Reference Story Number	Pool	Story Source	Story Summary
1	1 Points	EECS 581: Project 2	Minesweeper Easy AI: Simulated an simple AI assistance using Python for Minesweeper that randomly uncovered cells and takes turns with the user to complete the game.
2	1 Points	EECS 581: Project 1	Minesweeper Setup: Initialized Minesweeper using Python and PyGames data structures and components to lay a foundation for game implementation, by using pygames to create the basic grid UI layout.
3	1 Points	EECS 581: Project 1	GitHub Setup: Created a comprehensive GitHub README page highlighting project purpose, features, installation steps, usage examples, and contribution guidelines to enhance clarity, engagement, and collaboration.
4	1 Points	SELF Enterprise	Business Proposal Pitch: Created rigorous documentation that defined the scope, timeline, budget, and motivation for the software project. This acted as an anchor for the rest of the phases in creating the end-product.
5	2 Points	EECS 678: Quash Project	I/O Redirection Part 1: Implemented input redirection in C using the < operator, enabling programs to read data directly from files instead of standard input, improving automation and flexibility.
6	2 Points	EECS 678: Quash Project	I/O Redirection Part 2: Implemented output redirection in C using the > operator, allowing program output to be written directly to a file, overwriting existing content for streamlined data management.
7	2 Points	EECS 678: Quash Project	I/O Redirection Part 3: Implemented output redirection in C using the >> operator, enabling programs to append output to existing files without overwriting content, enhancing logging, persistence, and data tracking.
8	2 Points	Hackathon 2024: Wishmate	Embedded AI into UI Framework: Integrated the Amazon AWS Rekognition API to take pictures of items with the Kivy UI application to upload the pictures and store them in MongoDB.
9	2 Points	EECS 581: Project 1	Code Optimization: Refactored code to ensure appropriate and optimized logic which helped improve code efficiency and runtime, such as deleting unecessary variables and creating reusuable functions.
10	2 Points	EECS 348: Final Project	Documentation: Created the project plan, software requirements, software architecture, test cases, user manual, and project glossary documents for the arithmetic calculator to ensure all phases were properly recorded and used as reference in later steps.

Reference Story Number	Pool	Story Source	Story Summary
11	3 Points	Hackathon 2024: Wishmate Project	MongoDB Database: Created a database with multiple collections to store user information and allow efficient querying of data. Integregated to web app using Python scripts.
12	3 Points	EECS 348: Final Project	Quality Assurance Testing: Developed over 60 comprehensive test cases to validate all possible algebraic expressions, including edge cases, syntax errors, and complex user inputs, ensuring accuracy and reliability.
13	3 Points	EECS 348: Final Project	Parsing the Input for the Calculator: Programmed functions to handle various different cases of user input for the calculator, including both bad and good input, by parsing the string input and tokenizing each into a data structure to pass into other files in the project.
14	3 Points	SELF Community Service Project (CSP)	Subscription Management Database: Utilized PostgreSQL to automate user data updates, enabling the generation of timely notifications and ensuring real-time communication, accuracy, and efficiency across the system.
15	5 Points	EECS 447: Semester Project	Database Architecture: Designed database architecture using ER diagrams and logical models to define relations and entities that helped model a library system. It allowed for different tiers of members, creating accounts, renting and returning items, monitoring status of books, and charging accounts for overdue fees.
16	5 Points	EECS 581: Project 2	Minesweeper Hard AI: Simulated a sophisticated AI assistance using Python for Minesweeper than strategically uncovered cells, mimicking human intelligence. It flags mine cells, uncovered safe cells, and identified 1-2-1 patterns to make moves.
17	5 Points	EECS 447: Semester Project	Creating Queries: Designed a set of SQL queries for the Library Database Management System to handle core operations such as inserting new books, updating availability after checkouts, searching records by title or author, joining user and book tables to track issued items, etc. while ensuring accurate data retrieval process!
18	5 Points	SELF Community Service Project (CSP)	Subscription Management React App: Developed a full-stack subscription management module using React as the frontend framework, enabling dynamic user interactions, efficient state management, and seamless integration with backend services.

Reference Story Number	Pool	Story Source	Story Summary
19	8 Points	Personal Project: Heartland Linguistics	Cutomer Centered UI: Developed a website for a small business based on customer feedback and requirements. Highlighted the main critical parts of the website by using animations and graphics. Added navgation bar features and home page to help aquire customer attention, using TailwindCSS.
20	8 Points	EECS 581: Project 2	Minesweeper Medium AI: Simulated a medium AI assistance using Python for Minesweeper that had some intelligence. It flags mine cells by calculating the number of covered adjacent cells and comparing it with the cell's number. It then uncover cells remaining cells that were not flagged, indicating a safe cell.
21	8 Points	Hackathon 2024: TraWell	Website UI: Integrated new styles for the frontend of the website to help enhance user experience. Utilizes tailwind CSS to incorporate features such as responsive layouts, smooth hover transitions, modern card components, adaptive color schemes, and consistent spacing.
22	8 Points	EECS 678: Quash Project	Run Executables: Enabled the execution of programs with command-line parameters, searching the directories in the PATH variable if no absolute or relative path is given.
23	13 Points	Hackathon 2024: Wishmate	AWS AI Implementation: Utilized AWS Image Rekognition API along side Open CV to create an interaction feature for Wishmate. It allows users to take pictures and analyze the objects in the picture. Webscraping was used with this feature to look for associated Amazon links for the objects in the picture.
24	13 Points	San Francisco Hackathon	AI Land Recommender: Built an AI-driven recommendation engine that analyzes user preferences and farmland data to suggest optimal plots for fractional ownership on the Flow blockchain.
25	13 Points	Hackathon 2025: DreamDragon Project	Interactive UI: Created a interactive UI using HTML, CSS, and JavaScript for all parts of the web app, including the game component. This required designed graphics from scratch, specific for each page. Later, it was integrated into our Flask app.
26	13 Points	EECS 678: Quash Project	Foreground and Background Processes: Quash supports both foreground and background execution, where & indicates a background job. When a background job starts or completes, Quash displays messages showing the job ID, process ID, and command.