



STELLAR SURVIVAL

A Race Against the Cosmos



A PROJECT BY
FOUR LOOP
H.M.M.M. HERATH | H.M.S.B.B. HERATH
K.G.D.S. RANASINGHE | L.H.M.M.D. BANDARA



STELLAR SURVIVAL

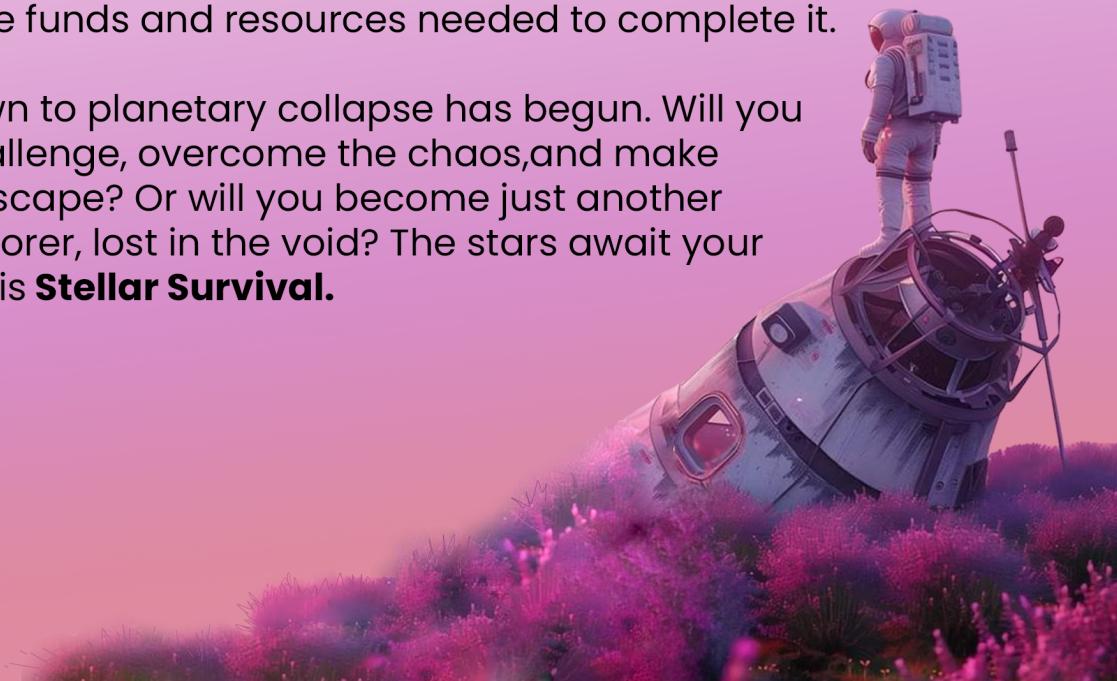
A Race Against the Cosmos

In the distant reaches of space, you find yourself stranded on an unstable alien planet, a world teetering on the edge of destruction. With each passing day, the planet's core grows more volatile, and your only hope of survival is to rebuild your damaged escape ship before time runs out. But survival won't be easy.

The land is rich yet unforgiving, filled with exotic crops like Glowberries, Starfruit, and Moonbeans, which could be your ticket to wealth and survival—if you manage to grow them in time. Resources are scarce, disasters strike without warning, and merciless space pirates lurk, waiting to pillage your farm. The elements are against you: meteors crash unpredictably, droughts turn fertile land into dust, and rogue solar flares threaten to erase your progress.

To outlast these cosmic threats, you must expand your farm, construct essential buildings like the Research Lab, and invest in protective technologies—Meteor Shields, Irrigation Systems, and Electric Fences—all while managing your limited credits. Your ship needs three vital parts to be spaceworthy again, and only through careful planning, strategic trading, and resilience can you gather the funds and resources needed to complete it.

The countdown to planetary collapse has begun. Will you rise to the challenge, overcome the chaos, and make your daring escape? Or will you become just another forgotten explorer, lost in the void? The stars await your decision. This is **Stellar Survival**.



STELLAR SURVIVAL

A Race Against the Cosmos

Introduction:

In Stellar Survival, you take on the role of an astronaut stranded on a mysterious planet. Your goal is to build a ship to escape the planet while managing a farm to sustain yourself. The game integrates various survival elements such as crop cultivation, building construction, resource management, and disaster prevention, providing a challenging experience filled with unpredictable events and strategic decision-making.



STELLAR SURVIVAL

A Race Against the Cosmos

Core Gameplay Loop:

The core gameplay revolves around balancing your resources and planning actions strategically, as your main objectives are to:

Plant Crops: Cultivate a variety of crops that grow over time, each with different growth durations and values.

Build Structures: Invest your credits to build structures such as a Research Lab and Ship Parts, which are required to escape the planet.

Prevent Disasters: Handle random disasters such as meteor strikes or droughts that threaten your crops. Disasters can be mitigated by purchasing items in the shop and building protective structures

Harvest Crops: Harvest grown crops to earn credits, which are crucial for both expanding your farm and purchasing needed items.

Shop Management: Visit the shop to purchase useful items like irrigation systems, meteor shields, and electric fences to safeguard your crops.

Manage Time and Resources: The game operates in turns, with time passing and resources being consumed or replenished based on actions. Managing turns and credits efficiently is key to progressing.

Mechanics Overview:

Player (Character) Mechanics:

Player Attributes: The player has several attributes that define their progress:

Credits: A form of in-game currency, used for planting crops, purchasing items, and building structures.

Turns Left: A limited number of turns to complete the game. Managing these turns wisely is essential.

Ship Parts: The player must acquire three ship parts to escape the planet.

Goal Credits: A target amount of credits the player must earn to fully succeed.

Crops and Farming:

Crop Classes: Different types of crops are available to plant. Each crop has:

Name: A unique identifier.

Growth Time: How many turns it takes for the crop to mature.

Value: How much the crop is worth once harvested.

Cost: How much it costs to plant the crop.

Requirements: Some crops require specific buildings to be planted.

Age: Tracks the current age of the crop.

Watered: A boolean flag indicating whether the crop has been watered during the current turn.

Overwatered: A counter to track how many times a crop has been overwatered.

Crop Growth: Crops grow over multiple turns, and each crop can be watered to accelerate its growth. However, crops that are overwatered too many times will rot and be destroyed.

Harvesting: Once a crop reaches its full growth time, it can be harvested for credits. Harvested crops no longer exist in the farm.

Disasters:

Disaster Types: Random events affect the farm, with a 30% chance of a disaster each turn:

Meteor Strike: Destroys crops if not prevented with a Meteor Shield.

Drought: Reduces crop growth if not prevented with an Irrigation System.

Pirate Attack: Damages crops and reduces progress if not prevented with an Electric Fence.

Acid Rain: Destroys crops if not prevented with a Shelter.

Disaster Prevention: Players can mitigate the effects of these disasters by purchasing items and building necessary structures. A warning for an upcoming disaster is given, and players need to act quickly to prevent damage.

Building Mechanics:

Building Structures: Players can invest credits to build essential structures:

Research Lab: Required for planting certain crops like Moonbean.

Ship Part: Each Ship Part contributes to building the escape ship. The player must collect three parts to win the game.

Other Buildings: Include structures like the Shelter, Meteor Shield, and Electric Fence, which help protect the farm from specific disasters.

Building Process: Each structure has a cost, and the player must have enough credits to construct them. Once built, these structures grant various benefits, such as crop protection or increased resources.

Shop Mechanics:

Shop Items: The shop offers the player a selection of useful items to purchase:

Irrigation System: Prevents drought damage to crops.

Meteor Shield: Protects crops from meteor strikes.

Electric Fence: Stops pirate attacks.

Shelter: Protects crops from acid rain.

Buying Items: Players can buy these items using credits. The items are essential for disaster prevention and improving the farm's sustainability.

Ship Mechanics:

Ship Building: The player needs to collect three ship parts to escape the planet.

Ship Parts: Each Ship Part costs a set amount of credits and can be obtained by building the corresponding structure.

Turn-based Mechanics:

Turn System: Each turn represents a passage of time in the game world, with the following actions available:

Plant Crops: Choose a crop to plant on the farm.

Water Crops: Water all crops to accelerate growth.

Wait: Let time pass, during which crops grow, and the player may receive bonuses or disasters.

Harvest Crops: Harvest mature crops for credits.

Build: Construct buildings to protect crops or acquire ship parts.

Shop: Purchase useful items to prevent disasters.

Disaster Randomization: A random event occurs at the end of each turn, with a 30% chance for a disaster to strike. If a disaster is imminent, players need to act quickly by buying items or building necessary structures.

STELLAR SURVIVAL

A Race Against the Cosmos

Victory and Game Over Conditions:

Winning:

The player wins if they manage to:
Earn enough credits to meet the goal (1000 credits).
Collect all three ship parts by building the necessary structures.

Losing:

The game ends if:
The player runs out of turns.
The player is unable to prevent too many disasters, resulting in the destruction of crops and a lack of resources to continue.

Object-Oriented Design Approach

The game follows an object-oriented design with the following key principles:

Encapsulation:

Each class encapsulates its own state and behavior, such as Crop, Player, Farm, Disaster, Building, and Shop.

Attributes like credits in Player and age in Crop are kept private to their respective objects and manipulated via class methods.

Abstraction:

The game models real-world farming mechanics while abstracting unnecessary details, making it easier to manage.

Disaster class abstracts different disasters by storing their effect as a Runnable, allowing flexibility in defining various disaster consequences.

Inheritance:

Instead of using a rigid inheritance structure, the design relies more on composition, which makes it more flexible.

Potential for future extension (e.g., subclasses of Crop for different types of plants).

Polymorphism:

The use of functional interfaces like Runnable (for disasters) and Consumer<Player> (for building benefits) enables dynamic behavior without requiring complex inheritance hierarchies.

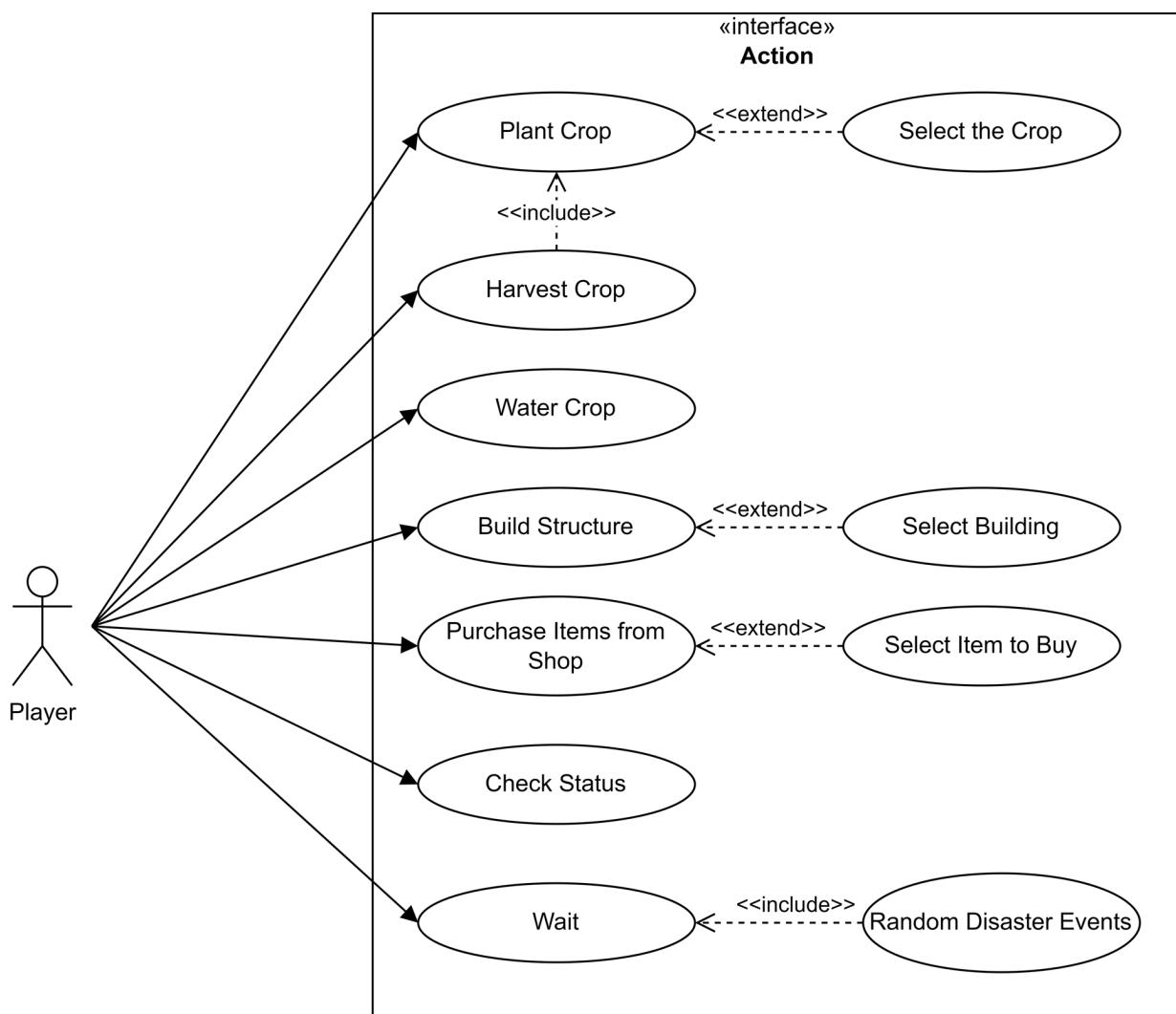
Composition over Inheritance:

The game uses composition extensively—for example, a Farm object belongs to a Player, and a Shop contains items stored as a Map<String, Map<String, Object>>.

The Ship class is an independent entity that interacts with the Player but is not a subclass of Player, keeping concerns separate.

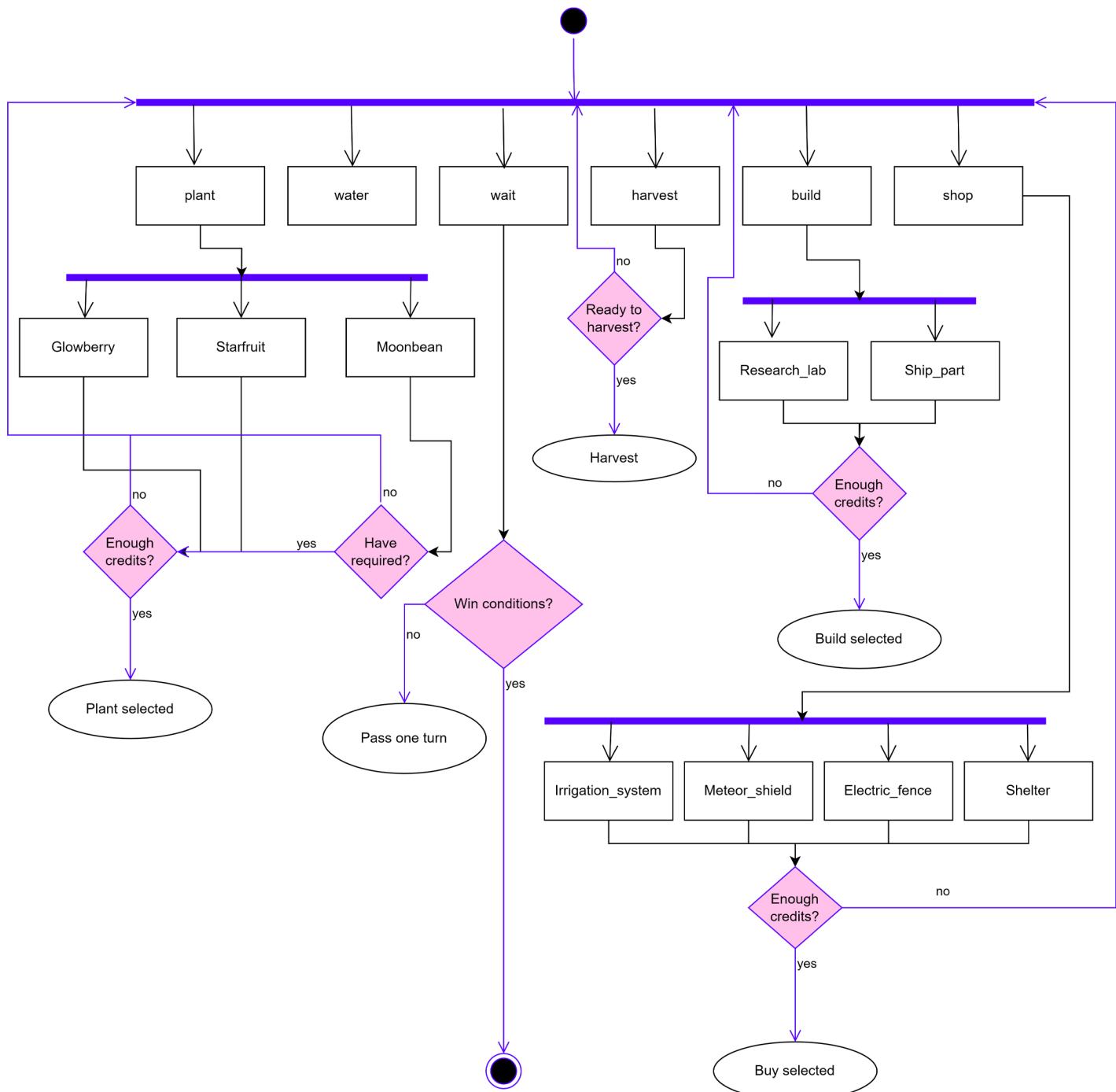
Use Case Diagram

This use case diagram outlines the different actions the player can perform in the game. It includes essential interactions such as planting, harvesting, watering crops, building structures, purchasing items, checking status, and waiting. Some actions extend into specific selections, like choosing a crop or building, while others include random disaster events.



Activity Diagram

Outline of the expected game flow, with the player actions such as planting, watering, waiting, harvesting, building, and shopping. It visualizes decision points like checking credits, meeting requirements, and determining win conditions.



Game Architecture Breakdown

The following is a breakdown of the core components in Stellar Survival, based on the class diagram. Each class plays a crucial role in the game, contributing to farming, resource management, survival, and the ultimate goal of escaping the planet.

Player Class

The Player is the main character, responsible for managing resources, farming, and ship construction.

Attributes:

name – Player's name.

credits – The in-game currency, earned from farming and used for purchases.

turns_left – The number of turns remaining before the planet becomes uninhabitable.

goal_credits – The amount needed to win the game.

ship – A reference to the Ship object.

farm – A reference to the Farm object.

shop – A reference to the Shop object.

Functions:

plant() – Allows the player to plant crops.

harvest() – Harvests fully grown crops to earn credits.

water() – Waters all crops, which helps them grow.

wait() – Passes time, growing crops and possibly triggering disasters.

build() – Constructs buildings or ship parts.

shop_menu() – Opens the shop to buy useful items.

status() – Displays the current game state.

check_win() – Checks if the player has met the winning conditions (enough credits + ship built).

Farm Class

The Farm is where crops are grown and disasters occur. The player expands, manages, and defends their farm against environmental threats.

Attributes:

crops – A list of planted crops.

buildings – A list of constructed buildings.

farm_size – Maximum number of crops that can be planted.

next_disaster – Tracks the next disaster (if one is scheduled).

random_disaster_chance – Probability of a random disaster occurring each turn.

Functions:

plant_crop(crop, player) – Plants a crop if space and requirements are met.

grow_crops() – Advances crop growth.

water_crops() – Waters all crops but overwatering may cause rotting.

harvest_crops(player) – Harvests mature crops, rewarding credits.

trigger_random_disaster() – Determines if a disaster happens in the next turn.

damage_crops(farm, damage) – Inflicts damage to crops during disasters.

build_structure(building, player) – Constructs buildings that improve survival chances.

Crop Class

Represents the different types of crops the player can grow. Each crop has growth requirements, costs, and rewards.

Attributes:

name – Crop name (e.g., Glowberry, Starfruit, Moonbean).

growth_time – Turns required to grow.

value – Credits earned upon harvest.

cost – Cost to plant the crop.

age – Current growth progress.

ready – Becomes True when the crop is fully grown.

watered – Tracks whether the crop was watered.

requires – Some crops need specific buildings to grow.

overwatered – Tracks consecutive overwaterings (can kill crops).

Functions:

grow() – Advances crop growth if watered.

harvest() – Rewards credits if the crop is ready.

☒ Disaster Class

Disasters are random space hazards that damage crops and slow progress unless countered.

Attributes:

name – Disaster name (e.g., Meteor Strike, Drought, Pirate Attack).

effect – The impact it has on the farm.

solution – The item or building needed to prevent it.

solution_hint – A hint for the player about how to avoid the disaster.

Functions:

occur(farm) – Triggers the disaster if the player isn't prepared.

Building Class

Buildings provide benefits and help the player overcome disasters and grow advanced crops.

Attributes:

name – Name of the building (e.g., Research Lab, Meteor Shield).

cost – Credits required for construction.

benefit – What the building does (e.g., unlocks new crops, prevents disasters).

Functions:

build(player) – Constructs the building if the player has enough credits.

Shop Class

The shop allows the player to buy protective items and other survival essentials.

Attributes:

items – A hashtable storing available shop items and their effects.

Functions:

display_items() – Shows items available for purchase.

buy_item(player) – Allows the player to purchase an item if they have enough credits.

Ship Class

The Ship represents the player's end goal. To win, they must collect all ship parts and earn enough credits.

Attributes:

parts – Number of ship parts collected (3 are required to escape).

Functions:

add_part(player) – Adds a part to the ship; when fully built, the player can escape!

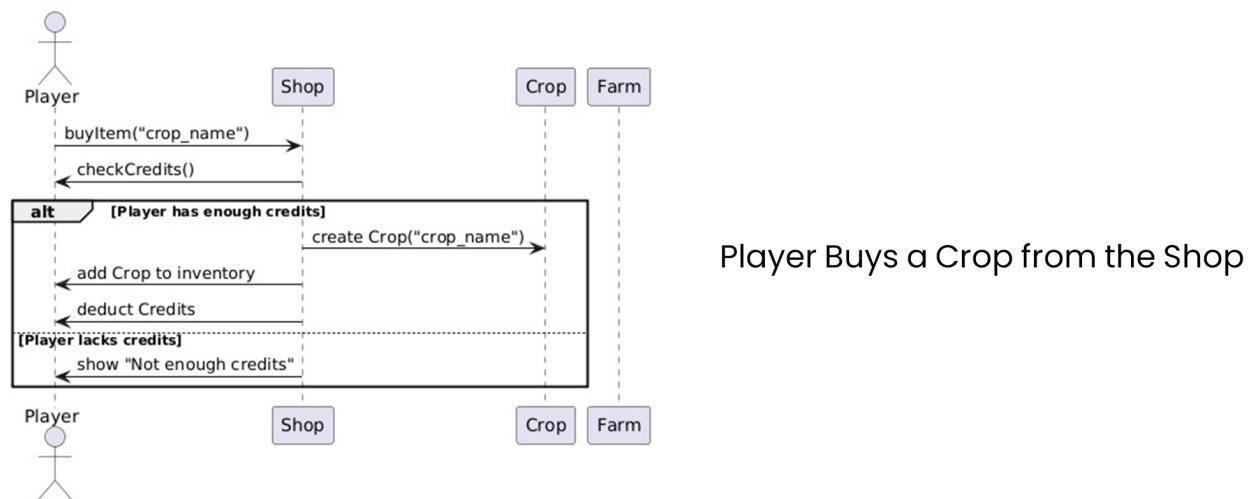
Class Structure

These class structure diagrams illustrate the expected object-oriented design for the game. They define key classes, their attributes, and methods, outlining how different game elements like players, farms, crops, shops, ships, buildings, and disasters interact. This serves as a blueprint for implementing the game's mechanics in a structured and modular way.

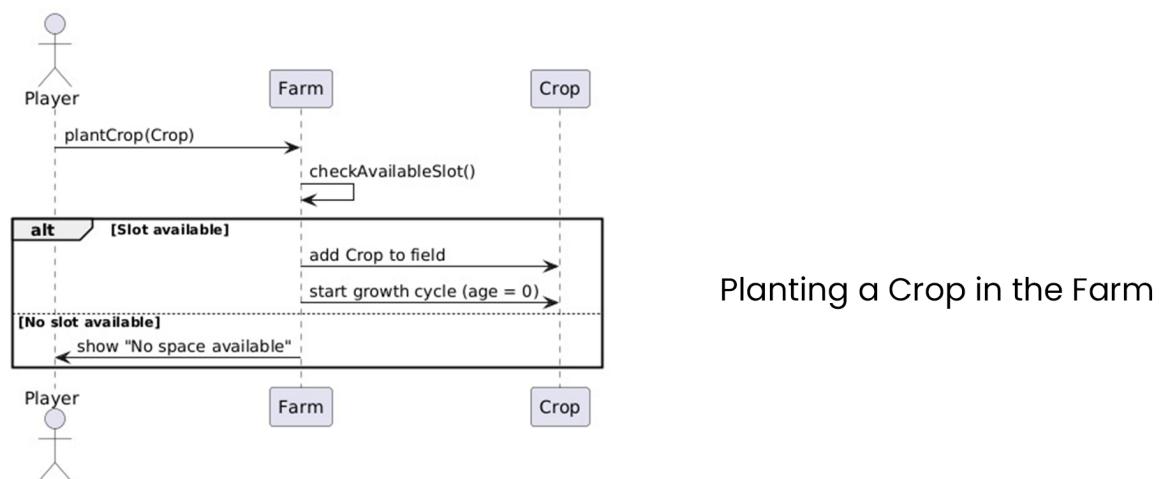


Sequence Diagrams

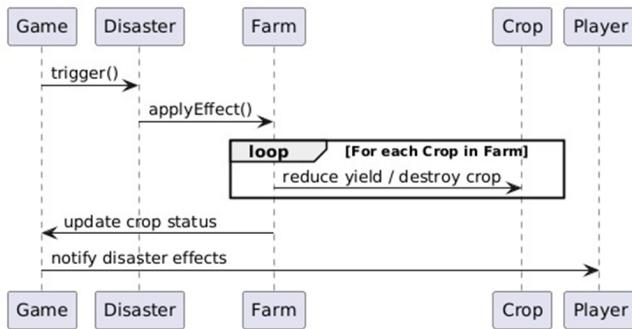
The following sequence diagrams outline the key interactions in the game, illustrating how different actions are processed step by step. These include buying and planting crops, handling disaster events, constructing buildings, and selling crops for credits. Each diagram represents the flow of interactions between the player and various game components, such as the farm, shop, and disaster system.



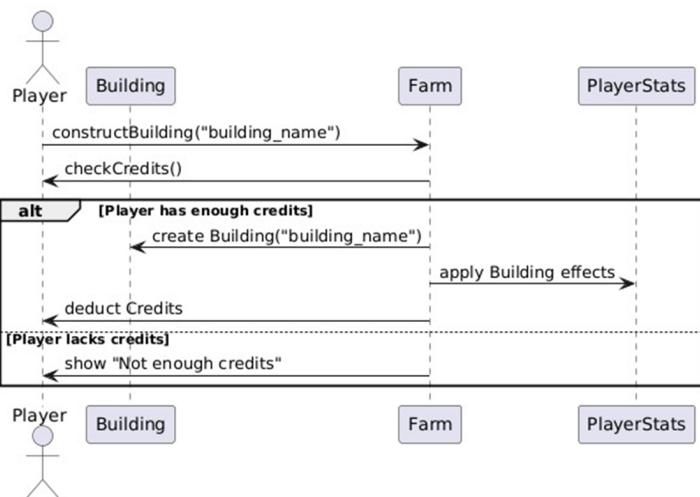
Player Buys a Crop from the Shop



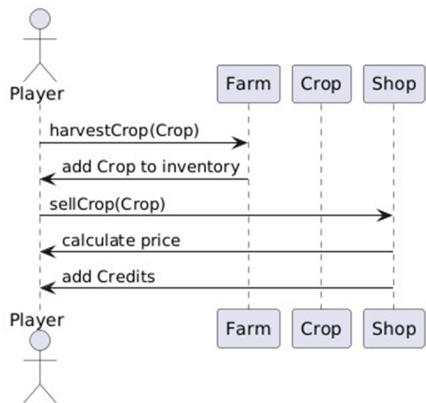
Planting a Crop in the Farm



Processing a Disaster Event



Constructing a Building and Applying its Effects



Selling Crops for Credits

Tools and Technologies Used

Java – The game is implemented in Java, leveraging its strong OOP capabilities, memory management, and robustness.

Java Collections Framework (JCF) – Used for managing game data efficiently:

List<Crop> for handling planted crops in Farm.

Map<String, Map<String, Object>> in Shop to store purchasable items.

LinkedHashMap ensures ordered insertion in menus.

Java Functional Interfaces – Runnable for disaster effects, allowing dynamic disaster behavior without requiring subclassing.

Consumer<Player> in Building to apply benefits to a Player instance.

Command-Line Interface (CLI) – The game is designed to run in a CLI environment, making it lightweight and easy to execute without GUI dependencies.

Uses Scanner for player input, making it interactive and menu-driven.

Randomization – The game includes random elements like: Math.random() to determine disaster occurrence.

Random().nextInt() to introduce variability in credit bonuses.

Unicode Symbols – Uses Unicode characters for a richer CLI experience, improving UI feedback.

STELLAR SURVIVAL

Upcoming Update: "Beyond the Stars - New Worlds to Explore"

We're excited to announce the upcoming Explorer's Survival update, version 2.0: Beyond the Stars! Get ready for an exciting expansion that will take you far beyond your current planet, introducing new challenging planets with unique and harsh environments, a space vendor, mysterious and rewarding plants, and survival mechanics that will keep you on your toes!

Here's a sneak peek into what's coming:

The Space Vendor

Discounted Parts and New Purchases!

Tired of struggling to gather resources for your ship? The new Space Vendor is here to help! This mysterious merchant roams the cosmos, offering discounted parts and rare technology upgrades that will make your survival much easier. For a limited time each turn, the space vendor will visit your farm and offer you exclusive deals, including ship parts, protective gear, and even rare plant seeds that you can't find anywhere else!

Discounted Ship Parts: The space vendor will provide ship parts at a fraction of the cost, speeding up your escape and giving you a break from the grind of collecting materials. However, dealership limits will apply, so choose wisely!

Space-Tech Gear: Need an edge in the new hostile environments? The vendor also offers specialized space suits, oxygen tanks, and heat shields that can protect you from extreme conditions on new planets.

Mysterious Plants with Immunities

A Reward for the Brave!

The universe is full of secrets, and mysterious plants are one of the most exciting new features in version 2.0! These unique crops offer immunities to various types of environmental damage and disasters. If you're lucky enough to find these rare seeds and cultivate them, you can grow plants that offer protection against specific natural calamities or even harsh climates.

Frost Fern: A plant native to icy planets, the Frost Fern gives you immunity to freezing effects, preventing your crops from being damaged in extreme cold.

Volcanic Cactus: Found on lava planets, this spiny plant shields your crops from high heat and lava flows, ensuring that your farm survives even the most intense conditions.

Radiant Bloom: A glowing plant that thrives in acid rain environments, this rare flower can neutralize the effects of toxic rainfall, keeping your crops safe and your farm thriving.

These plants will not only give you the ability to survive harsher environments, but they will also boost your crop value, making them highly valuable when harvested. Farmers who master these mystical plants will be rewarded handsomely with credits and extra resources.



New Planets to Explore – Survive the Elements!

Get ready to explore three brand new planets, each with its own set of challenges and survival mechanics. These planets are filled with environmental hazards, requiring new strategies to survive. Each planet will have unique resources, crops, and dangers, offering a diverse experience every time you travel.

Icy Planet – Frosthaven

Temperatures constantly dip below freezing, making it nearly impossible for crops to survive without Frost Ferns or specialized buildings. Build heating systems and insulated greenhouses to protect your crops from the extreme cold. Stock up on thermal suits and make sure to plant Frost Ferns to keep your crops safe from the freezing temperatures. Frostberry, a fruit that can withstand extreme cold, and Icevine, a hardy vine that grows even in freezing conditions.



Lava Planet – Inferno



Volcanic eruptions, scorching temperatures, and molten lava make this planet a dangerous place for crops. Only the hardest plants will survive in these extreme conditions. Build lava-resistant greenhouses and invest in Volcanic Cactus to ensure your crops are protected from the heat. A heat shield will also be necessary to protect you from the harsh climate while you work on the farm. Lavafruit, a plant that thrives in high temperatures, and Firewood Tree, a fast-growing tree that can withstand the heat and provide much-needed resources.

STELLAR SURVIVAL

We can't wait to bring these exciting updates to life and enhance your gameplay experience! Stay tuned for more adventures in Stellar Survival!

References & Inspiration

Game References (Mechanics & Design)

Starbound (2016) – Chucklefish

Official Wiki: <https://starbounder.org>

Inspiration: Farming, base-building, space exploration, and procedural planets.

RimWorld (2018) – Ludeon Studios

Official Wiki: <https://rimworldwiki.com>

Inspiration: Random events, survival colony management, farming, and disasters.

No Man's Sky (2016) – Hello Games

Official Site: <https://www.nomanssky.com>

Inspiration: Space farming, planetary exploration, and survival mechanics.

Stardew Valley (2016) – ConcernedApe

Official Wiki: <https://stardewvalleywiki.com>

Inspiration: Farming mechanics, economy system, and long-term progression.

Astroneer (2016) – System Era Softworks

Official Site: <https://astroneer.space>

Inspiration: Terraforming, resource gathering, and sci-fi exploration.

Other References and Tools

GeeksforGeeks UML Design Article

<https://www.geeksforgeeks.org/unified-modeling-language-uml-introduction/>

Draw.io and PlantUML for generating UML Diagrams

Pinterest for Visual references, concept art, and design aesthetics.