

TEAM SPROUT FINAL PROJECT REPORT

Submitted By:
Jayeeta Banerjee
Parvathy Mohan
Naga Suneela Gudise
Hanisha Patchineelam
Milton Clark
Sushmitha Batta

INSTRUCTOR: Prof. Liberty Page
Software Project Management - Fall 2018



University of
New Haven

Description

In this project, we implemented a web version of the two-player puzzle/game known as Sprouts. It was developed by two mathematicians at Cambridge in the 1960's and is well documented in a number of sites on the internet. [Wikipedia]

The game begins with a number of spots appearing on the board; the more the spots, the more difficult the game. In this implementation, the level of difficulty as easy translates into 3 initial spots on the board, and a difficult level translates into 5 spots.

Moves are taken by drawing a line between any two dots, but with each spot having a maximum of three lines adjacent to it. Each line cannot cross an existing line. After a line is drawn connecting two spots, a new spot is drawn anywhere along the line (preferably somewhere near the middle). This effectively splits the line into two lines and adds a spot with two lines intersecting it.

If the line forms a loop with both ends intersecting the same spot, each end contributes two ends to the line count for that spot.

The “move” alternates between the two players until there are no remaining spots with “unused” slots for a line end. When this occurs, the player making the last successful move wins the game.

There are a number of variations of this game, but these simple rules comprise the Sprouts game as implemented here.

Objective

The primary objective of implementing this game was not to build the game for the sake of the game, but rather to serve as a vehicle to utilize and experience first hand the Agile development process. In this case, the Agile methodology was followed, with sprints comprising the work units, scrums defining the framework within a spring, and the kanban/scrumban as the management framework providing visibility and flexibility needed throughout the process.

The secondary objective was to produce a product, a game, which could be used by professors in some of their classes. In this way, these professors acted as the end-user to define the requirements of the game and its environment. Game requirements are reflected in the prior description of the game, and additional requirements regarding the details of what should be displayed and when served to complete our understanding of what was needed in the final delivery of the game. It's possible that

Team Sprouts

this game could also serve as a learning tool for students. In fact, this game is actually a puzzle, and there are a number of truths/truisms that can become apparent after spending some time playing with it.

Perhaps a tertiary requirement was to stretch the technical skills of project members to acquire skills in technologies and delivery platforms which were unfamiliar up until this point.

Tools:

The following are the tools used in the project.

Tool	Description
Paper.js	Paper.js is an open source vector graphics scripting framework that runs on top of HTML5 Canvas. More details can be found in the link http://paperjs.org/
HTML5	HTML5 is the current major version of HTML standard. The new element <canvas> and SVG (scalable vector graphics) are used by Paper.js to create circles (spots/nodes) and paths (connection curves) for this project.
Brackets	Open source web development editor used for development of the code.
Heroku	Free cloud platform used to deploy the project.
GitHub	Software development platform used by team members for version control.
Trello	Project management tool used to assign tasks and monitor the task progress.
Safari browser in iPad	Browser and device used to perform testing as per customer requirement.

Technical architecture

Sprouts project is implemented using HTML5 and Paper.js. The first HTML is trial.html. This HTML page is automatically redirected to page2.html, which have text field options to enter player names, choose the level of the game and start button to play the game. Once the user clicks the 'Play me' button, the page is redirected to index.html, indexMedium.html or indexHard.html based on the level selected by the user. There is a JS file included for each of the index html files. They are app.js, appMedium.js and appHard.js. These javascript files contain the main logic of Sprouts. Paper.js is used to create circles and to create path objects for connecting the circles. Functions like onMouseDown(), onMouseUp() and onMouseDrag() are used to handle touch or mouse events. The following UML diagram represents the html files and the Javascript functions implemented for the project.

