**1. Draw the Work Breakdown Structure for the system to be automated**

**OBJECTIVE:** To **Draw the Work Breakdown Structure for the system to be automated**

**Theory :**

Work Breakdown Structure (WBS) :

In software engineering, a Work Breakdown Structure (WBS) is a project management tool that breaks down a software development project into smaller, more manageable components. It provides a hierarchical representation of the project's scope, allowing software teams to systematically plan, execute, and monitor the work required to complete the project successfully.

The WBS in software engineering serves several purposes:

**1.Task Decomposition:** It decomposes the project into smaller, more manageable tasks and subtasks. This breakdown allows project managers and team members to better understand the work involved and facilitates more accurate planning.

**2. Scope Definition:**The WBS helps define and clarify the scope of the software project. It identifies all the deliverables and work packages that need to be completed to achieve the project's objectives.

**3. Resource Allocation:**The WBS helps in assigning resources to specific tasks. This includes allocating development teams, testing resources, and other necessary personnel to different components of the project.

**4. Project Planning:** With a detailed breakdown of tasks and dependencies, project managers can create realistic project schedules. The WBS provides a foundation for creating timelines, estimating effort, and sequencing activities.

**5. Communication and Collaboration:** The hierarchical structure of the WBS provides a common framework for communication among team members, stakeholders, and project managers. It ensures that everyone involved has a shared understanding of the project's structure and components.

**6. Risk Management:** By breaking down the project into smaller elements, the WBS helps in identifying potential risks and dependencies early in the project lifecycle. This facilitates proactive risk management and mitigation strategies.

**7. Progress Tracking:** The WBS serves as a basis for tracking project progress. It allows project managers to monitor the completion of work packages and milestones, helping to ensure that the project stays on schedule.

**8. Cost Estimation:** The WBS can be used as a foundation for estimating costs associated with each work package. This aids in budgeting and financial planning for the software development project.

The WBS can be tailored to suit the specific needs of a software project. Each level of the hierarchy in the WBS represents a different level of detail, with higher levels representing major project phases or deliverables and lower levels representing individual tasks or features. The WBS is often created collaboratively with input from project managers, software developers, quality assurance teams, and other stakeholders involved in the project.

**Representing a Work Breakdown Structure (WBS):**

Representing a Work Breakdown Structure (WBS) can be done in various formats and visualizations. The choice of representation depends on the complexity of the project, the preferences of the project manager. Here are some common ways of representing a WBS:

1. **Indented Outline:** This is a simple and text-based representation where tasks and subtasks are indented to show their hierarchical relationships. Each level of indentation represents a different level in the WBS.

Advantages: Easy to create, straightforward, and suitable for smaller projects.

1. **Tree Diagram:** A tree diagram visually represents the hierarchical structure of the WBS. Nodes represent tasks, and the branching illustrates the relationships between them.

Advantages: Provides a clear visual representation, easy to understand, and shows the hierarchy.

1. **Tabular Format:**  A table with columns representing different levels of the WBS hierarchy. Each row corresponds to a task or work package, and columns may include task ID, description, start date, end date, etc.

Advantages: Organized and suitable for detailed information.

Let's create a simplified Work Breakdown Structure (WBS) for a basic software development project. In this example, we'll assume a simple project with three main phases: Planning, Development, and Testing

**Indented Outline Representation :**

1. Project: **Simple Library Management System Development (100%)**

**1.1. Planning Phase (15%)**

- 1.1.1 Define Project Scope (2%)

- 1.1.2 Identify Stakeholders (2%)

- 1.1.3 Gather Requirements (4%)

- 1.1.4 Develop Project Plan (4%)

- 1.1.5 Define System Features and Functionalities (3%)

**1.2. Development Phase (40%)**

**- 1.2.1 Database Design (10%)**

- 1.2.1.1 Define Database Schema (5%)

- 1.2.1.2 Entity-Relationship Diagram (ERD) (5%)

**- 1.2.2 Backend Development (15%)**

- 1.2.2.1 Implement Database Connectivity (8%)

- 1.2.2.2 Develop Core System Logic (7%)

**- 1.2.3 Frontend Development (15%)**

- 1.2.3.1 Design User Interface (8%)

- 1.2.3.2 Implement User Interface Components (7%)

**1.3. Testing Phase (25%)**

**- 1.3.1 Unit Testing (8%)**

- 1.3.1.1 Test Database Operations (3%)

- 1.3.1.2 Test Backend Logic (3%)

- 1.3.1.3 Test User Interface (2%)

**- 1.3.2 Integration Testing (8%)**

**-** 1.3.2.1 Test System Integration (4%)

- 1.3.2.2 Resolve Integration Issues (4%)

**- 1.3.3 User Acceptance Testing (9%)**

- 1.3.3.1 Involve Library Staff (5%)

