Lab Assignment NO: 01

AIM: Consider a Problem statement for software development and Prepare SRS for the same.

OBJECTIVES:

- To able to define and decompose the project scope into manageable and understandable parts.
- To keep track of project plan completion and ensure your project stays on track..

THEORY:

A work breakdown structure is a tool that helps you organize your project by hierarchy. With a WBS, you break down deliverables into sub-deliverables to visualize projects and outline key dependencies. Every work breakdown structure is made up of a few parts:

- A project baseline or scope statement, which includes a project plan, description, and name
- Project stakeholders
- An organized project schedule
- Project deliverables and supporting subtasks

Project managers use work breakdown structures to help teams to break down complex project scopes, visualize projects and dependency-related deliverables, and give team members a visual project overview as opposed to a list of to-dos.

From there, you'll organize your structure based on the hierarchical levels of sub-deliverables. Your project might also include phases based on the work needed and the overall project timeline.

Work Breakdown Structure (WBS) for ATM Withdrawal Software Development

Below is a hierarchical WBS structure for developing software for an ATM withdrawal system. It is divided into key deliverables, phases, and activities.

1. Requirements Analysis

- 1.1 Gather User Requirements
- 1.2 Define Functional Requirements
- 1.3 Define Non-Functional Requirements
- 1.4 Prepare Software Requirement Specification (SRS)

2. System Design

- 2.1 High-Level Design (HLD)
  - 2.1.1 System Architecture
  - 2.1.2 Module Design
- 2.2 Low-Level Design (LLD)
  - 2.2.1 Data Flow Diagrams
  - 2.2.2 Database Schema
  - 2.2.3 User Interface Design
- 2.3 Security Design
  - 2.3.1 Authentication Mechanisms
  - 2.3.2 Encryption Techniques

3. Development

- 3.1 Backend Development
  - 3.1.1 Implement Core Withdrawal Logic
  - 3.1.2 Integrate Banking APIs
- 3.2 Frontend Development
  - 3.2.1 Develop User Interface
  - 3.2.2 Implement User Feedback Mechanisms
- 3.3 Database Development
  - 3.3.1 Configure Database
  - 3.3.2 Implement Transaction Logging

4. Testing

- 4.1 Unit Testing
- 4.2 Integration Testing
- 4.3 Security Testing
- 4.4 Performance Testing
- 4.5 User Acceptance Testing (UAT)

5. Deployment

- 5.1 Prepare Deployment Plan
- 5.2 Deploy to Production Environment
- 5.3 Configure ATM Hardware Integration

6. Maintenance

- 6.1 Monitor System Performance
- 6.2 Bug Fixes and Updates
- 6.3 Feature Enhancements

7. Documentation

- 7.1 Technical Documentation
- 7.2 User Manuals
- 7.3 Maintenance Guides

Resource Estimation

1. Personnel

- **Business Analyst**: Requirements gathering and SRS preparation.
- **System Architect**: Design system architecture and module interactions.
- **Backend Developer(s)**: Implement withdrawal logic and integrate APIs.
- **Frontend Developer(s)**: Design and implement the user interface.
- **Database Administrator**: Configure and manage database systems.
- **Tester(s)**: Conduct unit, integration, and system testing.
- **Deployment Engineer**: Manage deployment and integration with hardware.
- **Technical Writer**: Prepare documentation.

2. Tools and Technologies

- **Programming Languages**: Java, Python, or C#.
- **Database**: MySQL, PostgreSQL, or Oracle DB.
- **Version Control**: GitHub/GitLab.
- **Development Environment**: IntelliJ IDEA, Eclipse, or Visual Studio.
- **Testing Tools**: Selenium, JMeter, or manual testing frameworks.
- **Security Tools**: SSL/TLS encryption libraries, penetration testing tools.

3. Time Estimation

- **Requirements Analysis**: 2 weeks
- **System Design**: 2-3 weeks
- **Development**: 6-8 weeks
- **Testing**: 3-4 weeks
- **Deployment**: 1-2 weeks
- **Maintenance**: Ongoing

4. Cost Estimation

- **Personnel Costs**: Salary or hourly rates of team members.
- **Hardware Costs**: Test environments and ATM hardware.
- **Software Costs**: Licensing fees for tools and libraries.
- **Miscellaneous Costs**: Training, documentation printing, etc.