Suggested Reference:

1. COCOMO
2. Halstead complexity measures
3. COCOMO (Constructive Cost Model), Seminar on Software Cost Estimation WS 2002 / 2003, presented by Nancy Merlo – Schett
4. The Halstead metrics
5. Software Engineering, National Program on Technology Enhanced Learning VI. Halstead Metrics, Verifysoft Technology

Rubric wise marks obtained:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | 4 | 5 | Total |
| Marks | Complete implementation as asked | Complete implementation as asked  Problem analysis | Complete implementation as asked  Problem analysis  Development of the Solution | Complete implementation as asked  Problem analysis  Development of the Solution  Concept Clarity  & understanding | Complete implementation as asked  Problem analysis  Development of the Solution  Concept Clarity  & understanding  Correct answer to all questions |  |

Signature of Faculty:

## **Practical – 3**

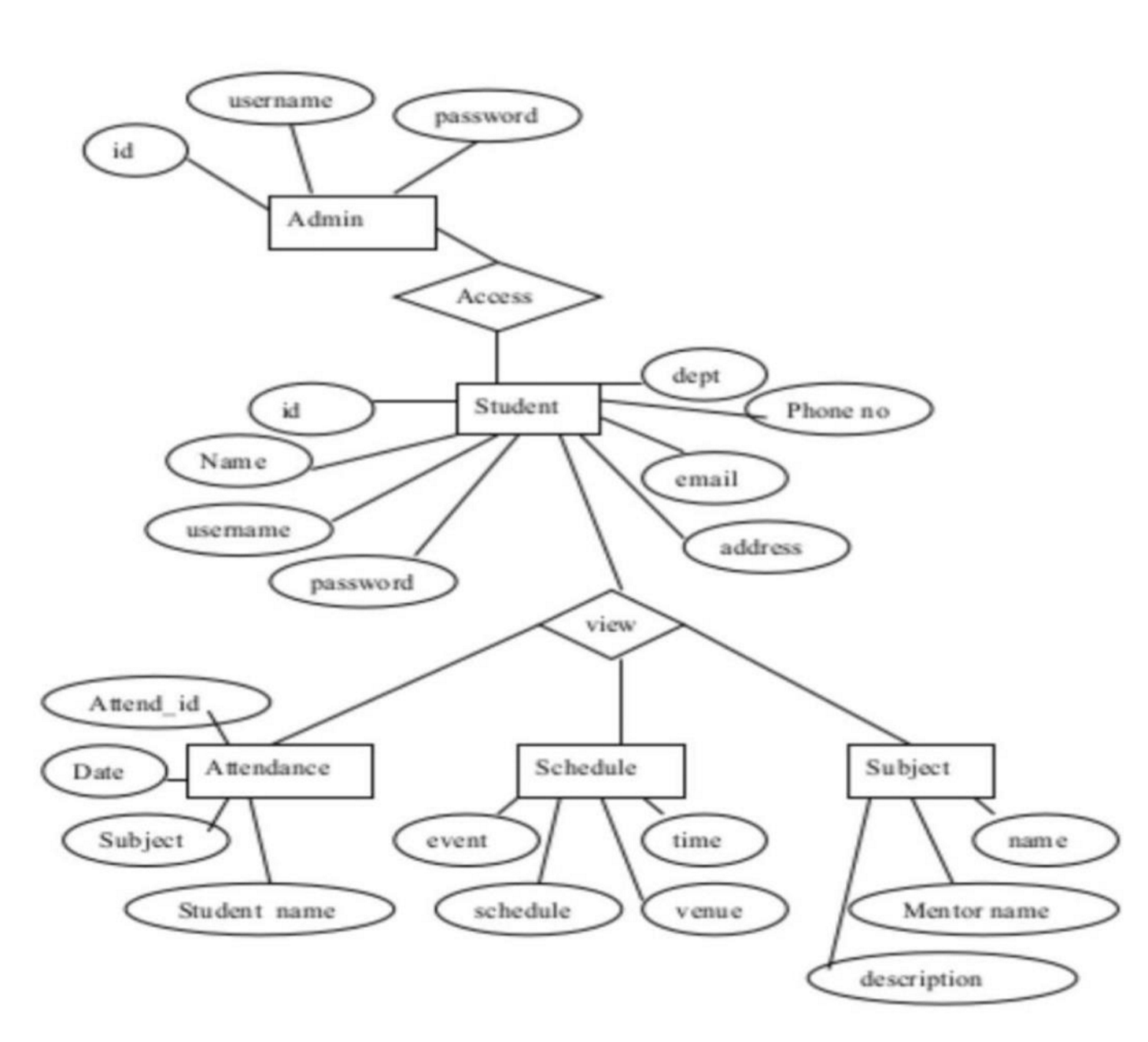
**AIM**: Prepare the following components of Data Flow Model: Data Dictionary, Data Flow Diagram and Structure Chart Objectives:

1. To learn how to prepare data dictionary.
2. To learn how to draw data flow diagrams.
3. To convert a DFD into structure chart.

