

# pixRUs Project Work Plan

## Project Overview

pixRUs is a web platform designed to connect sports pick sellers and buyers, fostering a transparent and trustworthy marketplace for sports predictions. The platform ensures secure transactions, highlights seller credibility, and plans for future integrations of AI and machine learning to enhance user experience. Since the team is unfamiliar with Django, PostgreSQL, Tailwind CSS, and JavaScript, time will be allocated for learning these technologies.

## Technologies & Learning Plan

- Backend: Django (1 week learning)
- Database: PostgreSQL (1 week learning)
- Frontend: Tailwind CSS, JavaScript (1.5 weeks learning)
- Future Integration: AI/LLMs for predictive analytics and suggestions

## Adjusted Milestone Calendar

Milestone	Due Date	Objective
Milestone 1	October 21, 2024	Learning and Initial Setup
Milestone 2	November 13, 2024	Backend Development & Database Setup
Milestone 3	December 2, 2024	Frontend Development & API Integration
Milestone 4	December 16, 2024	Core Features Implementation
Final Presentation	December 18, 2024	Project presentation, demo video, and public GitHub repository

## Milestone 1: Learning and Initial Setup (Due October 21, 2024)

Objectives:

1. Learn Django (1 week), PostgreSQL (1 week), and Tailwind CSS/JavaScript (1.5 weeks).
2. Set up the development environment.
3. Define the overall architecture.

Tasks:

- Complete tutorials for Django, PostgreSQL, and Tailwind CSS/JavaScript.
- Set up Git, Django, and connect the PostgreSQL database.
- Initialize Tailwind CSS and integrate JavaScript.

Deliverables:

- Fully configured environment.
- Initial database schema and backend architecture.

## **Milestone 2: Backend Development & Database Setup (Due November 13, 2024)**

Objectives:

1. Build foundational backend functionality.
2. Establish core API endpoints.

Tasks:

- Implement Django models for User, Seller, Buyer, Picks, Transactions.
- Set up user authentication (JWT or session-based).
- Set up API endpoints and test them.

Deliverables:

- Functional backend with API integration.
- Basic CRUD operations working.

## **Milestone 3: Frontend Development & API Integration (Due December 2, 2024)**

Objectives:

1. Develop frontend interfaces.
2. Integrate frontend with backend APIs.

Tasks:

- Create user interfaces for login, browsing, and transactions.
- Implement frontend API requests (GET, POST, PUT).
- Test API calls from the frontend.

Deliverables:

- Working UI for core features.
- Integrated frontend-backend functionality.

## **Milestone 4: Core Features Implementation (Due December 16, 2024)**

Objectives:

1. Finalize primary marketplace functionalities.
2. Implement trust mechanisms.

Tasks:

- Implement seller profiles, reviews, and ratings.
- Build a payment system (Stripe/PayPal).
- Ensure transaction logging and seller transparency.

Deliverables:

- Full marketplace with buying and selling functionality.
- Secure payment gateway and trust mechanisms.

## **Final Presentation: Demo and GitHub Repository (Due December 18, 2024)**

Objectives:

1. Prepare for the presentation and demo.
2. Release the project repository.

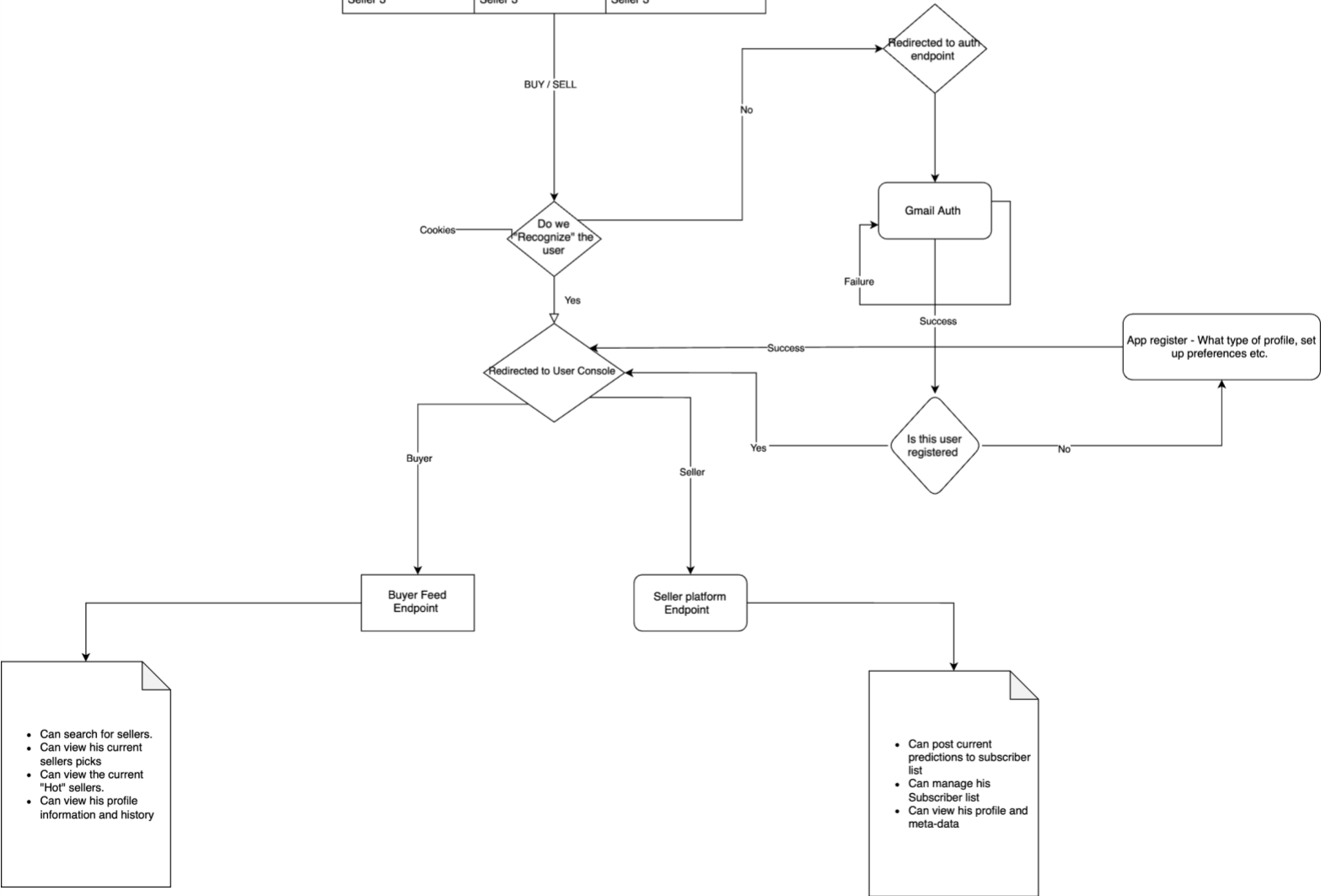
Tasks:

- Finalize and review the project.
- Record a demo video showcasing key functionalities.
- Push the final version to GitHub.

Deliverables:

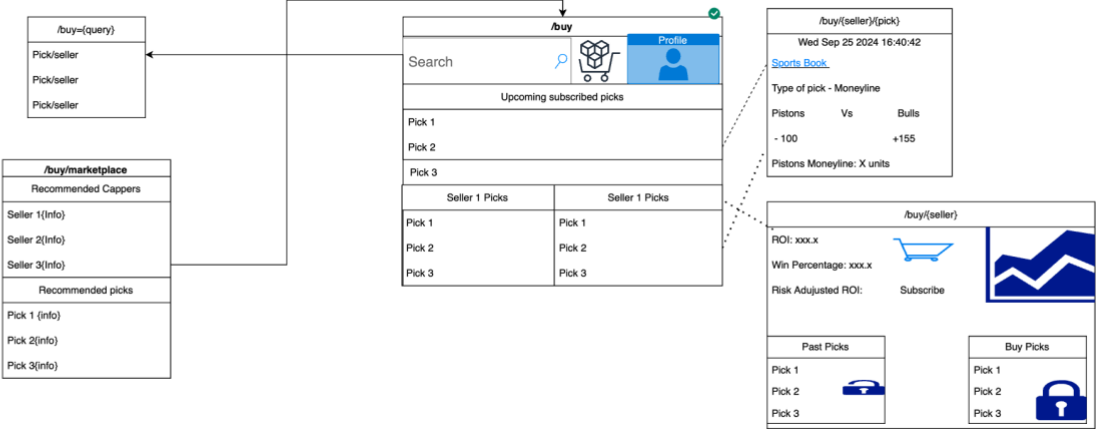
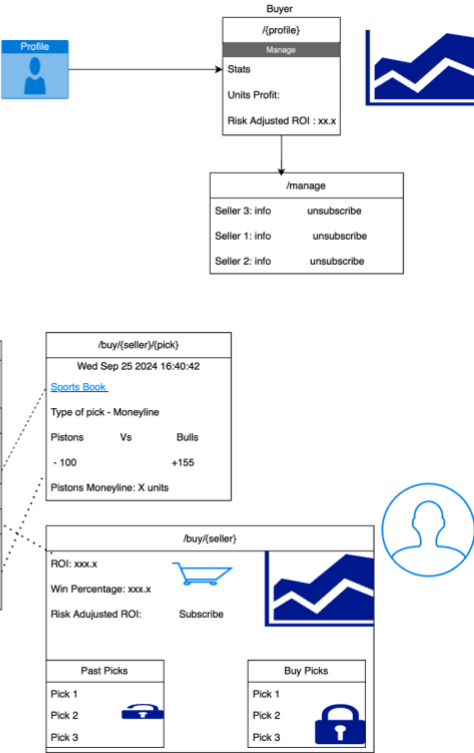
- Completed project presentation and demo.
- Public GitHub repository for pixRUs.

PIX		
Buy		Sell
<b>Buy</b>		<b>Sell</b>
Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.		Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.
Hottest	Recommended	CFB
Seller 1	Seller 1	Seller 1
Seller 2	Seller 2	Seller 2
Seller 3	Seller 3	Seller 3
NBA	CBB	Hockey
Seller 1	Seller 1	Seller 1
Seller 2	Seller 2	Seller 2
Seller 3	Seller 3	Seller 3



/Buy endpoints
/ -> shows current subscriptions(picks).
/profile_id shows profile info - stats, payment etc.
/marketplace -> displays available cappers
/manage -> endpoint to manage subscriptions.
/seller_id -> endpoint of current seller
/seller/pick_id -> endpoint of a pick

Buyer flow.



# Seller Flow

