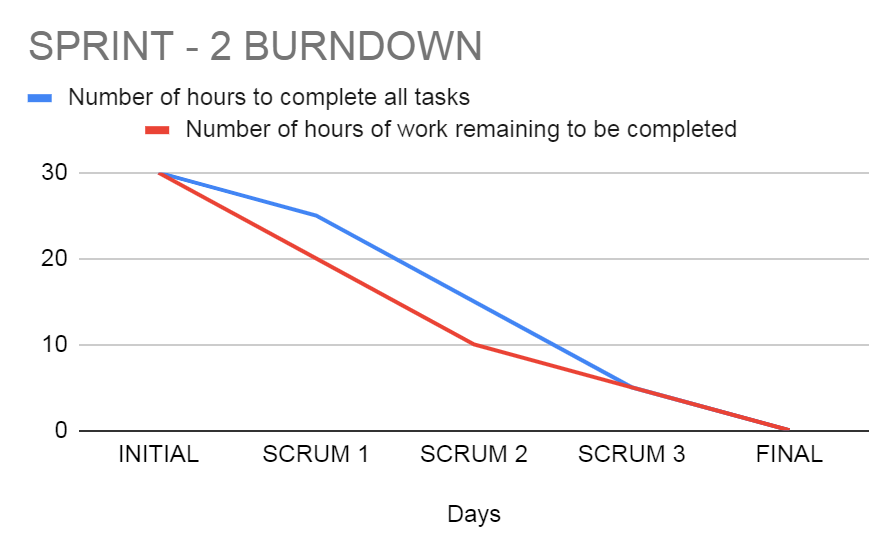
Retrospective document - Sprint 2

Sports Hub

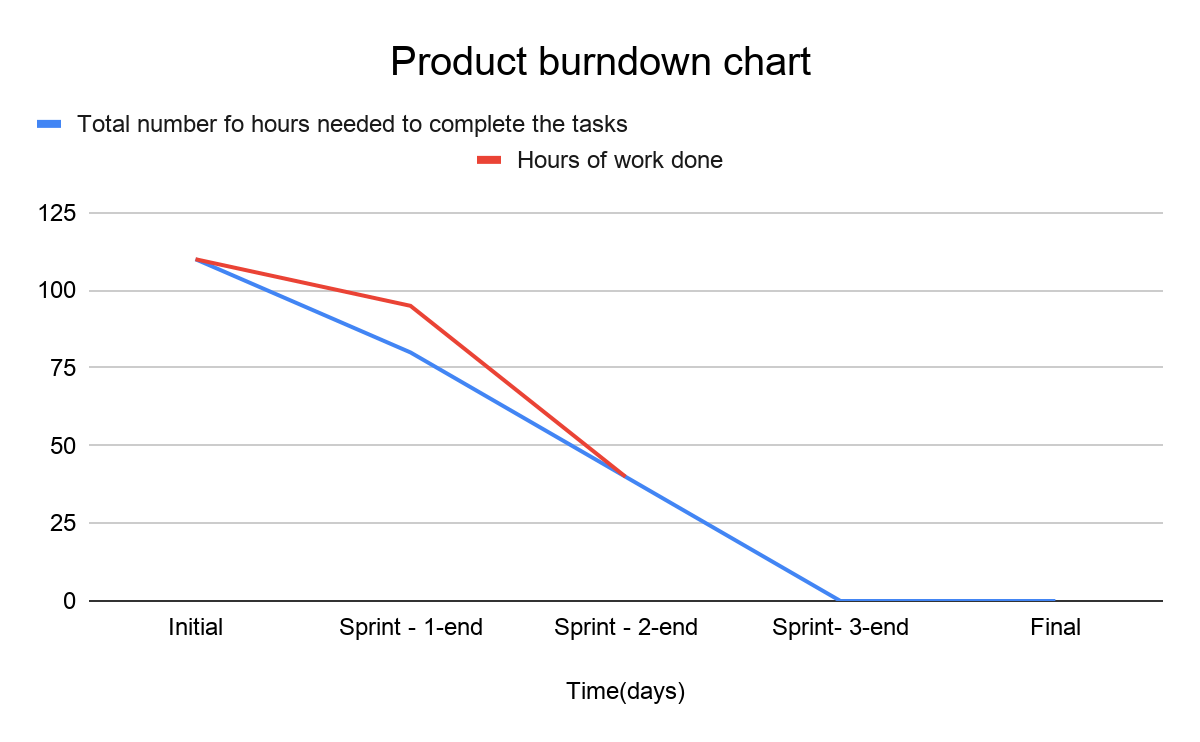
***Amarthya Sivakumar Annu, Austin Peterson, Yingjian Wang***

