

PocketSports: The Digital Coaching App

Garrett Gmeiner, Taylor Carlson, Tyler Ton, and Parker Cummings

Faculty Advisor(s): Dr. Fitzroy Nembhard, Dept. of Electrical Engineering and Computer Science, Florida Institute of Technology

Abstract:

Lots of apps only focus on one end of coaching. PocketSports aims to eliminate the need for multiple apps by delivering a cross-sport app with an increased app functionality

Coaches can...

1. Set and Track Goals
2. Design Practice Plans
3. Execute and Analyze Practice Plans

Players can...

1. Create Goals and Monitor Progress
2. View Practice Plans
3. Review Practice Results

System Architecture:

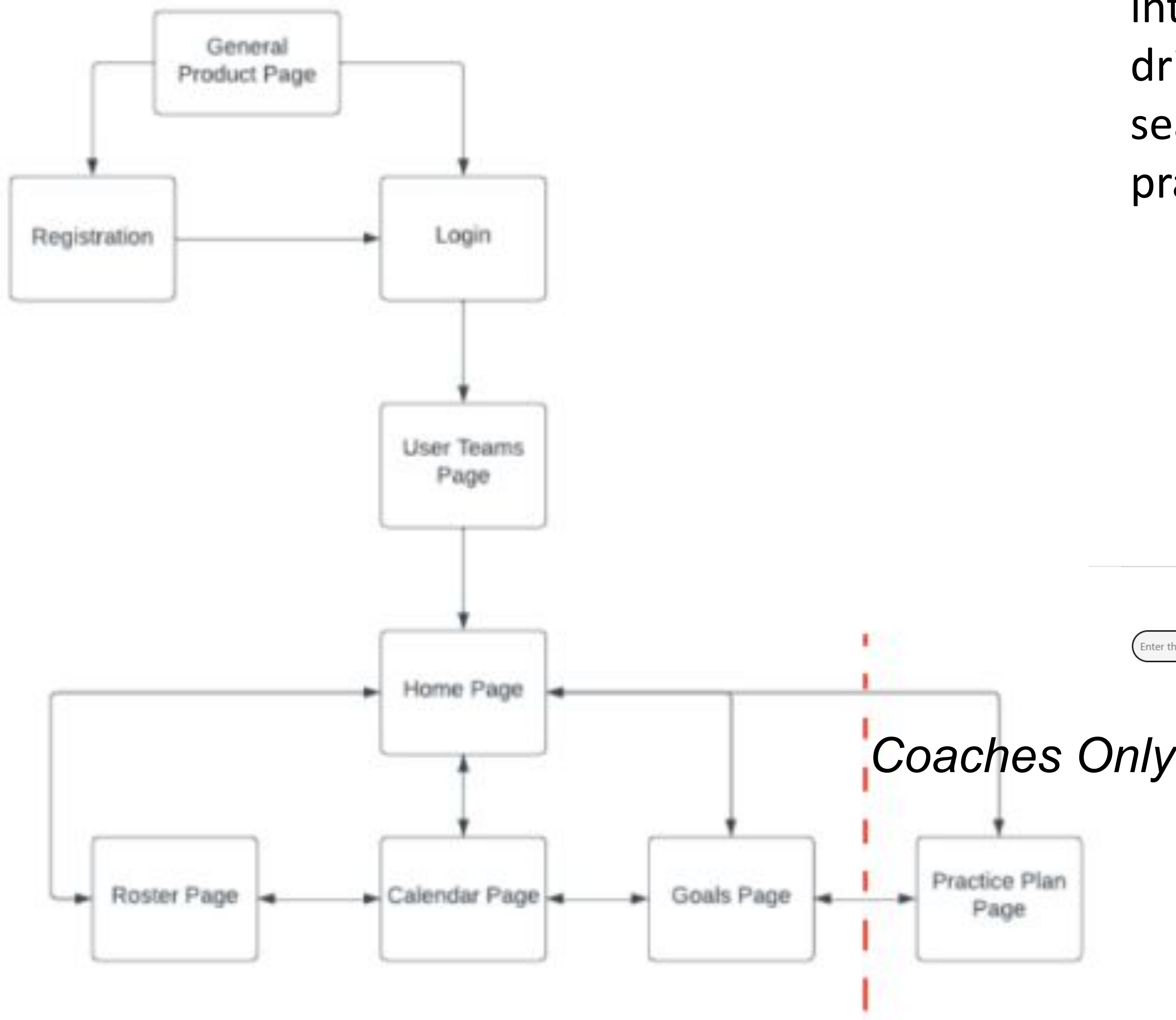
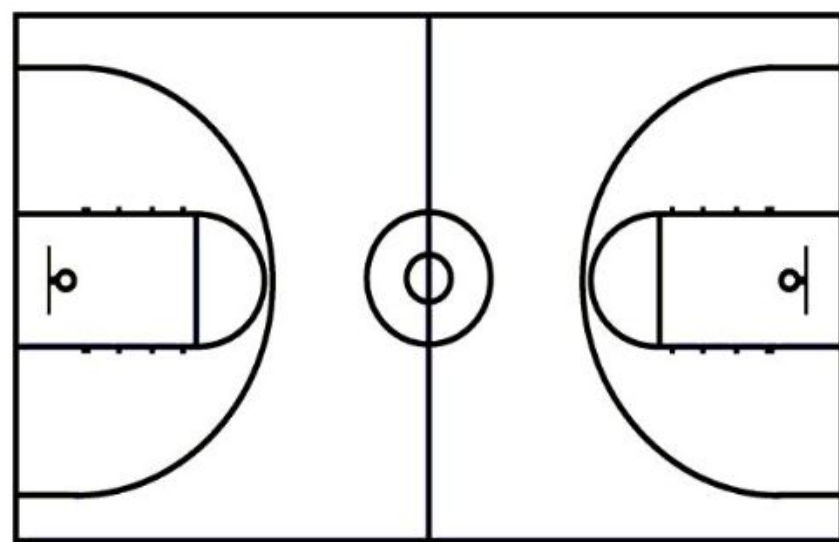


