## Planning

**Game theme:**
“Rocket Survival Challenge” is a fast-paced survival game where players need to control a rocket through dangerous meteorites in the universe while trying to survive for 30 seconds to land successfully. The game combines exciting evasion mechanisms with unique landing challenges, designed to test players' reaction ability and patience.

**Storyboard:**
Scene 1: Startup interface
Screen: Cosmic background, floating stars, and the central text shows "Press the space bar to start the game".
Player action: Press the "space bar" to start the game.

Scene 2: Game process
Screen:
The rocket is in the center of the screen, and the player controls the rocket to move horizontally.
Randomly generate meteorites moving from the top to the bottom.
The star background moves dynamically to add a sense of speed.
The "survival time" is displayed in the upper left corner.
Player goal: Move the rocket with the left and right keys to avoid meteorites.

Scene 3: Victory landing
Condition: After surviving for 30 seconds, the rocket automatically descends smoothly.
Screen: The rocket gradually rotates to adjust its direction and lands on the ground. The screen displays "Successful landing".

Scene 4: Failure interface
Condition: The rocket collides with the meteorite.
Screen: Rocket explosion special effects, the screen displays "Game over! Press R to restart".

**Timeline:**
Week 1: Conceptualization and Design
11/3: Brainstorm game theme and mechanics. Decide on a survival-based rocket landing game.
11/5: Outline gameplay objectives, controls, and progression (e.g., survive 30 seconds, land safely).

Week 2: Development
11/7: Design player assets (rocket), background (stars), and obstacles

(asteroids). Start coding the game environment using p5.js.
11/10: Implement rocket movement, asteroid generation, and collision logic.

Week 3: Integration and Testing
11/13: Add survival timer, landing mechanics, and end states (win/loss). Test gameplay for bugs.
11/15: Refine visuals and animations, ensuring smooth transitions and immersive experience.

Week 4: Finalization and Submission
11/18: Polish UI/UX, finalize game instructions, and ensure compatibility in different environments.
11/20: Submit completed game with documentation.