**Class diagram for our CricWare System :**

| **Sports System** |
| --- |
| * users : List<User> * admins : List<Admin> * externalSystems : List<ExternalSystem> * technicalServices : TechnicalServices |
| * addUser (user: User) * removeUser (user : User) * addAdmin (admin : Admin) * removeAdmin (admin : Admin) * addExternalSystem * removeExternalSystem * setTechnicalServices |

| **User** |
| --- |
| * userID : int * username : string * password : string * role : UserRole |
| * viewSchedule() * monitorFitness() * reviewPerformance() * receiveCommunication() |

| **Admin** |
| --- |
| * adminID : int * username : string * password : string |
| * manageTechnicalIssues() * manageSoftware() * handleUserFeedback() |

| **TechnicalServices** |
| --- |
| * backupFrequency * recoveryPlan |
| * setBackupFrequency * setRecoveryPlan |

| **ExternalSystem** |
| --- |
| * systemName : string * systemType |
| * getSystemName() * getSystemType() |

### **Explanation:**

* **SportsSystem:** Represents the main system and holds lists of users, admins, external systems, and technical services.
* **User** and **Admin:** Represent different types of system users with their respective roles and functionalities.
* **TechnicalServices:** Manages technical aspects of the system, including backup and recovery.
* **ExternalSystem:** Represents external systems integrated with the sports system.

This hierarchy is based on the identified stakeholders and their roles in the system. Further refinement may be required based on specific functionalities and interactions identified during detailed analysis and design. Additionally, the classes **BackupFrequency** and **EmergencyRecoveryPlan** are placeholders for potential future expansion based on the detailed requirements for backup and recovery.