

Kafes Game Design 1

Group Name: **KAFES-A**

Group Members:

- Abdullah Tunçer
- Ali Han Kılınç
- Engin Sühan Bilgiç
- Faruk Aksøy
- Kayrahan Yiice
- Sermet Arda Topal

Contents

1 UML Package Diagram	1
2 UML Sequence Diagrams	2
3 UML Communication Diagrams	12
4 UML Class Diagrams	16
5 Design Alternatives and Their Evaluation	17
5.1 Alternative 1: Observer Pattern	17
5.2 Alternative 2: Using Swing Layout Managers	17
5.3 Alternative 3: Using Online Real-Time Sync MongoDB	17

1 UML Package Diagram

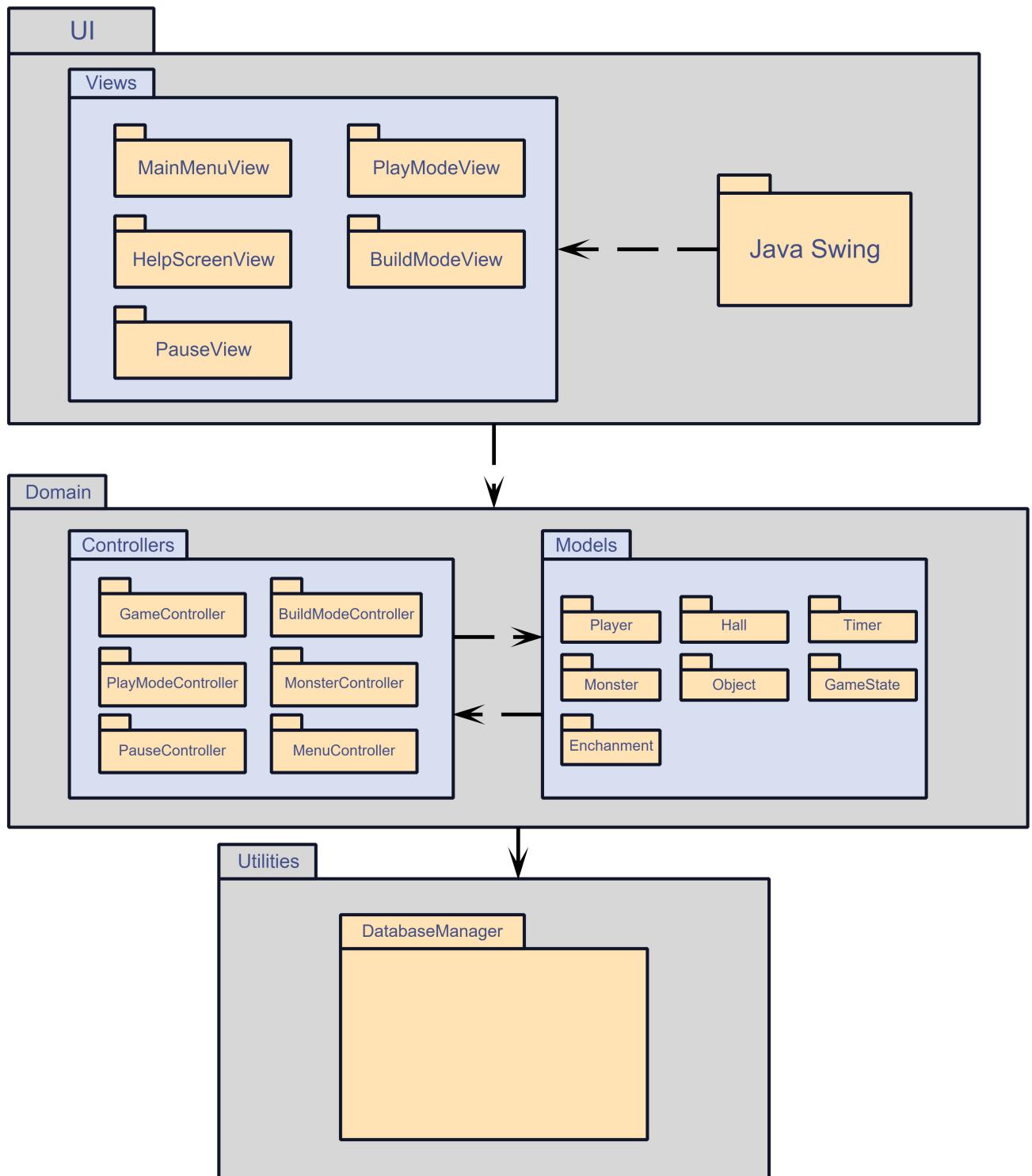


Figure 1: UML Package Diagram

2 UML Sequence Diagrams

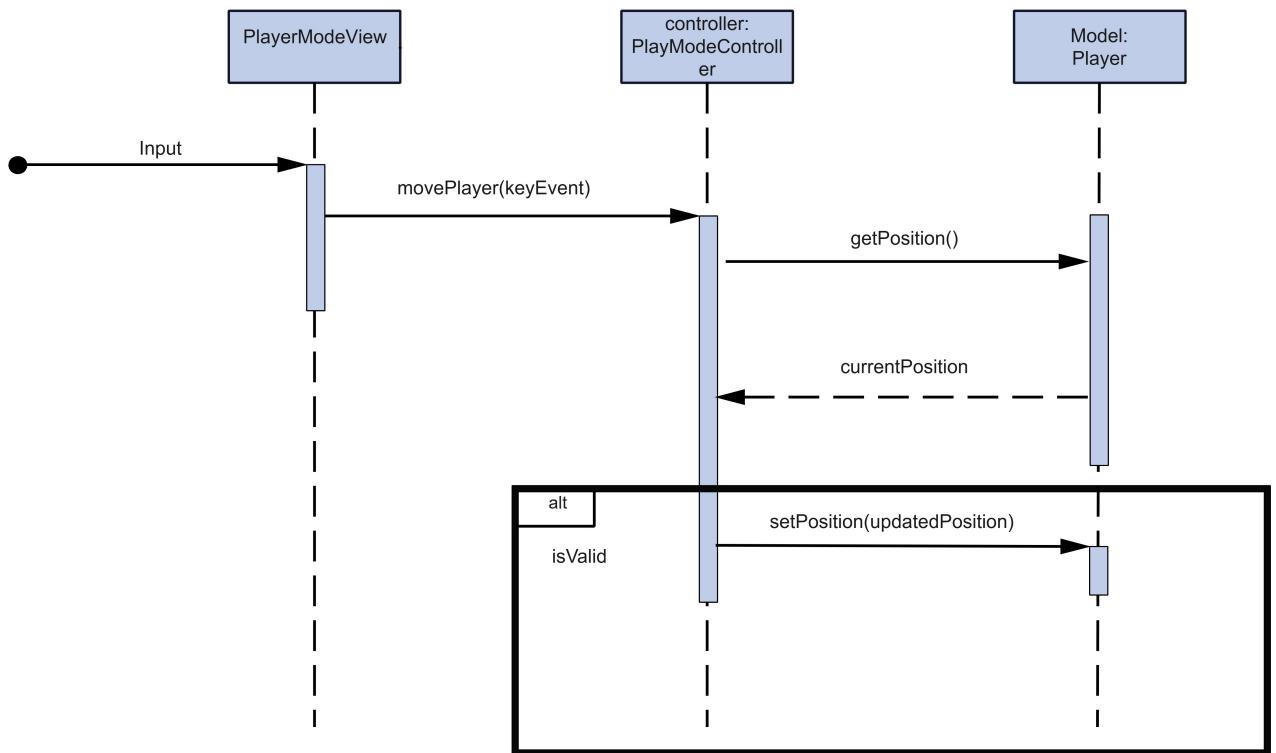


Figure 2: Move Character Sequence Diagram

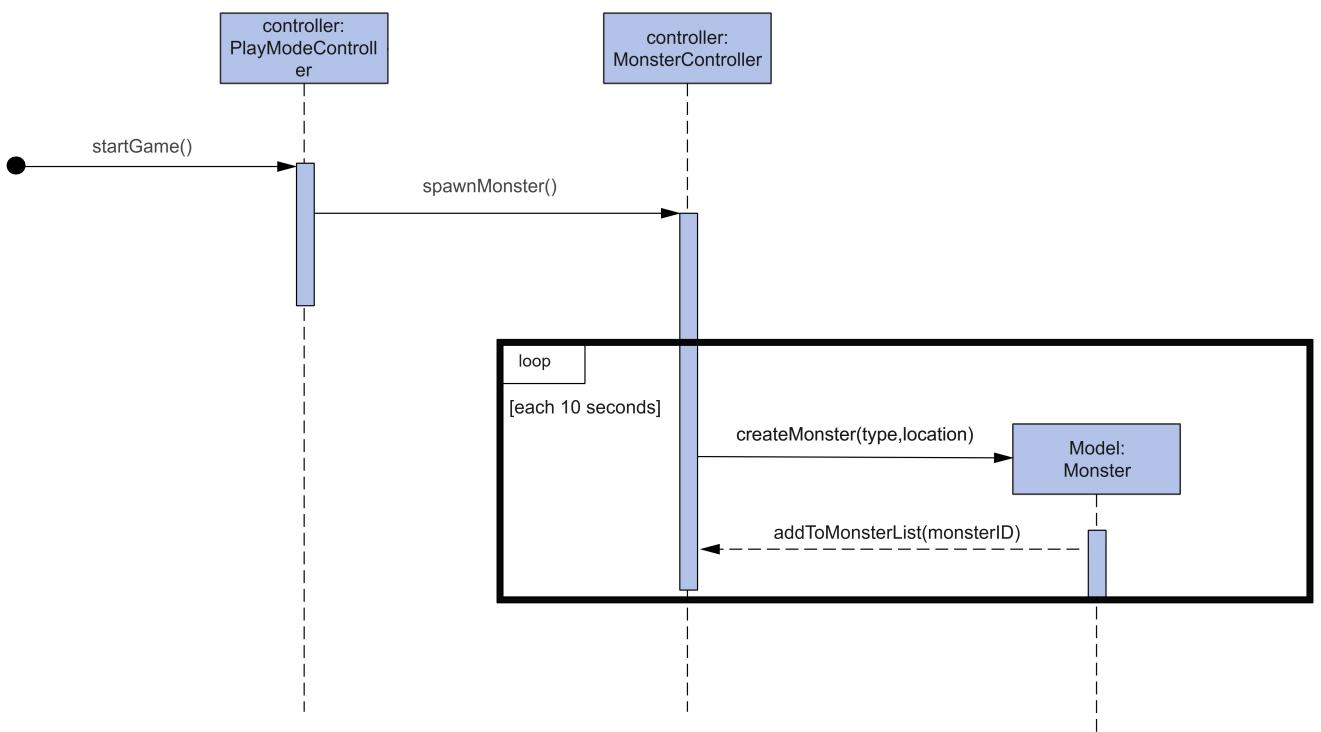


Figure 3: Spawn Monster Sequence Diagram

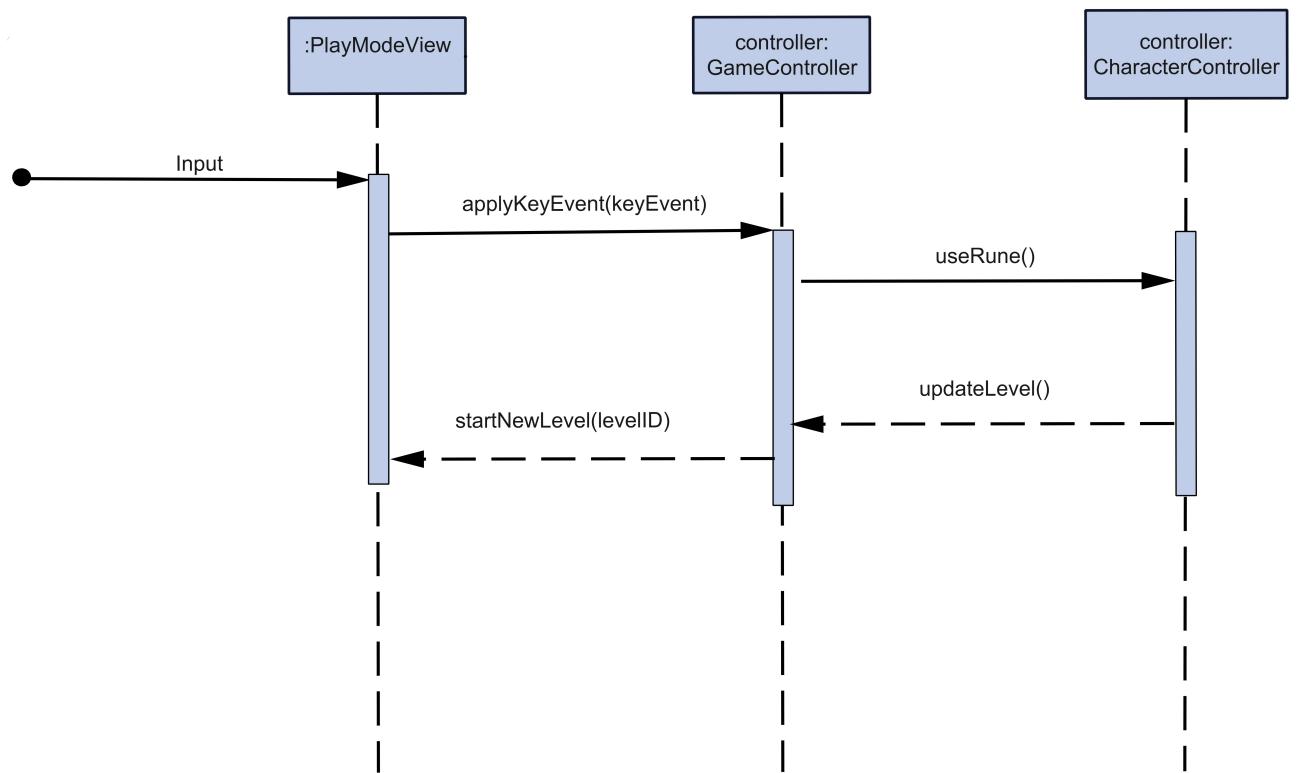


Figure 4: Use Rune Sequence Diagram

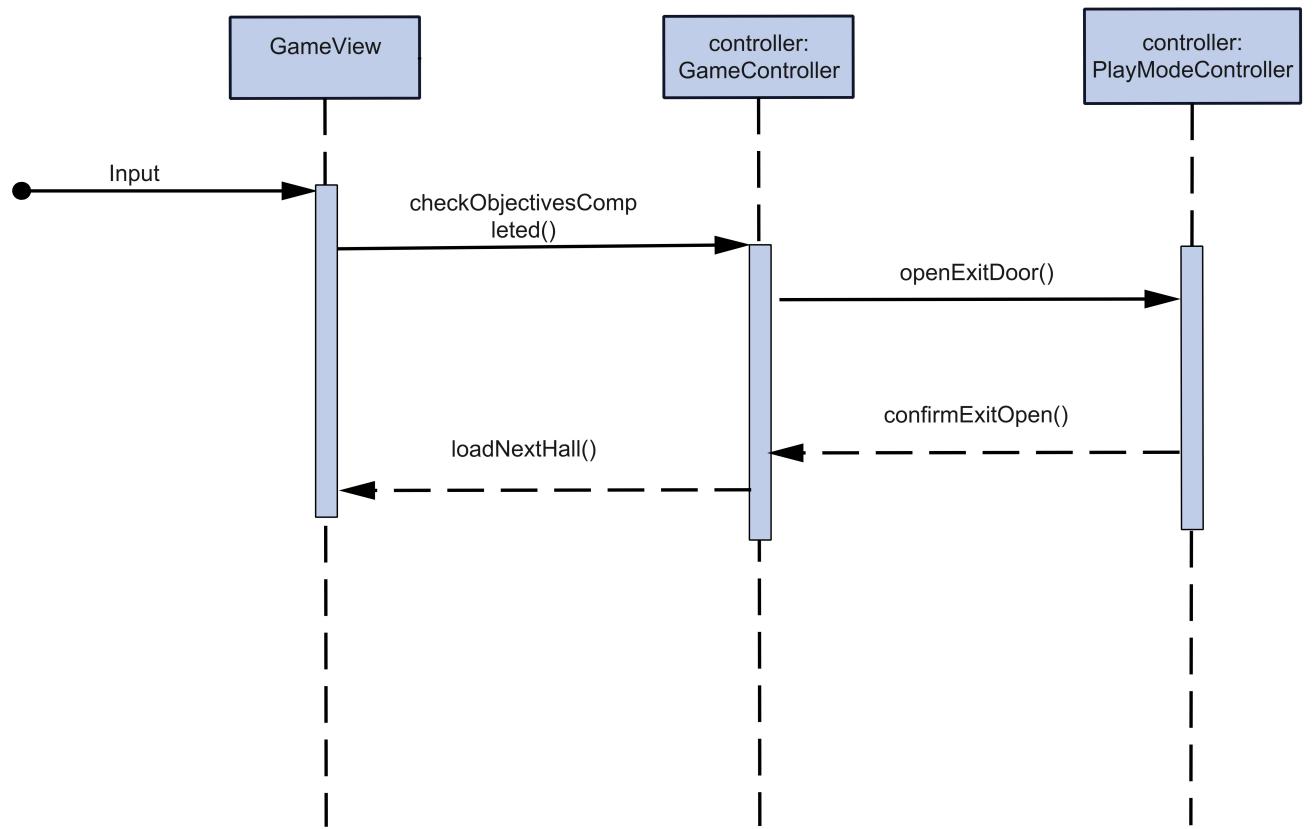


Figure 5: Progress to Next Hall Sequence Diagram