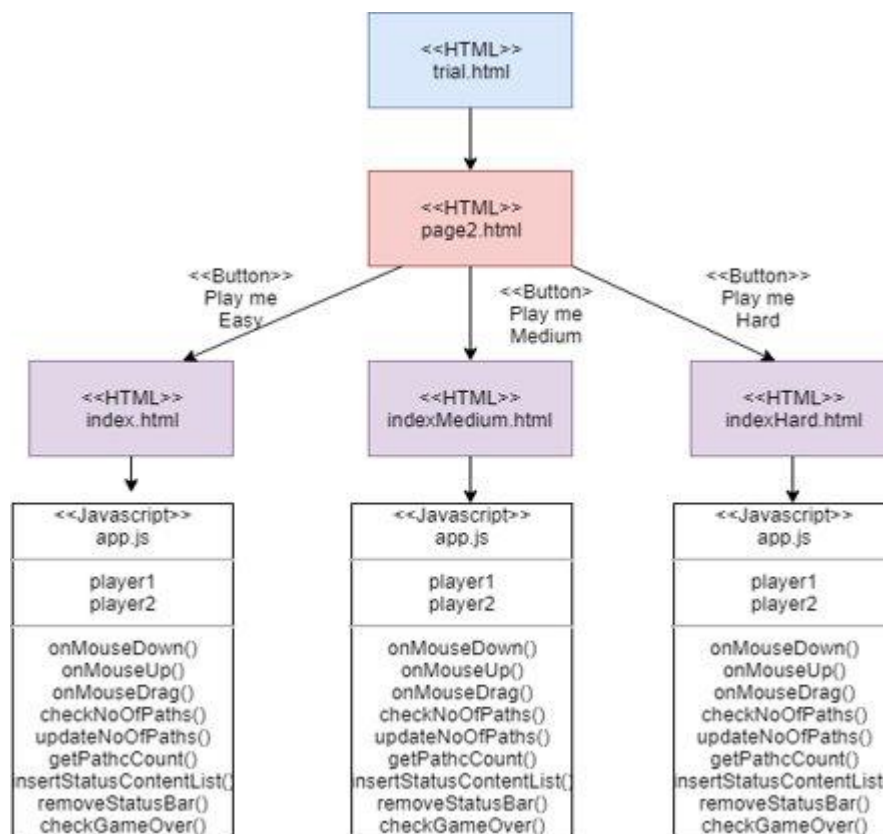


Figure: UML diagram

Test Design:

Test cases were designed and implemented based on the below user stories.

Tested User Stories	Description	Planned IN
4 User Stories	Login page for two players with start game button and game canvas screen with two initial red dots need to be created and integrated with graphics.	Sprint 1
2 User Stories	Player should access the game in safari with touch capability to draw a curve connecting 2 initial red dots.	Sprint 1
5 User Stories	Place only one dot on the line, curved lines should not cross each other, only allow 3 edges per dot, Initial dots position should be random	Sprint 2
4 User Stories	Curved lines of each player should be different color, dot should be highlighted once 3 edges were drawn, player's move status, play guide implementation	Sprint 2
4 User Stories	Implementation of medium and hard levels with page navigation, adding buttons for Play Again and Restart at the bottom of the game canvas	Sprint 3
3 User Stories	Play move updates on the right side of the panel, Display the player's name on the home screen to the game canvas, Implement the end of game and proclaim winner	Sprint 3

Test Data:

Below link was used to get the code and execute it based on user stories.

2) GitHub Link: <https://github.com/parvathycec/sprouts>

Test Results from system testing:

- **Sprint1:**

Page Navigation and Integration of game canvas with login screen. (Pass)

Able to enter player names and click play game button. (Pass)

To view two initial red dots and connect them with a curved line. (Pass)

User should be able to draw curves using touch ability. (Pass)

- **Sprint 2:**

Play guide to view game rules and restrictions. (Pass)

Adding player names from login screen to game canvas screen. (Pass)

To display a quote mentioning which player's turn. (Pass)

No two curved lines should be crossing each other. (Pass)

Only 3 edges per dot. (Pass)

Highlight dot in color once the three edges were drawn. (Failed and carried forward)

Able to draw self-loops (infinite loop). (Pass but not a project requirement)

Able to put only one dot in the middle of line. (Pass but misunderstood requirement carried forward)

Distinguish player's dots and curved lines with different colors (Pass)

Database implementation for player's score (Carried forward)

Initial position of the dots to random. (Failed carried forward)

- **Sprint 3:**

Elimination of self-loop. (Pass)

Put a dot in the middle of line as soon as curve is drawn. (Pass)

