SWEN30006 Software Modelling and Design
Semester 1 2023

Assignment 2 feedback for **Matthew Pham (1273607)**

Criterion	Available marks	Team submission
Build		
Build Success		TRUE
Tests		
Test Behaviour #1		TRUE
Overall mark for Assignment 2	**20**	**19**

Feedback on Team Submission

Design Report Evaluation

Extendable Design and Implementation:
The report discusses the strengths of their chosen designs, including better code cohesion, protecting variables from inadvertent changes, and establishing information experts (Section 1.3).
The limitations of the original designs are also clearly addressed, such as the low cohesion and unprotected variations in the original Controller class, and the lack of understanding of the game map in the original Autoplayer implementation (Section 1.1 and 2.1).

Rationale for Design Patterns:
The document provides robust justifications for their design decisions, with clear indications of how the selected patterns help address issues in the original codebase. This can be seen in their explanation of using the Singleton and Static Class patterns to ensure seamless conversion between different grid models and handling file-to-model conversions respectively (Section 1.2).
They have also justified the abstraction of GameMap to handle pathfinding for the Autoplayer, which allows for better extensibility (Section 2.2).

GRASP and GoF patterns Usage:
The design document clearly shows evidence of understanding and usage of GRASP principles in their design process, particularly in terms of high cohesion, protected variations, and the use of information experts (Section 1.1 and 1.3).
Similarly, the usage of the Singleton pattern (TileManager) and Static Class (MapFileLoader) as part of the GoF patterns is evident (Section 1.2).
The Autoplayer design leverages abstraction of GameMap into a grid of Spaces, a design inspired by the principle of abstraction from GoF patterns (Section 2.2).

Alternatives and Comparison:
The document includes a consideration of alternative solutions, discussing potential limitations and assumptions made during their design process. For example, they have addressed the limitations of the TileManager singleton (Section 1.2) and the consideration of a Space factory (Section 3).
The report explains why these alternatives were not chosen, such as the difficulty of creating Portals with a Space factory and the limitation on tile representation in the TileManager singleton (Section 3).
There are some additional scenario to be considered:

1) The usage of strategy pattern for the Autoplayer to allow for different pathfinding algorithms.
2) Using composite pattern for rules to allow for more complex rules (The team did mention the difficulty of using factory, but not rule checker).

Diagrams:
Good diagrams with precise notations.

Implementation:
The implementation reflected what they did and worked well