# Design Overview for Alien Blaster Program

Name: Hai Nam Ngo

Student ID: 103488515

# Summary of Program

“Alien Blaster” is an interactive C# gaming project where players are tasked with controlling a spaceship to shoot aliens within a specified time to accrue points. As players successfully shoot more aliens, the point tally increases. However, they must exercise caution to avoid contact with monsters and prevent monster invasions at the base.

Incorporating fundamental programming concepts, the game employs abstraction, allowing the spaceship to shoot without player knowledge of the shooting mechanism’s intricacies. The architecture is built around an Alien Class (parent), capable of moving and shooting. This foundational class spawns three distinct Alien Types (children), each inheriting the Alien Class's capabilities while introducing additional unique abilities, effectively demonstrating polymorphism in the game's design.



Figure Sample Output (Galaxy Attack: Shooting Game)

# Required Roles

Describe each of the classes, interfaces, and any enumerations you will create. Use a different table to describe each role you will have, using the following table templates.

Table 1: GameObject Class details

|  |  |  |
| --- | --- | --- |
| Responsibility | Type Details | Notes |
| \_Image | BitMap |  |
| \_x, \_y | Float |  |
| \_width, \_height | float |  |
| \_speed | Float |  |
| LoadImage() | <<abstract>> method | Abstract method that loads an image |
| Property X | Float | Get property for \_x |
| Property Y | float | Get property for \_y |
| Property Width | Float | Get property for \_width |
| Property Height | Float | Get property for \_height |
| IsAt(pt: Point2D) | boolean | Abstract method, check if object is at point pt |

Table 2: SpaceShip Class details

|  |  |  |
| --- | --- | --- |
| Responsibility | Type Details | Notes |
| \_ health | Int |  |
| \_bulletShape | Bullet |  |
| SpaceShip() | Constructor |  |
| LoadImage() | <<override>> method | Override abstract LoadImage() |
| Move() | void | Move the spaceship |
| Attack() | void | Spaceship attacks |
| IsAt(pt: Point2D) | boolean | Check if spaceship is at point pt |

Table 3: Alien Class details

|  |  |  |
| --- | --- | --- |
| Responsibility | Type Details | Notes |
| \_health | Int |  |
| \_bulletShape | Bullet |  |
| Alien | Constructor |  |
| Property Health | Int (read-only) | Get property for \_health |
| LoadImage() | <<abstract>> method | Abstract method that loads an image |
| Move() | Void | Abstract method, moves the Alien |
| Attack() | Void | Abstract method, Alien attacks |
| IsAt(pt: Point 2D) | boolean | Abstract method, check if Alien is at point pt |

Table 4: FireMonster Class details

|  |  |  |
| --- | --- | --- |
| Responsibility | Type Details | Notes |
| \_fireIntensity | Int |  |
| FireMonster() | Constructor |  |
| LoadImage() | <<override>> method | Override abstract LoadImage() |
| Move() | Void | Override abstract Move() |
| Attack() | void | Override abstract Attack() |

Table 5: AcidMonster Class details

|  |  |  |
| --- | --- | --- |
| Responsibility | Type Details | Notes |
| \_ acidStrength | Int |  |
| AcidMonster() | Constructor |  |
| LoadImage() | <<override>> method | Override abstract LoadImage() |
| Move() | Void | Override abstract Move() |
| Attack() | void | Override abstract Attack() |

Table 6: FlashMonster Class details

|  |  |  |
| --- | --- | --- |
| Responsibility | Type Details | Notes |
| \_flashTime | Int |  |
| FlashMonster() | Constructor |  |
| LoadImage() | <<override>> method | Override abstract LoadImage() |
| Move() | Void | Override abstract Move() |
| Attack() | void | Override abstract Attack() |

Table 7: GameSystem Class details

|  |  |  |
| --- | --- | --- |
| Responsibility | Type Details | Notes |
| Tick() | void |  |
| background | Bitmap |  |
| StartGame() | void | Starting the actual game |
| Quit | Void | Terminate the game |

Table 8: Alien details

|  |  |
| --- | --- |
| Value | Notes |
| Fire Monster | Slow, high damage |
| Acid Monster | Move zig zag, drain ship’s health |
| Flash Monster | High speed to crash the spaceship |

# Class Diagram

A screenshot of a computer

Description automatically generated

Although the acid, fire and flash look a little bit similar, but they have different movement and attack, even different health point, Fire and Acid were added special move for each of them, Flash remains the same.

# Sequence Diagram

Provide a sequence diagram showing how your proposed classes will interact to achieve a specific piece of functionality in your program.

A sequence event when player use spaceship to generate bullet, bullet come to acid monster, acid monster detech bullet and disapear

A diagram of a diagram

Description automatically generated