Implementation of game with medium [3 dots] and hard [4 dots] levels. (Pass)

Play again button at the end of game. (Pass)

Initial dots should be random when the game begins. (Pass)

Highlight dot in color once the three edges were drawn. (Pass)

Prompt to place a middle dot once the curve is drawn. (Pass)

Drawing a curved line and placing a dot should be considered as single turn. (Pass)

Turns of the player's and winner should be declared with score. (Pass)

Database implementation for players scores. (moved to backlog)

Customer Satisfaction

The team regularly met with the customer and got regular feedbacks. Customer is happy with the application. The details of the code and cloud link is delivered to the customer via email. An screenshot copy of that email is attached.

Team Sprouts

Team Sprout



Banerjee, Jayeeta <jbane2@unh.newhaven.edu>

6:50 PM



To: Fischer, Alice



TeamSprouts.docx
13.14 KB

Hello Dr. Fischer,

This is Jayeeta Banerjee from Team Sprouts. Our game is successfully deployed. You can find more details in this attached documents.

Thank you.

Best Regards,

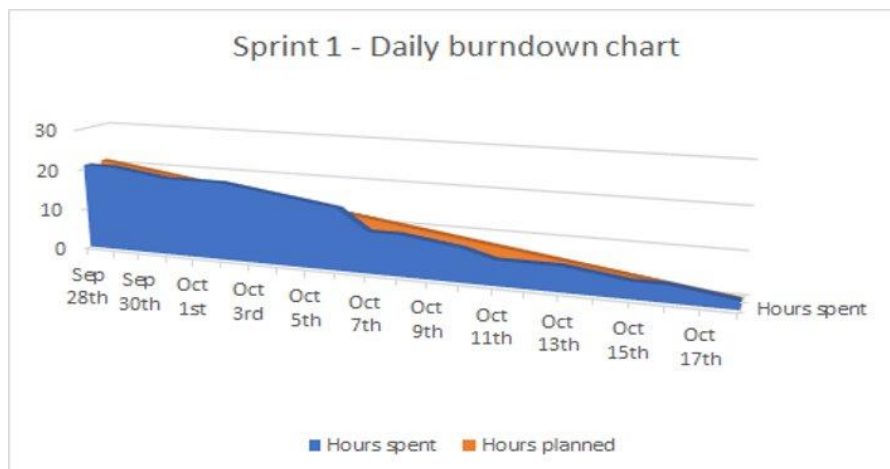
Jayeeta Banerjee

Sent from [Mail](#) for Windows 10

Sprint Retrospective - Project Artifacts

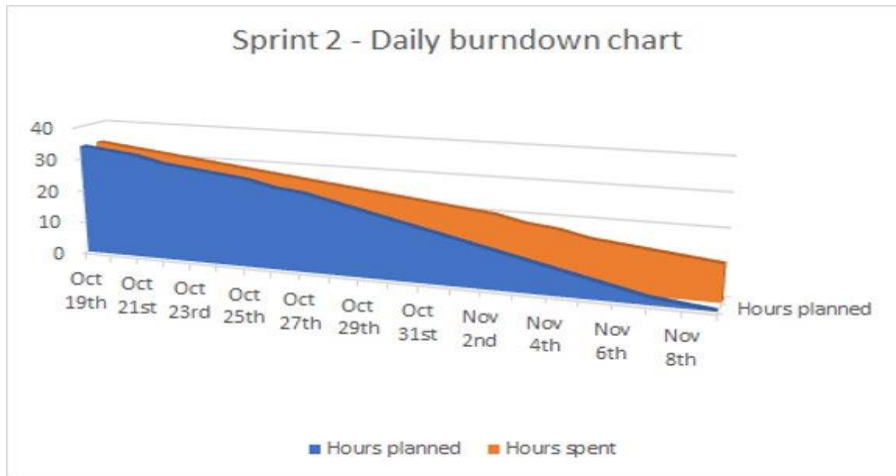
Sprints Burndown Charts:

- Sprint1:

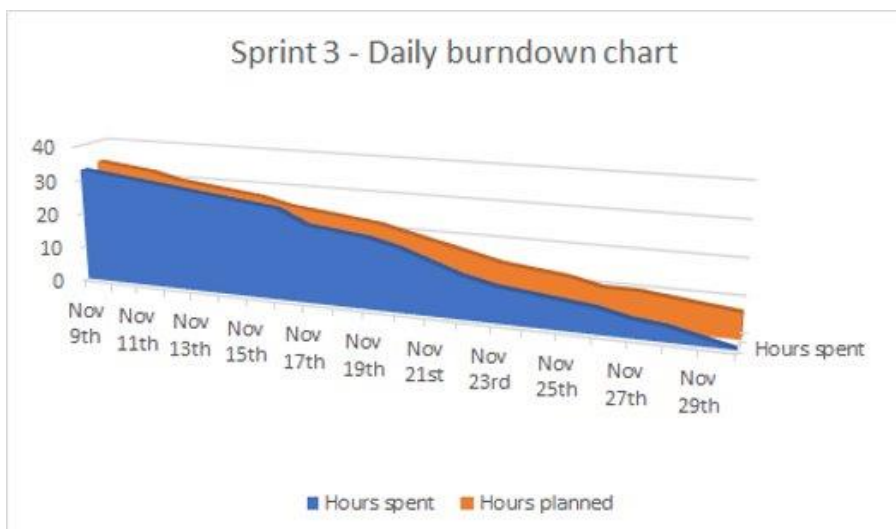


Team Sprouts

▪ Sprint2:

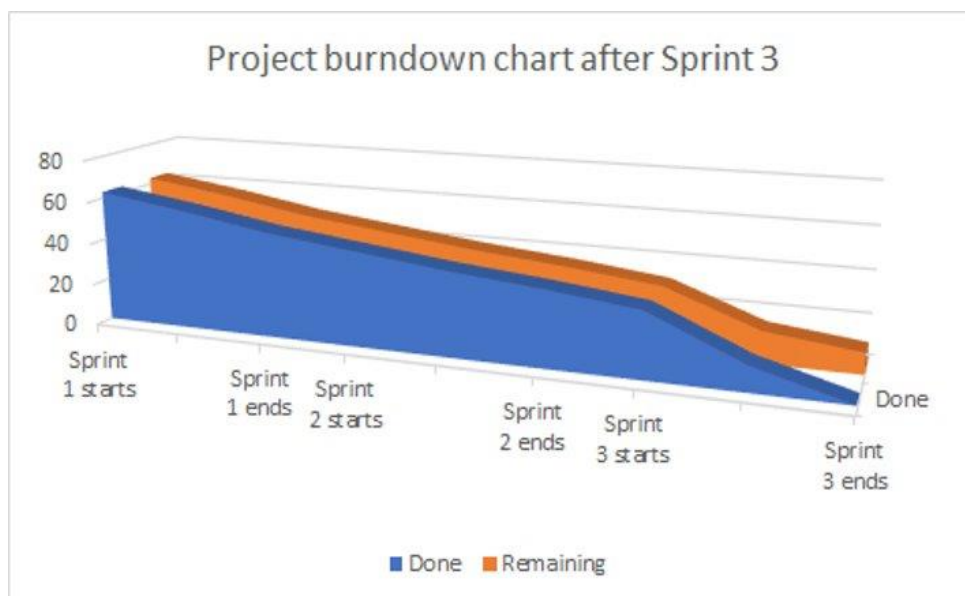


▪ Sprint3:



Team Sprouts

Project Burndown Chart:



Product Backlog:

Product Backlog of Game Sprout						
ID	AS a...	I want to be able to...	Priority	Sprint	Status	
1	User	View two dots when I start the game	1	Sprint1	Completed	
2	User	Connect two dot with curve	1	Sprint1	Completed	
3	User	Open the game in safari	1	Sprint1	Completed	
4	User	login with my Id	1	Sprint1	Completed	
5	User	Play touch screen	1	Sprint1	Completed	
6	User	View the highlight on dots with 3 edges	2	Sprint2	Completed	
7	User	Place a dot in the middle of the curve	2	Sprint2	Completed	
8	User	distinguish clearly between the participants lines based on color	2	Sprint2	Completed	
9	User	Not cross any curves in the canvas	3	Sprint2	Completed	
10	User	Enter the participant names	3	Sprint2	Completed	
11	User	View the score	3	Sprint3	Incomplete	
12	User	End the game when all possibles moves are completed	3	Sprint3	Completed	
13	User	Count Lines	4	Sprint3	Completed	
14	User	View the winner's name once the game ends	4	Sprint3	Completed	
15	User	Start the game and play with three dots	2	Sprint3	Completed	

