

EPIC RPG V2

A Text-Based Python Adventure

Submitted for Master of Computer Applications (M.C.A.) | Academic Year 2025–26

```
{  
    "Adil Belim": "(92400584032)"  
}
```

```
{  
    "Krupali Shah": "(92400584055)"  
}
```

```
{  
    "Niyati Bhanani": "(92400584019)"  
}
```

THE BUG

Beginner programmers often struggle to understand complex game logic—persistence, combat math, state management—because they are distracted by rendering graphics.



THE PATCH

A pure text-based RPG. By removing the GUI, the focus shifts entirely to Python programming concepts.

Core Objectives

- ✓ Implement secure User Management (Auth/Login)
- ✓ Create a persistent world using JSON file storage
- ✓ Develop modular game systems (Combat, Economy, Inventory)

SYSTEM SPECIFICATIONS & TECH STACK

ENVIRONMENT

Language: Python 3.x

Interface: Terminal / Console Window

OS: Windows 10+ / Linux / macOS 11+

Loading: Python 30.1

Code: Manua 36 MHz

HARDWARE

CPU: Intel/AMD Equivalent

RAM: 2 GB Min / 4 GB Rec

Storage: 200-300 MB (Log/Save Expansion)

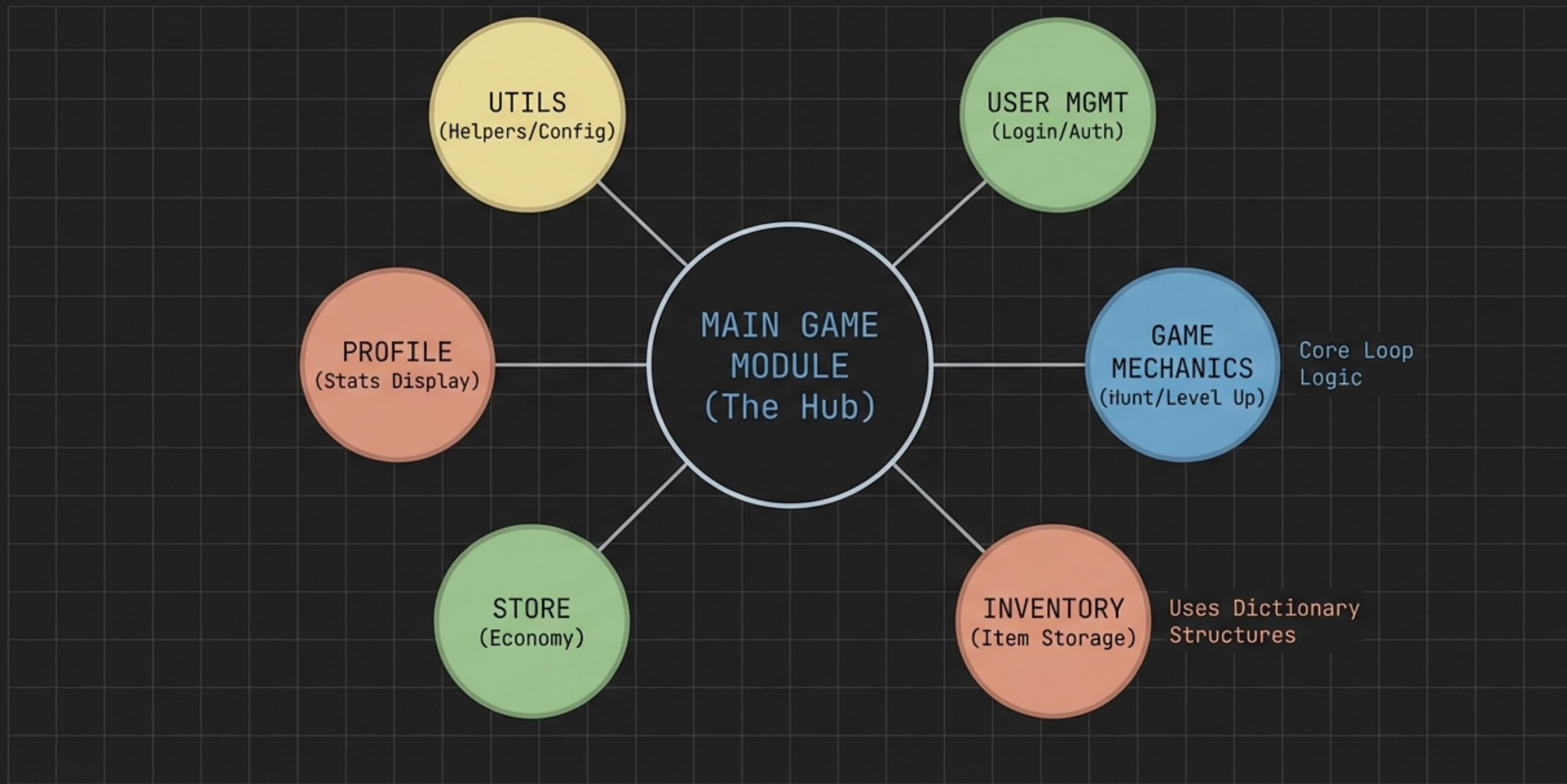
PERSISTENCE

Format: JSON Structure

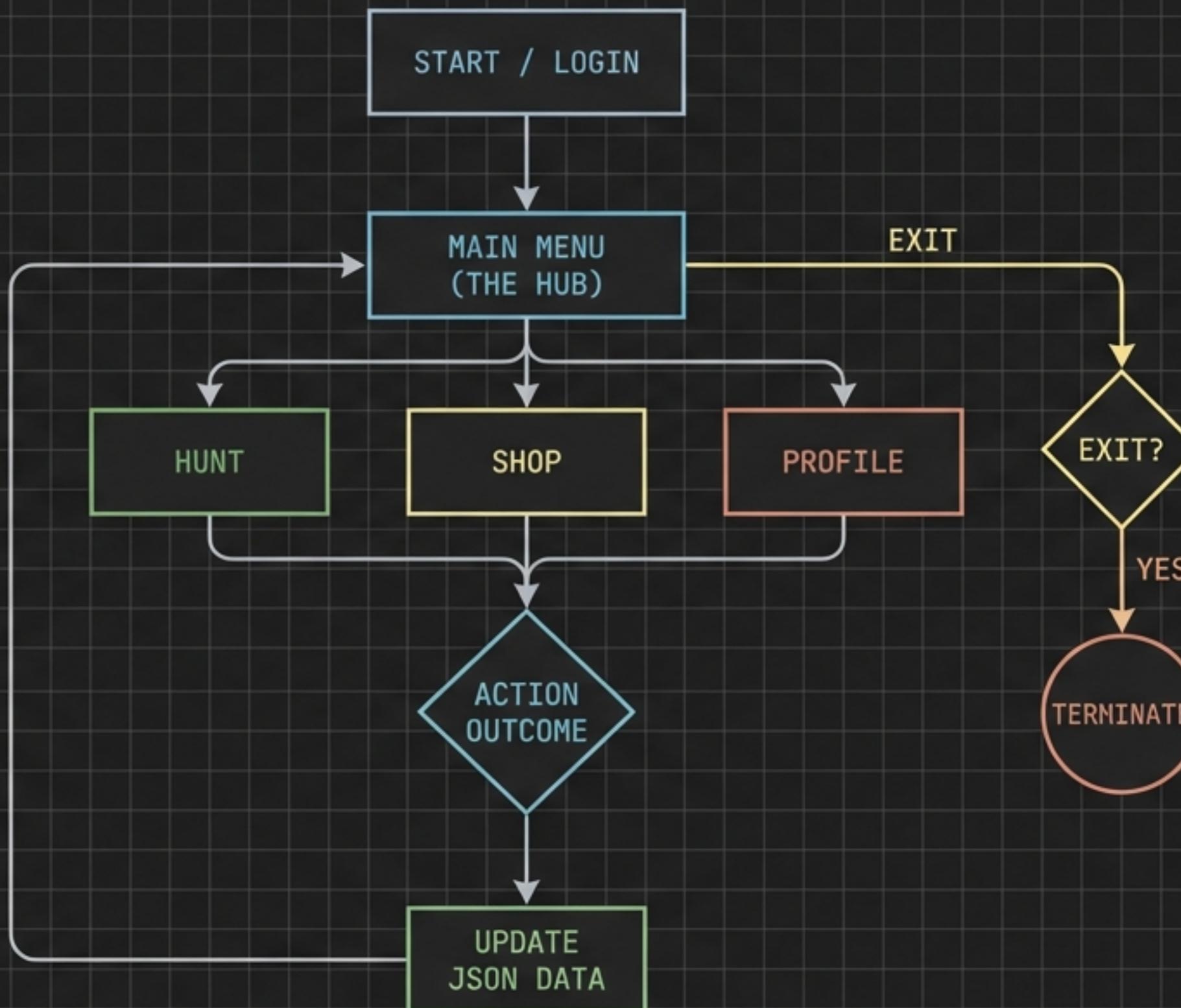
Method: File-Based I/O

Target: user_data.json

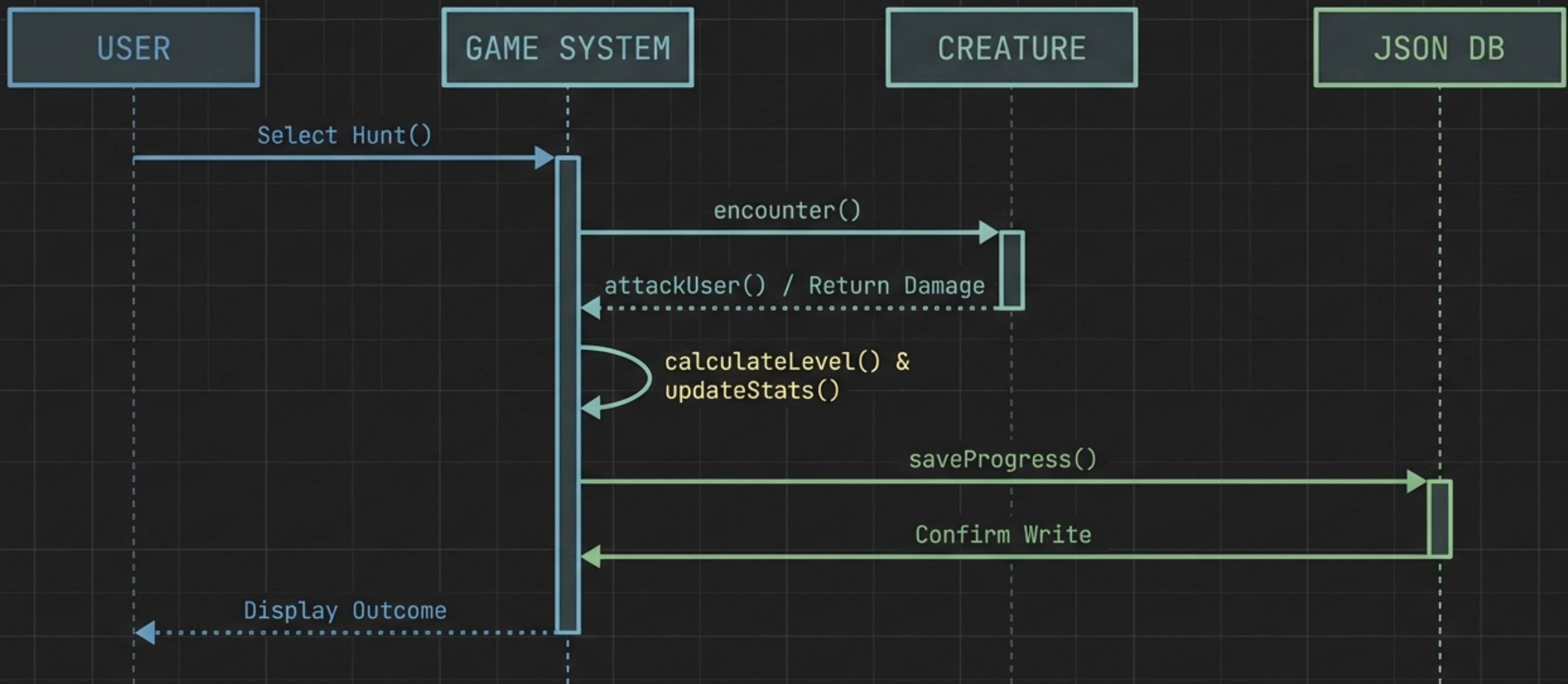
MODULAR ARCHITECTURE



THE LOGIC LOOP



DATA SEQUENCE: ANATOMY OF A HUNT



GAMEPLAY MECHANICS: RISK VS REWARD

THE HUNT



Standard expedition. Moderate risk, consistent XP gain.
Calls `rpg_hunt()`.

