

Name	#	Total hours	Tt Contributions	Date
Brandon Dodge		1	Inherited Project Application Testing	9/27/25
Team		1	Team Meeting	10/3
Brandon Dodge		3	Code Comments made to PysolverAI	10/4
Brandon Dodge		1	Documentation Changes to Adjust for Part 2	10/4
Asa Maker		0.75	Environment setup and code review: Cloned the forked repository, configured dependencies, and verified the game ran. Setup was faster than estimated due to familiarity with the original codebase.	9/28/25
Asa Maker		2.25	Implemented Hint feature core logic (safe cell reveal): Wrote the Hint system to select and reveal a safe cell. Required extra time testing logic to ensure it didn't trigger mine cells. Debugged minor index errors.	10/2/25
Asa Maker		1.25	Integrated Hint feature with GUI and limited usage: Added the Hint button, connected event handling, and restricted use to 3 per game. Integration went smoothly, with minimal UI bugs.	10/2/25
Asa Maker		2.5	Assisted in AI solver debugging (Medium difficulty): Verified solver logic on Medium difficulty and adjusted flagged neighbor counts. Time aligned with estimates.	10/2/25
Asa Maker		1.75	Playtesting and bug fixing: Ran multiple test sessions to confirm Hint and AI features worked together without crashes. Fixed small display timing issues and verified game resets.	10/2/25
Asa Maker		##	Contributions	
Zach Severt		2	Implement Easy/Hard AI solver for minesweeper. Partially contributed to medium AI difficulty. Easy and Hard implemented quicker because its random selection versus knowing all cell contents respectively	
Zach Severt		0.5	Github Management. Periodically checked files to ensure they were correct, and that no pushes or merges affected this.	
Josh Dwoskin		1.5	Updated the system architecture to align with our hint addition	
Josh Dwoskin		1	Created a new class diagram and added it to the system diagram document	
		##	Contributions	
		##	Contributions	
		##	Contributions	

Name	Total hours	T _T Contributions	Methodology	Date
Brandon Dodge	1	Original Code and Application Testing	Thoroughly tested the functionalities and capabilities of the project that our group has enherited.	9/27/25
Brandon Dodge	3	Comment Part 2 Coding	Used to better understand and describe the new code implemented into this preexisting product.	10/3
Team	1	Team Meeting	Time used for the entire team to gather and construct a projection for the entire project and recap all work that has already been done.	10/3
Brandon Dodge	1	Documentation Changes	Need to adjust the current documentation such as the readme to account for the new members who will be developing part 2 of this inherited project.	10/4
Asa Maker	1	Environment Setup & Code Review	Setting up the forked repository, verifying dependencies, and reviewing major modules typically takes about one hour for an experienced developer. The inherited project is known, reducing ramp-up time.	9/28/25
Asa Maker	2	Implement core Hint feature logic (safe cell reveal)	This involves analyzing how the game state is stored, adding logic to find a safe cell, and ensuring no interference with AI or game state transitions. Based on similar feature additions, 2 hours is appropriate for planning and implementation.	10/2/25
Asa Maker	1.5	Integrate Hint feature with GUI and limit usage	Adding GUI controls (e.g., button event, counter display) generally takes 1–2 hours, accounting for minor debugging. The estimate assumes familiarity with the UI event system and small-scale testing cycles.	10/2/25
Asa Maker	3	Assist in AI solver refinement and debugging (Medium difficulty)	Involves targeted testing rather than building from scratch. Based on expected need for console output analysis and small logic adjustments, 3 hours should suffice.	10/2/25
Asa Maker	1.5	Playtesting and fixing runtime issues	Final validation through full game runs. Estimate includes identifying minor bugs and ensuring Hint + AI work correctly under gameplay conditions.	10/2/25
Asa Maker	##	Contributions		
Asa Maker	##	Contributions		
Zach Severt	1.5	Easy/Hard AI	Implement Easy/Hard AI mode for minesweeper, easy will use random selection, and hard will "cheat" by knowing safe versus dangerous cells	10/3
Zach Severt	1	Github management	Verify correctness of github contents. This was an issue for our last project. Confirm that all files are up to date and ready for submission, and no pushes or merges are performed that affect this.	9/30 - 10/5
Josh Dwoskin	1	Class Diagram	Created a new class diagram to reflect our groups changes	10/2
Josh Dwoskin	2	System Architecture	Updated the system architecture with new hint addition	10/5
	##	Contributions		