SCAPES

Subsystem Decomposition and Design Patterns

By:

Redwan Wadud

100873111

Submitted to:

Dr. Christine Laurendeau

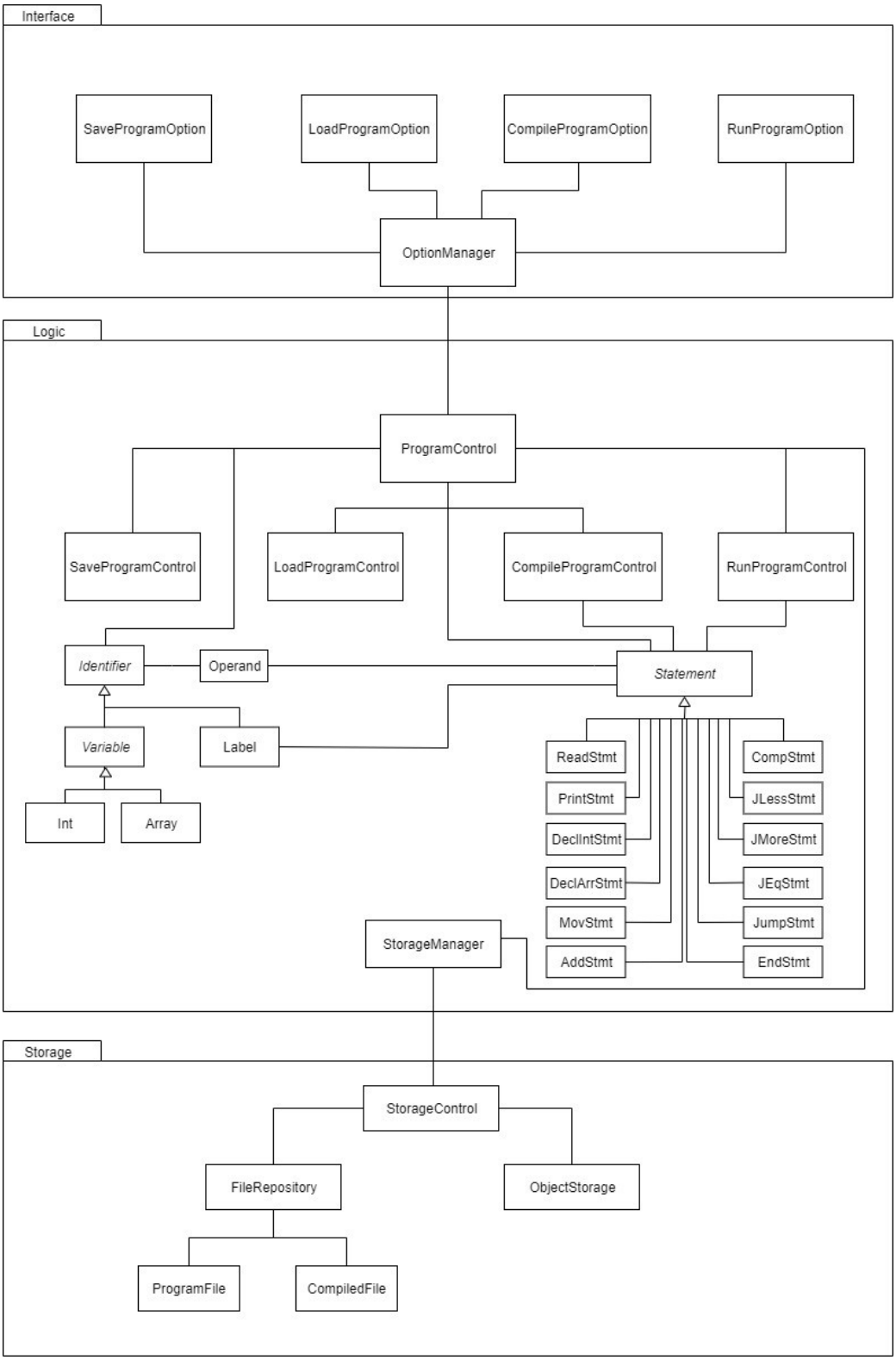
COMP 3004 - Object-Oriented Software Engineering

School of Computer Science

Carleton University

October 17, 2019

1. Subsystem Decomposition



By incorporating the Façade design pattern into the system, my design minimizes coupling and maximizes coupling.

