Assignment 1 - Requirements Elicitation and Documentation

Exercise 1 - Design and Documentation

You are given the following classes:

```java
public class Node{
  public Node(){...}
  ...
}

public class Edge{
  Node from;
  Node to;
  int distance;

  public Edge(Node from, Node to, int distance){...}
}

public class STGraph{
  List<Edge> edges;
  Node source;
  Node target;

  public STGraph(Node source, Node target){
    this.source = source;
    this.target = target;
    this.edges = new ArrayList<Edge>();
  }

  public void setST(Node source, Node target){
    this.source = source;
    this.target = target;
  }

  public void addEdge(Edge e){
    edges.add(e);
  }

  public List<Node> shortestPath(){...}
}
```

1. For which use cases would the above design decisions (passing the source and the target as arguments to the STGraph's constructor and using a setter method for modifying them afterwards) make sense?

2. Does the shortest path always exist? What information should be provided in the client documentation of the method shortestPath with respect to the return value?

3. Since computing the shortest path in a graph is an expensive operation, we would like to avoid re-computing it unless it is necessary.

   (a) Come up with a design to accomplish this.

   (b) How does this design influence the client-visible documentation for the class STGraph?

   (c) Are there alternative designs that achieve comparable results?

Exercise 2 - Design

You have seen different list implementations in the lecture, one of which uses reference counting to reduce the need for expensive cloning operations:

```java
class List<E>{
  E[] elems;
  int len;
  boolean shared;

  List(int l){
    elems = (E[]) new Object[l];
    len = l;
    shared = false;
  }

  private List(E[] e, int l){
    elems = e;
    len = l;
    shared = true;
  }

  void set(int index, E e){
    if(shared){
      elems = elems.clone();
      shared = false;
    }
    elems[index] = e;
  }

  List<E> take(){
    shared = true;
    return new List<E>(elems, len - 1);
  }
}
```

1. Come up with a scenario in which this implementation performs unnecessary cloning operations, when the set method is called.

2. How could you modify the implementation to eliminate this inefficiency?

3. Is your proposed solution efficient in the case in which unused objects are removed by the garbage collector? If yes, explain why. If not, explain how could you modify it to also efficiently handle this scenario.

Exercise 3 - Requirements Elicitation

You have been given the role of specifying the requirements document for an online flower shop.

> The client has been running this shop for many years and has become friends with both his customers and his suppliers. He currently has one messenger for the delivery of flowers and only three suppliers of bulk flowers that he receives from the Netherlands.

> His current customers appreciate his touch for determining the right flowers for the right moments in their lives. He therefore wants his online shop to have a personal feel to it, by showing logged-in customers a selection of seasonal flowers, their most recent purchases, as well as some offers based on personal data like gender and age on the homepage. For visitors that are not logged-in, the page should always show seasonal flowers.

> Customers need to create an account and be logged in to make purchases. Frequent customers (who have ordered flowers for more than 200 CHF in the previous year) are given an in-house account, which they may pay at the end of the month. Other customers must pay by credit card when making the purchase.

> He has found out that most of the mistakes in the process of ordering flowers happen with new or occasional customers. Therefore, customers have to provide their address and credit card details when creating an account. Both should be automatically checked for plausibility and rejected if there are obvious problems. When ordering flowers, customers can select either their home address or previously used addresses, or enter a new address. If there is a problem with payment, a user account is blocked and cannot make more purchases until it has been manually unblocked by the owner.

> One of the issues in working with his suppliers is to determine which flowers to get, and the right amount for each type of flowers. Therefore, he is interested in determining which flowers are popular during the major events of the year. However, he does not want to automate the process of ordering from his suppliers for the time being.

> His shop already has a point of sales system for in store sales, which should be integrated with the online shop. The administration of the online store will be handled by his daughter who is currently taking lessons in web design.

It is your responsibility to elicit the requirements and to write a document capturing them to begin the discussions with the customer. Start by determining the actors of the system. Write down the scenarios of the system (incl. exceptional one) and determine any open issues. Once you have determined the open issues, discuss these with the flower shop owner (your assistant). Think about how you could generalize the scenarios into use cases. Afterwards, identify the non-functional requirements of the system.