MIE – the 26th of May 2020

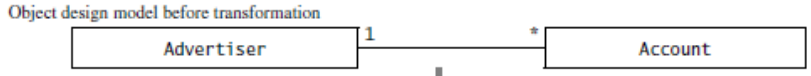
Working time 2h!

I. A person may be employed by companies. An employed person has a job at each employer. Any job is characterized by a title {designer, programmer, tester, other}, a start date and a salary. A person has a name, a birth date, a gender {female, male}, an information confirming that he/she is married. Each company is characterized by its name and maximum numberOfEmployee. Each married person knows the identity of the partner (wife or husband).

a. please represent by means of a UML class diagram a model complying with the above-mentioned description; 2.5pt

b. using OCL, please specify in the appropriate context an invariant checking that any employee of a company having a title different from designer cannot have a salary bigger than any other employee from the same company having the title designer; 1.5pt

II. Analyze the UML model above and explain using logical argues if the associated Java code was automatically generated or not. 1pt



Source code after transformation

```
public class Advertiser {
    private Set accounts;
    public Advertiser() {
        accounts = new HashSet();
    }
    public void addAccount(Account a) {
        accounts.add(a);
        a.setOwner(this);
    }
    public void removeAccount
            (Account a) {
        accounts.remove(a);
        a.setOwner(null);
    }
}
```

```
public class Account {
    private Advertiser owner;
    public void setOwner
            (Advertiser newOwner) {
        if (owner != newOwner) {
            Advertiser old = owner;
            owner = newOwner;
            if (newOwner != null)
              newOwner.addAccount(this);
            if (old != null)
              old.removeAccount(this);
        }
    }
}
```

III. The Figure 4 draw above presents by means of a diagram the behavior of a turnstile width diagnostic mode.

a. Please mention the type of the diagram, and all the components represented mentioning at the beginning the component type and after the list of all instances from the diagram 1.5 pt

- concrete states: Violation, Locked, Unlocked, TestCoin, TestPass
- composed states: NormalMode, DiagnosticMode
- pseudo states: 2 input states (one in each composed state) and a history state marked with H

b. In an informal manner (using your own words), please describe how the component behave 1.5 pt

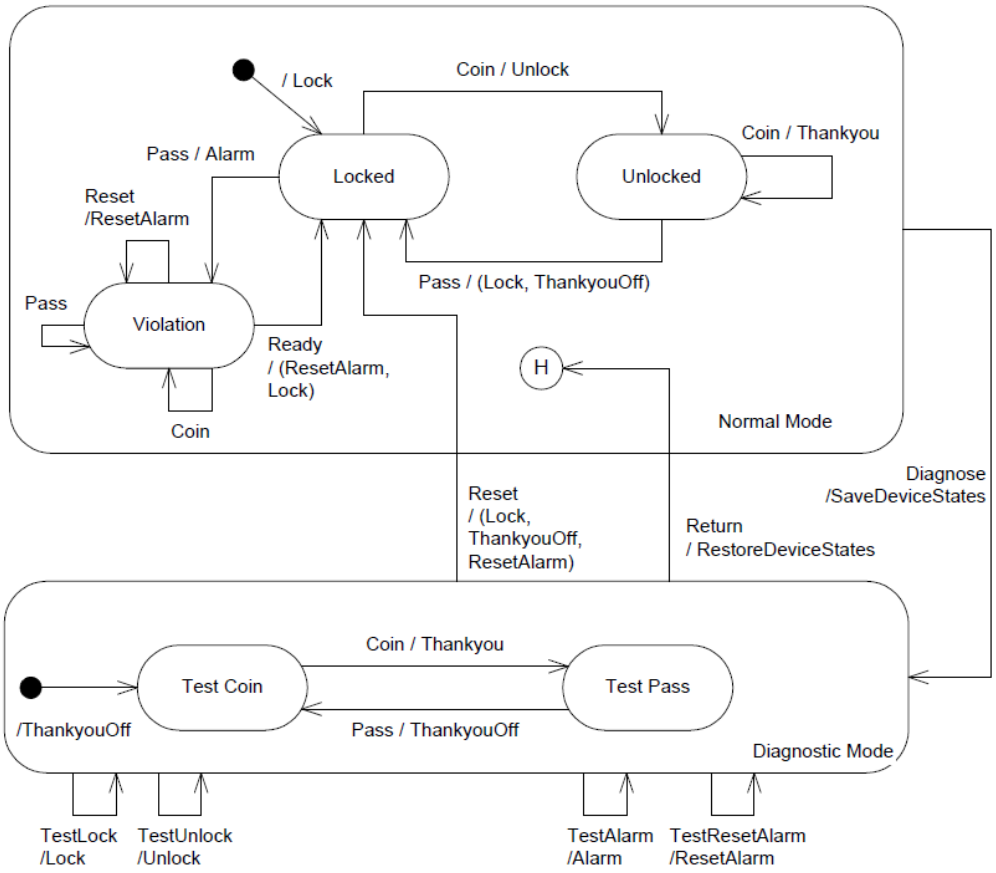


Figure 4: Turnstile with Diagnostic Mode.

IV. Analyzing the SimpleWatch use case diagram below, it is easy to notice that some usual relationships were not specified. Please includes these relationships and justify why these(this) are(is) needed. 1pt

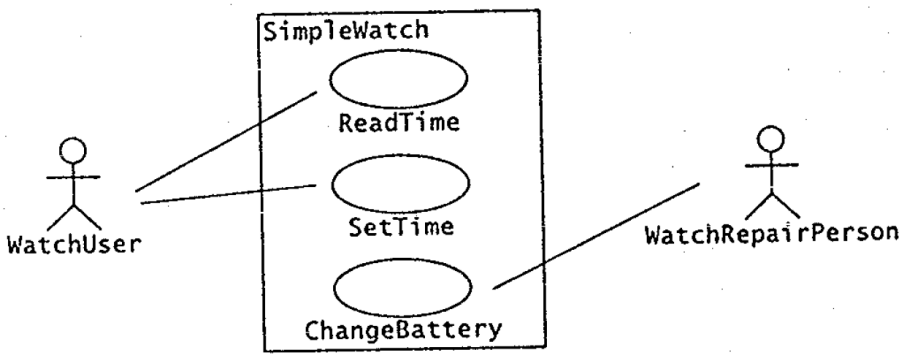


Figure 1 - An incomplete SimpleWatch use case diagram