Use Case Diagram

Your feature lists are all about understanding what your software is supposed to do. Your use case diagrams let you start thinking about how your software will be used, without getting into unnecessary details. The use case diagram contains all use cases within the system. See picture 1.

Detailed Use Case

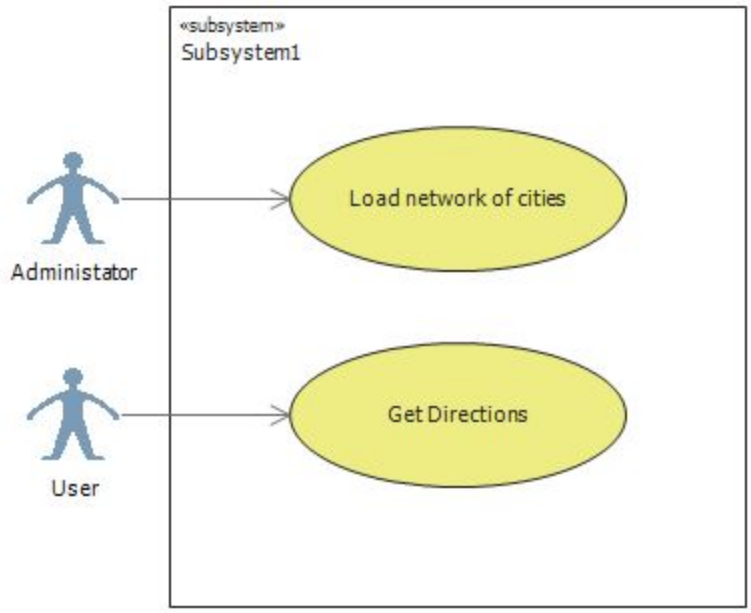
Each use case describes one and only one thing that your software needs to do. There are three basic parts to a good use case.

- Clear Goal
    - Every use case must have a clear goal to the system.If your use case doesn't help the customer achieve their goal, then the use case is not of much use.
- Start and Stop
    - Every use case must have a defined starting and stopping point. Something must begin the process, and then there must be a condition that indicates that the process is complete.
- External Actor
    - Every use case is started off by an external actor, outside of the system. Sometimes that actor is a person, but it could be anything outside of the system. The actor is actually a role, so that the same person can be different actors.

Example,

Let's say you want to make a program that can allow the administrator to store a network of cities, as well as all their edges that connect cities together. The program also allow any user to print out the route, indicating the starting city and the ending city.

Use Case Diagram (picture 1)



Detailed Use Case of Load network of cities

1. The administrator supplies a file of cities and paths.
2. The system reads in the name of city.
3. The system validates that the city does not exist.

4. The system adds the new city to the system.
5. The system repeats steps 2-4 until all cities are added.
6. The system reads in the name of a path to add.
7. The system reads in two cities that are connected.
8. The system validates that the cities exit.
9. The system creates a new connection between the two cities going in both directions on the current path.
10. The system repeats steps 7-9 until the path is complete
11. The system repeats steps 6-10 until all paths are entered.

Detailed Use Case of Get Directions

1. The user gives the system a starting city and a city to travel to.
2. The system validates that the starting and ending cities both exist on the network.
3. The system calculates a route from the starting station to the ending station.
4. The system prints out the route it calculates.