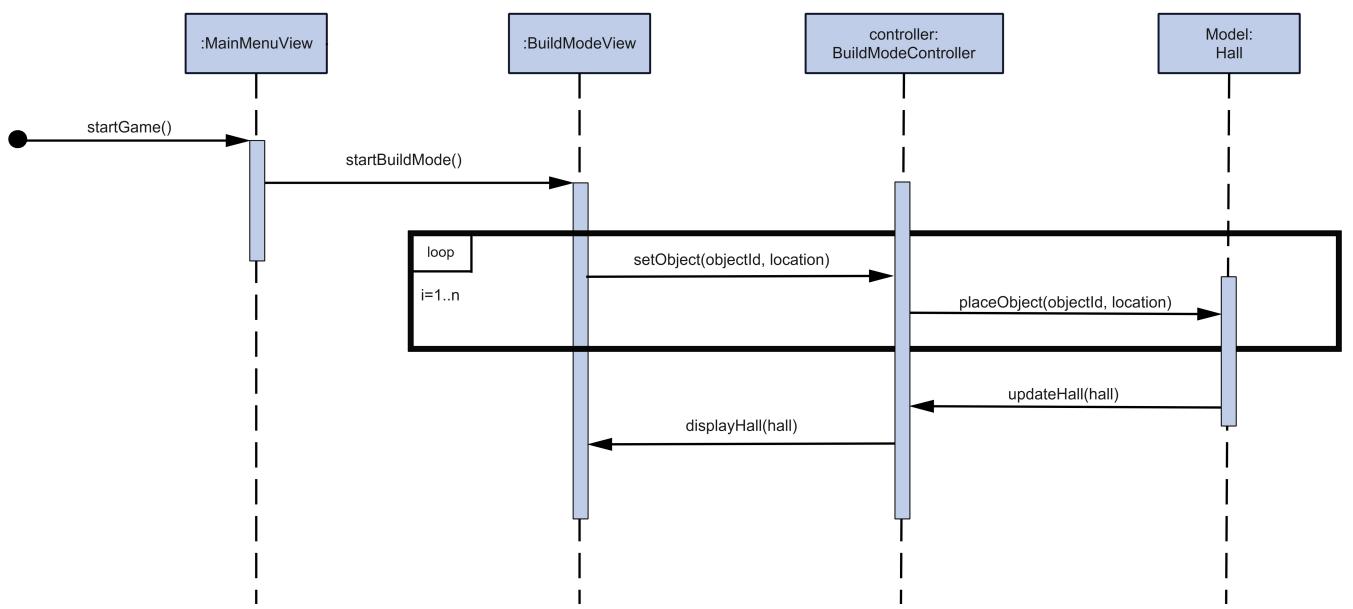


Figure 6: Build Mode Sequence Diagram

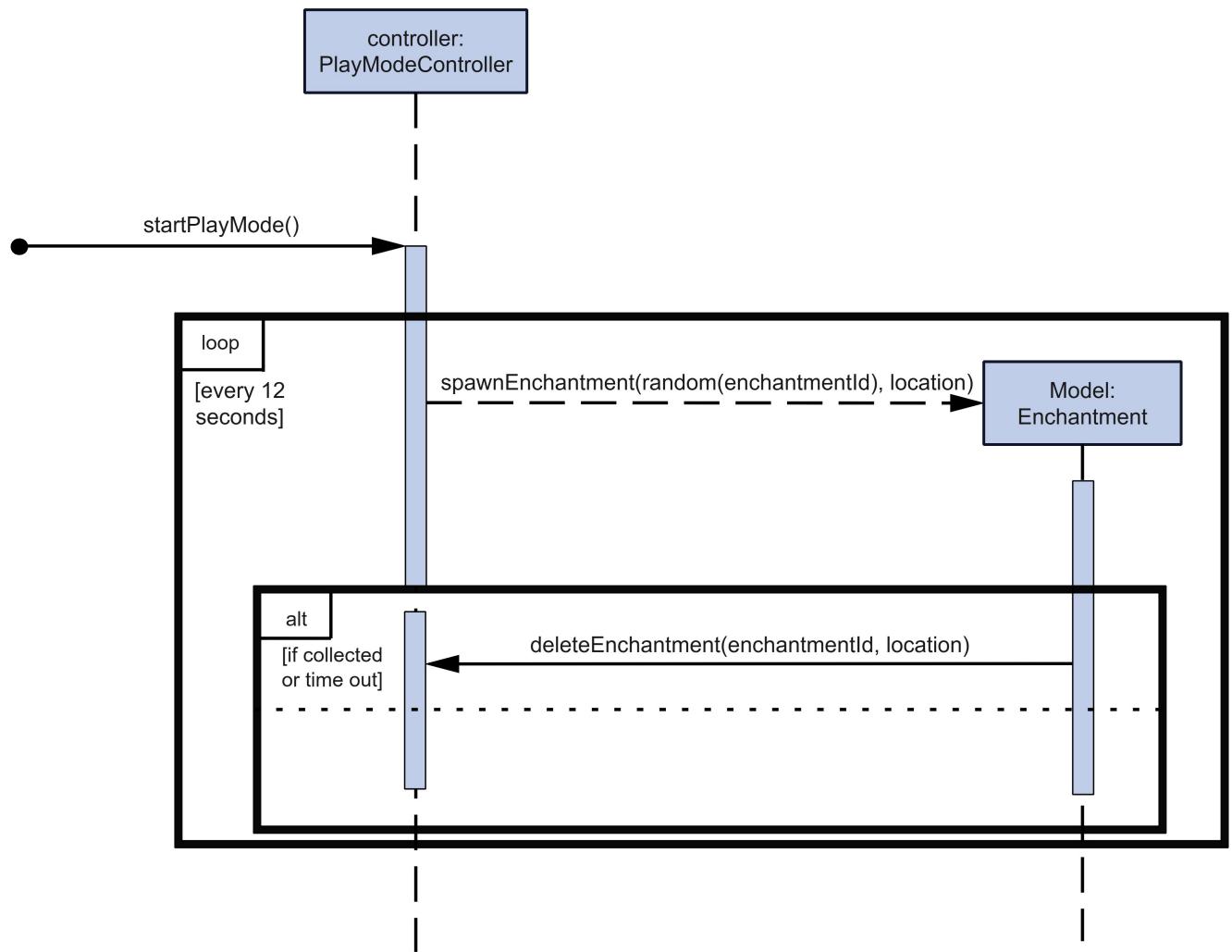


Figure 7: Spawn Enhanchment Sequence Diagram

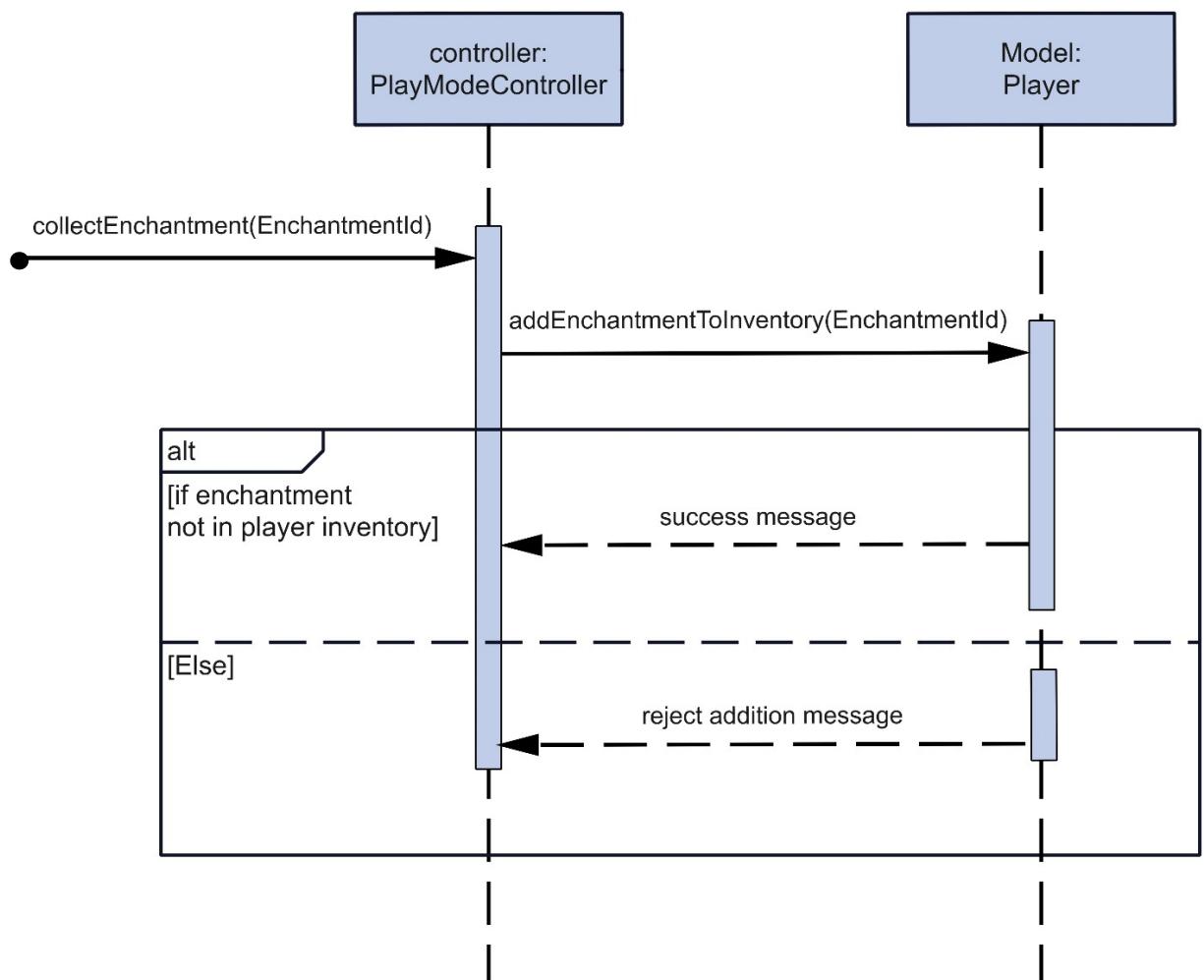


Figure 8: Add Enchantment Sequence Diagram

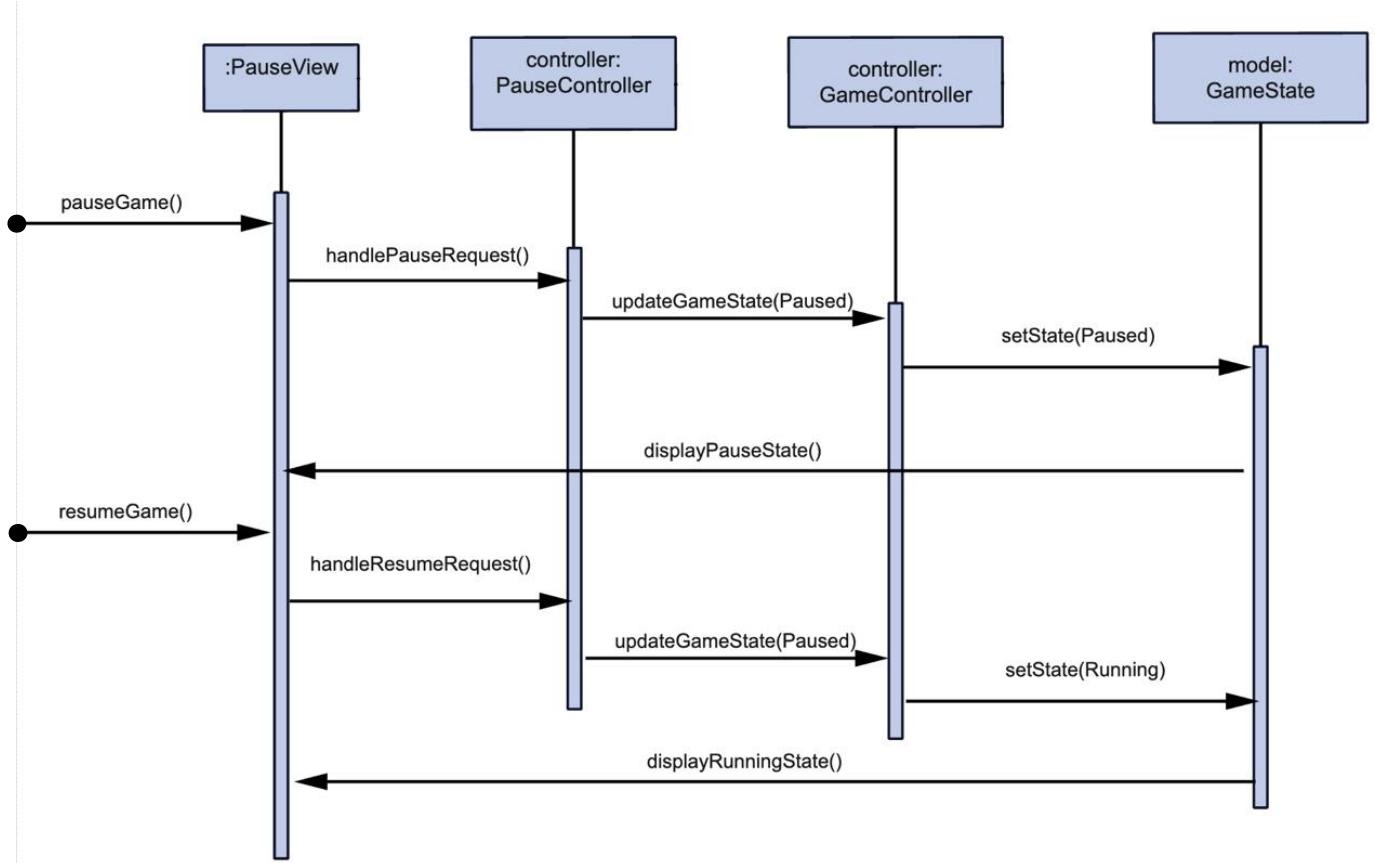


Figure 9: Pause Resume Sequence Diagram

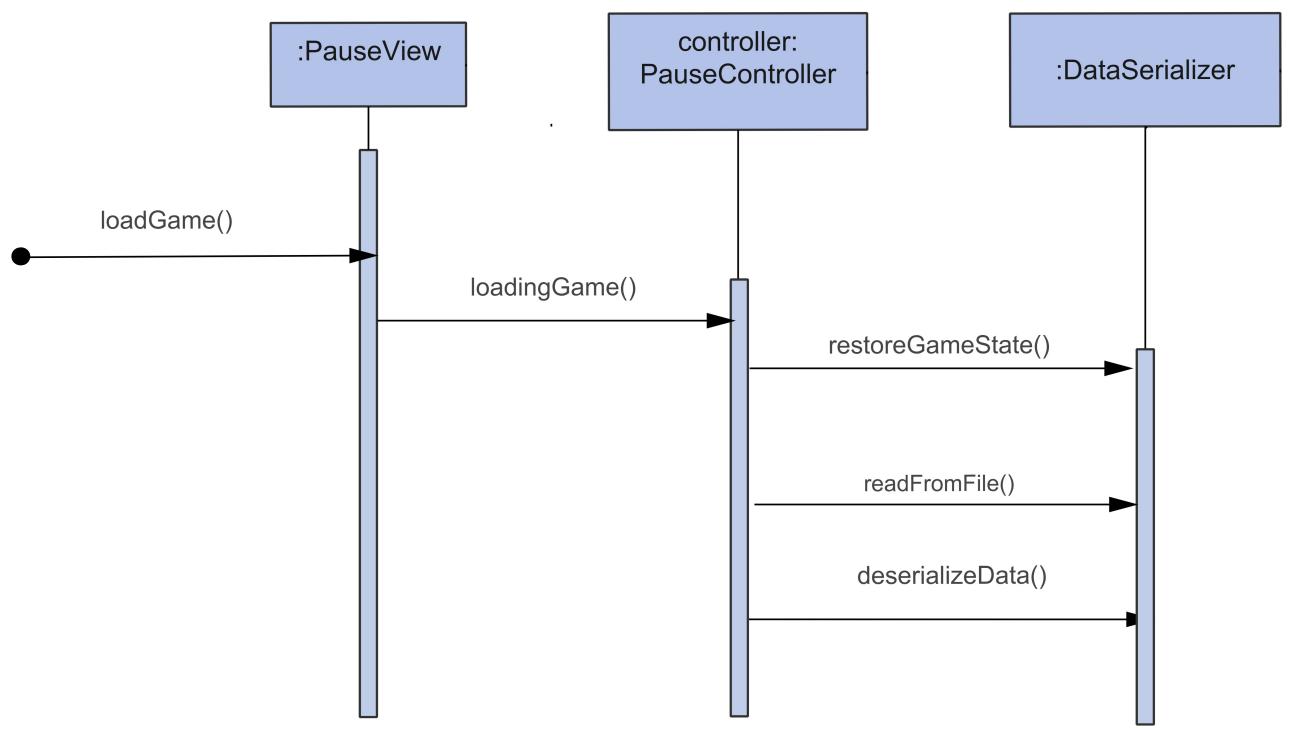


Figure 10: Load Game Sequence Diagram

