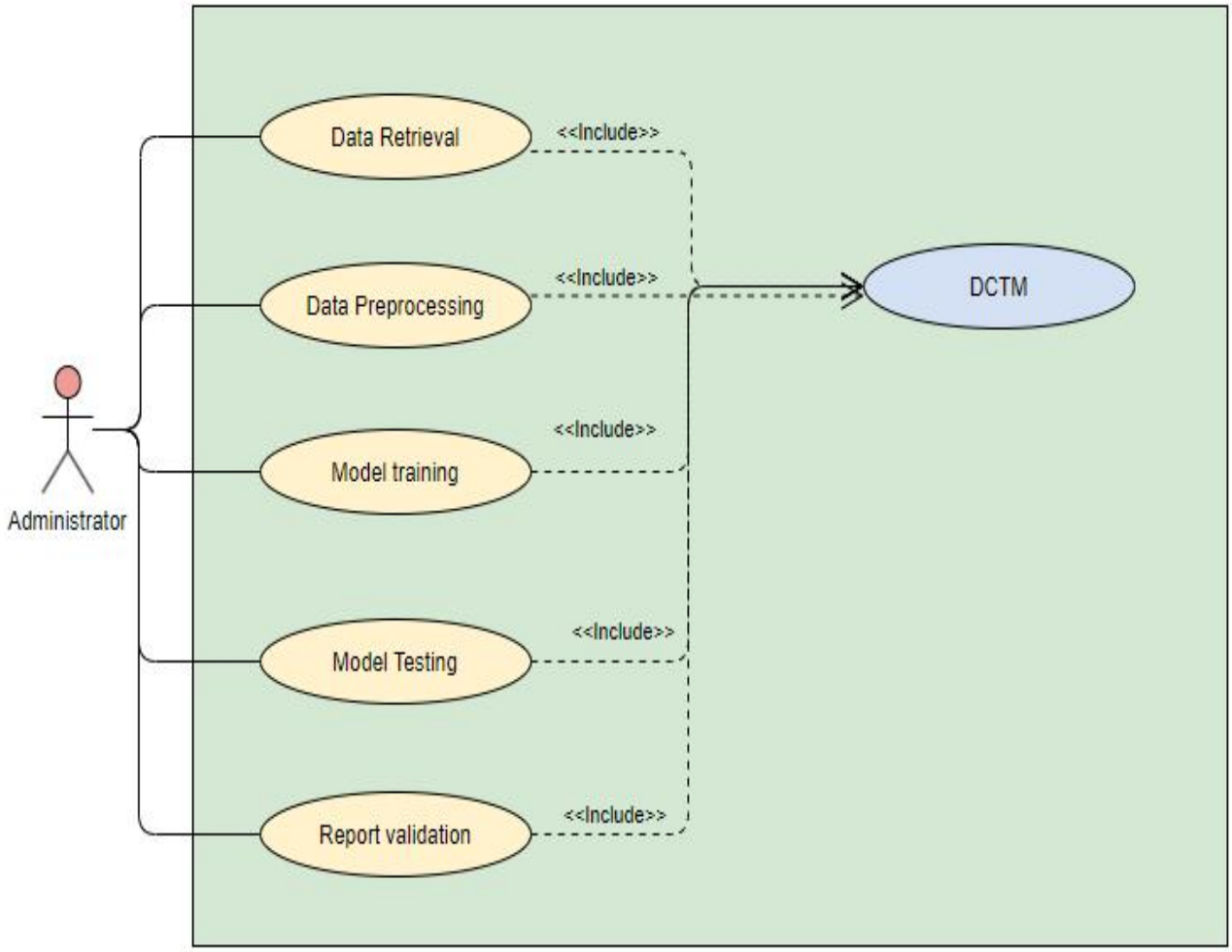
5. Profiled Threats
User Story: I want to be able to view profiled Threats.
Acceptance Criteria:
- The Cyber Security Team can view various threat scales.
- Get detailed profiles of any available threat reported in twitter.

