

# I Am EOD

## 1. Project Overview

You are an EOD (Explosive Ordnance Disposal) specialist, and your mission is to defuse a bomb before it detonates. The bomb consists of two modules:

1. **Password Module** – To access the bomb's main board, you must enter the correct password by solving a sequence of stages.
2. **Wire Module** – Once the main board is open, you must identify and cut the correct wire to stop the bomb.

You will receive a **Bomb Defusal Manual** that provides the logic for solving each module. However, the correct solution depends on the specific details observed in the current situation.

### Gameplay Rules:

- **Two-strike system:** You are allowed **up to two mistakes** before the bomb detonates on the third strike.
- **3-minute time limit:** Complete the defusal before time runs out.
- **Real-time interaction:** Players must make quick, logical decisions under pressure.

## 2. Project Review

### Existing Project: "Keep Talking and Nobody Explodes"

A popular bomb-defusal simulation that emphasizes teamwork, communication, and problem-solving. Players must work together, one defusing the bomb and the other providing instructions from a manual to solve various complex modules before time runs out.

### Improvements:

**Simplified Modules:** Instead of overly complex puzzles, this version will feature easier-to-understand mechanics, ensuring players can enjoy the challenge without feeling overwhelmed.

**Solo-Friendly:** While the original game heavily relies on multiplayer interaction, this project is designed to be solo-friendly, allowing players to defuse the bomb on their own if needed.

**However, it strongly encourages playing with friends to maximize the fun and challenge.** The game mechanics are balanced to ensure that while solo play is possible, the experience is more engaging and rewarding when players collaborate with others.

### **3. Programming Development**

#### **3.1 Game Concept**

##### **Mechanics and Objectives:**

**Password Module:** Players must press the correct buttons based on the displayed number to progress through stages.

**Wire Module:** Players must identify and cut the correct wire based on the given rules.

**Bomb Defusal Manual:** Players must download the defusal manual before playing the game.

##### **Key Features:**

Time-limited gameplay.

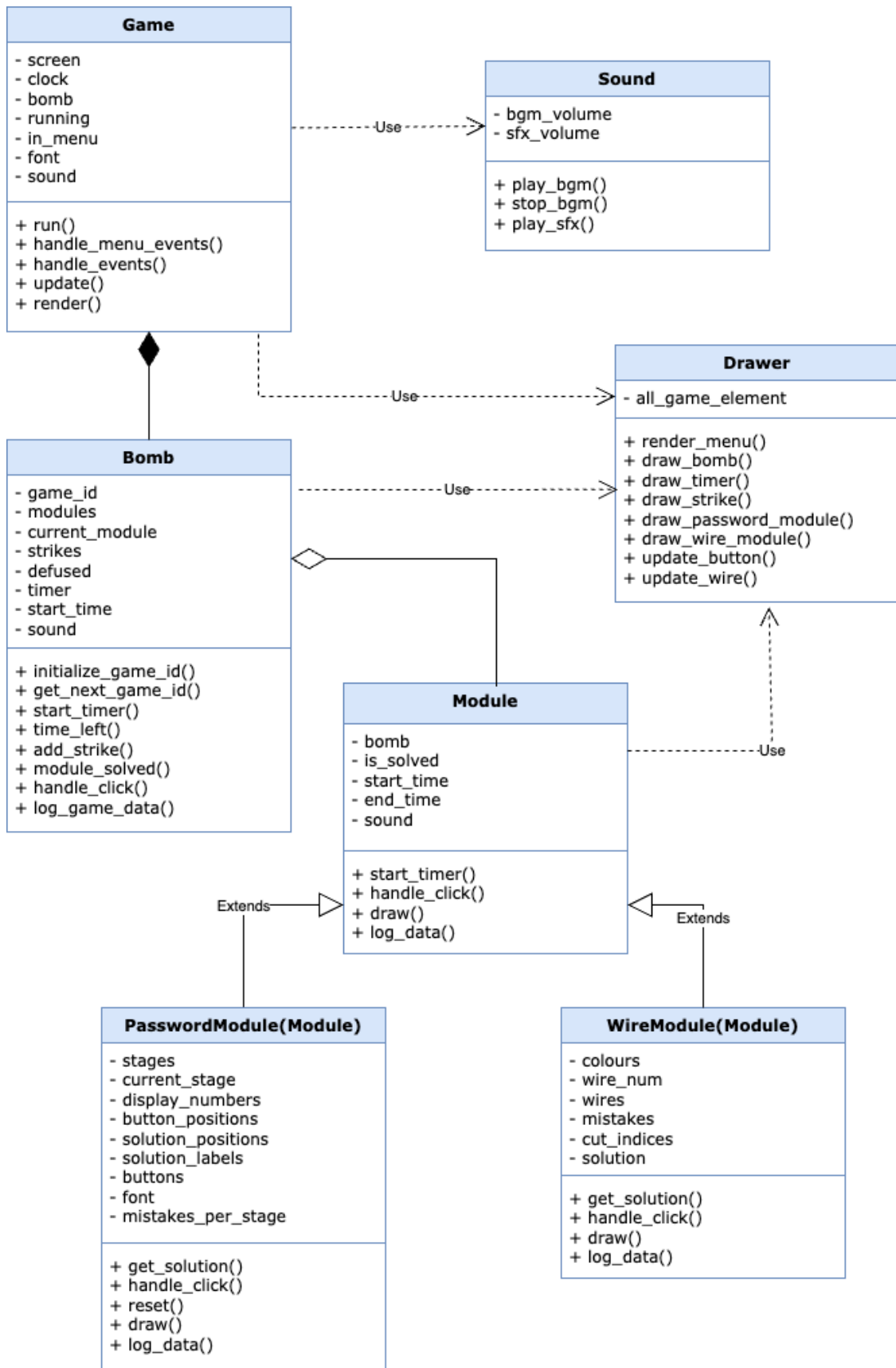
Multiple stages and increasing difficulty.