THE ADVENTURE



Standard expedition. High stakes.
Single powerful creature encounter.
`Hiraikankep`.
Calls `rpg_adventure()`.

PROGRESSION MATH

Level is calculated dynamically:
`get_level()` checks total XP against thresholds.
`check_level_up()` awards stat boosts.

```
if xp > threshold:  
    level += 1  
    stats['str'] += 2
```

THE ECONOMY: CRAFTING & COMMERCE

THE STORE



Exchange earned **Coins** for items.
Validates balance before transaction.

CRAFTING LOGIC



$$[\text{Logs}] + [\text{Coins}] = [\text{Wooden Sword}]$$

$$[\text{Gemstone}] + [\text{Coins}] = [\text{Clover}]$$

INVENTORY MANAGEMENT



The `view_inventory()` function bridges gathering and crafting, displaying current asset snapshot.

INTERFACE DESIGN: THE CLI DASHBOARD

- 1. 🎯 Hunt
- 2. 🗺 Adventure
- 3. 💊 Heal using Life Potion
- 4. 📦 View Inventory
- 5. 🧑 View Profile
- 6. 🏪 Store
- 7. 🌳 Chop Wood
- 8. 🎁 Reward
- 9. 🛠 Craft
- 10. 🗑 Delete User
- 11. 🚪 Exit

Visual Scanning: Emojis act as anchors in text-heavy UI.

Speed: Numeric input (1-11) for rapid execution.

Feedback Loop: Immediate return to menu after action.

QUALITY ASSURANCE: BLACK BOX TESTING

TEST CASE	ACTION	RESULT
TC-02	Valid Login Credentials	Pass: Access Granted
TC-03	Invalid Credentials	Pass: Access Denied
TC-09	Craft Lucky Charm	Pass: Stat Increased + Cooldown
TC-15	Exit Command	Pass: Safe Termination



Methodology: Manual terminal input validation covering boundary cases and logical errors.

DEVELOPMENT ROADMAP

PHASE 1: FOUNDATION

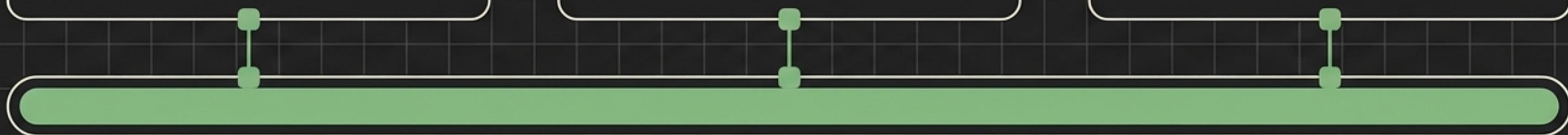
User Registration,
Login, Menu Navigation.

PHASE 2: THE CORE

Hunting, Adventure,
Inventory Handling.

PHASE 3: COMPLEXITY

Store, Crafting, Luck
Attributes, Cooldowns.



Strategy: Iterative implementation. Each feature tested in isolation before main module integration.

DEVELOPER'S LOG: MISSION DEBRIEF

LEVEL UP!

DATA STRUCTURES

Mastery of Dictionaries & Lists for inventory management.

FILE I/O

JSON implementation for persistent world state.

ARCHITECTURE

Modular design, breaking complex problems into 7 interacting components.

> Role-Playing relies on imagination and logic, not just graphics. We successfully built a stable, engaging world entirely in Python.

CREDITS & REFERENCES

TEAM

Adil Belim
(92400584032)

Krupali Shah
(92400584055)

Niyati Bhanani
(92400584019)

GUIDANCE

Internal Guide:
Prof. Dr. Ashwin Dobariya

HOD:
Dr. Sunil Bajeja

Dean:
Dr. R. Sridaran

Company Profile:
Galamine AI
(Amri Foundation)

REFERENCES

W3Schools
(Python Basics)

EPIC RPG Wiki
(Mechanics)

Certificate



Marwadi
University
Marwadi Chandarana Group