CHAPTER 2

ANALYSIS

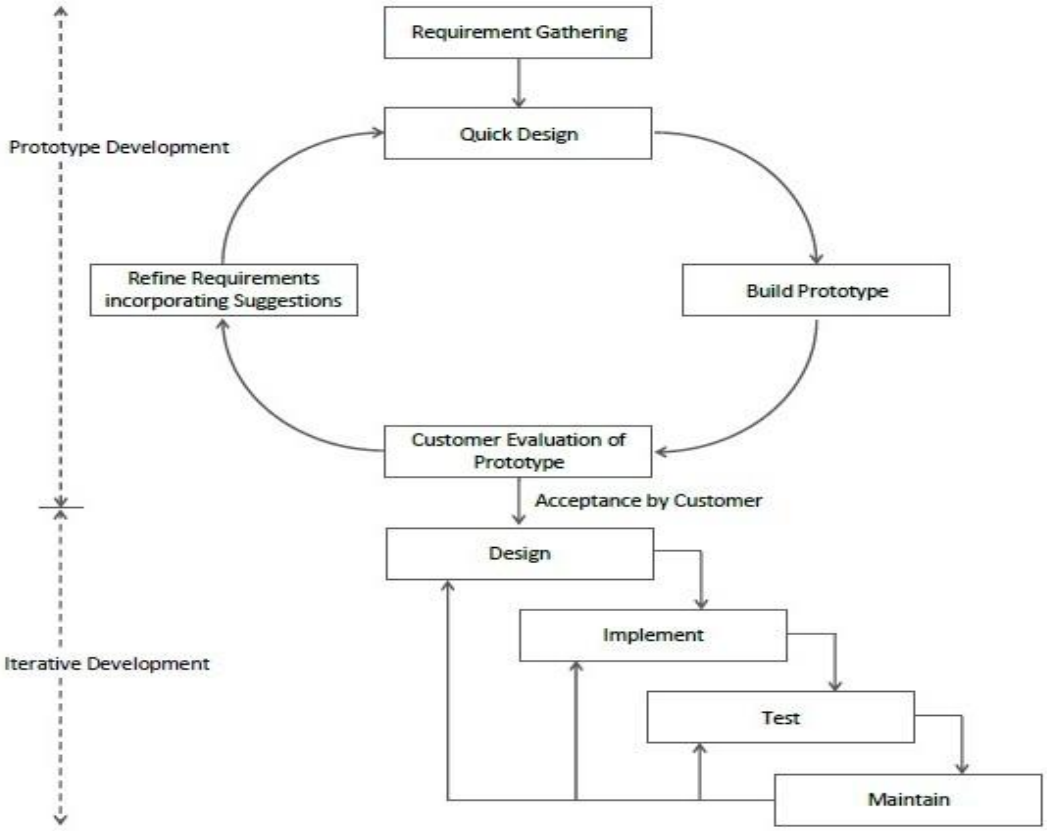
2.1 Requirement Analysis

- **Admin**
  - Owns and controls the entire platform.
  - Manages all registered institutes.
  - Configures system-level settings.
  - Monitors overall system activity.
  - Maintains database integrity.
  - Ensures platform security.
  - Handles technical and administrative issues.
- **User/Player**
  - Accesses personal profile.
  - Views match schedules.
  - Views tournament details.
  - Checks personal performance statistics.
  - Receives match notifications.
  - Participates in assigned matches.
  - Represents the institute in tournaments.
- **Organizer**
  - Registers the institute on the platform.
  - Manages institute profile and logo.
  - Adds or removes sports.
  - Creates and manages tournaments.
  - Schedules matches and events.
  - Creates and manages venues.

- Oversees teams and coaches.
- Controls institute-specific data.

- **Coach**
  - Manages team composition.
  - Selects player line-ups.
  - Assigns player roles.
  - Reviews team schedules.
  - Prepares teams for matches.
  - Monitors team performance.
  - Coordinates with players.
- **Viewers**
  - Views public tournament information.
  - Browses match schedules
  - Views basic match results.
  - Logs in for detailed statistics.
  - Accesses chat after login.
  - Has read-only system access.

2.2 Project Model



The Prototype Model is justified for this project because the University Sports Management Web Application is highly user-interface driven and requires continuous customization based on user needs. Features such as university registration, logo-based branding, dashboard layout, sports selection, and tournament management benefit from early visualization and feedback. By developing an initial prototype, the system requirements become clearer, design issues are identified early, and changes can be incorporated easily without affecting the entire system. This model reduces development risk, improves user satisfaction, and ensures that the final application closely matches user expectations. In conclusion, the Prototype Model provides flexibility, faster requirement validation, and iterative improvement, making it an effective and suitable approach for developing a customizable and scalable sports management web application.

2.3 Schedule Representation

Generalized project scheduling tools and technique can be applied with little modification to software projects.

Program evolution and review techniques (PERT) and critical path method (CPM) are two project scheduling method that can be applied to software development. Both techniques are driven by information already developed in earlier project planning activities:

- Estimate of effort.
- A decomposition of the product function.
- The selection of appropriate process model and task set.
- Decomposition of tasks.

1. Table Name: Schedule Representation

ACTIVITY	START DATE	FINISH DATE
Requirement Analysis		
System Analysis		
System Design		
System Coding		
Testing and Integration		

2.4 Feasibility Study