**Theory**:

* + A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It can be manual, automated, or a combination of both.
  + It shows how data enters and leaves the system, what changes the information, and where data is stored.
  + The objective of a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communication tool between a system analyst and any person who plays a part in the order that acts as a starting point for redesigning a system. The DFD is also called as a data flow graph or bubble chart.
  + The following observations about DFDs are essential:
    - All names should be unique. This makes it easier to refer to elements in the DFD.
    - Remember that DFD is not a flow chart. Arrows is a flow chart that represents the order of events; arrows in DFD represents flowing data. A DFD does not involve any order of events.
    - Suppress logical decisions. If we ever have the urge to draw a diamond-shaped box in a DFD, suppress that urge! A diamond-shaped box is used in flow charts to represents decision points with multiple exists paths of which the only one is taken. This implies an ordering of events, which makes no sense in a DFD.
    - Do not become bogged down with details. Defer error conditions and error handling until the end of the analysis.

**(ii)DFD Level - 1:**



Quiz:

1. Draw the Data Flow Diagram for Hotel Management System.
2. What is DFD? Give advantage and disadvantages of DFD.
3. Types and Components of Data Flow Diagram (DFD)

Suggested Reference:

1. NPTEL course material - System Analysis and Design
2. Booch, G. et al. The Unified Modeling Language User Guide. Chapters 15, 18, 27. AddisonWesley.
3. Jacobson, I. et al. Object-Oriented Software Engineering: A Use-Case Driven Approach. Addison-Wesley.
4. Fowler, M. UML Distilled: A Brief Guide to the Standard Object Modelling Language. Chapter

5. Addison Wesley.

Rubric wise marks obtained:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | 4 | 5 | Total |
| Marks | Complete implementation as asked | Complete implementation as asked  Problem analysis | Complete implementation as asked  Problem analysis  Development of the Solution | Complete implementation as asked  Problem analysis  Development of the Solution  Concept Clarity  & understanding | Complete implementation as asked  Problem analysis  Development of the Solution  Concept Clarity  & understanding  Correct answer to all questions |  |

Signature of Faculty:

Suggested Reference:

1. Use Case Diagrams.
2. Unified Modeling Language, Superstructure, V2.1.2.
3. "Functional Requirements and Use Cases", Ruth Malan and Dana Bredemeyer, Bredemeyer Consulting.
4. "A Use Case Template: draft for discussion", Derek Coleman, Hewlett-Packard Software Initiative X. J. Zheng, X. Liu, & S. Liu. (2010). Use Case and Non-functional Scenario

Template-Based Approach to Identify Aspects. Computer Engineering and Applications ICCEA 2010 Second International Conference on (Vol. 2, pp. 89-93).

Rubric wise marks obtained:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | 4 | 5 | Total |
| Marks | Complete implementation as asked | Complete implementation as asked  Problem analysis | Complete implementation as asked  Problem analysis  Development of the  Solution | Complete implementation as asked  Problem analysis  Development of the  Solution  Concept Clarity & understanding | Complete implementation as asked  Problem analysis  Development of the  Solution  Concept Clarity & understanding  Correct answer to all questions |  |

Signature of Faculty:

## **Practical – 5**

AIM: Prepare the structural view : Draw Class diagram and object diagram.

* Objectives: to learn about UML class diagram and object diagram components.

* Theory:

o Class diagram:

* + - * The class diagram depicts a static view of an application. It represents the types of objects residing in the system and the relationships between them. A class consists of its objects, and also it may inherit from other classes. A class diagram is used to visualize, describe, document various different aspects of the system, and also construct executable software code.
      * It shows the attributes, classes, functions, and relationships to give an overview of the software system. It constitutes class names, attributes, and functions in a separate compartment that helps in software development. Since it is a collection of classes, interfaces, associations, collaborations, and constraints, it is termed as a structural diagram.

