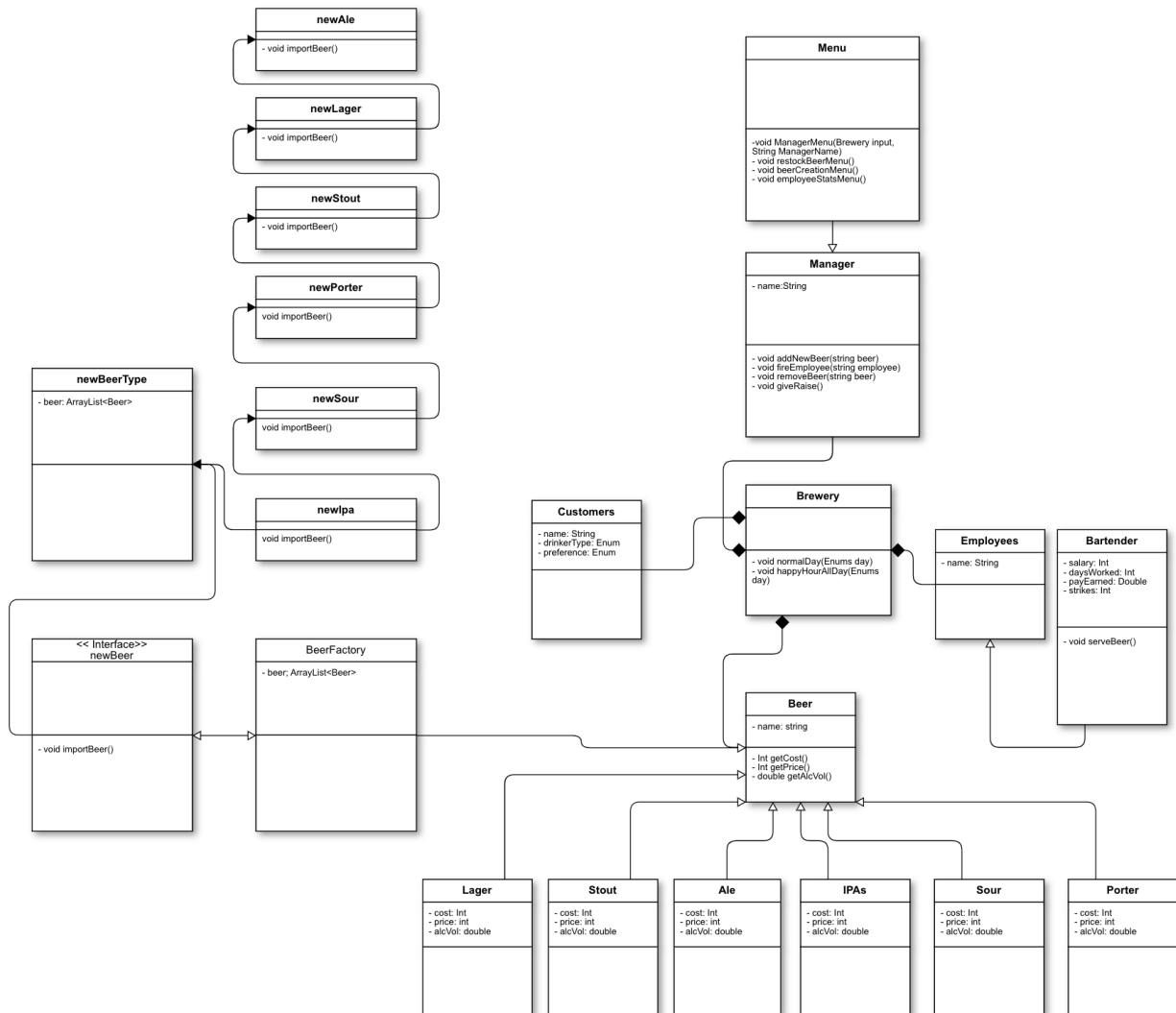


Team Members: Eric Morales and Rafael Basurto

Project Name: Nebula Brewing Co.

