Sprints

- Sprint 1 (Oct 20 - Nov 3)
    - Basic functionality (movability, interactive UI, basic game layout), working navigation to different sections of the game
    - Core systems & prototype
- Sprint 2 (Nov 3 - Nov 17)
    - Data management, rough implementation of games
- Sprint 3 (Nov 17 - Dec 1)
    - Nice to have but not necessary features (leaderboard, help button, etc), fine-tuning of games

User Stories

Game Functionality (Sydney)

- As a user, I want to be able to move around so that I am able to explore the map (4 hours)
    - Design an alien graphic which will function as the main character sprite
        - 2 hours
    - Design a character class which takes in specific attributes - character type, movement speed, etc - and has event listeners to implement a WASD system that will move a character object around
        - 2 hours
    - Write tests
        - 1 hour
    - Pass tests
        - 1 hour
- As a user, I want to be able to get help to learn the topics I don't already understand/need to work on (11 hours)
    - Design a help button graphic which will function as a “Need Help” button
        - 1 hour
    - Design a system for a popup help button
        - 2 hours
    - Create video or graphics for each class of problem
        - 5-8 hours
    - Write tests
        - 1 hour
    - Pass tests
        - 1 hour

- As a user, I want to be clearly informed when certain games or activities (like rapid fire math) cannot be redone, so that I can understand when incorrect inputs have consequences and don't accidentally miss important opportunities (4 hours)
    - Create a popup graphic for non-repeatable activities
        - 2 hours
    - Add the popup graphic to the game where needed
        - 2 hours
- As a user, I want to be prompted when minigames happen and what they are so that I can be prepared when it comes (4 hours)
    - Create a popup graphic for incoming games/activities
        - 2 hours
    - Add the popup graphic to the game where needed
        - 2 hours
    - Write tests
        - 1 hour
    - Pass tests
        - 1 hour
- As a user, I want to be able to **pause/resume/go back to the main menu/see the current process** so I can gain control of my progress (3 hours)
    - Implement an event listener for a pause/menu pop up
        - 1 hour
    - Design a menu pop up graphic
        - 2 hours
    - Write tests
        - 1 hour
    - Pass tests
        - 1 hour

Game State (Ethan)

Prerequisite: Data management is set up

Review mistakes

- As a user, I want to review the questions I got wrong with correct answers shown so I can understand my mistakes
    - Store all questions **and** answers presented to the player (on each unique planet, playthrough, etc.)
        - 1-2 hours
- As a user, I want to see brief explanations for why an answer is correct so I can learn from my mistakes
    - For each question in the dictionary/set, store a hard-coded (or generic relevant to the "class" of question) explanation.

        - 3-4 hours
    - After an incorrect answer, create an on.click() or on.hover() event handler to show the explanation on the correct answer
        - 1-3 hours
- As a user, I want a clear summary screen after each planet showing which questions I answered incorrectly
    - Design a summary splash screen

Retry Missed Questions mode

- As a user, I want to retry the questions I missed so I can improve my score
    - Implement a retry button
        - 1-2 hours
    - Fetch missed questions and their answers from data management, allow user to try them again
        - OR fetch from the same “class” of function to ensure that the player does not brute-force the game by memorizing answers
        - 2-3 hours
- As a user, I want the game to track whether I answered retried questions correctly so I can see my improvement
    - Work with data management to fetch missed questions and their answers.
        - 1-3 hours

Graphics & UX (Ethan)

Both Screens:

- As a user, I want to be able to **interact with objects** so that I have a more engaging game experience
    - Object interaction is a bit of a broad request(?), but we can use event handlers such as on.click() and various animations to create a sense of interaction.
        - 8-10 hours
- As a user, I want to see **on-screen indicators** showing which buttons to press (e.g., Space to continue, Return/Enter to go back) so that I always know how to progress through the game.
    - Hard code on-screen indicator types (e.g. moving, answering questions) and allow other classes to display these indicator types
        - 2-6 hours

UI (Miguel & Khoa)

- As a user, I want all texts, buttons, and panels to follow the same scheme and design so that the game looks cohesive and professional.
    - Create a global theme manager such that changing from one scene to another (e.g. switching planets) updates all components without manual changes.

    - Create UI factory functions such that every scene uses standardized components (e.g. buttons, pop-ups) as opposed to hard-coded Konva elements.
    - Create a centralized/config file such that fonts, colors, paddings, etc. are all defined in one place for easy changes.
        - 2-3 hours
- As a user, I want to see a **menu bar** so I can gain control of my progress.
    - Design and create a menu bar class, allowing other classes to display the menu bar over their content
        - 2-3 hours
- As a user, I want to see a **planet display** so I can see what my current planet and future planets are.
    - Make a mini display using the data from progress tracking (include on.hover() conditions for unlock and visuals)
        - 1-2 hours
- As a user, I want to see my **planet's gameplay** so I can begin to answer questions.
    - Design a mini-map showing user's checkpoints on the planet. Mini-map includes: back indicator, checkpoints button, planet in detail graphics, character on the checkpoint.
        - 2-3 hours
    - Display and Interactive Components Testing
        - 1-2 hours
- As a user, I want to see a **question and answer display** so I can see the question and solution after answering the question.
    - Design an interactive question display when users click into the level checkpoint on the planet. Question display includes: back indicator, textbox for answer, enter button to check the answer, and update player's progress.
        - 1-2 hours
    - Design an answer display prompting whether users are correct or incorrect then giving out the correct answer below. Answer display includes: correctness notification, correct answer, and button to continue.
        - 1-2 hours
    - Display and Interactive Components Testing
        - 1-2 hours

Audio (Ethan)

- As a user, I want to be able to hear game audio (e.g. music, action sounds) so that I have a more interactive game experience
    - Create a folder of audio clips for various actions
    - Allow other classes to access sounds for event handlers
        - 4-5 hours (for both)

Data Management (melanie)

Progress Tracking

- As a user, I want to see which planets I've completed and which ones are still locked so i can see my progress
    - Create progress.json structure
    - Display planet icons with lock/unlock visuals
    - Add condition for 80% accuracy unlock
        - 4-6 hrs
- As a user, I want the game to save my highest accuracy per planet so that I can beat my personal record
    - Add bestAccuracy field to localStorage data
    - Compare current vs. saved on level completion
    - Display best score on planet select screen
        - 3-4 hrs
- As a user, I want a summary at the end of each planet showing my accuracy, number of correct answers, time taken, and if necessary, a recap of the topic if I perform poorly
    - Create SummaryScreen UI in Konva
    - Implement timer tracking for inner planet play
    - Add recap text if <80% accuracy
        - 6-8 hrs