6. Patch Time and Analysis Map
User Story: I want to customize system get cluster maps, patch time and analysis graphs
Acceptance Criteria:
- The Cyber Security Team can update system to get patch time maps for vulnerabilities.
- Get alerts and profile data on latest report of cyber threats.

USE CASE DIAGRAM

EXP.NO: 4 **DATE: 04/04/2024**




Data Retrieval
<<Include>>
Data Preprocessing
<<Include>>
DCTM
Model training
<<Include>>
Administrator
Model Testing
<<Include>>
Report validation
<<Include>>

NON-FUNCTIONAL REQUIREMENTS

EXP.NO: 5 **DATE: 16/04/2024**

1. Performance

The DCTM should achieve high accuracy in detecting anomalies within a defined threshold of false positives and negatives.The model should operate with minimal latency to enable real-time or near real-time anomaly detection.The DCTP should be scalable to handle large volumes of network traffic data efficiently, potentially across distributed computing environments.The model training time should be reasonable, considering the size and complexity of the network data.

2. Security

The app must implement robust security measures to protect user data. All user information, including login credentials, personal details, and stored notes, must be encrypted both in transit and at rest. Access to the app should be controlled via secure authentication mechanisms, and administrators should have the ability to set and enforce password policies. The app should also include role-based access controls to ensure thatusers only have access to the features and data appropriate to their role.

3. Usability

The system should provide a user-friendly interface for monitoring system health, visualizing learned topics, and exploring anomaly detections.The system should allow for configuration of various parameters (e.g., anomaly thresholds, training settings) to adapt to specific network environments.The system should provide some level of explanation for the model's decisions, allowing security analysts to understand the rationale behind anomaly detections.

4. Reliability

The DCTM system should be highly available with minimal downtime to ensure continuous anomaly detection capabilities. The system should be designed to gracefully handle potential failures (e.g., hardware, software) with minimal disruption to its operation.

5. Maintainability

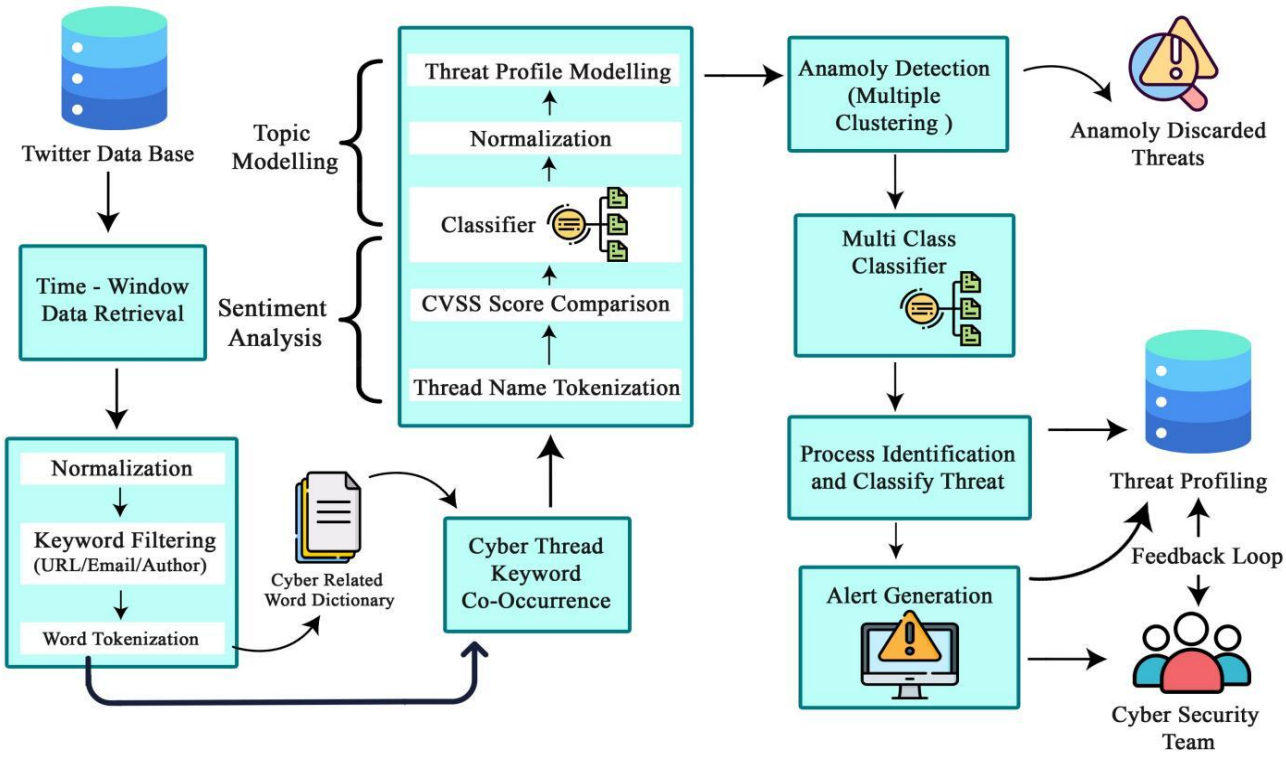
The system should be designed with modular components to facilitate easier maintenance, debugging, and future enhancements.Comprehensive documentation should be provided to explain the system's architecture, functionality, configuration options, and troubleshooting procedures.The system should implement robust logging and monitoring mechanisms to track system activity, identify potential issues, and facilitate performance optimization.

6. Compliance

The system must comply with relevant regulations and standards, including data privacy laws such as GDPR for users in Europe. The system should ensure that all data handling practices meet legal requirements for data protection and user privacy. Regular audits should be conducted to verify compliance, and any necessary adjustments shouldbe made promptly to address new regulatory changes.

OVERALL PROJECT ARCHITECTURE

EXP.NO: 6 **DATE: 25/04/2024**



STAGES :

STAGE 1: Data Collection and Data Preprocessing

- **Data Retrieval:** This layer is the first step where data is retrieved from a Twitter database. The data is retrieved for a specific period.This layer interacts with the data storage system (database) to retrieve, store, and manage data.
- **Data Preprocessing:** This section covers various techniques to prepare the raw data for further analysis.

STAGE 2: Feature Engineering

- **CVSS Score Comparison:** This layer is to comparing the Common Vulnerability Scoring System (CVSS) scores, which is a standard for assessing the severity of software vulnerabilities.
- **Sentiment Analysis:** This process analyzes the sentiment of the text data (positive, negative, or neutral).

STAGE 3: Topic Modeling and Clustering

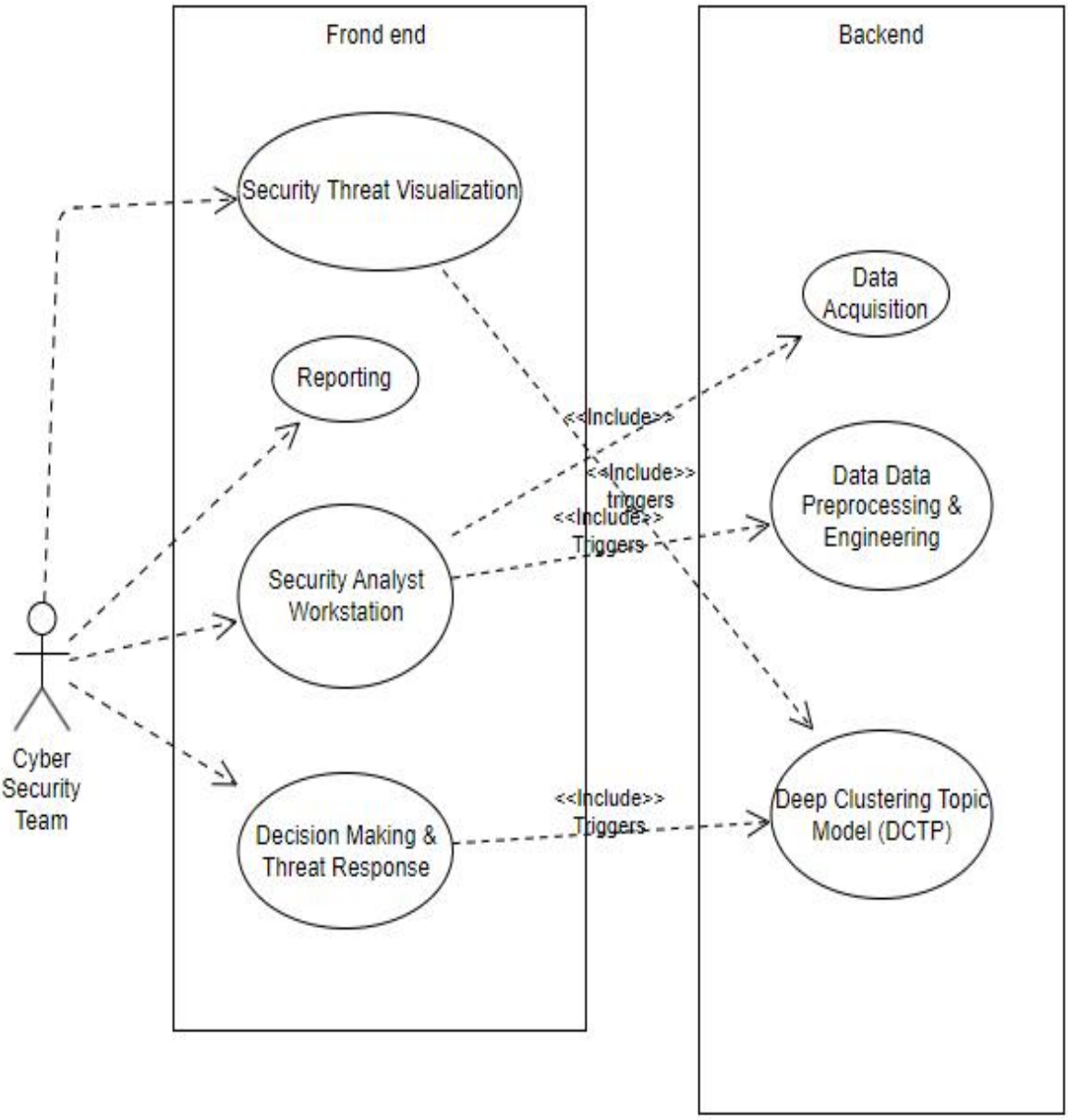
- **Topic Modelling:** This section refers to the process of automatically discovering hidden thematic structures within the data.These processes are likely applied to transform the data into a format suitable for topic modelling.
- **Clustering (Multiple):** This refers to grouping data points into multiple clusters based on their similarities. Here, it likely refers to using an anomaly detection algorithm to identify anomalous data points from multiple clusters.

STAGE 4: Anomaly Detection and Threat Profiling and alerts

- **Anomaly Detection:** This refers to the process of identifying data points that deviate significantly from the expected patterns some anomalies might be discarded based on certain criteria.
- **Threat Profiling:** This step likely involves using the learned topics to profile different types of cyber threats.This indicates that the system continuously learns and improves over time.
- **Alert Generation:** In this step frequent alerts are generated and sent to the cyber security team

BUSINESS ARCHITECTURE DIAGRAM

EXP.NO: 7 **DATE: 02/05/2024**



Actor:

- Cyber Security Team: Represents the primary user who interacts with the DCTM.This actor is involved in all frontend use cases.

Frontend Use Cases:

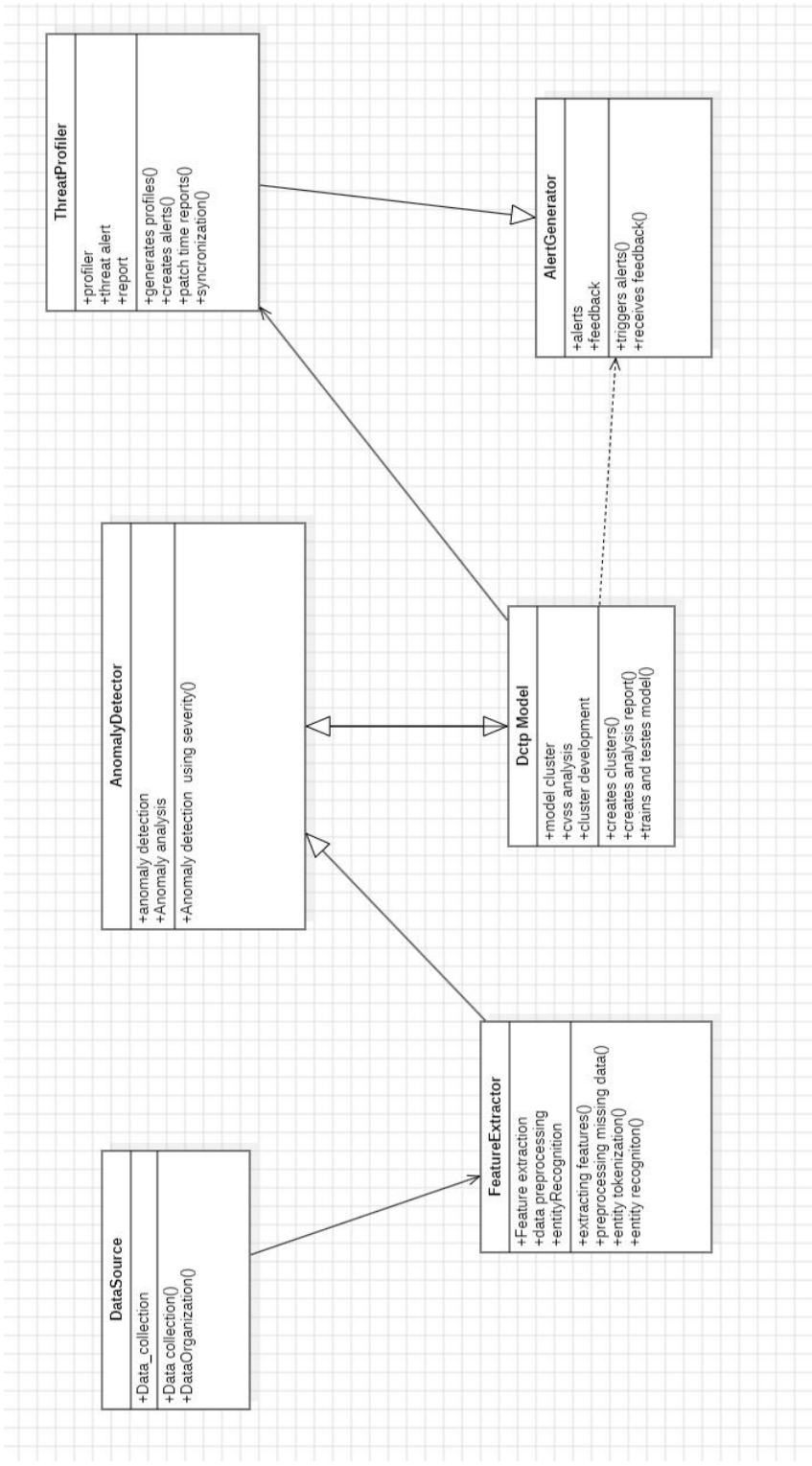
- **Security Threat Visualization**: his component provides security analysts with visualizations and reports to understand network activity, identified anomalies, and potential threats.
- **Reporting**: Creates reports to be view by the team
- **Analyst Workstation**: This represents the environment where security analysts interact with the system to investigate anomalies, manage alerts, and make informed decisions.
- **Decision Making & Threat Response:** leverage the insights from the DCTP system to make informed decisions regarding potential threats. This may involve further investigation, mitigation actions (e.g., isolating compromised systems), or reporting incidents.

Backend Use Cases:

- **Data Acquisition**: This component is responsible for collecting network traffic data from various sources within the organization's network.
- **Data Data Preprocessing & Engineering**: This component prepares the raw network traffic data for analysis by the DCTP model.
- **Deep Clustering Topic Model (DCTM):** This is the core component that utilizes deep learning techniques to identify hidden topics (patterns) within the network traffic data and performs anomaly detection..