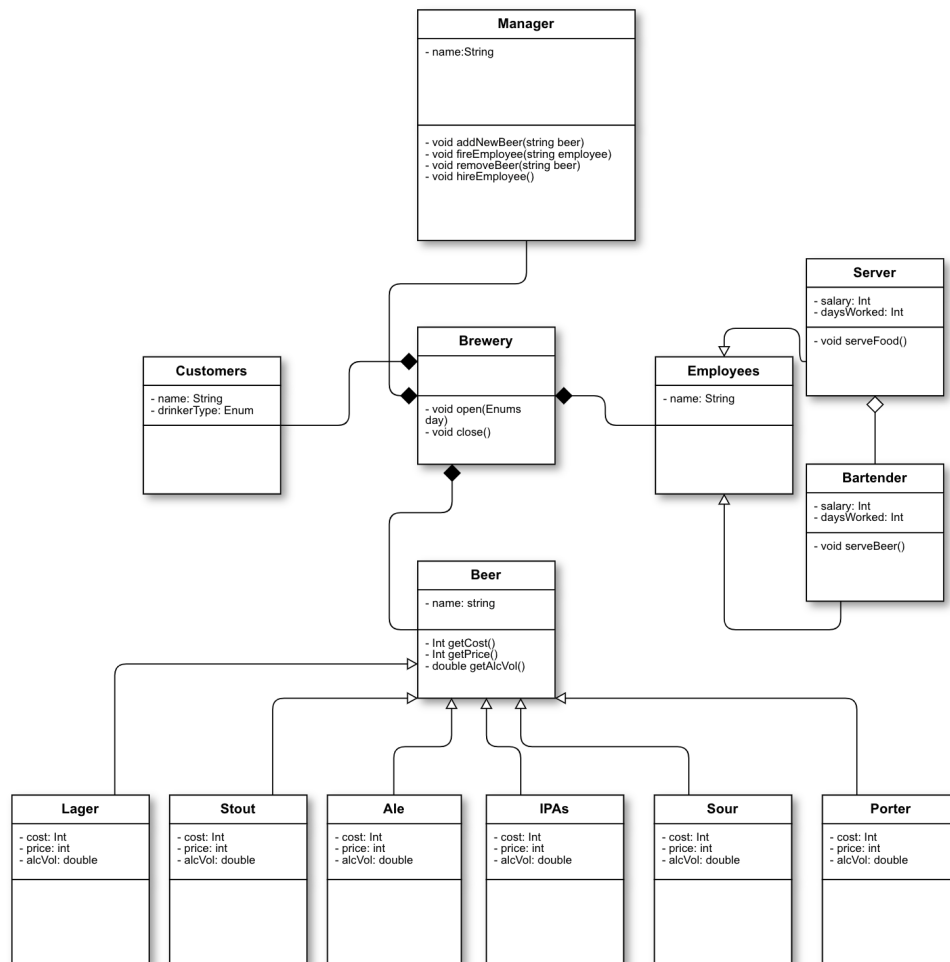
Final Class Diagram and Comparison Statement

Final UML class diagram:



In our design, our main focus was on two patterns, the factory pattern and command pattern. The factory pattern was a key part in initializing our beer instances at the beginning of our simulation, making sure that our beer count was accurate. The command pattern played a key role in our management of the user interface and its various executions. Other patterns that we have incorporated are the facade and iterator patterns. The facade pattern being used to simplify the system when executing and the iterator pattern being used to traverse through our various arraylists.

Diagram from project 5:



As we can see, our core design that was made in project 5 remains about the same. Our changes were incorporated into our original design. We incorporated a better user interface using the command pattern, a better way to initialize beer instances using the factory pattern, and we added a few more methods to a few classes.

To summarize, most of our key changes still kept our original design intact. The methods that were planned out in projects 5 and 6 remained about the same, while more methods were added to either accommodate those original methods or add more functionality to the program. A key change that we made is that in terms of design patterns, we did not end up using all of the ones that were previously planned, which are the composite and decorator patterns. Instead, we ended up using the facade and iterator patterns. Our original plans to implement the factory and command patterns remained the same. Finally, one key change that we made was that we

decided to not incorporate both bartenders and servers, so we only used bartenders for our employees class.

Third-Party code vs Original code Statement

Our project was a derivation of previous projects that we had throughout the semester, so there are a few similarities in classes and helper functions that were used here from previous projects. With that being said, most of our code for class initialization and implementation was originally written by us. Third-party code was mostly used to get helper functions that would allow us to execute various tasks within our program more efficiently. Here is a list to URLs of our third-party elements:

- <https://stackoverflow.com/questions/34159413/java-get-next-enum-value-or-start-from-first>
- <https://www.geeksforgeeks.org/how-do-i-generate-random-integers-within-a-specific-range-in-java/>

Statement on the OOAD process for your overall Semester Project

1. A key design process issue that we ran into was finding the appropriate design patterns to use for our semester project. As stated earlier, we believed that we had four solid patterns to use in our project, but later realized that they didn't work cohesively with the way we wanted them to.
2. A key design process element that helped us move constantly through the entire project was that even though we agreed to make a lot of changes throughout the development of the project, we stayed true to the original design of it.
3. Another key design process issue we ran into was figuring out exactly what frameworks or even coding languages we needed to use to move forward with the project