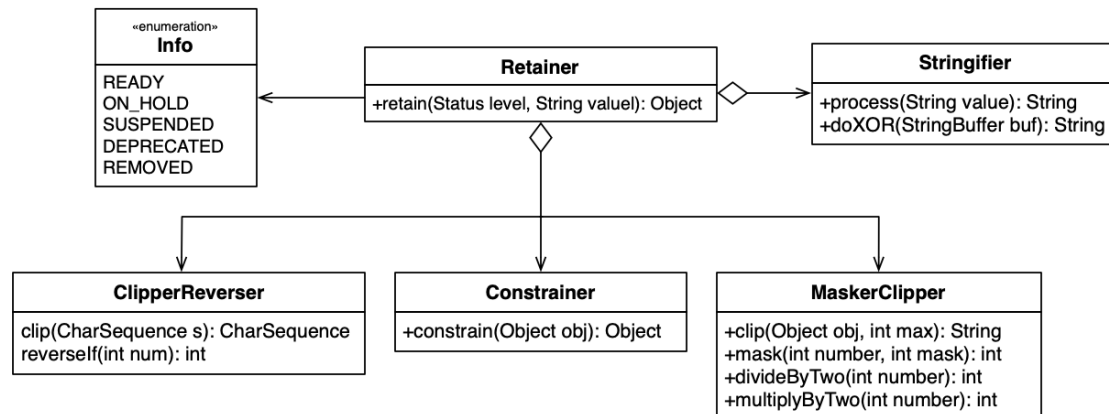




ASSESSMENT I (25%) – Group C

The following UML diagram shows a suite of classes in a poorly designed application:



You are required to provide a **redesign and refactoring** of the classes above that provides maximum reusability, extensibility and eliminates as many dependencies as possible in the design. **You are free to change any of the classes in any way you wish**, including renaming, changing method signatures and deriving new types. You must implement your design as a set of Java classes and **document your rationale in** a README file and in the source code where appropriate. The only constraint in the design is that there should only be one instance of the class **Retainer** and that it should be straightforward to get a handle on it from anywhere in the application if necessary.

Note that there is no single “correct” answer to this assessment – there are many possible solutions, all with their advantages and drawbacks. Any design patterns that may apply should already exist in the problem. State any **assumptions or known issues** relating to your design in comments at the top of your classes or in the README file.

INSTRUCTIONS FOR SUBMITTING YOUR WORK

Please **read the following carefully** and pay particular attention to the files that you are required to submit and those that you should not include with your work:

- A **set of stubs** for the classes in the UML diagram is available on Moodle.
- Use the module name **atu.software** and the package name **ie.atu.sw**.
- Instrument all methods with some minimal functionality or with `System.out.println()` statements. Note that the **focus of this assignment is on design, not on robustness**. Do not waste your time implementing features and functionality that are not needed.
- When you have finished the assessment, upload a **Zip** archive of the **src** directory and **README** to Moodle. The name of the Zip archive should be **<id>.zip** where **<id>** is your ATU student number.