Prompt Used
Create a single-page archery game using HTML, CSS, and JavaScript. The game should include a bow, arrow, and moving target created using div elements. The player shoots arrows by clicking, and the arrow moves straight from the bow's current position. The target moves continuously to increase difficulty. Implement a score system, timer, a goal for winning, bow movement with keyboard arrows, and a restart button. The game should have smooth animations, collision detection based on the arrow tip, and a dynamic difficulty system.

Analysis and Comparison
The AI-generated version of the game included detailed graphics, gradient effects, animated arrows, and a full game-over overlay with accuracy tracking. This made it visually polished and feature-rich but also added complexity and extra logic for arrow animation and collision detection.

My version focuses on a **minimal yet functional design**, keeping the bow, arrow, and target as images with simple rotation and straight-line shooting. It ensures:

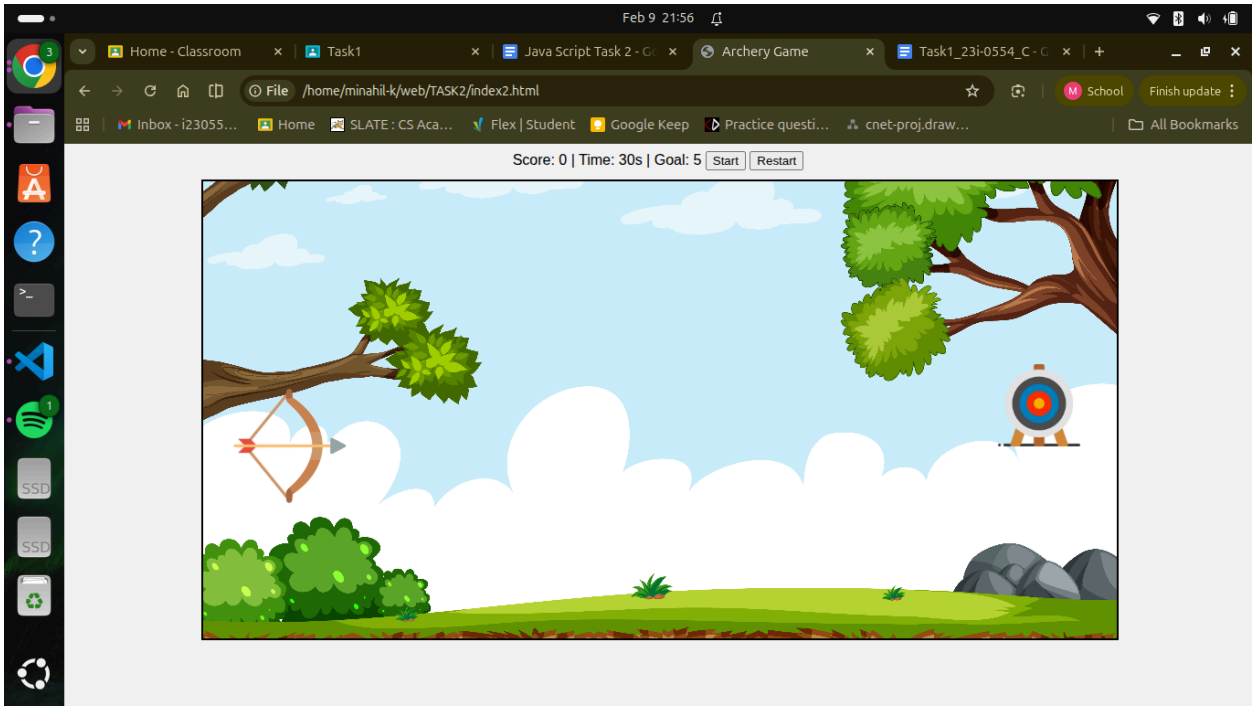
- Only one arrow can be active at a time.
- Arrow shoots from the current bow position.
- Collision detection is based on the **arrow tip** for accuracy.
- Score, goal, and game-over logic are clear.
- Bow can move up and down with arrow keys.
- Restart functionality works without reloading the page.

Through this comparison, I learned that **simplifying mechanics while keeping gameplay accurate** improves reliability and maintainability, whereas AI-generated versions can prioritize visuals and complex animations over lightweight code and academic clarity.

Git Link:
https://github.com/mina-hill/WebPTask2

My Version

- Minimal design for easy understanding and modification.
- Straightforward arrow movement and collision detection.
- Clear score, goal, and timer handling.
- Single active arrow logic prevents multiple shots issues.
- Keyboard control for bow adds interactive gameplay.



AI Version

- Visually rich with gradient backgrounds, animated arrows, and target.
- Includes accuracy percentage calculation and game-over overlay.
- More complex logic for arrow animation using `requestAnimationFrame`.
- Difficulty levels dynamically change target size and speed.