Design Document

Version 2.0 – 2023.12.05
Created 2023.11.11

Castaway Chronicles

Alexander Voidoc
Ali Zaeri
Sheryl Buttar
Jayant Arora

GitLab Repository:

https://mcsscm.utm.utoronto.ca/csc207_20239/group_64

Final Demo Video:

https://youtu.be/NTNzFGW1Y0E

SECTION 1: PROJECT IDENTIFICATION

This project is dedicated to designing an adventurous game with educational aspects that ensures a fun, enriching experience for children and students. At Castaway Chronicles, our aim is to cater to a diverse group of learners, including the visually impaired, students with ADHD and special-education students. Students with special learning needs and attention-deficit disorders often struggle to stay engaged in their learning and thereby require more creative teaching techniques that are thoughtfully designed to accommodate their learning styles, empowering and encouraging them to thrive in their educational journey. We intend to improve the previous assignments from the course by expanding upon the existing templates and incorporating additional functionalities. Through the completion of mini-games and challenges, centered around educational and recreational games, kids will be driven to learn on their own with Castaway Chronicles. This will allow them to win the game and collect coins. With this approach, the team at Castaway Chronicles aspires to make a meaningful impact in the lives of children that face difficulties in mastering foundational skills.

SECTION 2: USER STORIES

The following user stories were changed/deleted from the original phase 1 design document previously submitted. Reasoning for changing/deleting the following user stories is given below the table.

Name	ID	Owner	Description	Implementation Details	Priority	Effort
Manage Inventory	3.1	Jayant	As a player, I want to be able to view and manage my inventory so that I can see what items I've collected and if they will be useful on my journey.	Create a list attribute associated with the player that stores all items in player inventory	2	1
Island Objects	3.3	Sheryl	As a player, I want to be able to view items available on an island so that I can choose to pick up/drop something from my personal inventory.	Have images of these objects prepared for each island and create island objects for each respective island	3	1
Different biome types	4.7	Alex	As a player, I want certain grid areas to follow a certain theme and have more of certain types of objects in them and a different appearance.	Randomly select a theme that adjusts the appearance and the random chances of certain cells having certain events.	3	3

Randomly placed hidden and visible obstacles.	3.7	Alex	As a player, I want some cells to be obstacles that must be navigated around or some cells to have hidden traps.	In addition to randomly selecting the cells to have islands, also give them the potential of having hidden traps or an obstacle that the player must go around.	3	3
Pickup Item	2.3	Jayant	As a player, I want to be able to pick up items available on islands so that I can add them to my personal inventory, this will make them no longer available on the island.	Create a function that picks up an item, adding it to the player's item list and removing it from the items list for the island.	2	1
Drop Item	2.4	Jayant	As a player, when at an island, I want to be able to drop items in my inventory so that my inventory is specific to what I want, the items I drop should then be added to items available on the island I dropped them at.	Create a function that drops an item, removing it from the player's item list and adding it from the items list for the island.	2	1
Navigating through islands within the grid	1.8	Ali	As a player, I want to be able to easily select from a list or grid of available islands so I can navigate to them and explore the available options from that destination.	Implement a dropdown list or a list of buttons available on a grid, and implement a navigation method, changing the player's location to the destination and triggering any entrance cues.	1	2

Reasons for deleting the above stories:

- Story 3.3 - our team has decided to completely get rid of the idea of picking/dropping things on islands, including magic potions (this was mention in sprint 2 product backlog). As a result, multiple stories have been changed/deleted (3.1, 2.3, 2.4)
- Story 3.1 - as shown in the story 3.3 directly above, there is no longer a need to manage inventory, so this story 3.1 has been removed
- Story 4.7 - this story was also unnecessarily and "extra" and was not implemented
- Story 2.3 - deleted as mentioned above in story 3.3
- Story 1.8 - changed design of game, rather than having the player decide to go on a specific island from a list, the player follows the infinite grid to wherever it takes them
- Story 2.4 - removing the "drop item" concept as the team has decided to not include any items for functionality purposes. Pickup item (story 2.3) has been updated, shown in the

“changes made” section above

User Stories to be implemented (updated):

