Team Bracketed

Michael Rhode, Sam Winkelmann, William Braun, Luke Myers

Milestones

- User Research
  - Conduct interviews and gather insights from potential users to understand expectations, preferred features, and usability needs
- Design Concept
  - Create and refine wireframes and interface mockups. Review the design as a team to finalize the visual layout
- Framework and Tech Stack Finalization
  - Select the frontend, backend, and database frameworks. Install all necessary software for selected frameworks on developer environments
- Initial Backend Setup
  - Set up the backend server structure and connect it to a test database and the frontend
- Finish relational database setup
  - Create multiple database tables for different sections of our data: team data, user data, current and past leagues, etc. so that other functions can be implemented and tested properly.
- Create a minimum viable product
  - Finish a simplified version of the final application. Users will be able to create a challenge and select the winners for each game. 3rd party API's for live score updates will be used and results and records will be displayed.
- Set up user accounts and leaderboards
  - Finalize user account creation, authentication, and leaderboard ranking features. Verify scoring logic accuracy and display rankings dynamically
- Extend support for more leagues
  - Bring in other sports for a wider experience in the leagues.
- Testing and Validation
  - Conduct end-to-end testing across the frontend, backend, and database. Validate data accuracy, API performance, and usability feedback
- Final Review and Presentation Preparation

Review all components for consistency, prepare documentation, and create presentation materials for the final project demonstration

Team Bracketed	Luke Myers, William Braun, Sam Winkelmann, Michael Rhode		
Timeline			
Person	Task	Start Date	End Date
	Conduct interviews and gather insights from potential users to understand expectations, preferred features, and usability needs	10/5/2025	10/19/2025
Sam	Interview people to gain insight into what a user would want out of the game/UI.	10/5/2025	10/19/2025
	Create and refine wireframes and interface mockups. Review the design as a team to finalize the visual layout	10/19/2025	11/9/2025
Sam	Sketch mockups of the interface of the website.	10/5/2025	11/9/2025
Luke	Research potential third party services such as user authentication and automated email messaging, and decide if it would be better to implement our own or use the third-party providers.	10/19/2025	11/2/2025
Michael	Research statistics that would be important to calculate to show player/team skill.	10/19/2025	11/9/2025
	Select the frontend, backend, and database frameworks. Install all necessary software for selected frameworks on developer environments	10/5/2025	10/19/2025
Michael	Discover and decide on a database option for holding data to be used in internal data calculations.	10/5/2025	10/19/2025
Luke	Analyze benefits and drawbacks of different front and backend frameworks to determine what will work best for our application.	10/5/2025	10/12/2025
	Set up the backend server structure and connect it to a test database and the frontend	10/19/2025	11/23/2025
Michael	Learn how to connect a database to an API that has live updated information from the games being played.	10/5/2025	10/26/2025
Michael	Develop data tables in a database to hold data imported from an API.	10/26/2025	11/23/2025
William	Design and implement API endpoints that allow the frontend to retrieve and update user predictions, leaderboards, and match results.	10/19/2025	11/23/2025
Luke	Determine what information about sports leagues (Teams, Games, Schedules etc) should be stored locally vs gotten from APIs to maximize space and time optimization.	10/19/2025	11/2/2025
Sam	Implement decided upon design for the front-end of the website.	10/19/2025	11/23/2025
	Create multiple database tables for different sections of our data: team data, user data, current and past leagues, etc. so that other functions can be implemented and tested properly.	10/19/2025	11/23/2025
William	Integrate external sports and Esports APIs into the backend system to collect game data and make it accessible to the application.	10/10/2025	11/23/2025
	Finish a simplified version of the final application. Users will be able to create a challenge and select the winners for each game. Third party API's for live score updates will be used and results and records will be displayed.	11/23/2025	1/11/2025
Luke	Create a straightforward development environment where the front and backend can either run locally or online, and real vs test data can be used.	11/23/2025	12/7/2025
Sam	Develop initial barebones website so we can connect backend/database.	11/23/2025	1/4/2025
William	Develop logic for scoring and leaderboard updates to ensure user predictions are processed correctly and results are reflected accurately.	11/23/2025	1/11/2025
	Finalize user account creation, authentication, and leaderboard ranking features. Verify scoring logic accuracy and display rankings dynamically	1/11/2025	2/8/2026
William	Implement backend authentication to handle user login and registration.	1/11/2025	2/1/2025
William	Test backend to frontend communication by verifying that data transfers smoothly between the UI and database.	2/1/2025	2/8/2025
	Conduct end-to-end testing across the frontend, backend, and database. Validate data accuracy, API performance, and usability feedback	2/8/2026	2/22/2026
	Review all components for consistency, prepare documentation, and create presentation materials for the final project demonstration	2/22/2026	3/15/2026
Michael	Document all progress with respect to the database and its connection to the API and issues that arise and need to be solved.	10/5/2025	3/15/2026

Effort Matrix					
Primary Responsibility	Task	Sam	Michael	William	Luke
Sam	Interview people to gain insight into what a user would want out of the game/UI.	80%	0%	0%	20%
Sam	Sketch mockups of the interface of the website.	60%	0%	0%	40%
Sam	Implement decided upon design for the front-end of the website.	70%	0%	0%	30%
Sam	Develop initial barebones website so we can connect backend/database.	75%	0%	0%	25%
Michael	Discover and decide on a database option for holding data to be used in internal data calculations.	20%	60%	0%	20%
Michael	Develop data tables in a database to hold data imported from an API.	0%	80%	20%	0%
Michael	Research statistics that would be important to calculate to show player/team skill.	10%	70%	10%	10%
Michael	Learn how to connect a database to an API that has live updated information from the games being played.	10%	60%	20%	10%
Michael	Document all progress with respect to the database and its connection to the API and issues that arise and need to be solved.	0%	70%	30%	0%
William	Design and implement API endpoints that allow the frontend to retrieve and update user predictions, leaderboards, and match results.	10%	10%	60%	20%
William	Integrate external sports and Esports APIs into the backend system to collect game data and make it accessible to the application.	5%	15%	60%	20%
William	Develop logic for scoring and leaderboard updates to ensure user predictions are processed correctly and results are reflected accurately.	0%	10%	80%	10%
William	Implement backend authentication to handle user login and registration.	0%	10%	70%	20%
William	Test backend to frontend communication by verifying that data transfers smoothly between the UI and database.	10%	0%	50%	40%
Luke	Research potential third party services such as user authentication and automated email messaging, and decide if it would be better to implement our own or use the third-party providers.	10%	10%	10%	70%
Luke	Determine what information about sports leagues (Teams, Games, Schedules etc) should be stored locally vs gotten from APIs to maximize space and time optimization.	0%	30%	20%	50%
Luke	Create a straightforward development environment where the front and backend can either run locally or online, and real vs test data can be used.	10%	10%	10%	70%
Luke	Analyze benefits and drawbacks of different front and backend frameworks to determine what will work best for our application.	25%	0%	25%	50%