Estimates:

- **Sprint1:** The total estimate made for Sprint 1 is 21 story points.
- **Sprint2:** The total estimate made for Sprint 2 is 32 story points.
- **Sprint3:** The total estimate made for Sprint 3 is 31 story points.

Work Log:

ID	User Story	Priority	Schedule	Assignee	Total Hrs
1	View two dots when I start the game	1	Sprint1	Jayeeta, Parvathy	4
2	Connect two dot with curve	1	Sprint1	Jayeeta, Parvathy	3
3	Open the game in safari	1	Sprint1	Hanisha, Suneela	2
4	login with my Id	1	Sprint1	Sushmitha, Hanisha	4
5	Play touch screen	1	Sprint1	Parvathy	2
6	View the highlight on dots with 3 edges	2	Sprint2	Parvathy	4
7	Place a dot in the middle of the curve	2	Sprint2	Jayeeta	4
8	distinguish clearly between the participants lines based on color	2	Sprint2	Parvathy	3
9	Not cross any curves in the canvas	3	Sprint2	Jayeeta	4
10	Enter the participant names	3	Sprint2	Sushmitha	3
11	View the score	3	Sprint3	Sushmitha, Parvathy	5
12	End the game when all possibles moves are completed	3	Sprint3	Parvathy	4
13	Count Lines	4	Sprint3	Jayeeta	5
14	View the winner's name once the game ends	4	Sprint3	Parvathy	3
15	Start the game and play with thee dots and four dots	2	Sprint3	Jayeeta	4
16	Only three edges are allowed per dot	3	Sprint2	Jayeeta	4
17	Players status	2	Sprint2	Parvathy	3
18	Put a self loop	2	Sprint2	Jayeeta	3
19	Page navigation based on level	1	Sprint3	Hanisha	4
20	Game start animation	1	Sprint2	Parvathy	1
21	Deployment in heroku	2	Sprint3	Jayeeta	1
22	Testing	3	Sprint3	Jayeeta	5
23	Database Implementation	2	Sprint3	Milton	5
24	traceability	2	Sprint3	Milton	5

Completed Items:

Sprint1:

- Planning and designing
- Learn new Technology(paper.js)
- Login page and Welcome screen
- Iteration of canvas screen to welcome screen
- Generate two red dots in the canvas
- Draw a curve connecting the two dots
- Test Feasibility of touch capability in Safari.
- Graphics and design

Sprint2:

- Each player's curve lines are different color.
- Allow a new dot near middle of each curve.
- Display the players' name in game canvas
- Only one middle dot per curve. Remove if put another
- Allow three edges per dot
- The curves should not cross. Remove while crossing
- Play guide implementation
- Status of players' as showing 'player1's move.

Sprint3:

- Game over and declaring the winner.
- Self-loop count and fix infinite loops
- Highlighting a spot when there are three connections to it.
- Three levels of game with two, three and four spots.

Team Sprouts

- Display the status of the player in the right side of the canvas.
- Test plan.
- Page navigation.

Incomplete Items:

Sprint1:

- Planning Database for players and score cards.

Sprint2:

- Implement Hard level with three spots.
- Bug fixing of random dot position.
- Bug fixing on highlight.
- Database implementation of players' score.

Sprint3:

- Database implementation of players' score.

New items added:

Sprint1:

- Database implementation for the whole project.
- Added a login page to enter names of players.
- Adding graphics to loading screen.

Sprint2:

- A small game start animation.
- Only one middle dot per curve.
- Random dot position before it was fixed.
- Allow only three edges per dot.
- Play guide implementation.