Name	ID	Owner	Description	Implementation Details	Priority	Effort
Save Game	2.1	Jayant	As a player I want to be able to save my game progress so that I can continue my adventure game later on without losing my progress (coins, lives left, etc.) and game data.	Save game file by saving to serialized binary file	1	3
Lives Left	1.1	Jayant	As a player, I want to keep track of my remaining lives so that I know how many more chances I have to complete games.	Create an attribute associated with the player’s file that decrements by 1 every time the player loses a life, display on user screen	1	1
Coins Collected	2.2	Jayant	As a player, I want to keep track of the number of coins I’ve earned, so that I can know how many coins I require to reach the winning amount (500).	Create an integer attribute associated with the player class that increases/decreases depending on the events taking place with the player	2	2
Set Name	1.2	Jayant	As a player, I want the option to set my player name so that I can have a personalized gaming experience.	Create a rename function associated with the player class to change the player’s name that is displayed	3	1
Visited	4.1	Jayant	As a player, I want to know if I have visited a particular island already, so that I can focus on exploring new islands as new islands provide maximum rewards.	Create a boolean attribute that starts as false and changes to true once an island has been visited	2	2
Lose Entire Game	4.2	Jayant	As a player, I want the game to end/lose the game after I run out of my 3 lives, so that the game restarts fresh.	If the attribute introduced in “Lives Lost” reaches 0, take the game back to the home page giving the player the option to restart	1	2

Diverse Games	4.3	Sheryl	As a player, each time I arrive on an island I want to have the option of playing new, and a variety of mini-games, educational and adventurous, to keep the experience exciting.	Implement a randomizer to generate different pre-assigned games and play styles based on the island that the player is currently visiting. Each island would be a separate class and mini-games	1	3
Instructions	1.3	Sheryl	As a player, I want to know the specific rules/goals for each activity on each island so that I can fully understand the game I am playing and increase my chances of winning.	Create an instructions pop-up that is displayed before the start of each mini-game with short, yet understandable rules and guidelines for the game.	1	2
Inclusive Learning	4.4	Sheryl	As a player, I would like a certain number of attempts for each mini-game depending on the difficulty level chosen at the start of the game.	Create a predetermined number of tries per game as per the difficulty level chosen by the user and filter mini-games to have a certain number of problems/difficulty accordingly	3	1
Rewards	1.4	Sheryl	As a player, I would want to see additional rewards for completing minigames quickly or precisely, as well as within a set amount of time depending game to game.	Implement a reward system (a certain number of gold coins) per mini-game based on accuracy and time spent on the mini-game	2	1
Penalty	2.5	Sheryl	As a player who loves a challenge, I would like some consequences for losing a mini-game.	In the instructions text, notify a user the consequences of losing a game and in the implementation, predetermine penalties depending on the game and difficulty	2	2

Customize Difficulty	3.2	Sheryl	As a player with a short attention span, I want every island to provide distinct adventurous or educational opportunities with various themes and difficulties.	Create distinct island classes with corresponding mini-game options for each topic and difficulty level as a player progresses through the game	1	2
Accessible Gameplay	4.5	Sheryl	As a visually-impaired player I hope for accessible features in mini-games auditory prompts, and auditory as well as printable .txt file feedback.	Implement an auditory to feature and if a player clicks a particular key (ie, shift key) generate a .txt file which can then be transferred to a braille display, or for feature references as required by the user	1	3
Location indication	1.5	Ali	As a player, I want a clear indicator of my current location on the grid, so that I can easily plan my route and make informed decisions about where to go.	Implement a location indicator on the user interface, providing real-time updates on the player's current island or grid cell with audio and visual cues.	2	2
Customizable Grid Cells	3.4	Ali	As a player, I want the option to choose the size and difficulty of the grid cells for maneuvering, so that I can tailor the game's navigation to my preference and skill level.	Create a grid customization feature allowing players to choose grid cell size and difficulty settings via a settings menu. Implement system responsiveness to player selections which in turn modify game terrain and mechanisms.	2	3
Feedback on Island Arrival	4.6	Ali	As a player, I want to receive visual and auditory feedback when reaching a new island, so that I can feel a sense of accomplishment and progress in the game, as well as receiving confirmation of my arrival.	Implement triggers with audio and visual cues upon island arrival using event handlers. Add customizable settings to adjust celebration metrics. (e.g. different themes)	1	3

Return to Previously Visited Islands	2.6	Ali	As a player, I want the option to retrace my steps and return to islands I've visited before, so that I can improve my performance in the games on those islands and gain additional gold.	Save the previous island after visiting a new one. Add that to the history of each island, and make them accessible using a /back command.	1	1
Engaging Movement Experienc e	3.5	Ali	As a player struggling to pay attention, I want engaging animations and/or sound effects during island-to-island movement, so that I'm motivated to explore the game world.	Implement event triggers for movement between islands, presenting audio and visual effects. Potentially add customizable audio and images as a “loading screen”.	3	2
Randomly Generated Grid	1.6	Alex	As a player, I want there to be a grid of cells that I can navigate through with some cells having random islands on them.	Create a class that stores all of the cells for the grid. These cells may randomly have an island assigned to them so that when the player goes over this grid cell, they would enter that island. All randomly generated.	1	3
Procedural Generatio n of more grids	1.7	Alex	As a player, I want to be able to move off the current grid into another newly generated grid and be able to move back and forth between an unlimited amount of grids.	Have a controller that stores a position mapping for the different grids for memory when returning back to the grid. Everytime the player tries to move off the current grid, either make a new one or load a previously created one.	2	2
Pirates in open water	2.7	Alex	As a player, I want certain grid cells to have pirate ships that if I run into, will play a minigame to see if I win or lose gold.	Have certain cells randomly designated as pirate cells that if crossed over, will initiate a minigame for gold. If a minigame is won, the pirate will disappear, otherwise the pirate will still exist.	1	2

Pirates moving	2.8	Alex	As a player, I want the pirates to be able to move making it harder to avoid them.	Have pirates make a random move every time the player moves in some direction by moving to a nearby “open” cell or potentially the same cell the player is trying to move to. Pirates cannot move between grids, only within.	2	2
Fog around Unexplored areas	3.6	Alex	As a player, I want there to be fog in the cells I haven’t visited that clears and stays clear when I travel near. This way I can see the areas I’ve already visited.	Give the cells a fog attribute that visually occludes the cell when it’s enabled. Disable it once the player gets near allowing them to see whats in the cell if anything.	3	2
Randomly placed hidden and visible obstacles.	3.7	Alex	As a player, I want some cells to be obstacles that must be navigated around or some cells to have hidden traps.	In addition to randomly selecting the cells to have islands, also give them the potential of having hidden traps or an obstacle that the player must go around.	3	3

Acceptance criteria for these user stories are below:

Name	**ID**	**Description**
Diverse Games	4.3	When a player arrives on an island, they can navigate and select from a list of mini-games using keyboard-only controls. Users who rely on screen magnification receive an audible indication when the list of mini-games becomes visible on the screen. Users employing screen magnification are provided with an audible count of the number of mini-games in the list.
Instructions	1.3	The instructions should be available via auditory cues, making it clear and comprehensible for visually-impaired players. The instructions must be short and concise and should take no longer than 1 minute to read.

		The instructions must specify the type and magnitude of penalties associated with losing, taking into account the specific mini-game and chosen difficulty level.
Inclusive Learning	4.4	The mini-game's content, challenges, and problems should align with the difficulty level chosen by the player. The player should not be allowed more attempts than the predetermined number associated with the selected difficulty level.
Rewards	1.4	All rewards must be announced using auditory prompts, ensuring players using screen readers are aware of their achievements. The player's gold coin counter attribute should be updated to reflect the awarded gold coins.
Penalty	2.5	All penalties must be announced using auditory prompts, ensuring players using screen readers are aware of their losses. The player's gold coin counter attribute should be updated to reflect the lost gold coins.
Customize Difficulty	3.2	The player should find distinct island classes categorized by unique themes. Each island should either be a fun, challenge game or an education game based on common subjects (ie. math, science, etc.)
Location indication	1.5	The location indicator is visible and operable using keyboard navigation. The audio cue for reaching a new island or grid cell is clearly audible with a screen reader. The screen reader announces the name of the current island or grid cell when it changes. The location indicator is visible and functional with screen magnification tools.
Customizable Grid Cells	3.4	Grid customization options are accessible through the settings menu. Players can adjust grid cell size and difficulty settings in the customization feature. Changes in grid cell size are visually noticeable when navigating the grid interface. Changes in difficulty settings result in visible modifications to game terrain and mechanisms.
Feedback on Island Arrival	4.6	Auditory cues are triggered upon reaching a new island. Visual cues or animations confirm the player's arrival at a new island. Celebratory feedback options are customizable through settings, allowing players to choose from different themes. The celebration feedback can be adjusted based on player preferences.

Return to Previously Visited Islands	2.6	Player's progress on previously visited islands is saved in a history log. A "/back" command is implemented, allowing players to retrace their steps to previously visited islands. The history log includes details about each island, such as its name, games, and performance. Players can revisit islands from their history log to improve their performance and earn additional gold.
Engaging Movement Experience	3.5	Engaging animations and/or sound effects are triggered during island-to-island movement. Event triggers provide audio and visual effects to captivate players during transitions. Players have the option to customize audio and images as a "loading screen" for personalization. Customizable audio and image options are accessible through the game's settings menu.
Navigating through islands within the grid	1.8	Provide a dropdown list or a grid of buttons displaying available islands for player selection. Implement a navigation method that, when an island is selected, changes the player's location to the chosen destination. Trigger entrance cues or feedback corresponding to the selected island upon arrival. Players can easily select islands from the provided list or grid for seamless navigation. **modify before submission
Save Game	2.1	Given I am a player in the game And I want to continue my adventure later without losing my progress When I choose to "Save Game" Then the game should save all relevant player data, including coins, lives left, etc. And the game should store the data in a serialized binary file And I should have the option to load this saved game data when I return to the game
Lives Left	1.1	Given I am a player in the game When I lose a life during gameplay Then the "remaining_lives" attribute associated with my player file decreases by 1 And the updated value of "remaining_lives" is displayed on my screen
Manage Inventory	3.1	Given I am a player within the game When I lose a life during gameplay Then the "remaining_lives" attribute associated with my player file is decremented by 1 And the current value of "remaining_lives" is visibly displayed on my user screen
Coins Collected	2.2	Given I am a player in the game When I access my inventory Then I can see a list of all the items I've collected during gameplay And I can interact with the inventory to manage, use, or discard items as needed

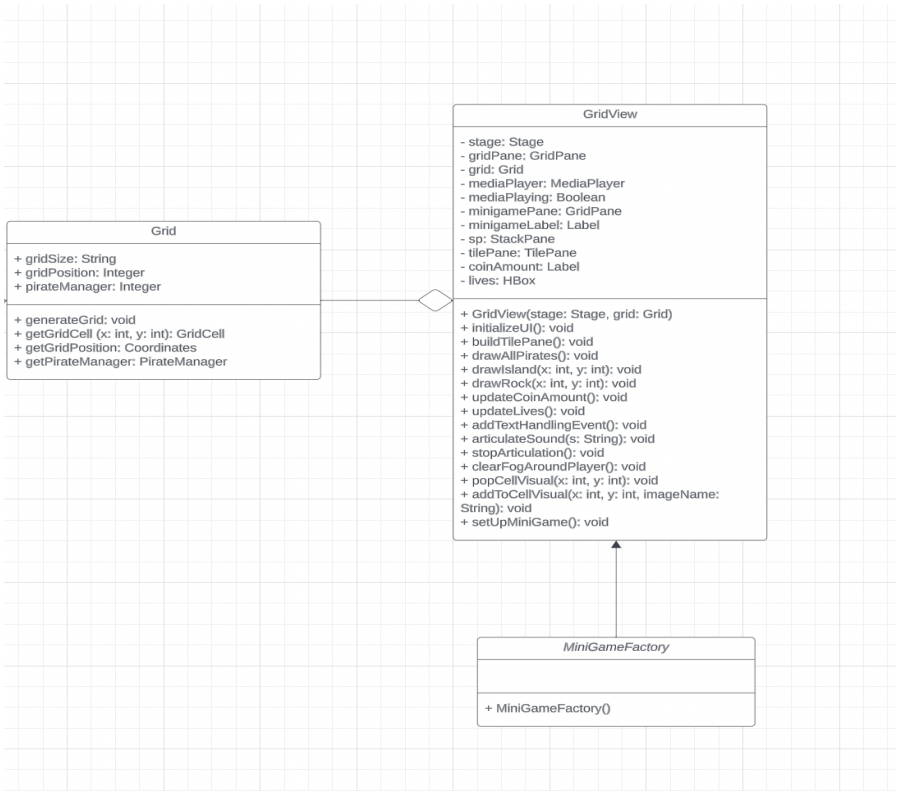
Set Name	1.5	Given I am a player within the game When I choose to update my player name Then the "rename" function is accessible to me And I can use it to change my displayed player name as desired
Visited	4.1	Given I am a player in the game When I want to determine whether I have visited a specific island Then I can access the "visited" attribute associated with that island And I can see whether it is marked as true, indicating that I have previously visited the island
Lose Entire Game	1.7	Given I am a player in the game When my "Lives Lost" attribute reaches 0 Then the game should conclude And I should be directed back to the home page And I should be given the option to restart the game for a fresh start
Pickup Item	2.3	Given I am a player exploring an island And there is an item available on the island When I choose to "Pick Up Item" Then the selected item should be added to my personal item list And the item should be removed from the island's item list And the item should no longer be available on the island
Drop Item	2.4	Given I am a player on an island And I have an item in my personal inventory When I choose to "Drop Item" on the island Then the selected item should be removed from my personal item list And the item should be added to the island's item list And the item should now be available for other players on the same island
Randomly Generated Grid	1.6	There should be a visible grid of water cells that the player is able to move through. Some of these cells should show themselves as having an island on them for the player to get to.
Procedural Generation of more grids	1.7	Moving to the edge of the current grid should move the player into an entirely newly generated grid. If the player reverses their direction, the grid they move into should be the exact same as it was before, not a new one.
Pirates in open water	2.7	There should be visible pirate ships on the map that if the player runs into, will initiate a sort of battle. If minigame is won, pirate should disappear off grid (sink).
Pirates moving	2.8	Every time the player moves, pirates should visibly move to a nearby cell. Pirates should only be observed to move

Fog around Unexplored areas	3.6	The map should start initially foggy except for the players immediate surrounding area. As the player moves, the area they move around should be cleared of fog and this should be saved whenever the player moves back and forth between grids. The fog should also hide pirate ships, islands, and anything else.
Randomly placed hidden and visible obstacles.	3.7	The map should be visibly populated with rocks or other obstacles that the player has to avoid. Some cells should also have hidden objects that the player will not see until crossing over that specific cell.
Different biome types	4.7	After moving through multiple grids, the player should notice a visible difference in the appearance of the grids corresponding to a specific biome with certain grids having special characteristics like “more pirates” or “more of a certain obstacle”.

SECTION 3: SOFTWARE DESIGN

Design Pattern #1: Strategy

Overview: This behavioral design pattern designs plans for different tasks separating algorithm from the user interacting with it.



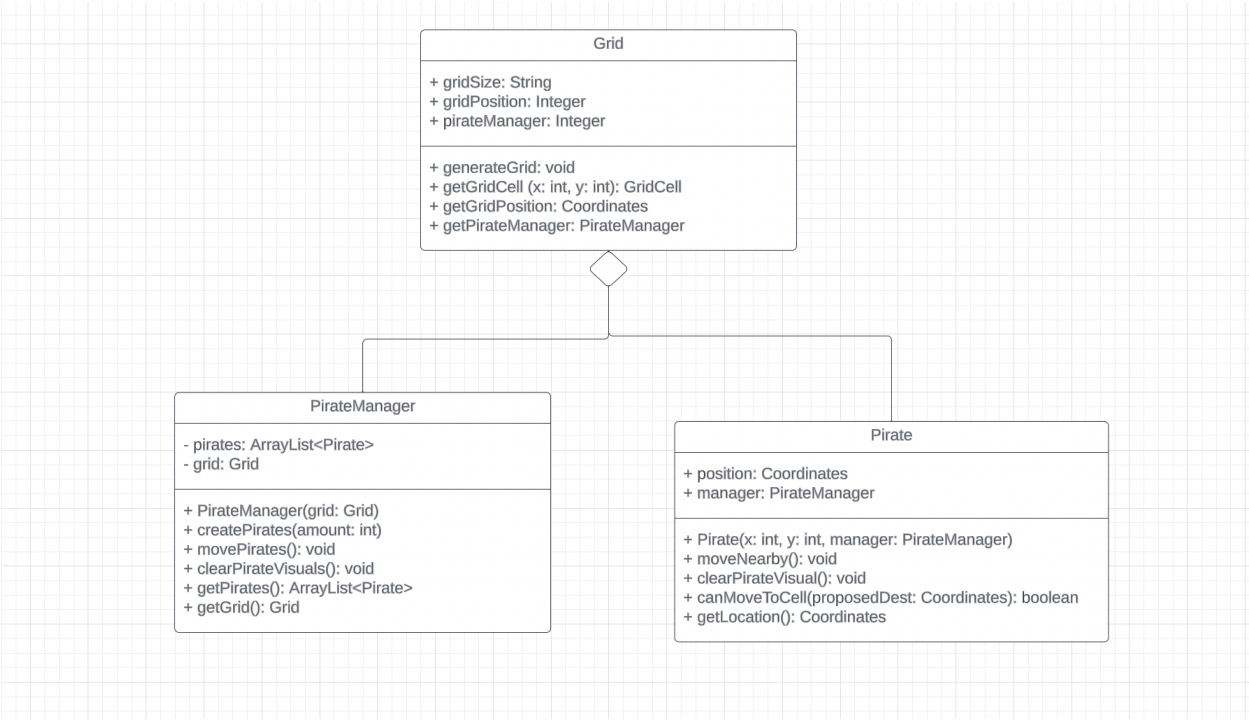
Implementation Details:

The UML diagram outlines The main components below:

- GridView uses Strategy Pattern for the color contrasts and sounds
- Based on a users selection, the color contrast of the entire game will change
- Based on a users actions, GridView interacts with the various classes to determine sounds

Design Pattern #2: Observer

Overview: This pattern provides a clear distinction between some subject and its observer, making it easy to make changes to the observer without modifying the subject. The fundamental principle is to create a relationship for communication without either subject or observer explicitly knowing.



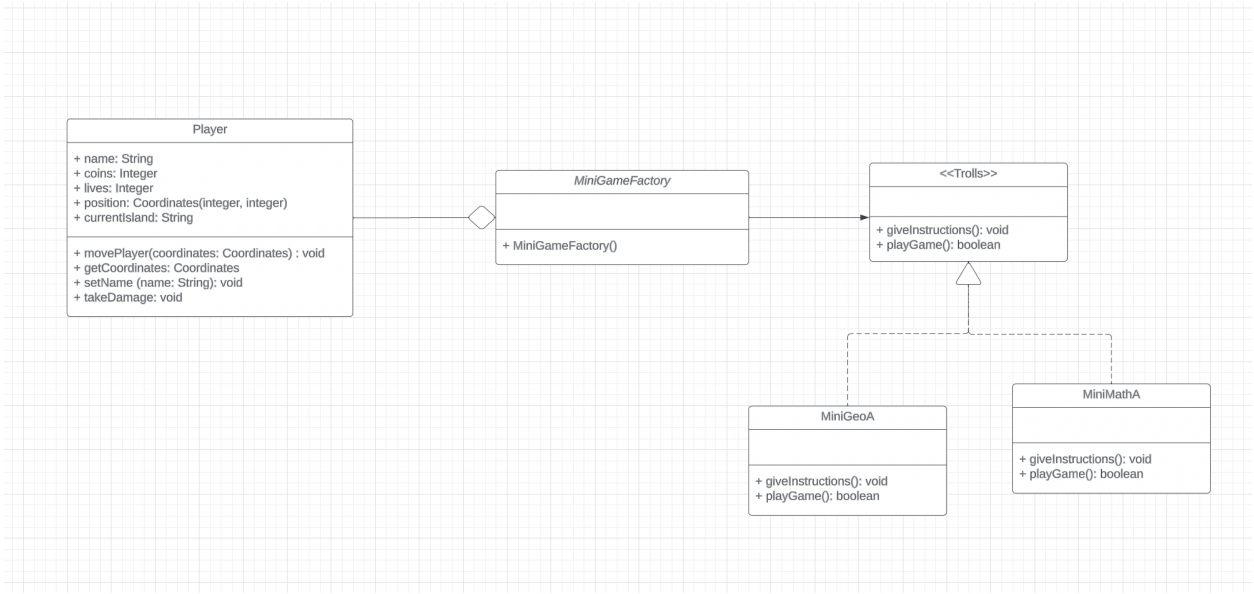
Implementation Details:

The UML diagram outlines The main components below:

- PirateManager acts as the subject which maintains the list of pirates as observers and sends notifications to registered pirates when changes occur
- Pirate acts as the observer which has the moveNearby and clearPirateVisual methods, specifying how the pirate reacts to changes.
- Pirate will also register to PirateManager to receive notifications or unregister to not receive notifications

Design Pattern #3: Factory

Overview: This creational pattern encourages flexibility and reusability in object creation by offering an interface to create objects while allowing subclasses to modify the kind of objects that will be created.



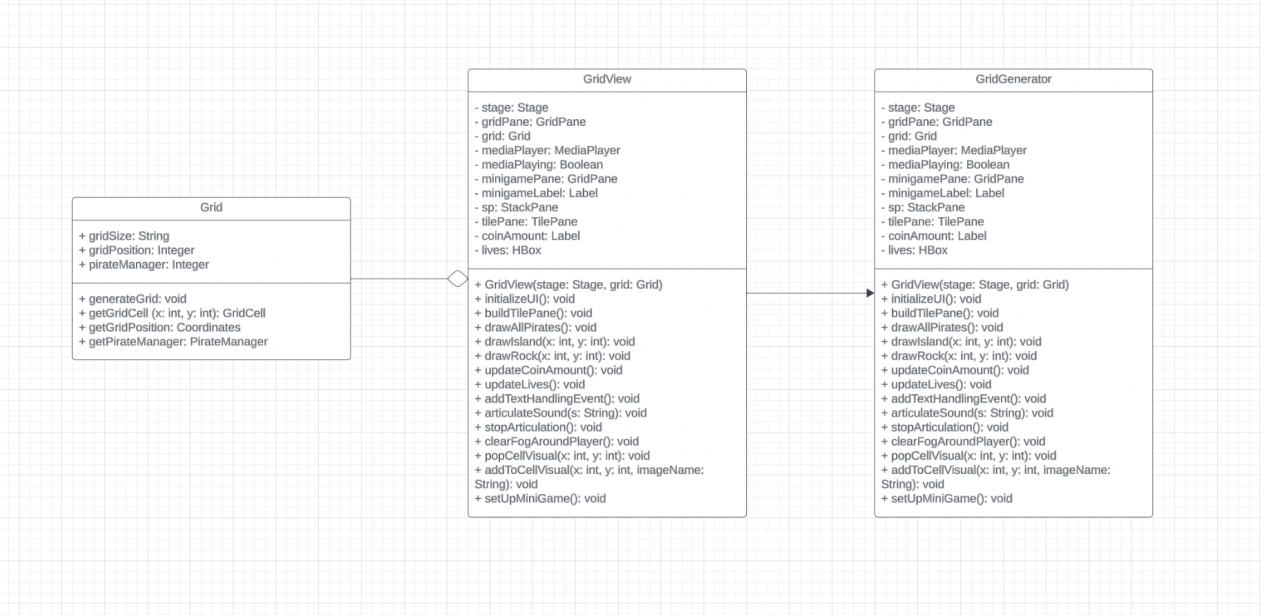
Implementation Details:

The UML Diagram Implements the main components below:

- The MiniGameFactory creates instances of various mini-games based on a specified Island given by the Player class, encapsulating the object creation process in a flexible and reusable manner.
- The Trolls interface is taken from Assignment 2 and is an abstract interface with an outline of what each MiniGame should have
- MiniGame's such as MiniGeoA and MiniMathA are just 2 of 13 smaller MiniGame's created that would inherit from the Trolls interface

Design Pattern #4: Model View Controller

Overview: This software design pattern divides program logic into three interconnected components: Model, View, and Controller. The Model oversees data management and business logic, the View displays information to users in a user-friendly graphical interface, and the Controller is like a communication pathway between the Model and View. This separation allows for a clear organization of responsibilities, with the Model handling data, the View managing user interface presentation, and the Controller handling user input and orchestrating updates to both the Model and View components.



Implementation Details:

The UML Diagram Implements the main components below:

- The model which is the Grid class manages the internal state of the grid, including information such as pirates
- Grid also provides methods to access and update the state of the grid
- GridView is the View aspect interacting with the JavaFX Framework to create a GUI which notices user inputs and updates the view accordingly
- GridGenerator is the Controller which manages the logic for generating, handles user input and translates it into actions that affect the model and view.