



CRYPTO©LEAGUE

Team 8: Sprint 2 Planning Document

Varun Gupta, Ritwik Bhardwaj, Utkarsh Jain, Nisarg Kolhe

February 26, 2018

SPRINT OVERVIEW

For this sprint, we hope to be done with the main logic of our game which includes the users being able to join and compete in leagues, create and edit their portfolio, and get the winnings once the league is over. The 'Join a League' tab will allow the users to select the available league types and either put them in a waiting room or assign them a league depending on league availability. The 'League' tab will show them the current league statistics, ranking of the users in the league, and a card to edit their portfolio. Clicking on the card to edit their portfolio will allow the users to create and manage their portfolio with either text input or a draggable pie chart.

Scrum Master:

Varun Gupta

Meeting Plan:

With team: Every Monday and Wednesday 6 - 8 PM

With team coordinator: Every Wednesday 1:30 - 2:00 PM

Risks and Challenges:

Now that we are in the second sprint, we are going to implement the leagues feature. This is the most important part since it contains the essence of our project. This poses a challenge for us as there are many logical things in the implementation which can be easily overlooked. We will have to be careful about this and make sure that there are no issues that the user may face while joining a league and being in the league. Since we have lot of features in the sprint, i.e. to implement the league and portfolio features, we fear that we might not be able to meet the non-functional requirements which we might push to sprint 3. We have a running risk of not finishing everything in the given time frame. A lot of the tasks we have in this sprint, are dependent on each other. Due to this, being unable to finish one might have an effect on a lot of other tasks too. We have a challenge of finishing tasks which are a bottleneck for the other tasks. We are going to implement some tasks with which none of us have experience with, such as implementing watchers to automatically change the status of the league.

CURRENT SPRINT DETAILS

User Story #1

As a user, I would like to be able to see various leagues and their rules.

#	Description	Time	Owner
1	Implement league selection UI	6hr	Utkarsh
2	Implement route to get league-types	4hr	Utkarsh
3	Implement warning modal before joining a league	2hr	Utkarsh
4	Manually test the league-types API	2hr	Utkarsh
5	Write unit tests to check if the views are being populated in the front-end	2hr	Utkarsh

Acceptance Criteria:

1. Given that user is logged in, when the user clicks on 'Join A League', then they should be presented with all the available league types.
2. Given that the API route for league types is successfully implemented, when the data is received from the API, then the data for various league types should be displayed in their respective cards.
3. Given that the user is on the league selection UI, when the user selects a league, then they should be shown a warning message to inform them that they cannot leave a league once they join in.

User Story #2

As a user, I would like to be able to join a league and see the status of my current league.

#	Description	Time	Owner
1	Implement league waiting room UI	4hr	Nisarg
2	Implement automatic status change for leagues	6hr	Nisarg
3	Implement route to get the league	3hr	Varun
4	Implement league locking logic	4hr	Nisarg
5	Write unit tests to check if the views for league waiting room UI are being populated in the front-end	2hr	Utkarsh
6	Manually test the route for joining a league and if time allows add unit tests	2hr	Varun

Acceptance Criteria:

1. Given that a user is logged in, when they are currently not in a league, then they should be able to join a league.
2. Given that a user is logged in, when they are trying to join a league, then they should be able to see the various types of leagues.
3. Given that a user has joined a league, when they are on the league page, then they should be able to see the status of the league and the time left for the league to start.

User Story #3

As a user, I would like to be able to see the leaderboard of my current league.

#	Description	Time	Owner
1	Implement league leaderboard route in the backend	4hr	Varun
2	Manually test automatic changes to the leaderboard and if time allows add unit tests	2hr	Varun
3	Implement the UI of the leaderboard card	4hr	Nisarg
4	Write unit tests to check if the views for leaderboard card are being populated in the front-end	2hr	Nisarg

Acceptance Criteria:

1. Given that a user is in a league, when the user navigates to their league view, then the user should see a leaderboard card on the screen.
2. Given that a user has joined a league, when they are on the leaderboard view, they should be able to see the leaderboard of the league showing the users based on their ranks.
3. Given that a user has joined a league, when they are on the leaderboard view, they should be able to see their personal rank on all pages of the leaderboard card.

User Story #4

As a user, I would like to be able to create my cryptocurrency portfolio.