Randomized challenges for replayability.

Detailed performance tracking.

#### **3.2 Object-Oriented Programming Implementation**

1. **Bomb** - Manages game state, timer, strikes, and modules.
2. **Module (Base Class)** - Abstract class for all bomb modules.
3. **PasswordModule (Inherits from Module)** - Implements password-solving logic.
4. **WireModule (Inherits from Module)** - Handles wire-cutting mechanics.
5. **Game** - Controls the game loop, rendering, and events.
6. **Drawer** - Renders UI elements (menu, bomb, modules).
7. **Sound** - Manages background music & SFX.
8. **Stats** - Generates performance visualizations.



### **3.3 Algorithms Involved**

- **Rule-Based Logic:** Each module follows specific rules that dictate the correct solution.
- **Randomization:** Bomb configurations, including passwords and wire colors, are randomized in each game to ensure replayability.
- **Decision Trees:** Used to determine outcomes based on player choices (e.g. whether the bomb is successfully defused or detonates).

## **4. Statistical Data (Prop Stats)**

### **4.1 Data Features**

<b>Metric</b>	<b>Purpose</b>
Game Completion Rate	Measures success vs. failure rates.
Time Taken vs. Mistake Rate	Correlates speed & accuracy.
Most Common Failure Reasons	Identifies key failure points.
Mistake Distribution by Stage	Highlights difficult password stages.
Module Completion Time	Compares time taken per module.
Attempts vs. Success (Wire Module)	Shows a learning curve.

### **4.2 Data Recording Method**

Data is logged into CSV files (game.csv, wire.csv, password.csv) after each game session.

### **4.3 Data Analysis Report**

Feature	Graph Type	X-Axis	Y-Axis
Game Completion Rate	Pie Chart	Success/Failure	Percentage
Time Taken vs. Mistakes	Scatter Plot	Time (s)	Mistakes
Failure Reasons	Bar Chart	Failure Type	Frequency
Mistake Distribution	Stacked Bar	Stages	Mistake Count
ModuleTime	Boxplot	Module Type	Time (s)
Attempts vs. Success	Line Graph	Attempts	Success Rate (%)

### **5. Project Timeline**

Week	Task
1 (10 March)	Proposal submission / Project initiation
2 (17 March)	Full proposal submission
3 (24 March)	Statistic visualization plan submission
4 (31 March)	Final UML & data collection refinement
5 (7 April)	Draft TKinter visualization
6 (14 April)	Submission (Draft)
7(11 May)	Final submission (100%)

### **6. Document version**

Version: 5.0

Date: 11 May 2025

