1) Statement of Work

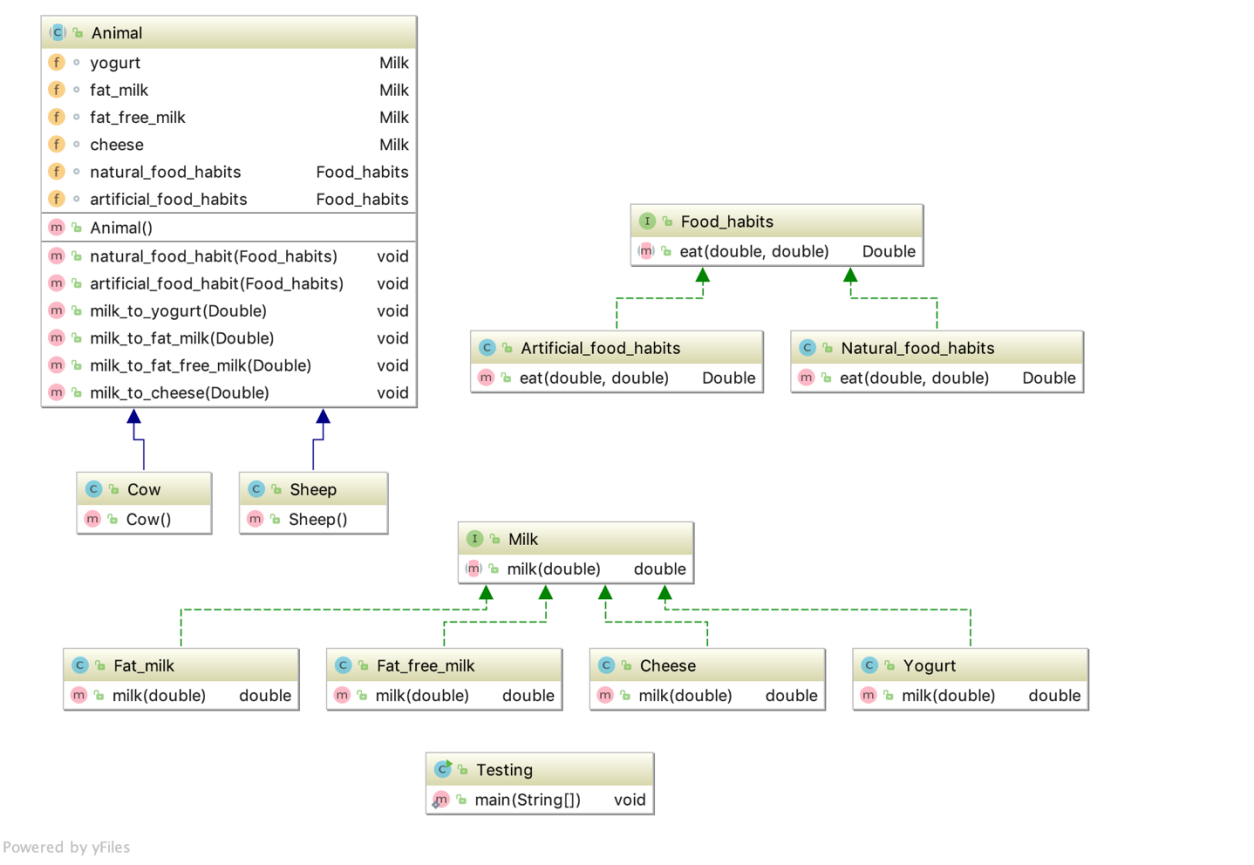
We have a milk company. We are working with farmers. All farmers have animals as a cow and a sheep and all of them are has one or more than one animals. All the animal's milk liters calculate from farmers. Each animal are eating natural food and artificial food in different time in a day. They record animal's food that are eating daily and animal's total milk daily, also calculate drinking water by each animal's. As a total information recorded by farmers and they give them when we pick up the milk. We pick up the milk weekly. The gotten milk is processed and splitted by us. We are doing cheese, fat milk, fat-free milk and yogurt with this milk. The milk is splitting before processing and after splitting and processing milk's liter is decreasing because of processing. Sheep's milk is using for doing yogurt and cheese. Cow's milk is using for doing cheese, yogurt, fat-free and fat milk. When doing each process (milk, cheese, yogurt) first milk is splitting with constant counts and milk is multiply by the constant count. These counts are defined by us.

With my application firstly calculate the foods amount with drinking water. And shows the milk liter base on this food. After splitting is starting and continue by processing. For each process(yogurt, cheese, fat milk and fat-free milk) is showing with diverging liter before processing and composed amount of new composed product after processing.

2) Explanation on Utilized Design Patterns

I choose Strategy Pattern for this application. I had a animal class, I had to use this class as abstract class and animal class had two class for cow and sheep, and I need to food's habit and milk class and I used them as an Interface class they had subclasses which using same method. I'm thinking strategy pattern was being usefull for me when was doing this application.

3) UMLClassDiagram



4) Research

The application has an Animal class as abstract. It has several methots both food and milk. These are natural\_food\_habits(), artificial\_food\_habit() for food habits, milk\_to\_yogurt(), milk\_to\_cheese(), milk\_to\_fat\_milk(), milk\_to\_fat\_free\_milk() for milk they are using when milk is processing. Animal class has Cow and Sheep class. these class extends Animal class. Both Cow and Sheep class can do yogurt and cheese but just Cow class can do milk. Than there is Milk class as an Interface, it has milk method with parameter double milk, this parameter for milk liter. And there are Fat\_milk, Fat\_free\_milk, Cheese, Yogurt class these are implements the Milk Interface, both of them use the milk method which we reach thanks to Milk Interface. All of milk methods is doing different processes. They are using when milk is splitting and procesessing. They calculate splitting milk count and being processed milk amount. And there are also Food\_habit class as an Interface. It has eat method with double water and double food parameters. There are Artficial\_food\_habits and natural\_food\_habits class both of them implement Fodd\_habits Interface. These class using eat method for different process. With this method calculate food amount with drinking water. I used Interface to represent both foot and milk behavior. At the end tehre is Testing class, this class is main class. This task is calculate food and water ratio and according to this ratio calculate milk liter, when liter is finding, milk is splitting and processing and all of these operations writing.