During this sprint we have almost completed our user interface and we have also almost completed our work with our first API. We have finished our home page, players page, upcoming games page. We only have a few small bugs to finish which we will be doing during our next spring. For the other two API’s we have finalized our ideas and they will be implemented during the next sprint. We have also got access to the other API’sThe other important part of this sprint was doing the user study and writing the user study report. The only changes we have made during this sprint is that we will be using seatgeek to display the number of seats available in a game instead of actually allowing the user to book tickets on our website. We believe that we have performed more effectively during this sprint and have made all the changes that we wanted to make from the previous sprint. We also believe that all three of us worked and contributed equally to the project. The only problem we faced during this sprint was that we had used all the queries that were available for free this month from the NBA API. Therefore we had to email the team and resolve the issue. Other than that this sprint went as per plan. As for now we also have no changes in our product backlog. I have also attached all the scrum meeting documents which gives detailed information on how the work was divided between the team during this sprint.

SPRINT - 2 BURNDOWN CHART:



PRODUCT BURN-DOWN CHART AFTER SPRINT-2



UPDATED PRODUCT BACKLOG

Green - All three members

Red- Amarthya Sivakumar Annu

Blue - Austin Peterson

Yellow - Yingjian Wang

---------------------------Initial Design---------------------------------

1. Brainstorm application ideas
2. Make a final design decision
3. Discuss requirements
4. Create product backlog
5. Create a burndown chart
6. Create a design document
7. Sketch initial design
8. Delegate jobs (writing reports, PM)
9. Design Sprint 1 backlog
10. Determine code methodology/language
11. Delegate backlog tasks
12. Make official design drawings
13. Complete SCRUM documentation for this sprint

--------------------------------Sprint 1----------------------------------

1. Brainstorm application ideas
2. Make a final design decision
3. Discuss requirements
4. Create product backlog
5. Create a burndown chart
6. Create a design document
7. Sketch initial design
8. Delegate jobs (writing reports, PM)
9. Design Sprint 1 backlog
10. Determine code methodology/language
11. Delegate backlog tasks
12. Make official design drawings
13. Determine website creation method (template, pure HTML/CSS, website element designer)
14. Planning for the user study
15. Coming up with user stories
16. Create the website files
17. Create site navbar
18. Create upcoming games section
19. Create a recent games section
20. Add league/sport pages
21. Add league/sport logos
22. Implement interactive aspects (scrolling, dynamic game block creation)
23. Search for the APIs
24. Choose the appropriate APIs
25. Format API return data
26. Link API data with website elements (game blocks)
27. Filter hot games for “home” page
28. Filter games by sport for sports pages
29. Get images for each game block
30. Design game block link page

--------------------------------Sprint 2----------------------------------

1. Turn in sprint 1 retrospective
2. Create sprint 2 backlog
3. Create sprint 2 initial burndown chart
4. Design game block link page
5. Continue to build product backlog
6. Link recap video (through API?)
7. Format game time data
8. Format game date data
9. Format teams involved data
10. Retrieve player data
11. Algorithm to show team rankings.
12. Create twitter feed website section
13. Learn twitter api
14. Get access to twitter API
15. Link twitter API
16. Filter the tweets.
17. Find function to fetch the feed.
18. Decide what to show
19. Decide on how to form hashtags to search.
20. Display the tweets.
21. Front end to display twitter feed.
22. practice/familiarize with javascript
23. Integrate javascript code into website
24. edit/develop javascript
25. Include previous game data if multiple games played in a series (fallback)
26. Format final score data
27. Format score throughout the game data (by quarter, inning, etc)
28. Add design elements - pictures
29. Add general team data
30. Include next game between the teams if possible (fallback)
31. Add key moments/highlights from the game
32. Add MVP data if possible (kind of the same as above)
33. Team color code
34. Complete SCRUM documentation for this sprint
35. Design user study
36. Start user testing
37. Collect user data/feedback
38. Complete user study report
39. Make appropriate changes based on user feedback
40. Learn seatgeek API
41. Link seatgeek API
42. Front-end for booking tickets
43. Display link to booking tickets
44. Try to make it possible to buy tickets on the applications itself
45. Is it possible to display booked seats?
46. Try to make it possible to choose seats.

--------------------------------Sprint 3----------------------------------

1. Conduct sprint - 2 retrospective survey
2. Finish sprint - 2 retrospective
3. Create sprint - 3 backlog
4. Find API’s for two other games
5. Front end for two other games
6. Create a new page that combines information from all three games
7. Incorporate information from the other two games also on the existing code.
8. Start filter by team process (search for a team, get data for their games)
9. Create a modified sport page for individual teams
10. Find team info API
11. Link team info with team page
12. Develop search function for teams
13. Make search results page
14. Build team player grid(?)
15. Finish team pages
16. Clean up site/make final design changes
17. Test links/pages.
18. Try to make the website mobile friendly.
19. Check and link work of all members.
20. Clean up.
21. Make project burndown chart
22. Make sprint burndown chart
23. Analyze stretch and backfall.
24. Create project presentation.
25. Finish team retrospective.