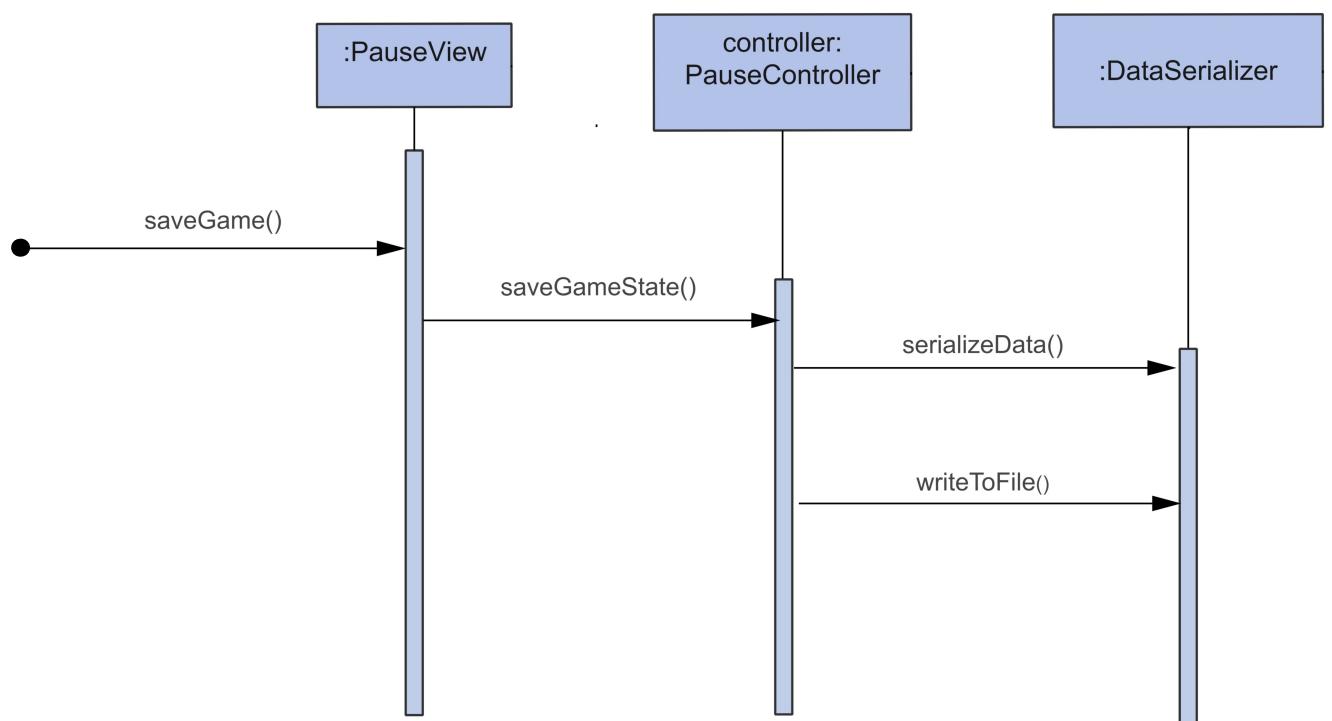


Figure 11: Save Game Sequence Diagram

3 UML Communication Diagrams

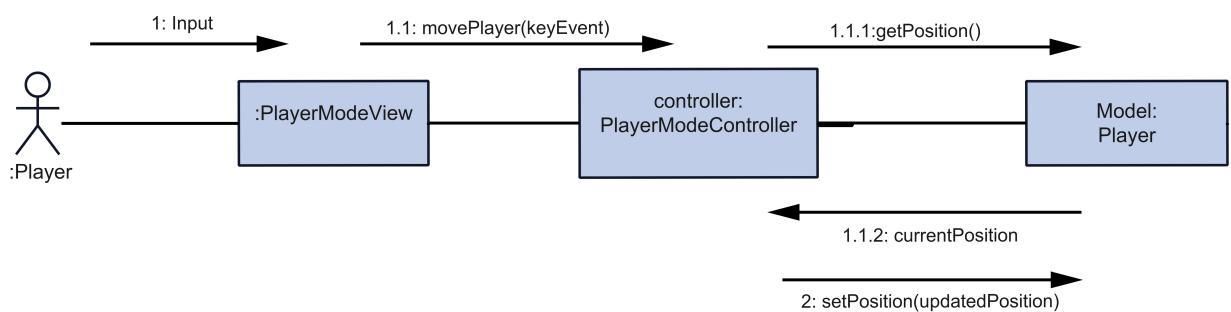


Figure 12: Move Character

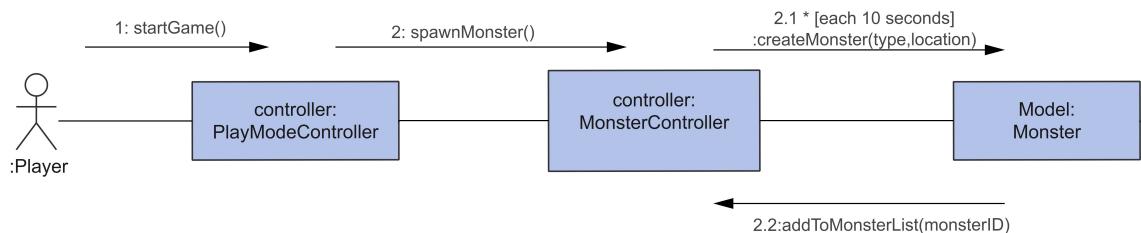


Figure 13: Spawn Monster

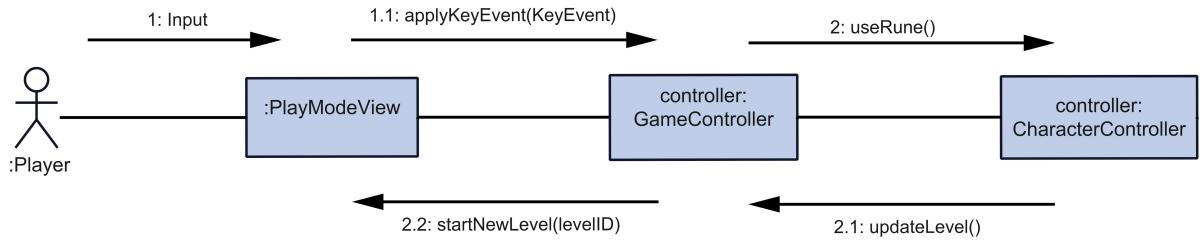


Figure 14: Use Rune

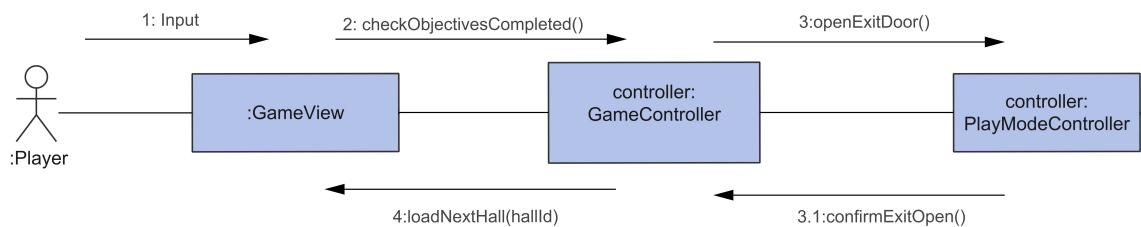


Figure 15: Progress to Next Hall

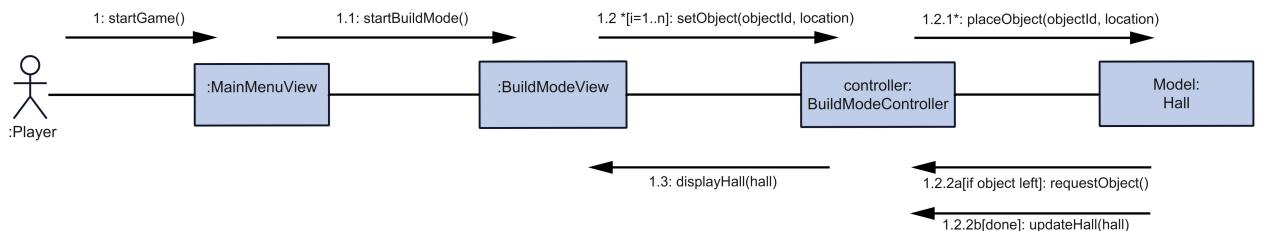


Figure 16: Build Mode