Game State

- As a user, I want my scores to save automatically so I can return where i left off
    - Implement autoSave() on state change
    - Store JSON in localStorage
    - Add try/catch for save errors
        - 3-5 hrs
- As a user, I want to continue my journey from the same planet where I left off when I reopen the game
    - Read progress on game load
    - Load player's last planet scene dynamically
        - 3-4 hrs
- As a user, I want to enter my username at the start of the game so my progress and scores are personalized
    - Create username input screen
    - Save to localStorage object
        - 3-5 hrs
- As a user, I want my username to be remembered between sessions so i don't have to re-enter it every time
    - Auto-load last used username
    - Add "switch user" button

        - 2-3 hrs

Question and Content Management

- As a user, I want questions randomized each session so gameplay feels fresh and not repetitive
    - Implement shuffle function for question array
    - Add unique seed per playthrough
        - 2-3 hrs
- As a user, I want the game to track which questions I missed so I can review them later in a “Retry Missed Questions” mode
    - Add missedQuestions[] array per planet
    - Log question IDs of incorrect answers
        - 3-4 hrs

Scoring and Accuracy

- As a user, I want to be able to see which questions I get wrong, so that I know which questions I have to work on.
    - Add “Review Mistakes” overlay with icons or color markers of some sort
    - Store wrong question data temporarily
        - 4-5 hrs
- As a user, I want to see my overall score and accuracy after finishing each planet and mini-game so I can track improvement.
    - Add scoring formula (correct/total * 100)
    - Show at planet summary + mini-game end
        - 3-4 hrs

Analytics & Feedback

- As a user, I want to view a performance dashboard summarizing accuracy by topic to see what I’ve learned
    - Create dashboard canvas (bar chart using Konva shapes)
    - Fetch average accuracy per concept
        - 5-7 hrs
- As a user, I want a ‘Review Mistakes’ option showing incorrect answers with correct explanations so I can learn from errors
    - Load missedQuestions[] data
    - Display question, user’s answer, correct answer, and hint
        - 6-8 hrs

Game Logic (melanie)

Math Questions:

- Mercury: add, subtract, multiply, divide SMALL integers
    - As a user, I want to answer addition, subtraction, multiplication, and division questions so I can practice small-number operations
    - As a user, I want to replay Mercury if I score below 80% so I can improve and unlock Venus
        - Implement arithmetic question generator (1–2 digit)
        - Add validation & scoring logic
            - 4-6 hrs
        - Add accuracy check, replay condition, and reset logic
            - 2-3 hrs
- Venus: add, subtract, multiply, divide LARGE integers
    - As a user, I want to answer addition, subtraction, multiplication, and division questions so I can practice operations with larger numbers
    - As a user, I want to replay Venus if I score below 80% so I can improve and unlock Earth
        - Reuse Mercury logic, adjust ranges (2–4 digit)
        - Update question generation
            - 3-5 hrs
- Earth: add, subtract TIME
    - As a user, I want to practice adding and subtracting time so I can apply math to real-world contexts
    - As a user, I want to replay Earth if I score below 80% so I can improve and unlock Mars
        - Write time-format question generator (HH:MM)
        - Implement time arithmetic functions
            - 6-8 hrs
- Mars: find prime number, factors or multiples of a number
    - As a user, I want to practice identifying prime numbers and factors so I can have a stronger understanding of number theory
    - As a user, I want to replay Mars if I score below 80% so I can improve and unlock Jupiter
        - Generate random numbers, test primality
        - Create “choose all factors” interface
            - 6-8 hrs
- Jupiter: find a pattern of sequence
    - As a user, I want to find the next number in a sequence so I can recognize arithmetic and geometric patterns
    - As a user, I want to replay Jupiter if I score below 80% so I can improve and unlock Saturn
        - Generate arithmetic/geometric sequences
        - Add multiple-choice options for next term

            - 4-6 hrs
- Saturn: simple metric units conversion
    - As a user, I want to convert between metric units so I can practice real-world measurements
    - As a user, I want to replay Saturn if I score below 80% so I can improve and unlock Uranus
        - Implement unit conversion formula bank
        - Randomly pick conversions to quiz
            - 5-7 hrs
- Uranus: add, subtract, multiply, divide with like denominators
    - As a user, I want to add and subtract fractions with like denominators so I can master foundational fraction operations
    - As a user, I want to replay Uranus if I score below 80% so I can improve and unlock Neptune
        - Create fraction object representation
        - Implement addition/subtraction logic
            - 6-8 hrs
- Neptune: add, subtract, multiply, divide with unlike denominators
    - As a user, I want to combine fractions with unlike denominators so I can practice finding common denominators
    - As a user, I want to replay Neptune if I score below 80% so I can improve and unlock the final planet (?)
        - Extend Uranus logic with LCM denominator finder
        - Add mixed number support (optional)
            - 7-9 hrs

Mini Game Logic (melanie)

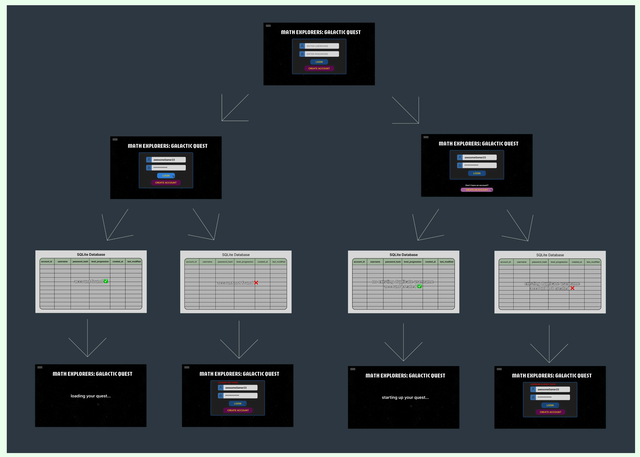
Mini games: 1 after Mars (Half-way), 1 after Neptune (Finale)

- Rapid-fire Math Questions (Option for Finale): Timed questions
- Fuel Fractions
    - As a user, I want to solve fraction problems quickly to refuel my ship before time runs out
        - Fraction equation generator
        - Implement timer countdown & fuel bar mechanic
            - 6-8 hrs
    - As a user, I want time-based rewards lie extra fuel for streaks of correct answers
        - Add streak counter + reward logic (extra fuel/time)
        - Display reward animation
            - 4-5 hrs
- Asteroid Factor Field (Option Half-way Assess)
    - As a user, I want to control a spaceship that blasts asteroids with correct factors/multiples so I can apply what I learned on Mars

        - Build simple Konva space field scene
        - Random asteroid generation w/ numbers
        - Collision detection + input check
            - 8-10 hrs
    - As a user, I want points for each correct asteroid destroyed so the game feels rewarding
        - Add score variable + on-screen display
        - Update scoring after hit
            - 3-4 hrs

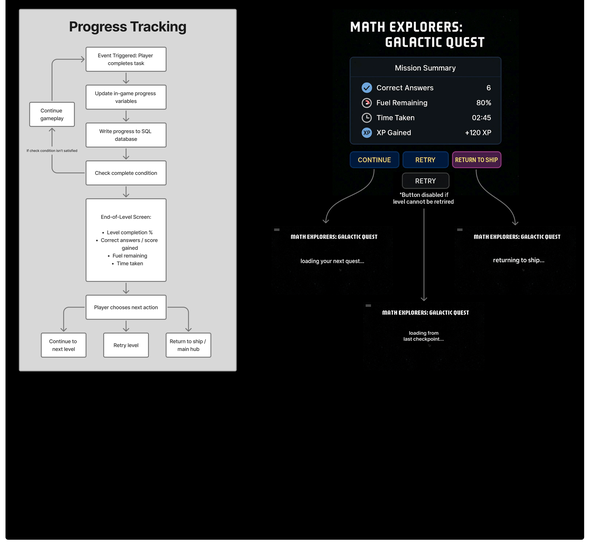
Design Mockups:

User Login / Account Creation



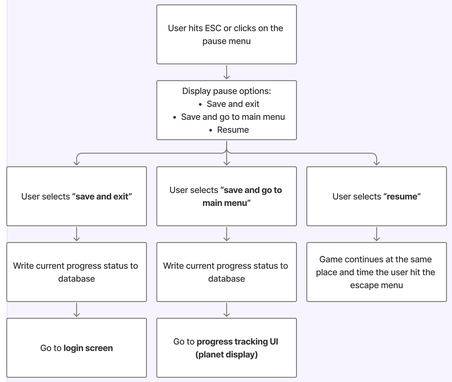
Progress Tracking

- Progress tracking graphic/UI
- End of level / end of game summary



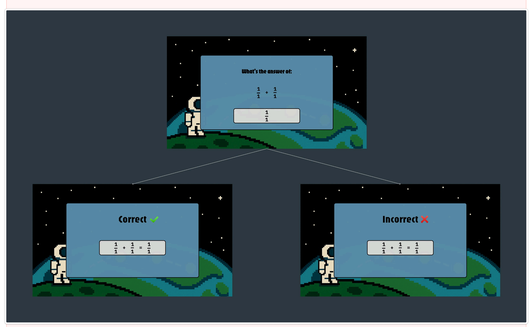
General Game UI

- Escape menu UI
- What does the general user interface look like when on the ship
- What does the general user interface look like when you are controlling the alien on a planet



General Math Questions

- General game flow



Rapid Fire Math Questions

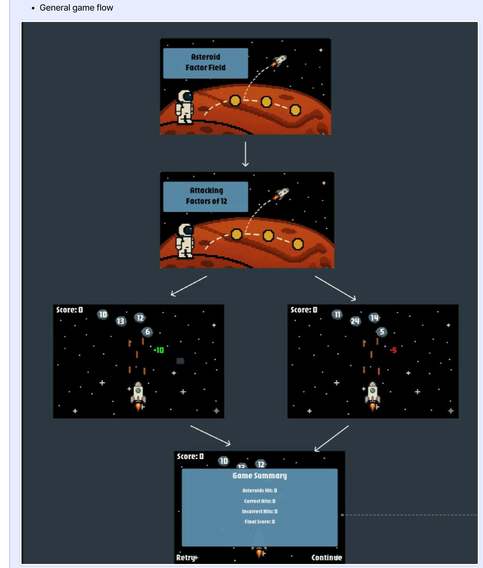
- General game flow

Fuel Fractions

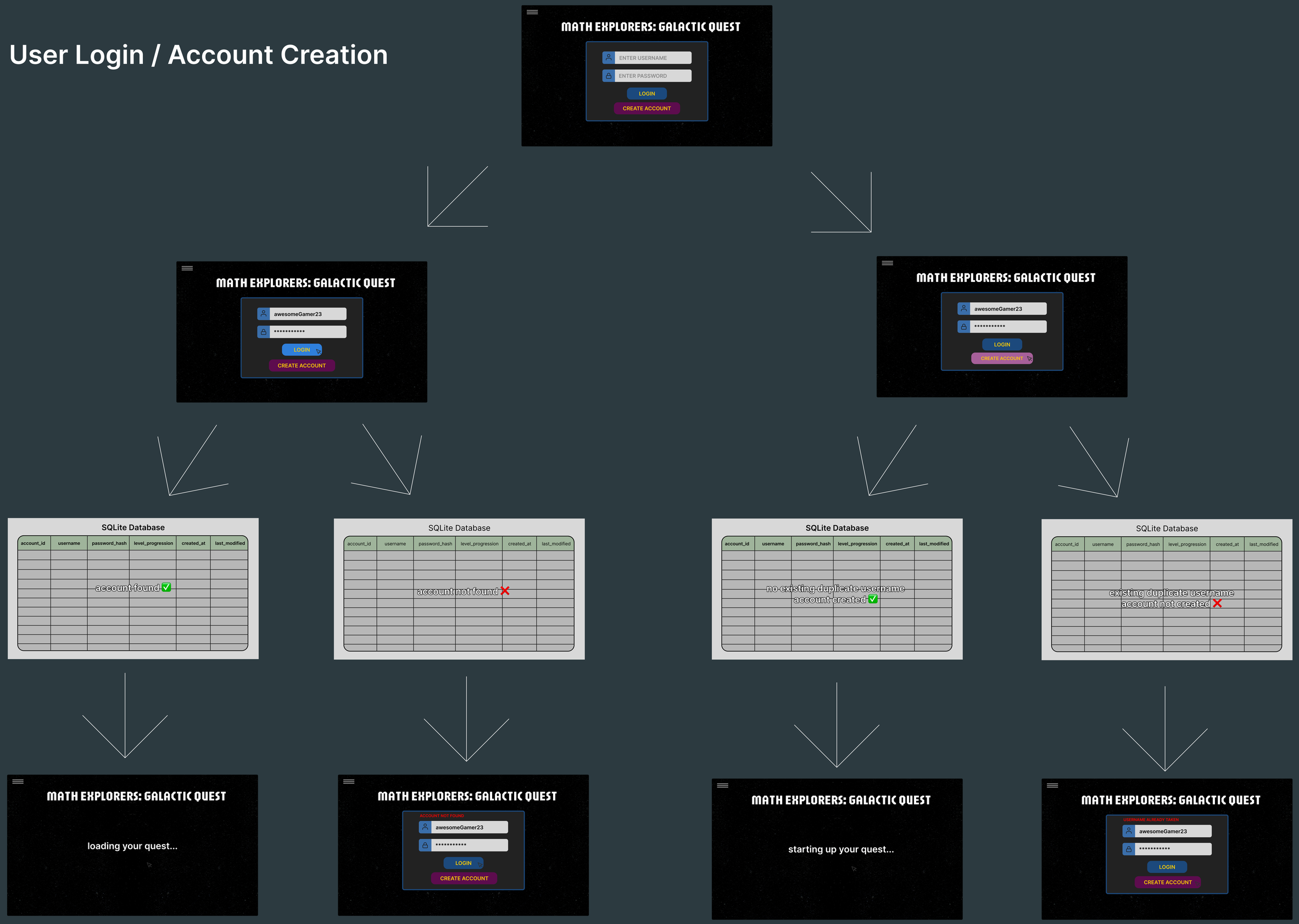
- General game flow

Asteroid Field Factors

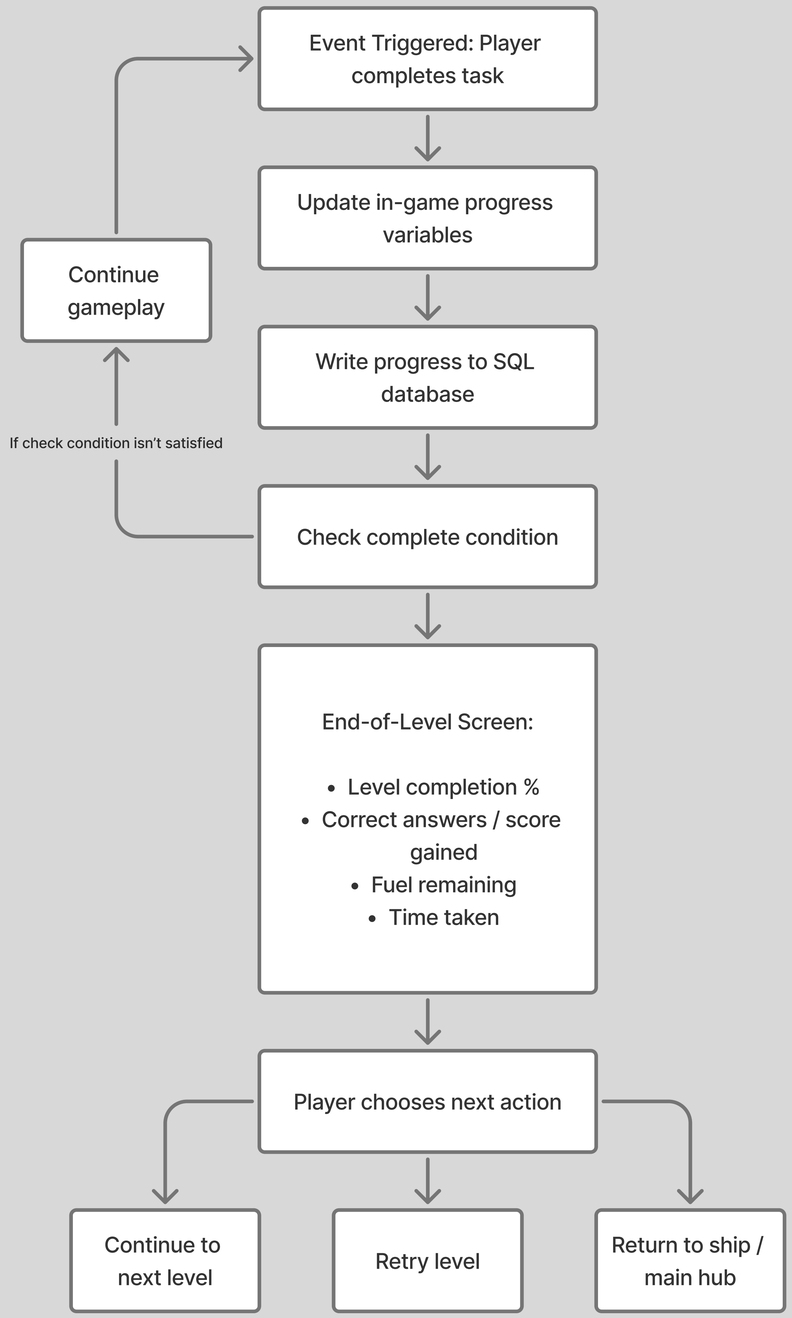
- General game flow

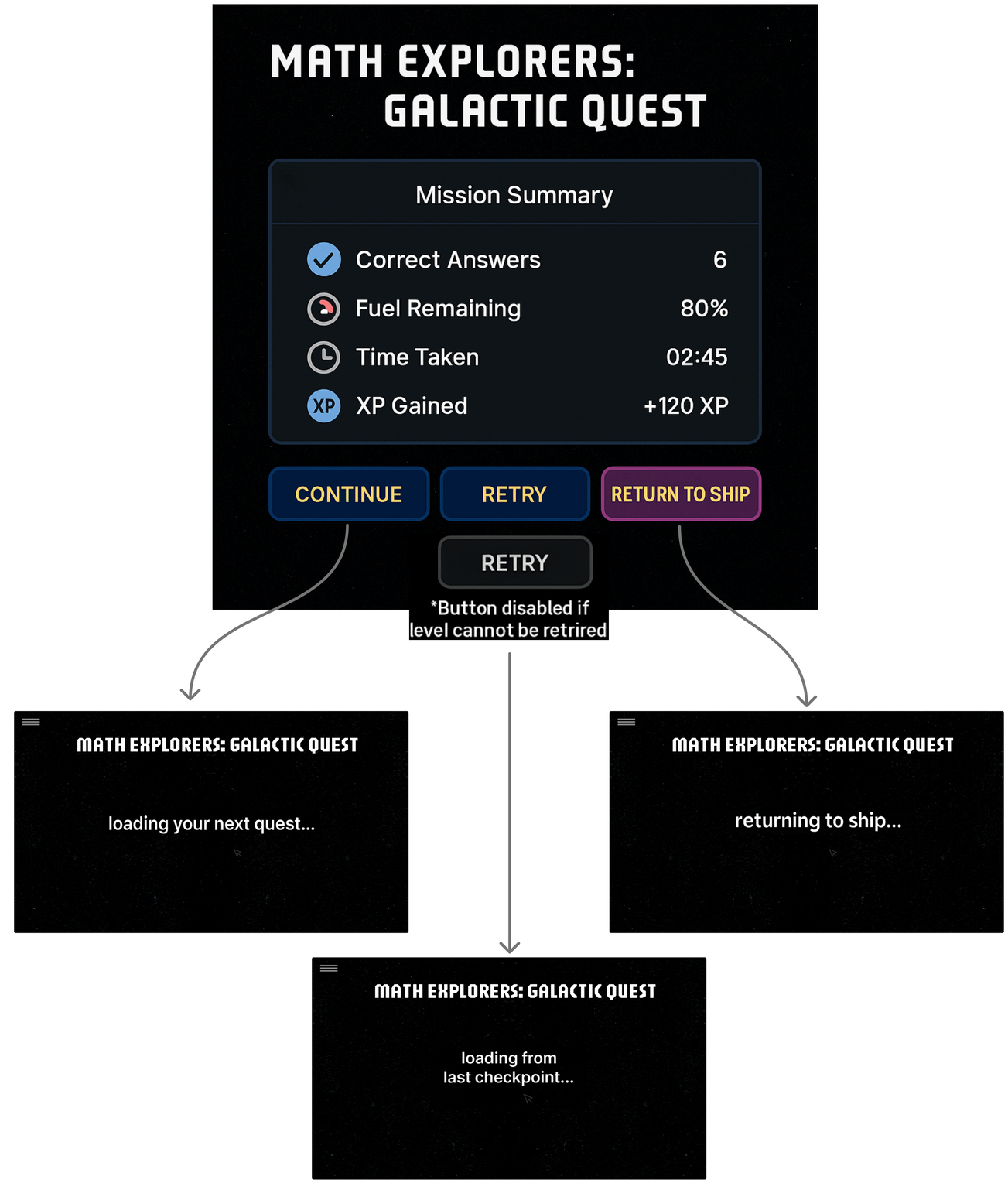


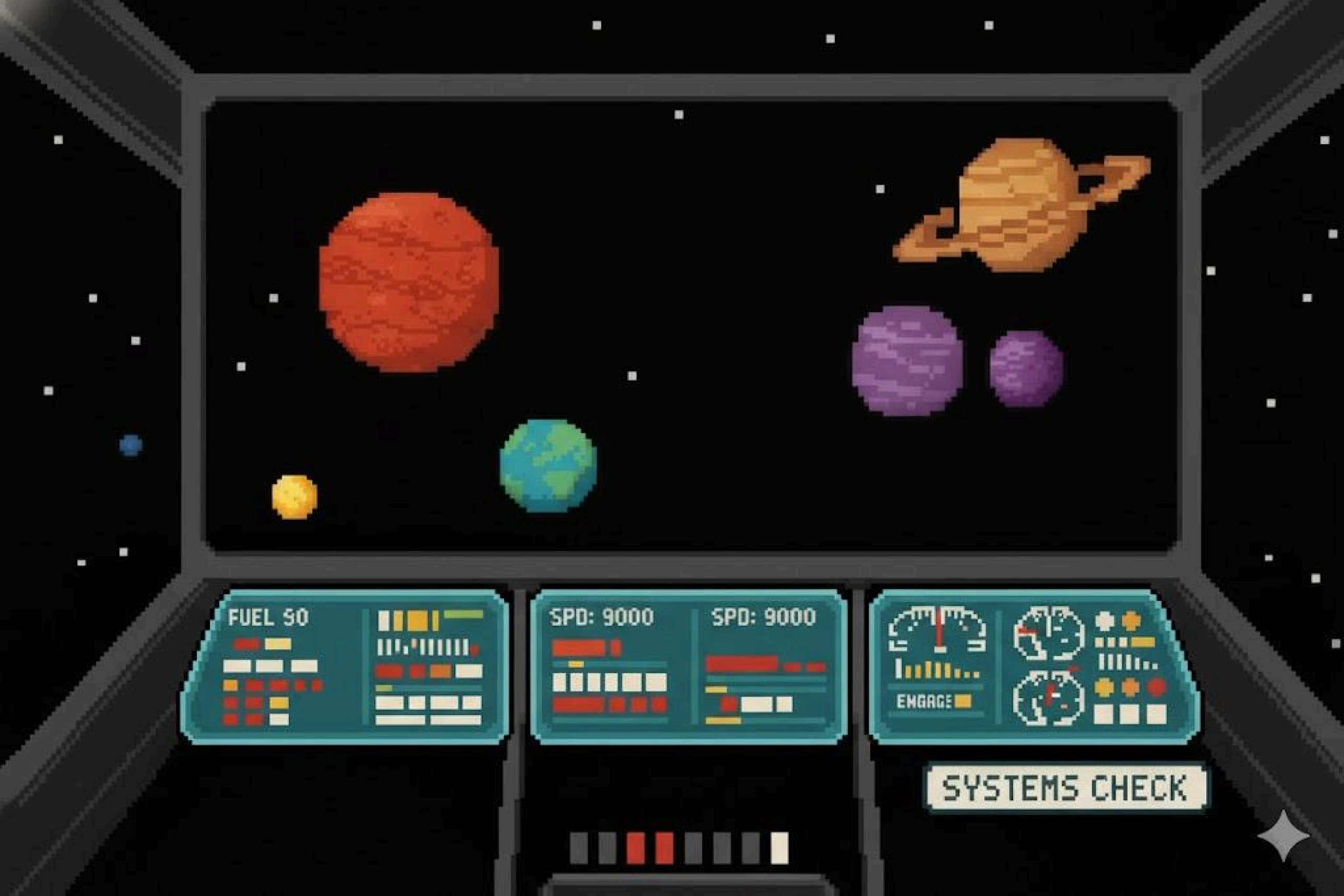
User Login / Account Creation

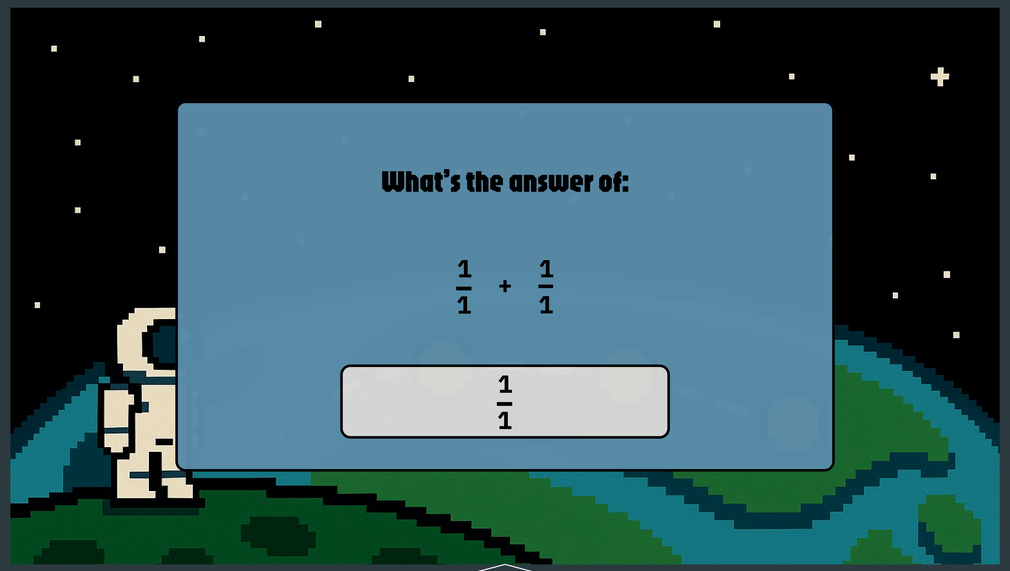


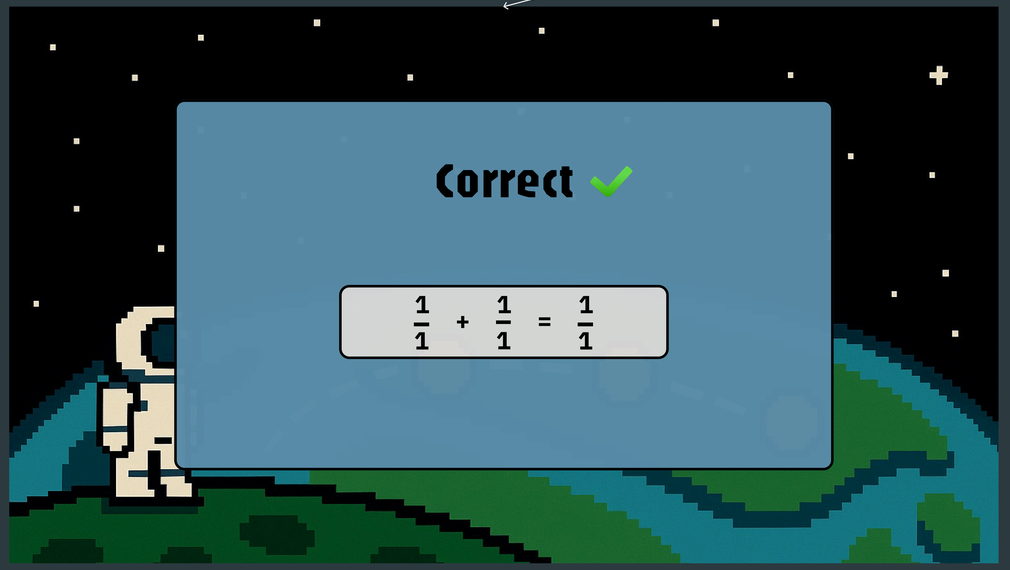
Progress Tracking

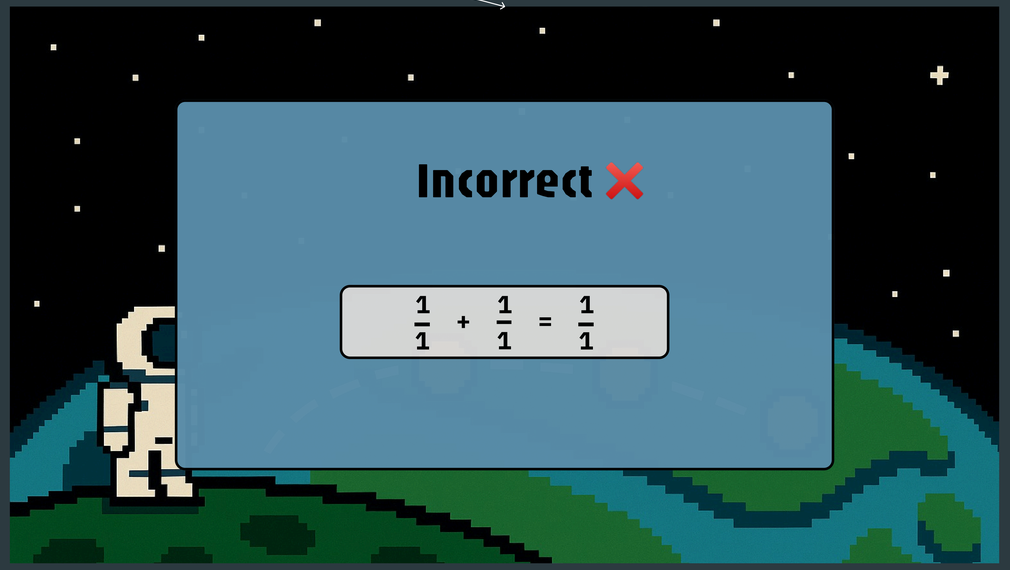


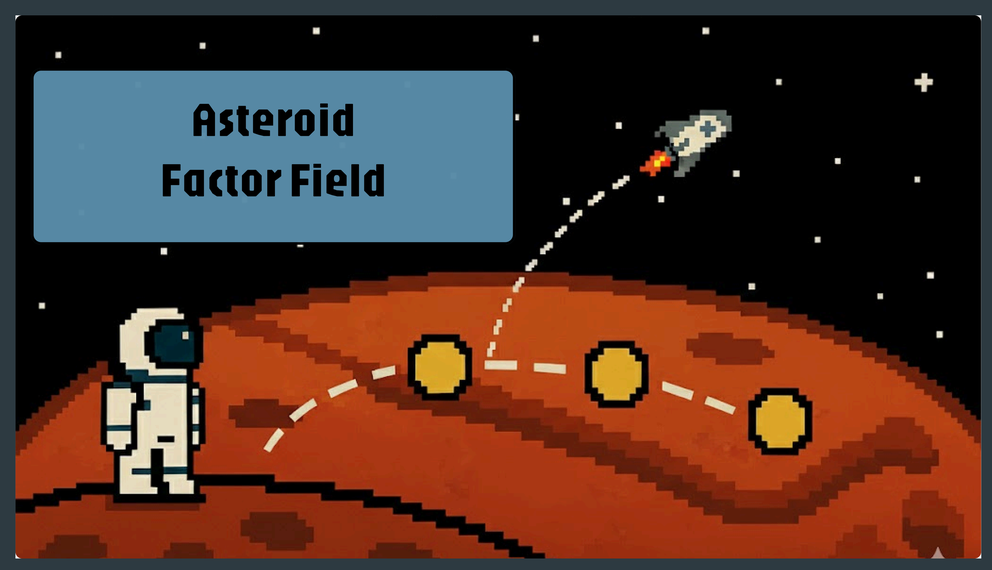



FUEL 50
SPD: 9000
SPD: 9000
ENGAGE
SYSTEMS CHECK

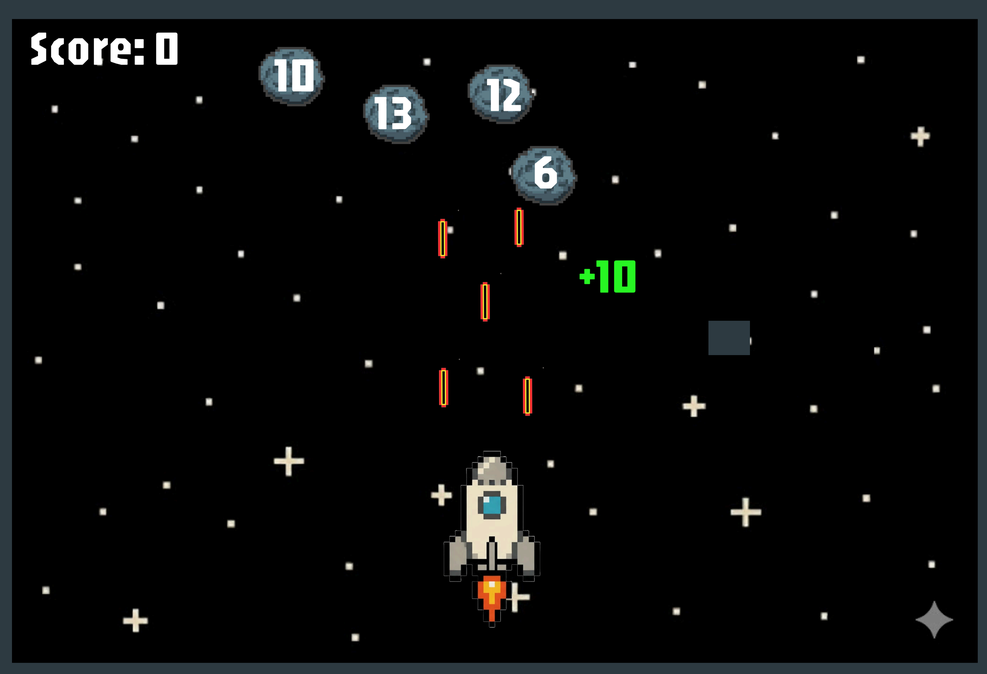

What's the answer of:
$\frac{1}{1} + \frac{1}{1}$
$\frac{1}{1}$

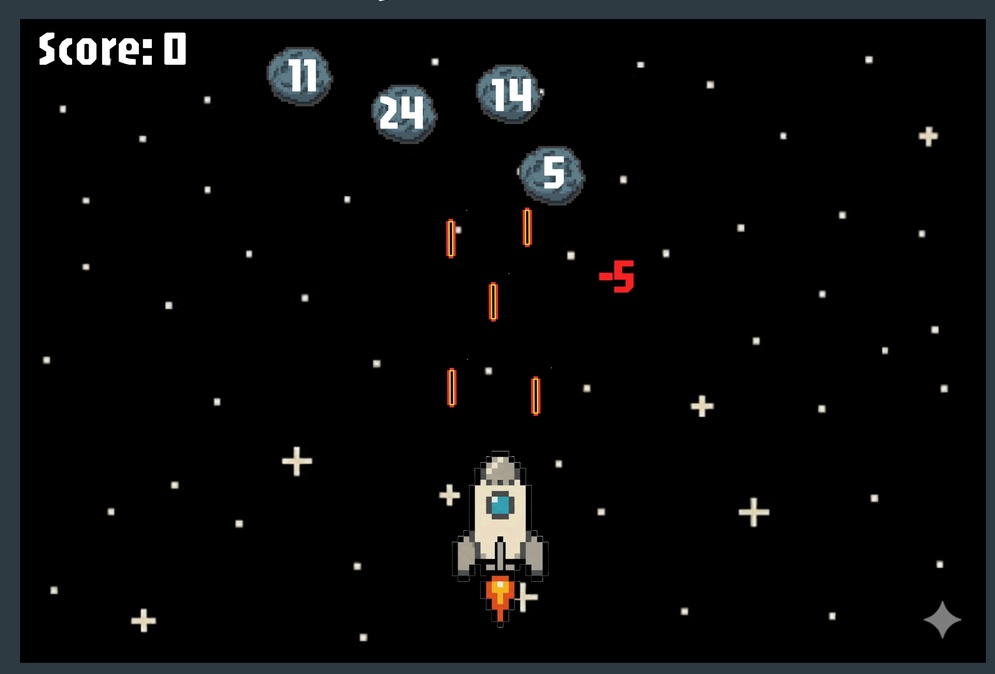

Correct
$\frac{1}{1} + \frac{1}{1} = \frac{1}{1}$

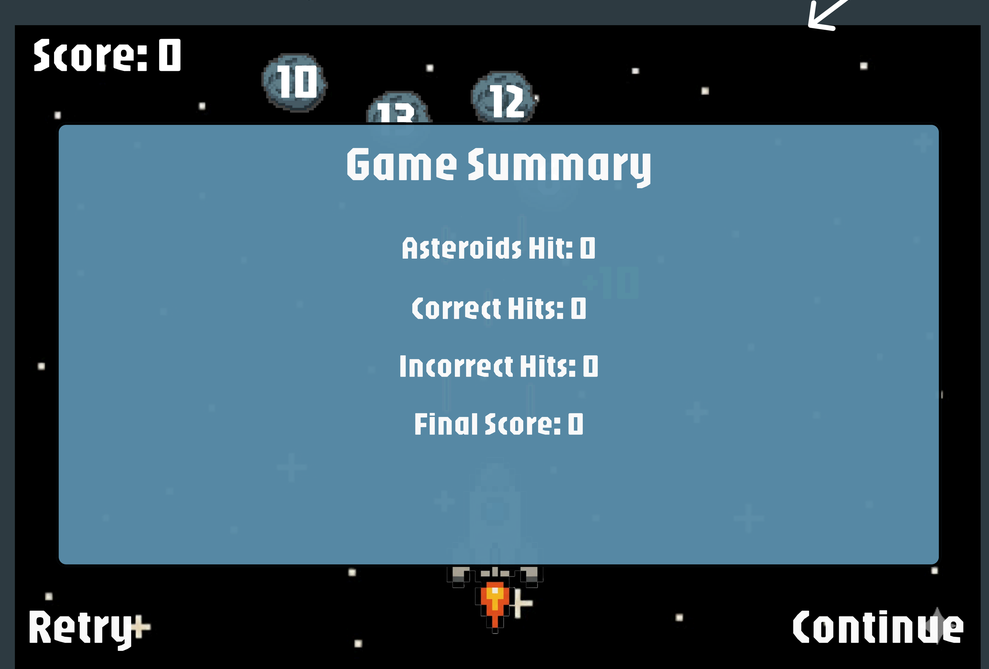

Incorrect
$\frac{1}{1} + \frac{1}{1} = \frac{1}{1}$


Asteroid
Factor Field

Attacking
Factors of 12


Score: 0
10
13
12
6
+10

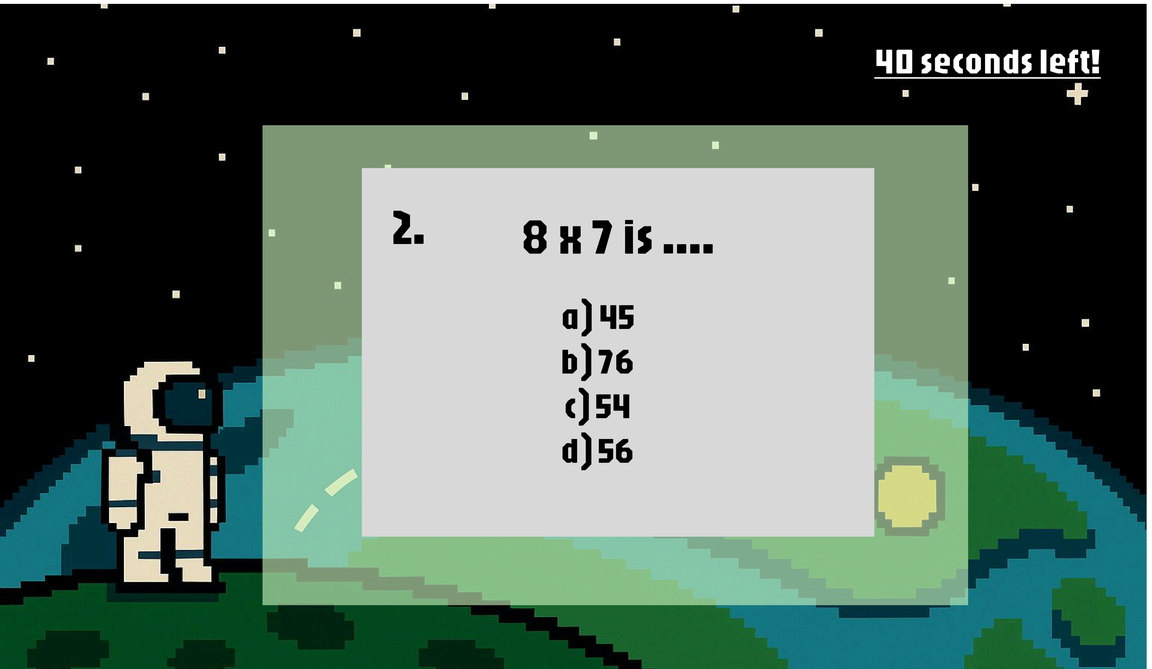

Score: 0
11
24
14
5
-5


Score: 0
10
12
Game Summary
Asteroids Hit: 0
Correct Hits: 0
Incorrect Hits: 0
Final Score: 0
Retry
Continue

Pause Menu UI

RAPID FIRE MATH

Answer as many questions as you can!



TIME'S UP!

Total answered: 8
Accuracy: 100%
Streak: 8

RETURN TO MAP

RESTART