- 1.3.3.2 Address User Feedback (4%)

**1.4. Deployment (12%)**

**- 1.4.1 Prepare for Deployment (4%)**

- 1.4.1.1 Create Deployment Plan (2%)

- 1.4.1.2 Ensure Compatibility with Library Systems (2%)

**- 1.4.2 System Deployment (5%)**

- 1.4.2.1 Install and Configure System (3%)

- 1.4.2.2 Data Migration (if applicable) (2%)

**- 1.4.3 Training (3%)**

- 1.4.3.1 Conduct Staff Training (2%)

- 1.4.3.2 Provide User Manuals (1%)

**1.5. Documentation (5%)**

**- 1.5.1 User Manuals (3%)**

- 1.5.1.1 Create User Guides (2%)

- 1.5.1.2 Provide Training Material (1%)

**- 1.5.2 Technical Documentation (2%)**

- 1.5.2.1 Document Database Structure (1%)

- 1.5.2.2 Code Documentation (1%)

**1.6. Project Closure (3%)**

- 1.6.1 Finalize Project Documentation (2%)

- 1.6.2 Evaluate Project Success (1%)

- 1.6.3 Lessons Learned (0.5%)

This WBS provides a breakdown of tasks involved in developing a Library Management System, starting from planning and progressing through development, testing, deployment, documentation, and project closure. Each task can be further detailed, and dependencies between tasks can be identified. It's important to note that the level of detail can be adjusted based on the project's complexity and requirements.

**Tabular Format:**

|  |  |  |  |
| --- | --- | --- | --- |
| **LEVEL** | **WBS CODE** | **TASK** | **% of WORK** |
| 1 | 1 | **Simple Library Management System Development** | 100 |
| 1.1 | 1.1 | **Planning Phase** | **(15%)** |
|  | 1.1.1 | Define Project Scope (2%) |  |
|  | 1.1.2 | Identify Stakeholders (2%) |  |
|  | 1.1.3 | Gather Requirements (4%) |  |
|  | 1.1.4 | Develop Project Plan (4%) |  |
|  | 1.1.5 | Define System Features and Functionalities (3%) |  |
| 1.2 | 1.2 | **Development Phase** | **(40%)** |
|  | 1.2.1 | **Database Design (10%)** |  |
|  | 1.2.1.1 | **Define Database Schema (5%)** |  |
|  | 1.2.1.2 | **Entity-Relationship Diagram (ERD) (5%)** |  |
|  | 1.2.2 | **Backend Development (15%)** |  |
|  | 1.2.2.1 | **Implement Database Connectivity (8%)** |  |
|  | 1.2.2.2 | **Develop Core System Logic (7%)** |  |
|  | 1.2.3 | **Frontend Development (15%)** |  |
|  | 1.2.3.1 | **Design User Interface (8%)** |  |
|  | 1.2.3.2 | **Implement User Interface Components (7%)** |  |
| 1.3 | 1.3 | **Testing Phase** | **(25%)** |
|  | 1.3.1 |  |  |
|  | 1.3.1.1 |  |  |
|  | 1.3.1.2 |  |  |
|  | 1.3.1.3 |  |  |
|  | 1.3.2 |  |  |
|  | 1.3.2.1 |  |  |
|  | 1.3.2.2 |  |  |
|  | 1.3.3 |  |  |
|  | 1.3.3.1 |  |  |
|  | 1.3.3.2 |  |  |
| 1.4 |  | **Deployment** | **(12%)** |
|  | 1.4.1 |  |  |
|  | 1.4.1.1 |  |  |
|  | 1.4.1.2 |  |  |
|  | 1.4.2 |  |  |
|  | 1.4.2.1 |  |  |
|  | 1.4.2.2 |  |  |
|  | 1.4.3 |  |  |
|  | 1.4.3.1 |  |  |
|  | 1.4.3.2 |  |  |
| 1.5 | 1.5 | **Documentation** | **(5%)** |
|  | 1.5.1 | **User Manuals (3%)** |  |
|  | 1.5.1.1 | Create User Guides (2%) |  |
|  | 1.5.1.2 | Provide Training Material (1%) |  |
|  | 1.5.2 | **Technical Documentation (2%)** |  |
|  | 1.5.2.1 | Document Database Structure (1%) |  |
|  | 1.5.2.2 | Code Documentation (1%) |  |
| 1.6 | 1.6 | **Project Closure** | **(3%)** |
|  | 1.6.1 | Finalize Project Documentation (2%) |  |
|  | 1.6.2 | Evaluate Project Success (1%) |  |
|  | 1.6.3 | Lessons Learned (0.5%) |  |

**TREE (hierarchical structure) : Draw your self**

**Result : Work Breakdown Structure Created for Simple Library Management System Development**