#	Description	Time	Owner
1	Implement the portfolio UI	5hr	Ritwik
2	Implement search and add coins to the portfolio	5hr	Ritwik
3	Write unit tests to check if the views for portfolio UI are being populated in the front-end	2hr	Ritwik
4	Implement the portfolio object in the backend	2hr	Varun
5	Add the post and get user portfolio routes in the backend	4hr	Varun
6	Create the user portfolio in the database appropriately	1hr	Varun
7	Manually test the backend route for posting and getting the user portfolio and if time allows add unit tests	2hr	Varun

Acceptance Criteria:

1. Given that the user has joined a league, when the user clicks on their portfolio, then they should be able to create their portfolio.
2. Given that the user is creating their portfolio, when the user tries to search or add a coin to their portfolio, then they should see a search bar which will show them appropriate coin data.
3. Given that the backend routes are implemented, when the user submits their portfolio, then all of their portfolio information should be saved appropriately in the database and the changes should be reflected on the frontend as well.

User Story #5

As a user, I would like to be able to easily modify my portfolio before the league starts.

#	Description	Time	Owner
1	Implement updating the user portfolio via the pie chart and the list view	4hr	Ritwik
2	Implement portfolio validation	3hr	Ritwik
3	Manually test the validation for the updating the user portfolio	2hr	Ritwik
4	Update the user portfolio in the database appropriately	1hr	Varun
5	Manually test the backend and check if the user portfolio is getting updated in the database appropriately and if time allows add unit tests	3hr	Varun

Acceptance Criteria:

1. Given that the user has already created a portfolio, when the user tries to edit their portfolio, then they should be able to do so using the pie chart or by manually entering the data.
2. Given that the user has made edits to their portfolio, when the user tries to submit the changes, then their portfolio should get validated by both the frontend and the backend (server).
3. Given that the user has entered valid information on their portfolio, when the validations are done, then all of their portfolio information should be saved appropriately in the database and the changes should be reflected on the frontend as well.

User Story #6

As a user, I would like to be able to visualize my current portfolio in a chart as well as a list view.

#	Description	Time	Owner
1	Implement a Pie chart in the Portfolio card on the league view	3hr	Nisarg
2	Implement list view in the Portfolio card	3hr	Nisarg
3	Write unit tests to check if the views are being populated in the front-end	2hr	Nisarg

Acceptance Criteria:

1. Given that a user is in a league, when the user navigates to their league view, then the user should see a portfolio card on the screen.
2. Given that the route for the portfolio is implemented in the backend, when the API is called from the frontend, then the portfolio object should successfully be received and stored by the frontend.
3. Given that the frontend has the portfolio object stored, when the user navigates to the league view, then the chart and the list view should be populated automatically.

User Story #7

As a user, I would like to be able to see the real-time market data of cryptocurrencies in my portfolio.

#	Description	Time	Owner
1	Display market data of the user's coins on the Portfolio view.	3hr	Ritwik
2	Implement route to get the Market data of the users coins.	2hr	Ritwik
3	Write unit tests to check if the coin data is being populated in the front-end	2hr	Nisarg

Acceptance Criteria:

1. Given that a user is in a league, when the user has not created a portfolio, then the market data of none of the coins should be shown in the portfolio view.
2. Given that the user is in a league, when the user has already created a portfolio, then the user should see the market data of their coins in the portfolio view.
3. Given that the route for the market data is implemented in the backend, when the API is called from the frontend, then the market data should successfully be received and stored by the frontend.

User Story #8

As a user, I would like to be able to see my league statistics.

#	Description	Time	Owner
1	Implement the UI to view all of the statistic for the league	6hr	Utkarsh
2	Calculate the statistics appropriately on the frontend and display them to the users	4hr	Utkarsh
3	Write unit tests to check if the views for statistics UI are being populated accurately in the front-end	2hr	Utkarsh

Acceptance Criteria:

1. Given that the user has joined a league, when the user clicks on their league, then they should be able to view the league statistics.
2. Given that the user has joined a league, when the league is locked, then the user should be able to see more data on the UI card for example the final payouts for each rank.
3. Given that the user has joined a league, when the league has started, the user should be able to see the current leader of their respective league on the UI card.

User Story #9

As a user, I would like to be able to receive payouts for my current league.

