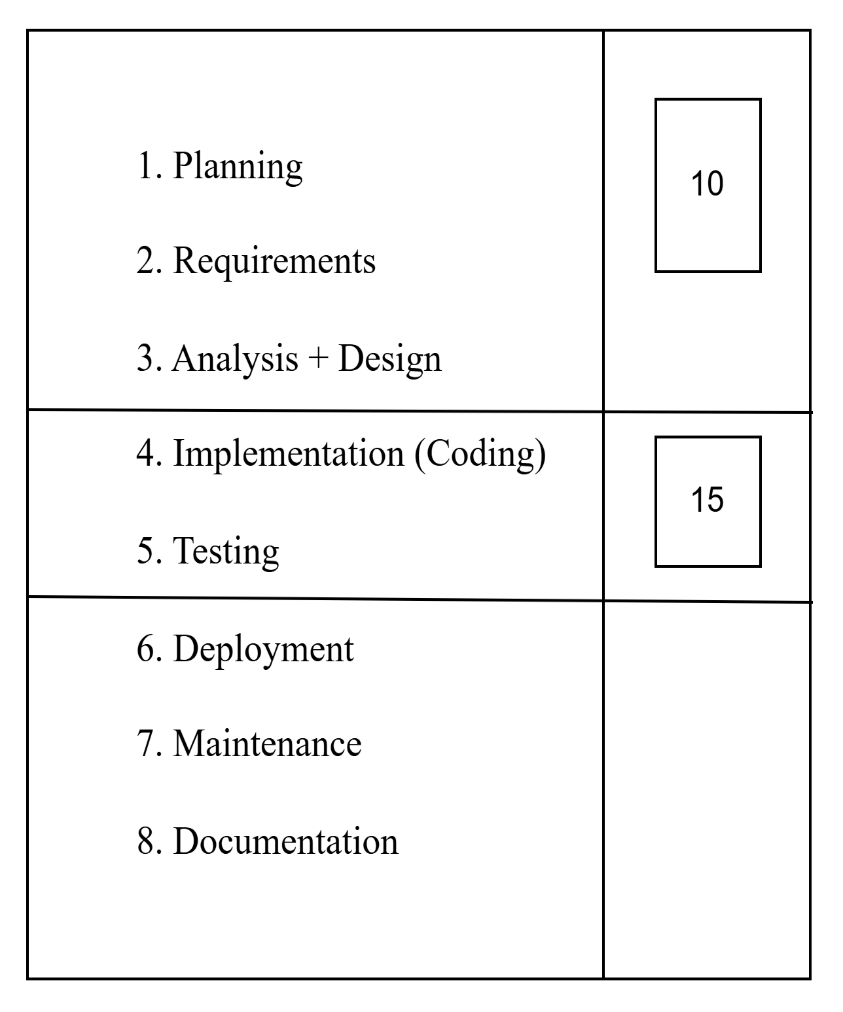
**1st Lab – Online Shop Management System**

**Marks Distribution:**

1. **Scope**

* **Users:**
  + Customer
  + Admin / Shop Owner
* **Core Functionalities:**
  + **Customer Side (Front-End):**
    - Browse products (with categories)
    - Add to cart
    - Place order
    - User Registration and Login
  + **Admin Side (Back-End):**
    - Manage product list (Add, Edit, Delete)
    - Manage customer orders (view, update status)
    - Inventory control
    - View customer list
* **Requirement Overview:**
  + Role-based login system (Customer/Admin)
  + Product database with name, price, image, category, and stock
  + Cart and order system for customers
  + Admin dashboard to manage shop activities
* **Validation:**
  + All input fields should be properly validated (e.g., email format, password strength, stock availability)
  + Features should be implemented according to the shop owner's requirements
* **Control:**
  + If any requirement is technically not possible, suitable alternatives should be implemented to ensure usability

1. **Time Planning**

* Total Project Duration: **7 Days**

|  |  |
| --- | --- |
| **Task** | **Duration** |
| Requirement analysis & Planning | 1 day |
| Front-end development | 2 days |
| Back-end development | 2 days |
| Testing and debugging | 1 day |
| Final adjustments & Deployment | 1 day |

1. **Cost Estimation (Approximate)**

|  |  |
| --- | --- |
| **Category** | **Estimated Cost** |
| Front-end | 3000 BDT |
| Back-end | 4000 BDT |
| Development Time | 3000 BDT |
| **Total Estimated** | **10,000 BDT** |

1. **Quality and Testing**

* All code should be clean, readable, and well-commented so that others can understand it easily
* Proper folder structure and consistent naming conventions must be followed
* Functionality-wise testing should include:
  + Login/Signup flow
  + Add to cart and order placement
  + Admin actions on product and orders
* User Interface testing across devices for responsiveness
* Bug fixing based on testing feedback

1. **Implementation and Testing**

**Implementation (Coding)**

* **The system will be developed using full stack technologies.  
  Example stack:**
  + **Front-End:** HTML, CSS, JavaScript (React.js)
  + **Back-End:** Node.js with Express
  + **Database:** MongoDB or MySQL (based on design)
* **Key functionalities will be implemented module-wise:**
  + User registration & login system
  + Product browsing, cart, and order placement
  + Admin dashboard for product and order management
* **Code will follow best practices:**
  + Clean structure
  + Reusable components/functions
  + Proper error handling

**Testing**

* **Unit Testing:** Small units like functions and components will be tested individually
* **Integration Testing:** Ensuring that modules (e.g., login → cart → order) work together as expected
* **System Testing:** Full system will be tested on a staging environment
* **Cross-browser testing:** To check performance across Chrome, Firefox, Edge, etc.
* **Bug Tracking & Fixing:** Any found bugs will be logged and resolved before deployment

1. **Risk Management**

* **Team Communication:**
  + Regular updates and clear task division among team members
* **Ownership:**
  + A project owner will guide the development and feature priorities
* **Budget Tracking:**
  + Expenses will be tracked based on progress and adjusted as needed