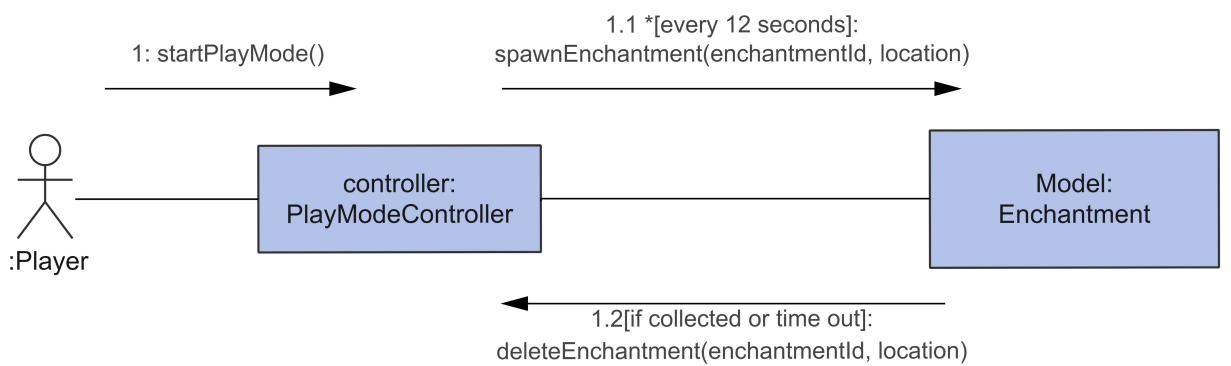


Figure 17: Spawn Enchantment

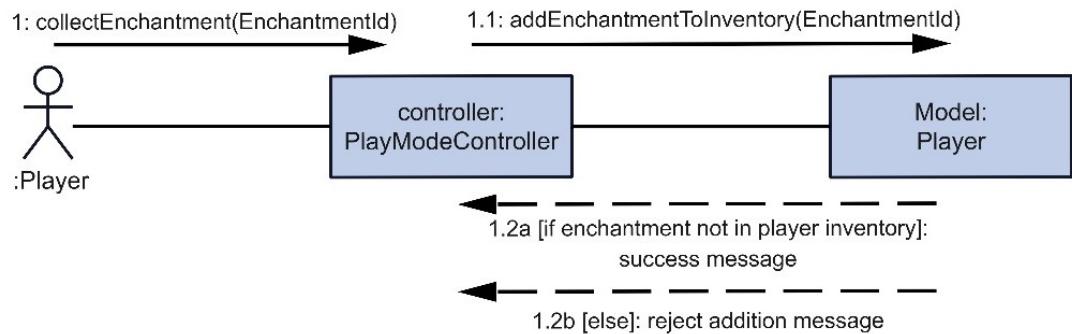


Figure 18: Add Enchantment

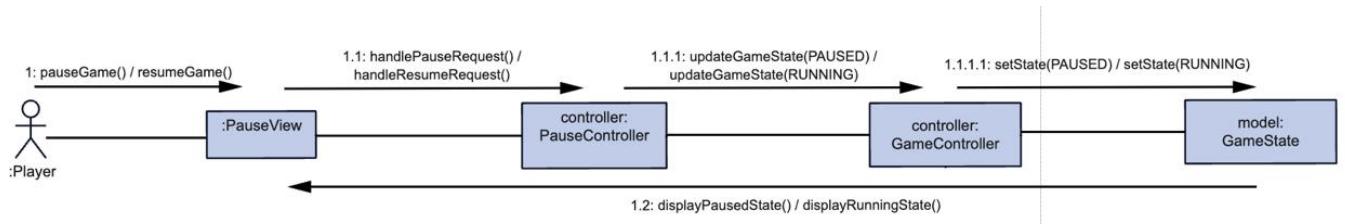


Figure 19: Puase Resume

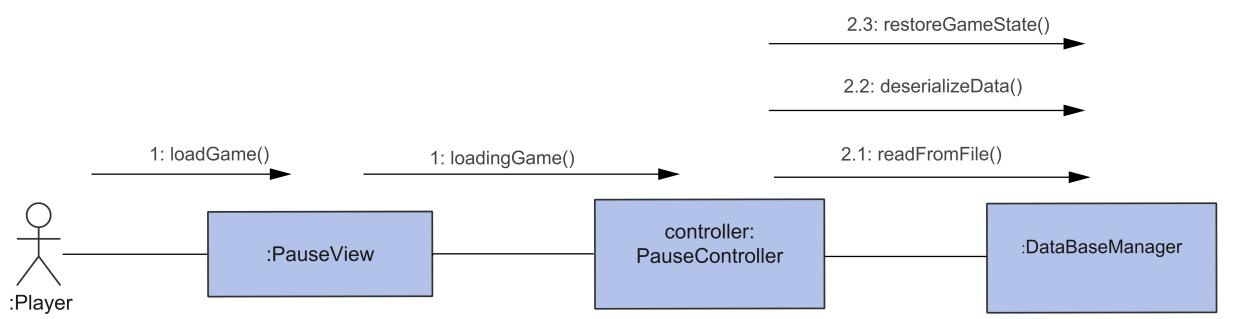


Figure 20: Load Game

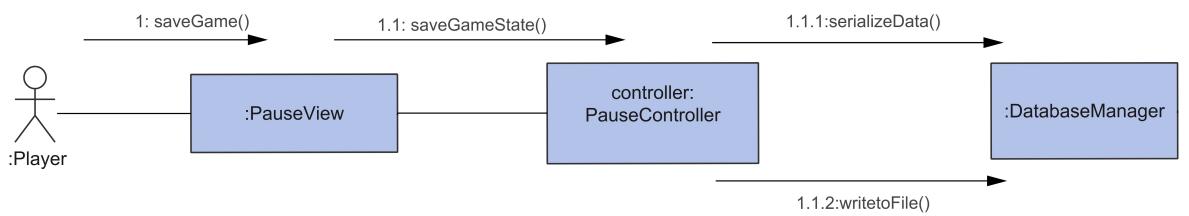


Figure 21: Save Game

4 UML Class Diagrams

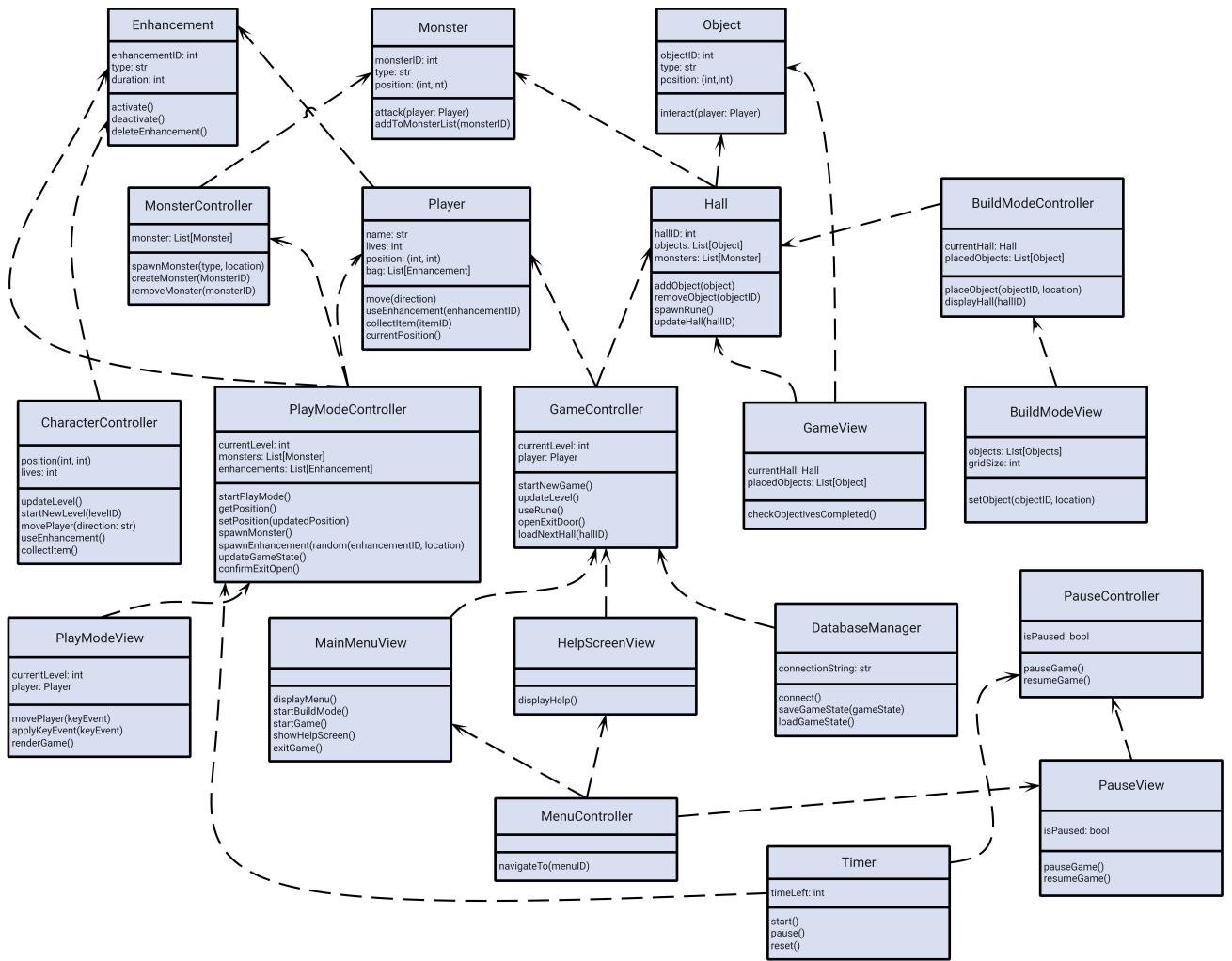


Figure 22: UML Class Diagram

5 Design Alternatives and Their Evaluation

In this section, we discuss three different design alternatives for the system along with their respective pros and cons.

5.1 Alternative 1: Observer Pattern

Implement the Observer pattern where views subscribe to model updates. When the model changes, all subscribed views are automatically notified.

Pros:

- Strict adherence to the model-view separation principle.
- Scales well for multiple views observing the same model.
- Decouples the model and views, making the system more modular and easier to maintain.

Cons:

- More complex to implement, requiring observer and observable interfaces.

5.2 Alternative 2: Using Swing Layout Managers

Use built-in layout managers (e.g., BoxLayout, FlowLayout) to organize components dynamically.

Pros:

- Automatically adjusts to resizing, providing a responsive layout.
- Easier to modify or expand the GUI without recalculating positions.

Cons:

- Might be difficult to adjust everything as desired.
- Can involve trial and error to achieve the desired layout.

5.3 Alternative 3: Using Online Real-Time Sync MongoDB

Perform all model data updates directly on MongoDB, syncing views as needed.

Pros:

- Ensures data persistence at all times.
- Simplifies synchronization in multi-user scenarios.

Cons:

- Limited number of queries per minute due to free MongoDB version.
- Slower operations due to free version of the database.