o Purpose of Class Diagrams

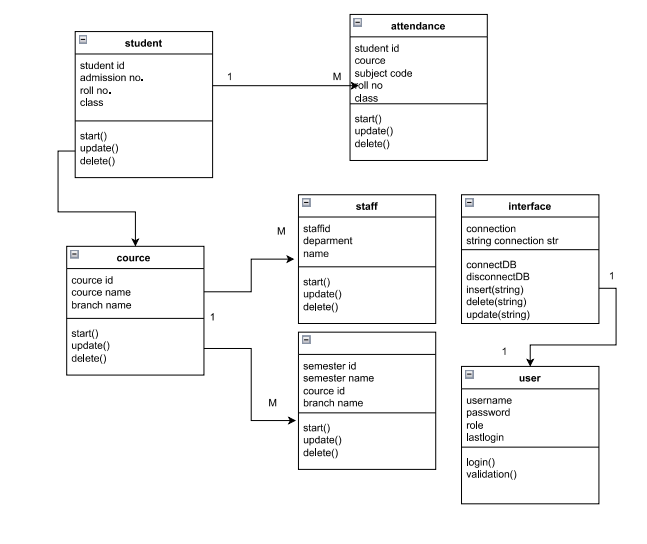
The main purpose of class diagrams is to build a static view of an application. It is the only diagram that is widely used for construction, and it can be mapped with objectoriented languages. It is one of the most popular UML diagrams. Following are the purpose of class diagrams given below:

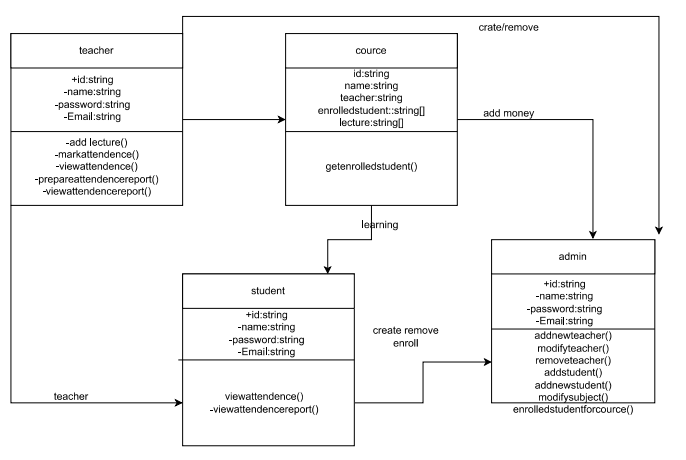
* + - 1. It analyses and designs a static view of an application.
      2. It describes the major responsibilities of a system.
      3. It is a base for component and deployment diagrams.
      4. It incorporates forward and reverse engineering.

o Benefits of Class Diagrams

* + - 1. It can represent the object model for complex systems.
      2. It reduces the maintenance time by providing an overview of how an application is structured before coding.
      3. It provides a general schematic of an application for better understanding.
      4. It represents a detailed chart by highlighting the desired code, which is to be programmed.
      5. It is helpful for the stakeholders and the developers.

Vital components of a Class Diagram

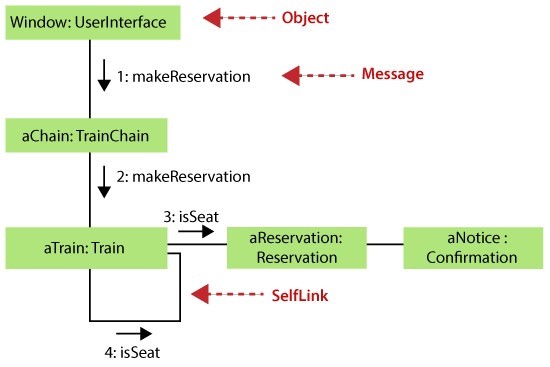
**Class diagram :**

**Object diagram:**

Rubric wise marks obtained:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | 4 | 5 | Total |
| Marks | Complete implementation as asked | Complete implementation as asked  Problem analysis | Complete implementation as asked  Problem analysis  Development of the Solution | Complete implementation as asked  Problem analysis  Development of the Solution  Concept Clarity  & understanding | Complete implementation as asked  Problem analysis  Development of the Solution  Concept Clarity  & understanding  Correct answer to all questions |  |

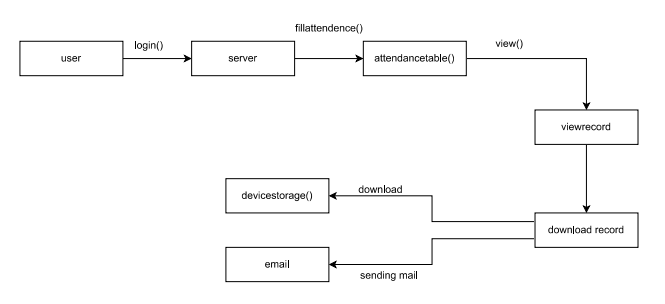
Signature of Faculty:



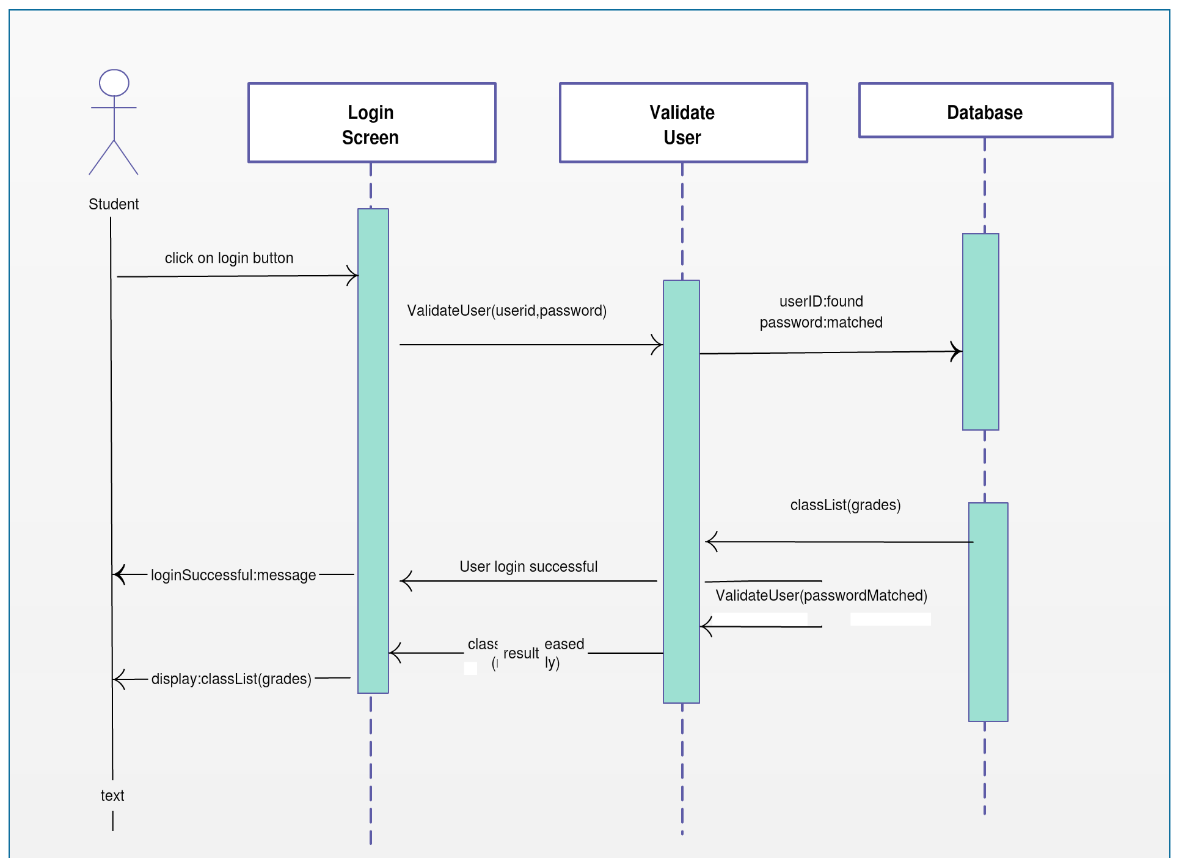
Suggested Reference:

1. Booch, G. et al. The Unified Modeling Language User Guide. Chapters 15, 18, 27. AddisonWesley.
2. Jacobson, I. et al. Object-Oriented Software Engineering: A Use-Case Driven Approach. Addison-Wesley.
3. Fowler, M. UML Distilled: A Brief Guide to the Standard Object Modelling Language. Chapter

5. Addison Wesley.



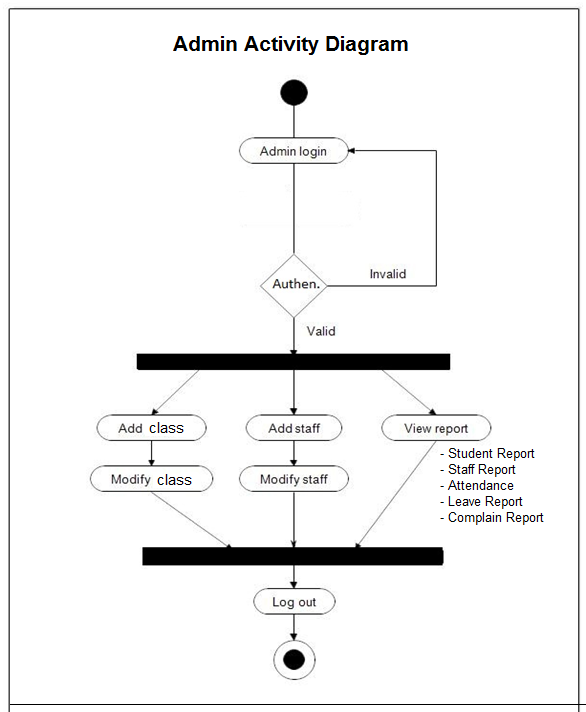
(Fig. Collaboration diagram )