CLASS DIAGRAM

EXP.NO: 8 **DATE: 07/05/2024**



Classes:

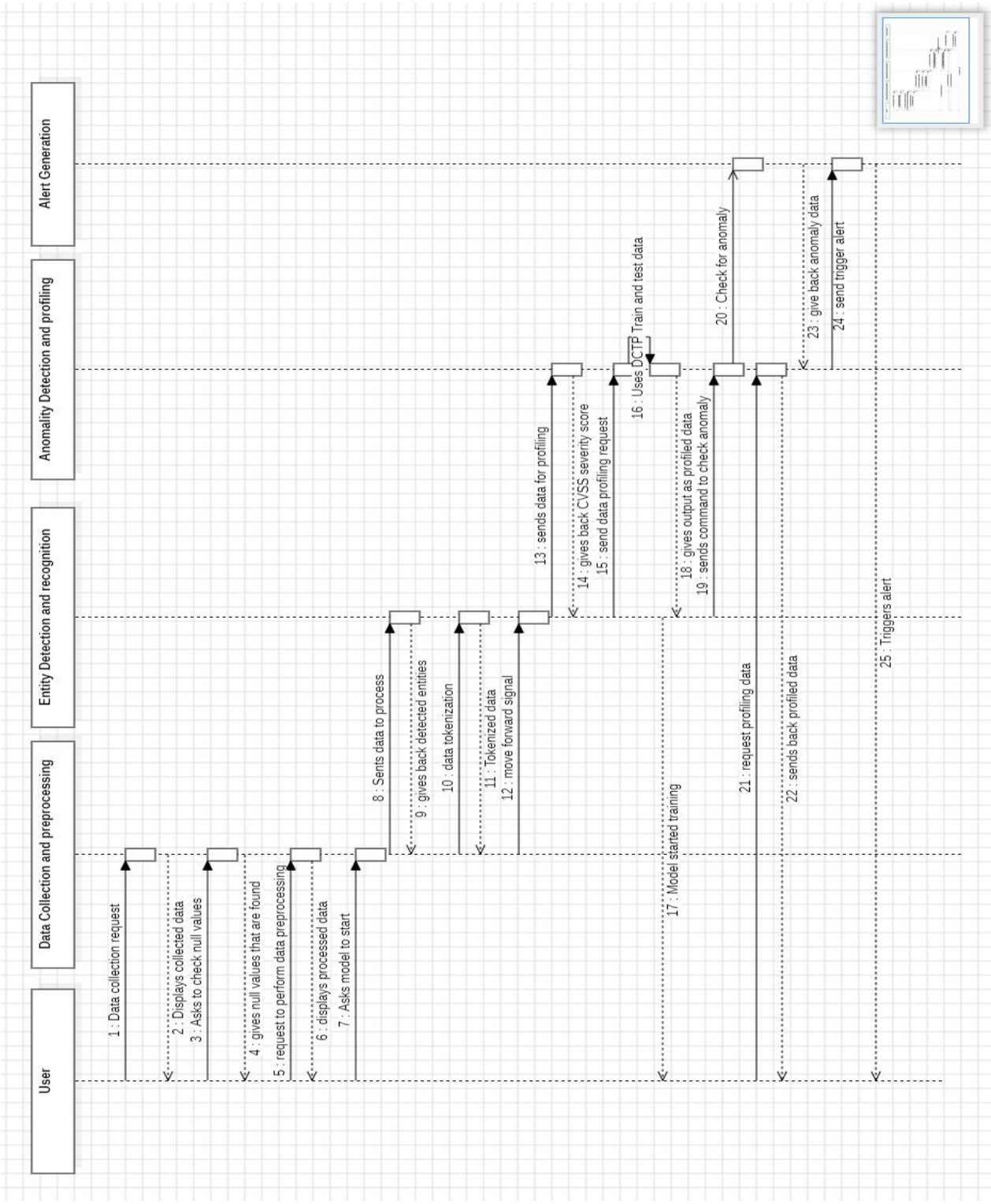
1. **DataSource:** Manages access and retrieval of network traffic data.
2. **FeatureExtractor:** Inherits from DataSource and extracts relevant features from the data.
3. **DctpModel:** The core deep learning model responsible for topic discovery and anomaly detection.
4. **AnomalyDetector:** Utilizes the DctpModel to analyze data and generate anomaly reports.
5. **ThreatProfiler:** Analyzes anomaly reports and generates threat profiles.
6. **AlertGenerator:** Generates alerts for security analysts based on threat profiles.

