**LAB 1 – MOBILE**

**1. Project Idea**

This project is a **mobile application for managing basketball teams, players, transfers, and training sessions**. Coaches and managers can easily organize their teams, track player details, record transfers, and schedule training sessions. The app is simple to use and supports both online and offline functionality — allowing users to make changes anytime. Once an internet connection is available, all local updates are synced automatically with the server to keep information consistent.

**2. Domain Details**

**Entities**

**Team**

* **TeamID** – unique identifier for each team.
* **Name** – name of the basketball team (required).
* **City** – city where the team is based.
* **CoachName** – name of the team’s coach.
* **Budget** – total financial budget of the team.

**Player**

* **PlayerID** – unique identifier for each player.
* **Name** – player’s full name (required).
* **Age** – player’s age.
* **Height** – player’s height.
* **Weight** – player’s weight.
* **Salary** – player’s salary.
* **Position** – player’s role (Guard, Forward, Center).
* **TeamID** – foreign key linking the player to their team.

**Transfer**

* **TransferID** – unique identifier for each transfer.
* **PlayerID** – player being transferred.
* **FromTeamID** – current team of the player.
* **ToTeamID** – destination team.
* **TransferDate** – date of the transfer.
* **Fee** – transfer fee.

**Training**

* **TrainingID** – unique identifier for each training session.
* **TeamID** – team associated with the session.
* **Date** – scheduled date and time.
* **Type** – training focus (e.g., Shooting, Defense, Conditioning).
* **Duration** – duration of the session.

**3. CRUD Operations**

* **Create** – Add a new team, player, training, or transfer.
* **Read** – View all existing teams, players, ongoing transfers, and scheduled trainings.
* **Update** – Edit teams (e.g., players, budget), player information (e.g., height, weight, salary), transfer details (e.g., fee, date), or trainings (e.g., type, team, schedule).
* **Delete** – Delete a team, player, training, or transfer.

**4. Persistence Details**

**Local Database (on device):**

* **Create** – Store new teams, players, transfers, and trainings locally.
* **Read** – Display cached data even when offline.
* **Update** – Save offline edits and apply them once reconnected.
* **Delete** – Mark deleted items locally for removal when synced.

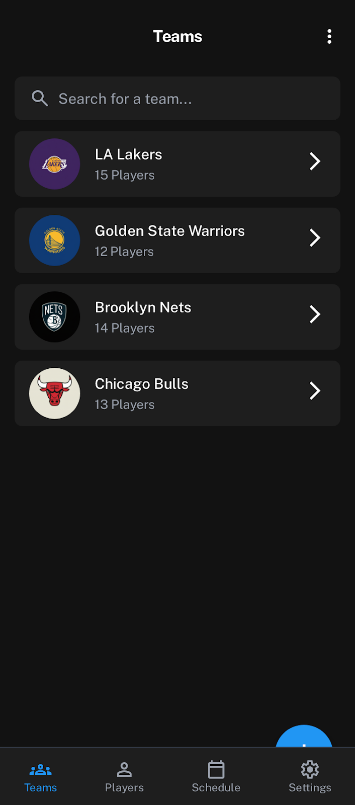
**Server (cloud database):**

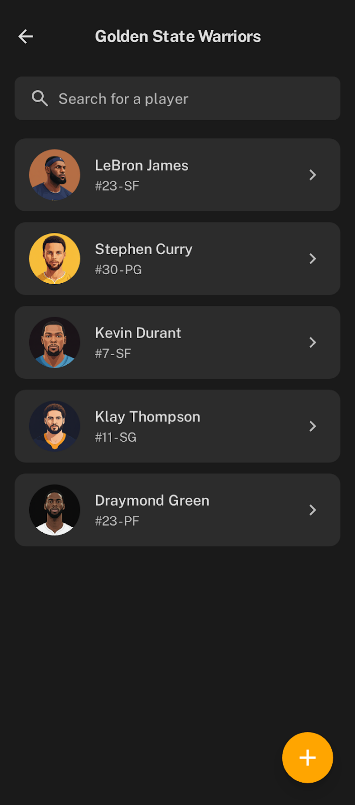
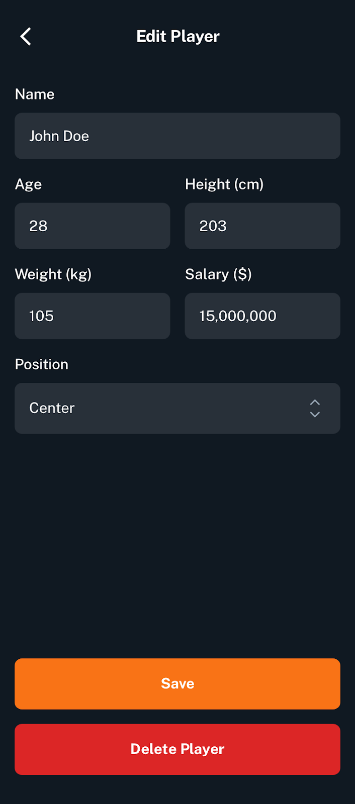
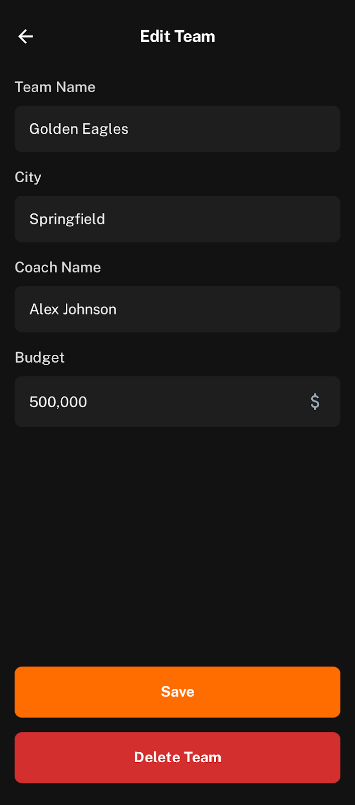
* **Create** – Upload new data to the cloud once online.
* **Read** – Fetch the latest version of all data from the server.
* **Update** – Sync all modified records with the cloud database.
* **Delete** – Permanently remove deleted data from the server.

