**Key Points**

* The scheduler program will have secure account creation and sign-in for coaches and players, with role-based access.
* It includes calendar views (daily, weekly, monthly) and a to-do list for managing tasks.
* Coaches can add, edit, and remove team events, while players can manage their personal events and submit attendance and fitness feedback.
* Coaches can set required items for events and send team messages, with optional notification toggling for users.
* The program ensures data security with encrypted storage and role-based permissions.

**Account Management**

The scheduler will allow users to create accounts by entering a username, password, and role (coach or player), with team affiliation. Passwords will be encrypted and stored securely. Sign-in will validate credentials against this database, granting access to the main dashboard upon success.

**Event and Task Management**

Users can view their calendar in daily, weekly, or monthly formats, showing team events for all members and personal events for individual users. Coaches can add team events visible to all, while players and coaches can add personal events visible only to themselves. A to-do list will display pending tasks, sortable by date and time, with completion options.

**Role-Specific Features**

* **Coaches**: Can add, edit, and remove team events, set required items (e.g., cleats, jerseys), and send messages to the team.
* **Players**: Can indicate attendance (yes/no/maybe) for team events and submit fitness feedback (fatigue, mental state) via questionnaires, linked to specific events.

**Notifications and Security**

Users can toggle notifications on or off, affecting message and event alerts. All sensitive data, especially passwords, will be encrypted, with access controls based on user roles to ensure security.

**Detailed Implementation Specification**

This implementation specification outlines the design of a scheduling application, "TeamSync Scheduler," tailored for coaches and players in sports teams. It provides a comprehensive set of features for event management, communication, and player feedback, with role-based access controls to ensure functionality aligns with user needs. The specification is based on the requirements for a convenient scheduler that supports to-do lists, calendar views, secure account management, player attendance and fitness tracking, event management with differing permissions, item requirements, notification toggling, event notes, and coach-to-team messaging.

**System Overview**

The application will be a web or mobile-based platform where users (coaches and players) can log in to access their personalized dashboards. The system will differentiate functionalities based on user roles, ensuring coaches have tools for team management and players have tools for personal tracking and feedback. Data will be stored in a secure database with encryption for sensitive information, and the system will support real-time updates for calendar views and notifications.

**User Roles and Permissions**

* **Coaches**: Primarily use the app to schedule team events (matches, trainings, community events) and communicate with players. They can add, edit, and remove team events, set required items, and send messages to the team.
* **Players**: Use the app to view their schedule, manage personal tasks, indicate attendance for events, and submit fitness feedback. They can add, edit, and remove their own personal events but cannot modify team events.

**Functional Requirements**

**1. Account Management**

* **Account Creation**:
  + Users input username, password, role (coach/player), and team affiliation.
  + System validates username uniqueness and password strength (e.g., minimum length, special characters).
  + Passwords are encrypted using a secure hashing algorithm (e.g., bcrypt) before storage.
  + Store user details in a secure database with fields: user\_id, username, hashed\_password, role, team\_id.
  + Upon successful creation, redirect to login screen.
* **Account Sign-In**:
  + Display login form for username and password.
  + Validate credentials against the database, comparing hashed passwords.
  + On success, grant access to the main dashboard; on failure, display error message and allow retry.

**2. Dashboard and Calendar Views**

* **Calendar Display**:
  + Offer three views: daily, weekly, monthly, selectable via navigation.
  + Retrieve events from the database:
    - For coaches: Show all team events (added by them) and their personal events.
    - For players: Show all team events (added by coaches) and their personal events.
  + Display events with details (date, time, location, notes, required items) in the selected view format.
  + Include navigation to switch between views and access event details.
* **To-Do List**:
  + Retrieve pending tasks from the database, sorted by date/time.
  + Display as a list with checkboxes for completion.
  + Allow users to mark tasks as complete, updating the database.

**3. Event Management**

* **Add Event**:
  + Input parameters: date, time, location, type (team or personal), notes, required items (for team events, coach-only).
  + If user role is "coach":
    - Option to add as a team event (visible to all team members) or personal event (visible only to them).
    - For team events, save to database and sync to all team members' calendars.
  + If user role is "player":
    - Only allow adding personal events, saved to their own calendar.
  + Store event details in database with fields: event\_id, date, time, location, type, notes, required\_items, creator\_id, team\_id (for team events).
* **Edit Event**:
  + Verify user has permission:
    - Coaches can edit any team event or their personal events.
    - Players can edit only their personal events.
  + Display event details in editable form, allow changes to date, time, location, notes, required items (for team events).
  + Save updates to database, notify affected users (team members for team events).
* **Remove Event**:
  + Verify permission as per edit rules.
  + Delete event from database, update affected calendars, and notify users if necessary.

**4. Player-Specific Features**

* **Submit Attendance**:
  + For team events only, players can select attendance status: Yes, No, Maybe.
  + Save response to database, linked to event\_id and user\_id.
  + Update coach's view to show participation levels for planning.
* **Submit Fitness Feedback**:
  + Linked to specific team events, accessible day before or day of the event.
  + Display quick questionnaire: fatigue level (1-5), mental state (1-5), additional notes.
  + Store responses securely, linked to event\_id and user\_id, accessible to coaches for gauging player condition.

**5. Coach-Specific Features**

* **Set Required Items**:
  + For team events, coaches can input list of items (e.g., cleats, red jerseys) to bring.
  + Save to event details in database, visible to players in event view.
* **Send Team Message**:
  + Input message text, optional attachment to specific event.
  + Send to all team members, store in message history with fields: message\_id, sender\_id, team\_id, content, timestamp, event\_id (if applicable).
  + Notify users based on their notification preferences.

**6. Notifications**

* **Toggle Notifications**:
  + Display toggle switch in user settings.
  + Save preference (on/off) in database, linked to user\_id.
  + System respects preference for sending push notifications for new events, messages, etc.

**7. Security and Data Storage**

* **Secure Data Storage**:
  + Encrypt sensitive data (passwords, personal info) using industry-standard encryption (e.g., AES-256 for data at rest, TLS for transmission).
  + Implement role-based access controls: coaches can view team data, players can view only their data and team events.
  + Use secure database protocols (e.g., SQL injection prevention, prepared statements).

**Data Model**

The following table outlines the proposed database structure:

A screenshot of a black screen

AI-generated content may be incorrect.

**User Experience**

* **Player Experience**: Upon logging in, players see the monthly calendar view with planned trainings. They can RSVP to events (yes/no/maybe) ahead of time for coach planning. They can view event notes and, day before or day of, submit fitness feedback via questionnaires for coaches to assess their condition.
* **Coach Experience**: Coaches can add team events, set required items, and send reminders. They can view attendance and fitness feedback to plan accordingly, ensuring effective team management.

**Technical Considerations**

* The system should support real-time updates for calendar views and notifications, possibly using WebSocket or push notification services.
* Scalability should be considered for teams with large numbers of players, ensuring database queries remain efficient.
* User interface should be intuitive, with mobile responsiveness for on-the-go access.

This specification provides a foundation for implementing the TeamSync Scheduler, ensuring all required features are addressed with clear role-based permissions and secure data handling.