Sequence-diagram:

Rubric wise marks obtained:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | 4 | 5 | Total |
| Marks | Complete implementation as asked | Complete implementation as asked  Problem analysis | Complete implementation as asked  Problem analysis  Development of the Solution | Complete implementation as asked  Problem analysis  Development of the Solution  Concept Clarity  & understanding | Complete implementation as asked  Problem analysis  Development of the Solution  Concept Clarity  & understanding  Correct answer to all questions |  |

Signature of Faculty:

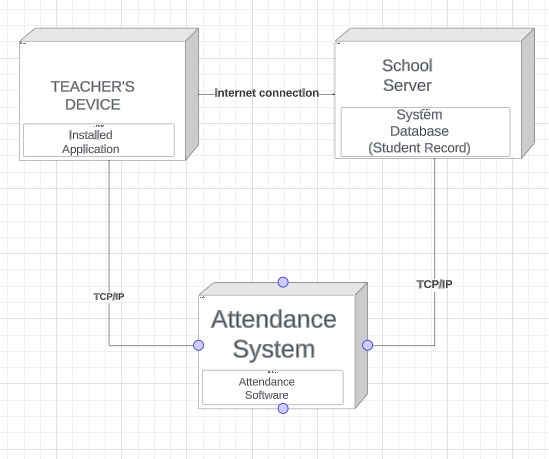
**Activity diagram**:

Rubric wise marks obtained:

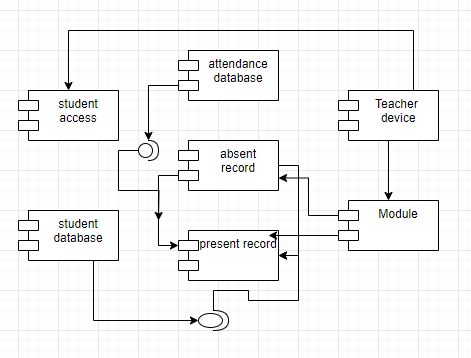
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | 4 | 5 | Total |
| Marks | Complete implementation as asked | Complete implementation as asked  Problem analysis | Complete implementation as asked  Problem analysis  Development of the  Solution | Complete implementation as asked  Problem analysis  Development of the  Solution  Concept Clarity & understanding | Complete implementation as asked  Problem analysis  Development of the  Solution  Concept Clarity & understanding  Correct answer to all questions |  |

Signature of Faculty:

Deployment diagram:



Component diagram:



-Alerting Grafana can integrate with various monitoring system. include Prometheus.

1. **ELK Stack for Log Management**:

**Description**: The ELK Stack consists of three open-source tools: Elasticsearch, Logstash, and Kibana. It is used for log management, log analysis, and log visualization.



**Key Features:**

* + Elasticsearch: Indexes and stores log data for efficient searching and analysis.
  + Logstash: Collects, processes, and enriches log data before sending it to Elasticsearch.
  + Kibana: Provides visualization and dashboards for log analysis.

1. **Zabbix for Infrastructure Monitoring:**

**Description**: Zabbix is an open-source monitoring solution that can be used for network monitoring, server monitoring, and application monitoring.



**Key Features:**

* + Auto Discovery: Zabbix can automatically discover and monitor devices and services.
  + Custom Monitoring Items: It allows you to create custom monitoring items for specific metrics.
  + Alerting: Zabbix supports alerting and notification when predefined thresholds are breached.

Suggested Reference:

1. Deepak Gaikwad, Viral Thakkar, DevOps Tools from Practitioner’s ViewPoint, Wiley.
2. The DevOps Handbook - Gene Kim et. al.

Rubric wise marks obtained:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Rubrics | 1 | 2 | 3 | 4 | 5 | Total |
| Marks | Complete implementation as asked | Complete implementation as asked  Problem analysis | Complete implementation as asked  Problem analysis  Development of the  Solution | Complete implementation as asked  Problem analysis  Development of the  Solution  Concept Clarity & understanding | Complete implementation as asked  Problem analysis  Development of the  Solution  Concept Clarity & understanding  Correct answer to all questions |  |

Signature of Faculty: