SWEN 262 State Machine Refactoring

**INSTRUCTIONS:** Analyze the Java class provided to you and read the instructions below. Submit a ZIP archive containing a PDF version of this document that includes the required artifacts in the space after each question and your refactored source code.

1. A **finite state machine diagram** that captures the possible states of the class. If you need a refresher, there are [lots](http://www.uml-diagrams.org/state-machine-diagrams.html) [of](https://www.visual-paradigm.com/solution/statedgm/statemachine/) [tutorials](https://www.lucidchart.com/pages/how-to-draw-a-state-machine-diagram-in-uml) online.
2. Fill out the **GoF Pattern Card** below to describe the classes that you would use to refactor the Java class into the State design pattern. You should write at least 2-3 sentences per class to describe its role in the pattern in the context of the application.

|  |  |  |
| --- | --- | --- |
| **Gof Pattern Name: State** | | |
| **Class** | **GoF Participant Name** | **Participant’s activity within the pattern in the context of the application.** |
|  |  |  |
|  |  |  |
|  |  |  |
| **Deviations from the standard pattern:** | | |

1. Provide a UML diagram of the classes and interfaces described in your GoF pattern card. While you may submit a photo or scan of a hand drawn diagram, it must be LEGIBLE.
2. Refactor the source code according to your design above. The code must pass any tests that have been provided. Include your refactored code as part of your submission.