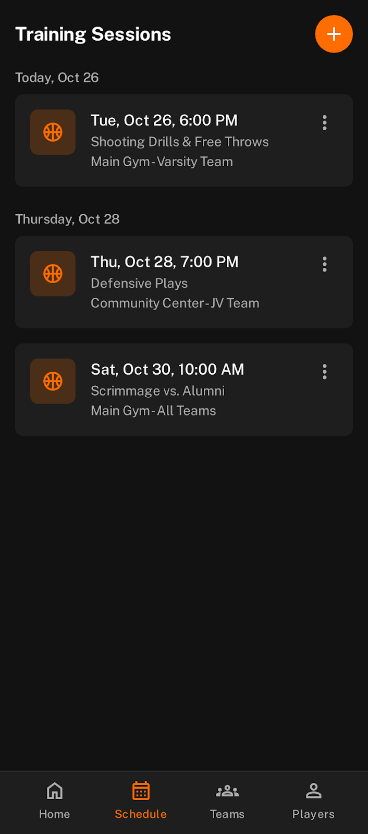
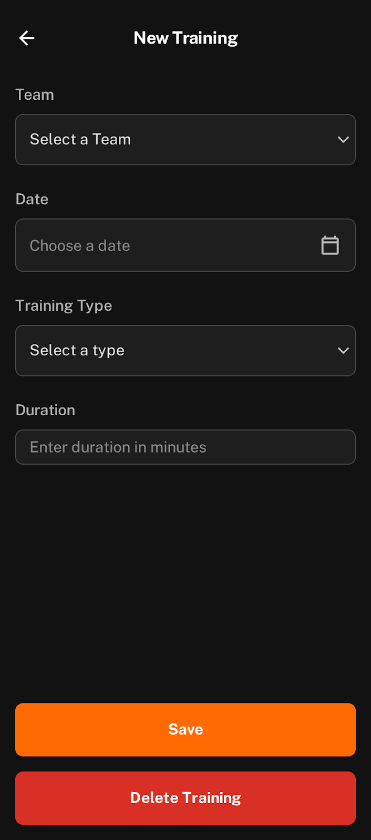
**5. Offline Scenarios**

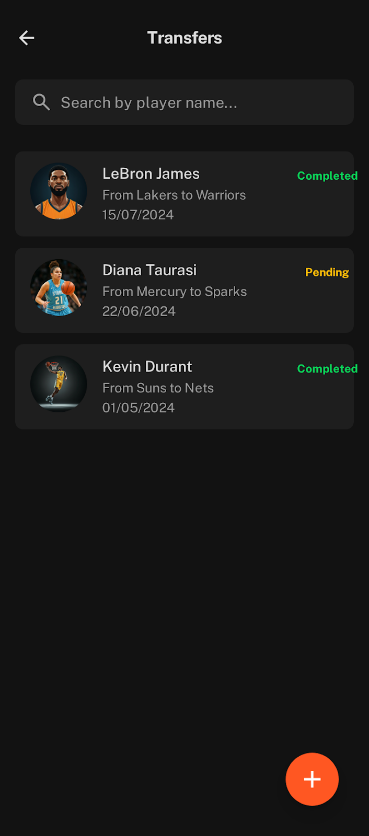
* **Create (offline):** Add a new player, team, transfer, or training → saved locally → synced when connected.
* **Read (offline):** View all previously stored information about teams, players, and trainings.
* **Update (offline):** Edit details (e.g., player stats, training info) → stored locally → updated on the server later.
* **Delete (offline):** Deletion actions are recorded locally and applied to the server after reconnection.

**6. App Mockup**

****Teams List Team Add/Edit Players List Player Add/Edit

****

A screenshot of a phone

AI-generated content may be incorrect.Transfer List Transfer Add/Edit Trainings List Training Add/Edit