Figure 1: System Architecture

Methods:

- MERN Stack Implementation: Using MongoDB for data persistence, Express and Node.js for server-side logic, and React for the front end, ensuring a modular, scalable architecture.
- Real-Time Features: Employing websockets to allow coaches to collaboratively design and execute practice plans, track live drill times, and update player stats on the fly.
- Data Visualization: Integrating libraries such as Chart.js to display progress toward goals and performance metrics, enabling coaches and players to monitor improvements visually.
- Role-Based Access: Implementing secure login flows and role-specific dashboards for owners, coaches, players, and parents, ensuring that each user only sees data relevant to their role.
- Goal Setting & Practice Planning: Providing interactive design tools for creating animated drills, storing them in a library, and seamlessly integrating them into daily practice schedules.



Add O Add X Add ...

Delete Selected

Drill Name:

Drill Name:

Enter the drill's name

Select an Existing Tag

Enter a new tag

Add Drill Tag

Selected Tags:

Save Drill

Close

Acknowledgements:

Brad MacArthur, Men’s Lacrosse
Jordan Willis, Women’s Volleyball
Michaela Gelbaugh, Women’s Basketball

Tools Used:

- GitHub: We used GitHub as our code repository, to conduct code reviews, and host the website
- Git: We used Git as an intermediate with local and remote branches and to push our work to GitHub.
- Jira: We used Jira for project management
- Node: We used Node as the application’s server environment
- MongoDB: We used MongoDB for the application’s database
- Google Docs/Slides: We used Google Docs/Slides for document collaboration
- HTML: We used HTML for creating the webpage’s structure
- JavaScript: We used JavaScript for creating the webpage’s functionality
- CSS: We used CSS for creating the webpage’s styling

Limitations and Improvements:

- Limitations
 - Drill libraries and performance metrics are currently basic
 - Partial offline functionality still relies on stable network access for optimal data sync
 - Lack of advanced AI-driven analytics for personalized training recommendations
- Improvements
 - Expand sports-specific drill libraries for broader coverage
 - Enhance offline modes with robust caching and synchronization
 - Integrate AI/ML capabilities to deliver more customized and data-driven coaching insights