OOP Project Sprint 1
Outline and Contribution Report

PlayOps - A retro videogame rental system

Blevis Allushi, Kristjan Seraj

University Metropolitan Tirana

BSc in Artificial Intelligence

Assigning / Accepting Professor: Prof. Evis Plaku

November 4, 2025

Contents

1 Project Overview

The **PlayOps** is a management system for game rental stores, specifically targeting the dwindling vintage games scene. It allows users to:

- Add, remove, and search for customers and games.

- Display game inventories and customers.

- Manage game rentals and calculate rental costs.

PlayOps development implements OOP prinicples such as encapsulation, modular design, and data abstraction.

2 UML Class Diagram

The UML diagram explains the structure and relationships of the components within this project.
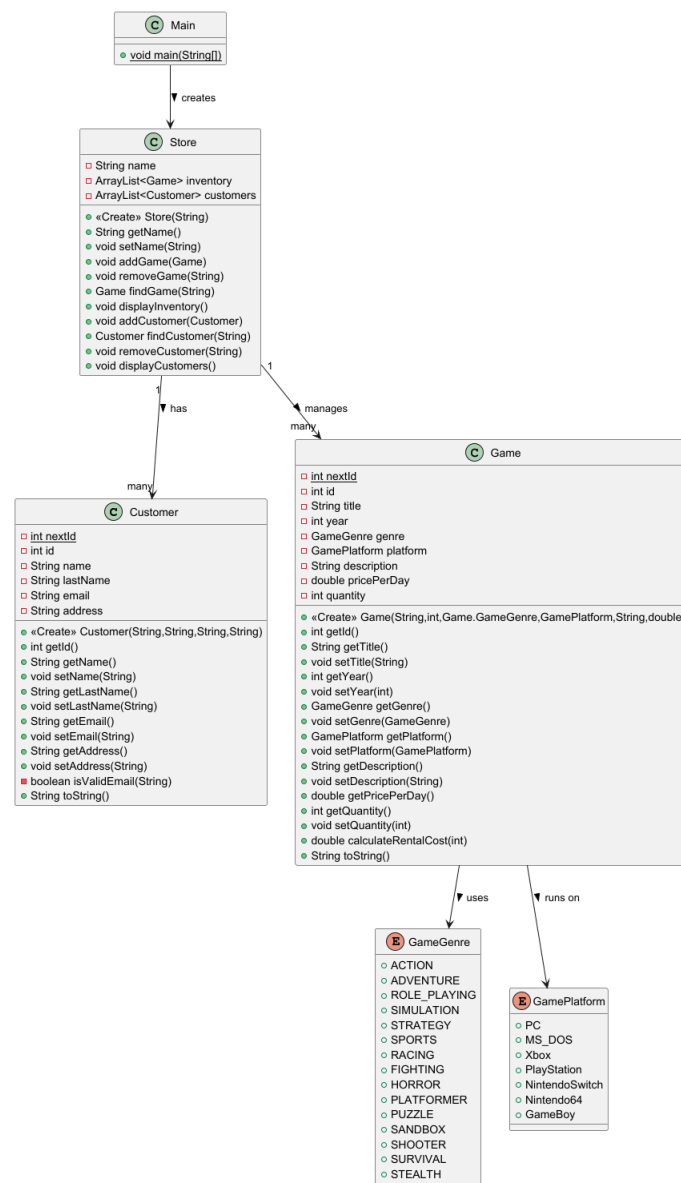
3 Project Structure and Design

The project follows a modular object-oriented structure:

- **Customer.java** — Manages customer data and an instance of validation in the email entries.

- **Game.java** — Represents the game properties and their attributes.

- **Store.java** — Manages lists of customers and games.

- **Main.java** — Entry point that handles user interaction.

Each class was designed with primary OOP philosophy of dividing work into classes with reusability and user convenience.

Team Roles and Contributions

Name	Responsibilities and Contributions
Blevis Allushi	Co-author and developer. Designed the overall class architecture and implemented (`Customer`, `Game`) classes, implemented key methods and restrictions via enumeration, finalized (`Main`) after all work was submitted. Contributed to UML modeling and documentation structure.
Kristjan Seraj	Co-author and developer. Implemented core functionality for connecting (`Customer`, `Game`) classes through the (`Store`) class, implemented ArrayLists, developed search and display methods. Contributed equally to debugging, testing, and UML refinement.

Collaboration Process

Blevis and Kristjan collaborated equally throughout all phases of the project. From the initial idea and planning to programming, testing and final documentation. Tasks were divided among the two, but major decisions and features were collectively discussed and decided upon.

4 Reflection

This project was assigned to the 2025 class of AI & Software Engineering at University Metropolitan Tirana, Tirana, Albania. First and foremost, this project was fun and entertaining. Although Sprint 1 was evidently not the most complex, it initiated enjoyable discourse not only within groups, but among them as well. It laid foundations for what it is like to work in pairs for the successful delivery of a project where each member's implementations are crucial and important in their respective ways. It suffices to say, no longer speaking for myself in third person, that I believe it engaged the class, hardened the knowledge gained up to this point in this course, and served as an entry to real-world expectations. In myself's and Kristjans group each of us gained experience in:

- Designing maintainable class hierarchies.

- Intertwining classes to achieve workflow and functionality.

- Using collections effectively in Java.

- Collaborating through Git-based workflows.

- Documenting work through designing UML files and using LaTeX for official documentations.