Connecting Intelligence

Institute of Computer Engineering Technology

iCET Certified Developer

Coursework

Assignement	Object Oriented Programming
Batch No	iCD 113
Name	Defense System
Ass. Date	16th November 2024

Defense System

Introduction:

The Defense System project aims to create a simulated system that manages and monitors various defense units, including Helicopters, Tanks, and Submarines. The project emphasizes communication, coordination, and information exchange between these defense units.

Project Overview:

The project seems to simulate a defense system with various components, such as helicopters, tanks, and submarines. These components are represented as Swing GUI windows (Helicopter, Tank, and Submarine classes) and interact with a central controller (MainController) and an observer pattern (Observer).

Project Requirements

- Simulate a defense system with different types of units.
- Each unit has unique features and attributes.
- Implement communication and coordination between defense units.
- Visualize the system with graphical user interfaces for each defense unit.
- Provide a central controller (Observer) for managing and observing the system.
- Simulate strength updates, area messages, and message broadcasts.

Case Study

The Defense System project can be applied to real-world scenarios where military or defense systems need a centralized control mechanism to coordinate various units. The use of graphical interfaces for each unit enhances user interaction and monitoring capabilities. The project allows for scalability by easily adding new defense units and extending functionalities. It promotes modularity, making it adaptable for different defense strategies and scenarios.

Helicopter, Tank, Submarine (SuperDefence)

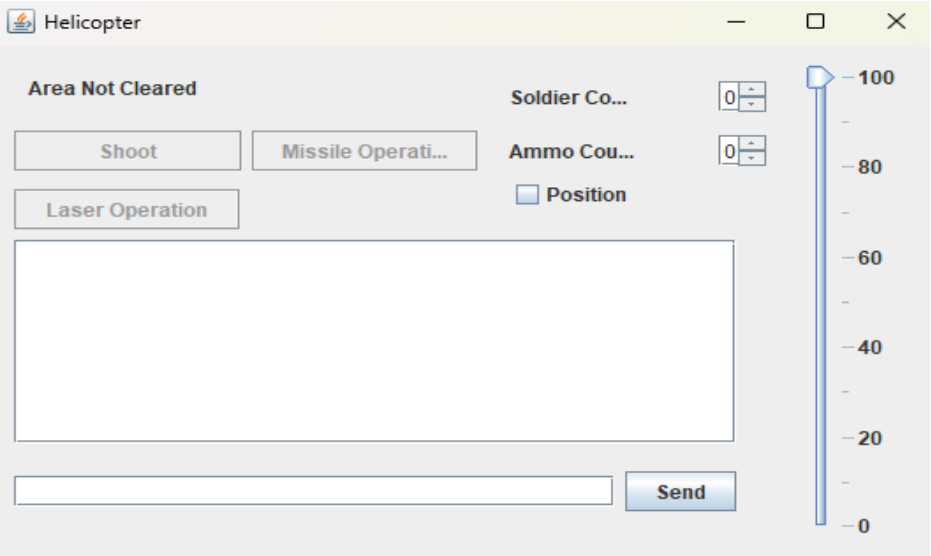
- These classes represent different defense components.
- Each component has GUI elements for controlling soldiers, ammunition, and displaying messages.

Strength Enum

- Represents different strength levels (LOW, MEDIUM, HIGH, STRONG, CLOSED).

Components

1. Helicopter



Role and Characteristics

- Visualize the Helicopter as a defense mechanism with aerial capabilities.
- Consider its unique characteristics, such as mobility, firepower, and capacity for aerial operations.

User Interaction

- Conceptualize how a user interacts with the Helicopter through a graphical user interface.
- Define actions like shooting, missile operations, and laser operations, considering their impact on the Helicopter's state.

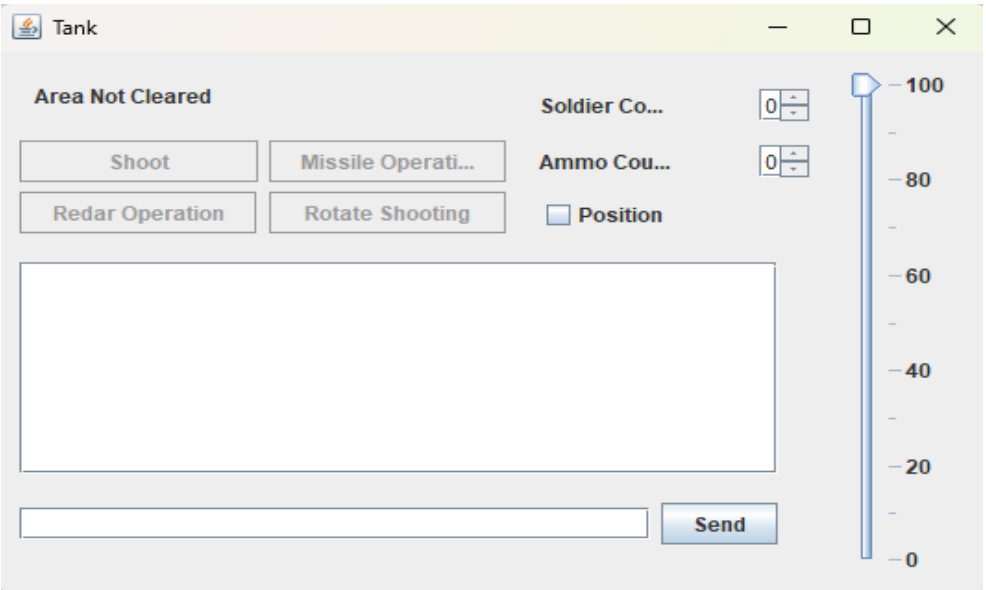
Observer Interaction

- Understand how the Helicopter acts as an observable entity, notifying the MainController about changes in its area, war strength, and messages.
- Define events triggering updates and communications with the Observer pattern.

Communication with MainController

- Determine how the Helicopter reports area messages, updates the message box, and communicates its war strength to the MainController.
- Define how the Helicopter sends messages to the main controller, contributing to the overall defense system.

2. Tank



Role and Characteristics

- Envision the Tank as a ground-based defense mechanism with significant firepower and armored capabilities.
- Consider its role in ground operations, emphasizing strength, durability, and effectiveness in combat scenarios.

User Interaction

- Conceptualize user interactions with the Tank, focusing on shoot operations and other tank-specific functionalities.
- Define how actions impact soldier count, ammo count, and other Tank-specific attributes.

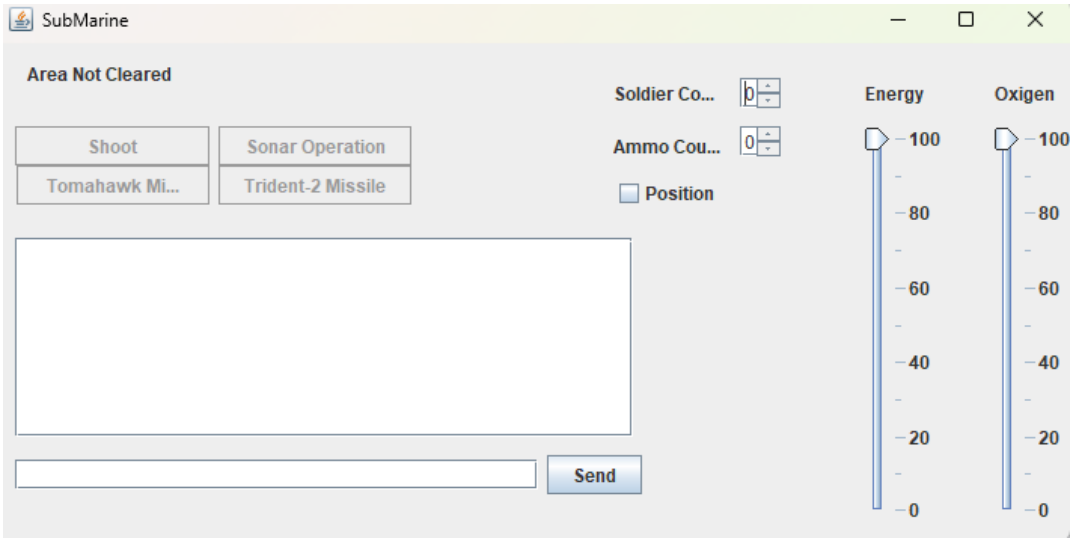
Observer Interaction

- Understand the Tank's role as an observable entity, communicating with the MainController about changes in its area, war strength, and messages.
- Define events that trigger updates and interactions with the Observer pattern.

Communication with MainController

- Determine how the Tank reports area messages, updates the message box, and communicates its war strength to the MainController.
- Define how the Tank sends messages to the main controller, contributing to the overall defense system

3. Submarine



Role and Characteristics

- Envision the Submarine as a defense mechanism specialized for underwater operations.
- Consider its unique capabilities, such as deploying missiles, sonar operations, and shooting underwater.

User Interaction

- Conceptualize user interactions with the Submarine, focusing on underwater shoot operations, missile deployments, and sonar activities.
- Define how these actions impact energy, oxygen levels, and other Submarine-specific attributes.

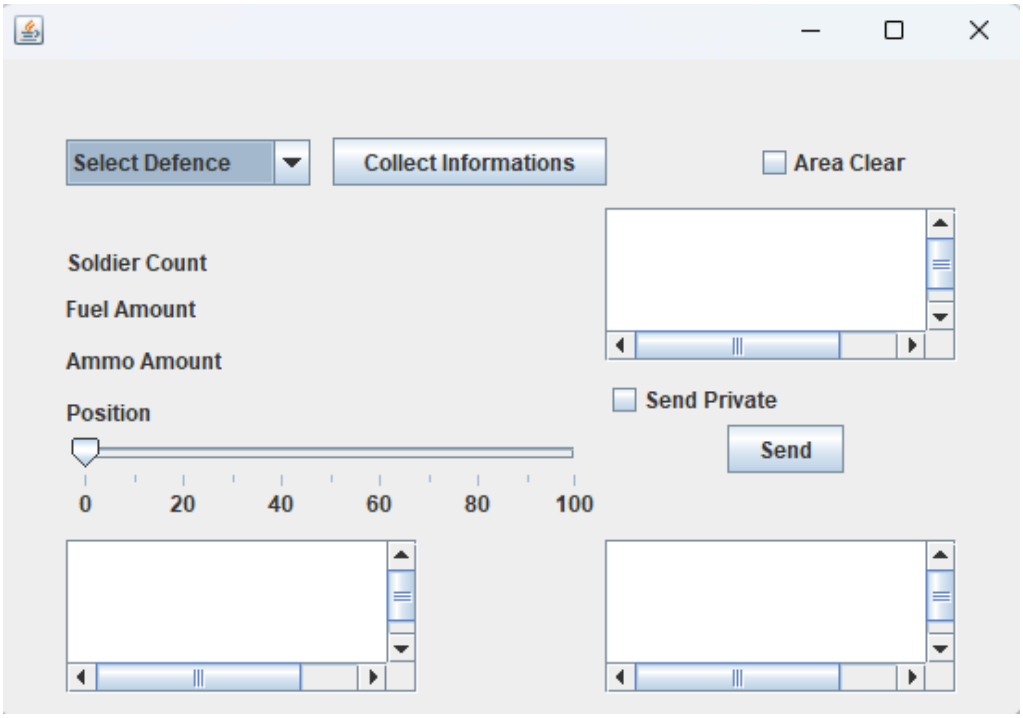
Observer Interaction

- Understand the Submarine's role as an observable entity, communicating with the MainController about changes in its area, war strength, and messages.
- Define events that trigger updates and interactions with the Observer pattern.

Communication with MainController

- Determine how the Submarine reports area messages, updates the message box, and communicates its war strength to the MainController.
- Define how the Submarine sends messages to the main controller, contributing to the overall defense system.

4. MainController



Observer Management

- Visualize the MainController as the central coordinator, managing interactions among different defense mechanisms.
- Understand how it maintains a collection of observables (Helicopter, Tank, Submarine) and facilitates communication.

Strength Calculation

- Conceptualize the MainController's role in calculating overall defense strength based on updates from individual defenses.
- Utilize the Observer pattern to broadcast strength changes to all defenses, maintaining a cohesive defense system.
- Envision the MainController as the hub for area messages and global communication.
- Relay area messages, update the message box, and facilitate message exchanges among different defense mechanisms.

Global Operations

- Define how the MainController handles global operations that involve multiple defenses working together.
- Act as a mediator for interactions among different defense mechanisms, ensuring coordinated responses.

Flow

Initialization

- The Starter class initializes the observer and the main controller.
- Instances of Helicopter, Tank, and Submarine are created and associated with the observer.

User Interaction

- Users interact with the GUI of each defense component (e.g., shooting, sending messages).
- GUI actions trigger events that are handled within each defense component.

Strength Signal

- The Starter class sends an initial strength signal to all observables, simulating a change in the defense system's strength.

Defense Units: Helicopter, Tank, Submarine

- Each defense unit extends the SuperDefence class, providing common functionalities.
- The Helicopter, Tank, and Submarine classes implement the Observable interface.
- They have specific features such as shooting, missile operations, laser operations, and unique attributes like soldier count, ammo count, energy, and oxygen levels.