Sprint3:

- Current player's turn ends only after the player puts a spot on the curve that was just made.
- This is a case of misunderstanding of the user case.
- Changes made as per customer feedback.
- Player move updates in the right side of the canvas.
- Level-wise implementation for hard, medium and easy.
- Self-loop calculation.
- Play again button to restart the game.

Issues and Solutions:

• Sprint1:

Issues faced:

- Joining dot with curve - initially curve is not touching dots
- In touch screen mode, curve is getting removed after one touch event.
- Hosting in safari.

Solutions:

- By creating new path object while defining new curve

Team Sprouts

- Condition to check if path exists or not
- Heroku cloud hosting

- **Sprint2:**

Issues faced:

- Finding Co-ordinates on paths & dots
- Calculating number of edges
- Showing the players' name in canvas
- Random dot position
- Bug fixing

Solutions:

- Use log to print the position and alert method.
- Use console of console.log to verify
- Using 'location' attribute of paper.js
- Using proper range of random co-ordinates
- Manual testing and task moved to Sprint3

- **Sprint3:**

Issues faced:

- Misunderstanding of the requirement.

Solutions:

- Reworked for few tasks.

Lesson Learned:

Sprint1:

- Experienced developers set a good velocity.
- Use of Trello helped in assigning tasks and viewing the status of tasks.
- Deploying the code to Heroku helped in testing and demo purposes.

Sprint2:

- Re-evaluated velocity based on wishful thinking.
- Plan next sprint based on a sustainable velocity.
- Unit testing for complex problems needed a better test plan.
- Complex bugs identified as new tasks and estimates are made based on Fibonacci Series.
- Difficult to get everyone together physically for team meetings.

Sprint3:

- Needed a more regular client-developer meetings to avoid misunderstanding and rework.
- Developer should always take the latest before making updates.
- Planned the velocity (put a margin) based on the factors like holidays.
- More soapbox

Reflection on the project

Customer product:

All the requirements by the customer was implemented. The main requirement by the customer was to have a touch-enabled product that can be accessed online. This is ensured by testing in iPad using touch and deploying in the cloud, so that any user can directly access it. We, as a team, have also added value to the product by going one step ahead and making it more user-friendly with instructions and display event texts. One of the team members have tested the application thoroughly to ensure usability.

Team structure:

Most of the times, a democratic project management structure was followed. However, there were a few instances, where one or two team mate(s) acted as a chief programmer to increase the velocity of the project. At the beginning of the project, pair-to-pair programming was adopted, because Paper.js was a new platform for all the teammates. Pair-to-pair programming was a good change, in the sense that it reduced bugs.

The most efficient team structure followed was democratic. It reduces ego clashes and multiple people taking responsibility for a task.

Agile Sprint:

The SPM method adopted by the team was Sprint. This was a very effective process, as it was iterative and regular customer feedback avoided rework at the end. As a team, we would continue with this method for software implementations.

Sprint Meetings Timeline

SPRINTS	REVIEW	RETROSPECTIVE	REMARKS
SPRINT1	10/25/2018, 10/23/2018	10/26/2018	Good velocity, accomplished all task
SPRINT2	11/07/2018, 11/05/2018	11/08/2018	Few stories added to sprint3
SPRINT3	12/05/2018, 12/03/2018	12/06/2018	Project successfully deployed, customer is satisfied

Customer Meeting Timeline:

SPRINTS	DATE	REMARKS
SPRINT1	10/09/2018, 10/13/2018	Explained requirements
SPRINT2	NA	NA
SPRINT3	11/29/2018, 12/04/2018	Few bug fixes, overall satisfied

Deployment:

Heroku link: <https://sproutgame.herokuapp.com>

GitHub Link: <https://github.com/parvathycec/sprouts>

SPRINTS	DATE
SPRINT1	10/26/2018
SPRINT2	11/08/2018
SPRINT3	12/06/2018

Conclusion:

The team used agile concepts to successfully manage the implementation of Sprouts application. The iterative approach of agile and the regular feedback from the customer after each Sprint, ensured that the quality of the product matched the customer expectations. The customer is very happy with the final product. The product development using software project management techniques were very useful in learning the practical aspects of team management. It can be concluded that the experiential learning achieved by the team by implementing this project shows that the key to success of software project management is people, process and product.