Implementation of Soda Machine is follows below classes, interfaces and design patterns.

Writing API’s for Soda machine actions.

**SodaController**

This is the starting point for UI request/response. We need to define multiple API calls to get/post the data from UI application.

**SodaService**

This is the interface. We can follow Factory design pattern here. This interface is used for delegating the controller request/response to respective implementation classes. All main action methods are defined here.

1. Insert a quarter
2. Remove a quarter
3. Push the desired soda button
4. Dispense the soda

**SodaServiceImpl**

All the action methods are implemented here. We have to handle all the states here.

1. Sold out
2. No quarter
3. Has quarter
4. Sold

We have to send back the status to the UI for different states codes. So that UI can handle the response easily.

And also we have to store the purchase and other data to generate the reports.

**Repository Interfaces**

This one is used for Data operations (store, retrieve, delete etc).

**Models**

We need to define multiple models to handle the persistence operations.

**UI**

UI can build on any technology. Since we are building actions through API calls. API’s are flexible and can use any UI framework/other services etc.

**Reports**

Reports can build based on stored information easily.

Based on above information, it is easy to adopt new technologies, UI, Different Services, Code maintenance, deployment etc. Factory and Proxy design patterns can be used to implement the solution.