2. Design Patterns

2a. Facade

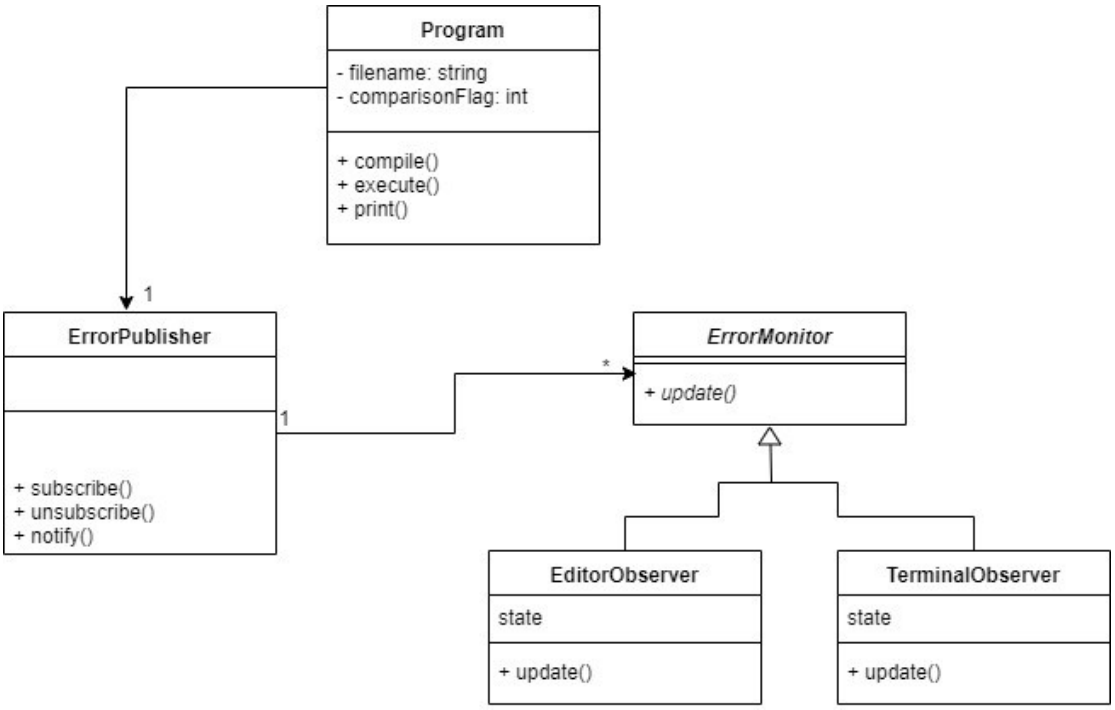
The façade design pattern hides away the complexity of a system and provides a simpler interface to use. The façade design pattern reduces the complexity and minimizes communication between between subsystems. It makes life easier for the programmer by providing a higher level interface without needing to go into dealing with all the low level stuff.

My system uses the façade design pattern to minimize the coupling between the subsystems.

Refer to the 1. Subsystem decomposition diagram above on how it will be implemented into the system.

2b. Observer

The observer design pattern uses a publisher/subscriber relationship. A publisher has many subscribers. A change in state in the publisher will notify all of its subscribers to take action as necessary. This design pattern can be implemented into my system to observe for errors during the compilation/execution process and notify the Programmer via the Editor boundary object and Terminal boundary object. The syntax errors would be highlighted for the Programmer to see and runtime errors would be printed to the terminal window.



UML Class diagram of how the observer pattern will be implemented.