#	Description	Time	Owner
1	Implement league ending logic	5hr	Varun
2	Manually test automatic status change and league locking and if time allows add unit tests	2hr	Varun

Acceptance Criteria:

1. Given that the league's status is not a waiting room, when the league's status changes to being locked or the league ends, then the watcher function should be called for the respective status change.
2. Given that the league's status is locked, when the league ends, then the league's status should be changed to ended and the respective winning user's should get their payout tokens.
3. Given that the user is in a league, when the league ends, then the final statistics of the league should be shown to the user.

User Story #10

As a developer, I would like to be able to hold concurrent leagues and let users join only one league at a time.

#	Description	Time	Owner
1	Implement functionality to allow users to join only one league at a time	3hr	Ritwik
2	Implement functionality to allow concurrent leagues to run at the same time in the backend	2hr	Ritwik
3	Manually test if the user is being able to join only a single league at a time and if there are concurrent leagues in the backend and if time allows add unit tests	2hr	Varun

Acceptance Criteria:

1. Given that a user is already in a league, when an API request to join another league is made to the server, then the server should return an error message.
2. Given that a user is logged in, when the user has already joined a league, then the user should not be able to join another league on the frontend.
3. Given that a league is already full or has already started, when a new user tries to join a league, then a new league should be created to accommodate that user.

REMAINING BACKLOG

30/42 Stories Complete

Functional Requirements

1. As a user, I would like to be able to register for a CryptoLeague account with a social platforms like Facebook or Google.
2. As a user, I would like to be able to login and manage my CryptoLeague account.
3. As a user, I would like to be able to change my password.
4. As a user, I would like to be able to see my current tokens.
5. As a user, I would like to be able to watch real-time market prices of cryptocurrencies.
6. As a user, I would like to be able to see current cryptocurrency related news.
7. As a user, I would like to be able to see current leaderboard of top performing players.
8. As a user, I should be able to see my rank in the global leaderboard.
9. As a user, I would like to be able to see profiles of people in the leaderboard.
10. As a user, I would like to be able to see various leagues.
11. As a user, I would like to be able to see the rules for different leagues.
12. As a user, I would like to be able to see the buy-ins and payouts for different leagues.
13. As a user, I would like to be able to join a league.
14. As a user, I would like to be able to see the status of my current league.
15. As a user, I would like to be able to see the leaderboard of my current league.
16. As a user, I would like to be able to make my cryptocurrency portfolio.
17. As a user, I would like to be able to add various cryptocurrencies to my portfolio.
18. As a user, I would like to be able to select a captain coin whose gains/losses will be doubled in the final score.
19. As a user, I would like to be able to easily modify my portfolio before the league starts.
20. As a user, I would like to be able to visualize my current league portfolio.

21. ~~As a user, I would like to be able to see my portfolio in a list view.~~
22. ~~As a user, I would like to be able to see the real-time market data of cryptocurrencies in my portfolio.~~
23. ~~As a user, I would like to be able to see the the time for the league to start and end.~~
24. ~~As a user, I would like to be able to see the prize distribution of the league.~~
25. ~~As a user, I would like to be able to my current rank in the league.~~
26. As a user, I would like to be able to see my previous leagues.
27. As a user, I would like to be able to see my current statistics.
28. ~~As a developer, I would like to be able to restrict users to join only one league at a time.~~
29. As a developer, I would like to be able to replenish user's tokens if they do not have enough to join the lowest league.
30. ~~As a developer, I would like to be able to have multiple leagues running at the same time.~~
31. If time allows, as a user, I would like to be able to generate private leagues.
32. If time allows, as a user, I would like to be able to join private leagues.
33. If time allows, as a developer, I would like to be able to make the tokens tradeable on the Ethereum platform.

Non-functional Requirements

1. ~~As a developer, I would like to make the UI seamless and easy to use.~~
2. ~~As a developer, I would like to make the frontend responsive such that users can access it from their mobile phones.~~
3. ~~As a developer, I would like to have my backend host concurrent leagues.~~
4. As a developer, I would like to make sure that a player can join a league within 500 milliseconds.
5. As a developer, I would like to assign unique wallet addresses to the users to secure their tokens.
6. As a developer, I would like to prevent the users from seeing the portfolio of other users in the league until the league is over.
7. ~~As a developer, I would like to be able to host my frontend and backend separately.~~

8. As a developer, I would like to deploy on AWS after testing.
9. If time allows, as a user, I would like to protect my account with two factor authentication.