

Minesweeper Task Tracker

Created on 8/29 by A.W.I.

Requirement ID	Description of Requirement	Status	Story Points	Priority	Sprint No.
1	Create the game's core backend, and logic	Done	5	5	1
1.1	Implement 10x10 board data structure with row/col labels	Done	2	3	1
1.2	Implement mine placement (user specified, 10-20, safe first click)	Done	3	2	1
1.3	Implement uncover cell logic with recursive reveal	Done	3	4	1
1.4	Implement flag toggling (covered to flagged, vice versa)	Done	2	2	1
1.5	Implement win/loss detection, with conditions	Done	2	3	1
2	Create the game's user interface	Done	3	5	1
2.1	Render the grid UI (covered/uncovered/flagged cells)	Done	3	4	1
2.2	Display mine and flag counters dynamically	Done	2	2	1
2.3	Add status indicator ("Playing", "Win", "Loss")	Done	1	1	1
2.4	Handle user input (clicks/keypresses should uncover/flag)	Done	1	3	1
2.5	Reveal all mines on loss	Done	1	1	1
3	Create the system architecture and documentation	Done	2	5	1
3.1	Draft system architecture diagram	Done	2	4	0
3.2	Write architecture overview (components and data flow)	Done	3	4	0
3.3	Document assumptions (fixed grid, mine count, etc.)	Done	3	2	0
3.4	Write prologue comments in all files	Done	3	2	1
3.5	Add inline comments explaining major code blocks	Done	2	2	1
4	Create project management and process documentation	Done	3	4	0
4.1	Estimate person-hours with clear methodology	Done	2	3	0
4.2	Track actual person-hours (day-by-day)	Done	3	2	1
4.3	Set up GitHub repo and branching workflow	Done	1	1	1
4.4	Weekly demos with GTA/team	Done	2	3	1
5	Create testing and validation methods	Done	3	5	1
5.1	Unit test mine placement	Done	2	4	1

Minesweeper Task Tracker

Created on 8/29 by A.W.I.

Requirement ID	Description of Requirement	Status	Story Points	Priority	Sprint No.
5.2	Unit test uncovering logic	Done	2	4	1
5.3	Stress test for crashes/abnormalities in multiple instances	Done	2	2	1
5.4	Find a user to test UI intuitiveness	Done	1	1	1
6	Create our submission and relevant artifacts	Done	3	1	1
6.1	Code freeze on master branch by due date	Done	1	1	
6.2	Prepare final demo build	Done	2	1	1
6.3	Submit peer evaluations on Canvas	Done	2	1	
6.4	Verify attribution of external sources	Done	1	1	1