Relationships:

Network traffic data flows from the Data Acquisition component to the Data Preprocessing & Engineering component.Preprocessed data is fed into the DCTP model for analysis. The DCTP model outputs anomaly scores and potential threat classifications.
Threat Profiling & Alert Generation analyzes these outputs and generates alerts for security analysts. Security analysts interact with the Security Threat Visualization & Reporting component to investigate anomalies and make decisions.

SEQUENCE DIAGRAM

EXP.NO: 9 **DATE: 16/05/2024**



Actor:

Cyber Security Team: The user interacting with the model.

System Components:

- **Data collection and preprocessing:** Performs data collection and handling operations.
- **Entity detection and Recognition :** A system that performs entity detection and recognition.
- **Anomaly Detection and Profiling:** The component responsible for using DCTM approach model to detect anomaly and profile threats

  Alert generator: The component responsible triggering alerts

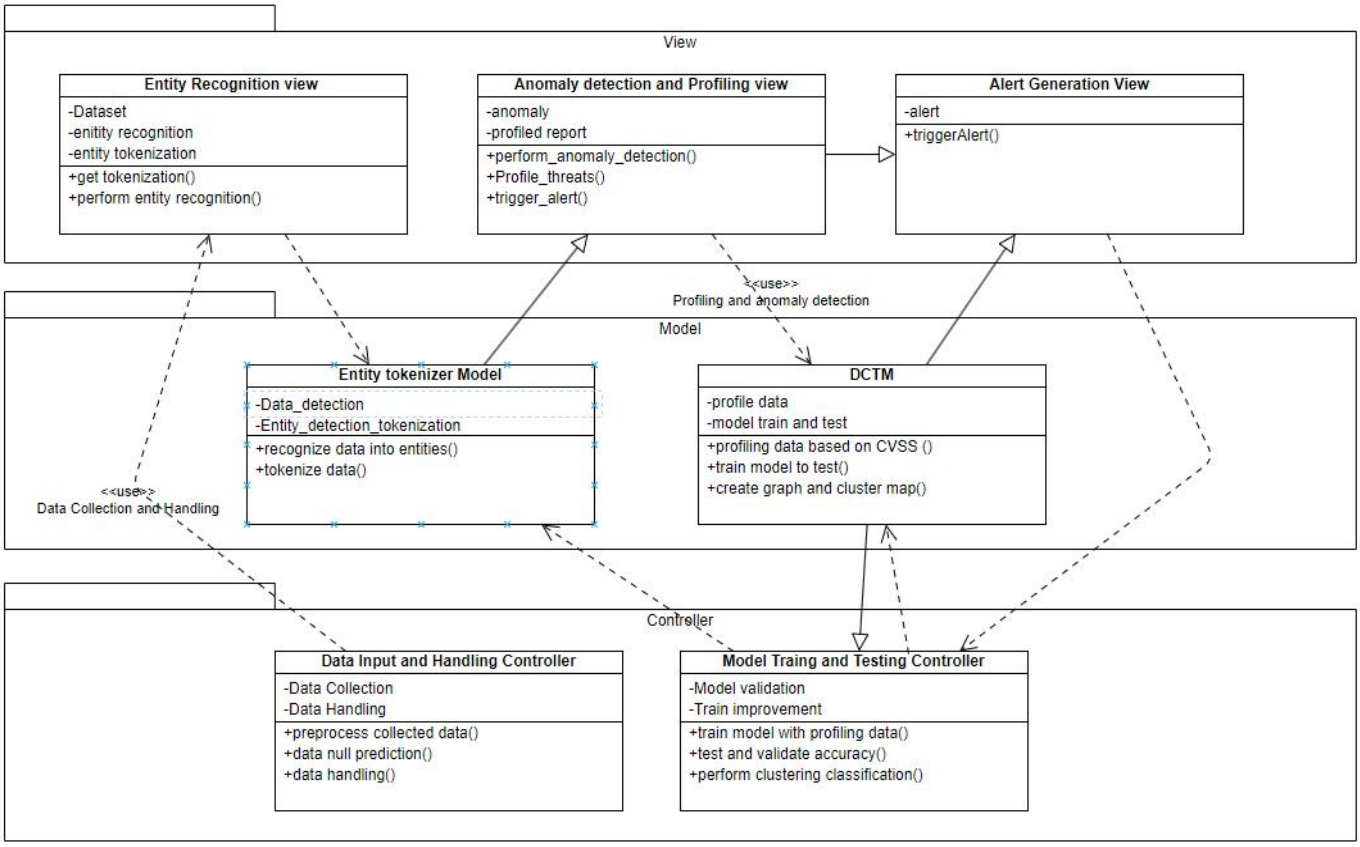
Sequence of Events:

- The user initiates the sequence by requesting a website creation.
- It then displays the collected data.
- The user asks the Data Collection and Preprocessing object to check for null values in the data.
- The Data Collection and Preprocessing object finds the null values and sends them back to the user.
- The user makes a request to perform data preprocessing.
- The Data Collection and Preprocessing object preprocesses the data and displays the processed data.
- The user then asks the model to start.
- Potentially, the Data Collection and Preprocessing object sends the preprocessed data to the Entity Detection and Recognition object.
- Entity Detection and Recognition recognizes entities and sends them back.
- The Data Collection and Preprocessing object performs data tokenization and sends the tokenized data.
- After processing, a signal is sent to move forward.
- The tokenized data is then likely sent to the Anomaly Detection and Profiling object.
- Anomaly Detection and Profiling provides a CVSS severity score, indicating the Common Vulnerability Scoring System (a standard for rating software vulnerabilities).

- Anomaly Detection and Profiling also sends a request for data profiling
- It potentially uses training and test data to build a model.The model is then started for training.
- After training, a command is sent to check for anomalies .Anomaly Detection and Profiling requests the profiled data.
- The Anomaly Detection and Profiling object receives the profiled data and sends back the anomaly data.
- Finally, an alert is triggered based on the anomaly data , and the user is likely notified.

ARCHITECTURAL PATTERN (MVC)

EXP.NO: 10 **DATE: 28/05/2024**



Model:

- **Entity Tokenizer Model:** This likely represents the data detection and tokenization of the model report and it has methods for recoginze data into entity and tokenize data
- **DCTM Model:** This likely manages all Natural Language Processing ML model operation for clustering data based on profiling data and it has methods for profiling data based on cvss , train model to test and create graph and cluster map

View:

- **Entity Recognition view:** This component displays the entity recognition data along with token representation. It has methods for get tokenization, perform entity recognition.
- **Anomaly Detection and Profiling View:** This component displays information related to anomaly detection and profiling threats. It has methods like perform_anomaly_detection ,Profile_threats, trigger_alert**.**
- **Alert Generator:** This component displays alerts. It has methods for trigger_alerts

Controller:

- **Data Input and Handling Controller:** This component handles user interactions related to the data handling . It has methods for processing collected data, data null prediction

Model Training and Testing Controller: This component handles user interactions related to model training and testing. Ithas methods for train model with profiling data, test and validation accuracy and perform clustering

Relationships:

The Model, View, and Controller components interact as follows:

- The Model provides data and logic to the Controller.
- The Controller receives user input from the View and manipulates the Modelaccordingly (e.g., adding a new event).
- The Controller instructs the View to update itself based on changes in the Model(e.g., displaying a newly added event).
- The View interacts directly with the Model. It communicates with the Model through the Controller.