Assignment 1

Group 33

17. September 2021

1 C Sharp

1.1 Generics

In the first method the type T must implement IComparable. In the second method T must be of type U and therefore both must implement iComparable.

2 Software Engineering

2.1 Exercise 1

What is meant by "knowledge acquisition is not sequential"? Provide a concrete example of knowledge acquisition that illustrates this.

It is not sequential since new knowledge you obtain may invalidate the previous. Galileo Galilei discovered that the earth was not the center of the solar system, by acquiring new knowledge. The new knowledge concerned the observation he made about how Jupiter's moons orbited. As such he invalidated previous knowledge by obtaining new.

2.2 Exercise 2

Specify which of the following decisions were made during requirements or system design:

- *"The ticket distributor is composed of a user interface subsystem, a subsystem for computing tariff, and a network subsystem managing communication with the central computer."*

  (This decision was made during system design)

- *"The ticket distributor will use PowerPC processor chips."*

  (This decision was made during system design)

- *"The ticket distributor provides the traveler with an on-line help."*

  (This decision was made during requirement design)

2.3 Exercise 3

In the following description, explain when the term account is used as an application domain concept and when as a solution domain concept: "Assume you are developing an online system for managing bank accounts for mobile customers. A major design issue is how to provide access

to the accounts when the customer cannot establish an online connection. One proposal is that accounts are made available on the mobile computer, even if the server is not up. In this case, the accounts show the amounts from the last connected session."

"Assume you are developing an online system for managing bank accounts for mobile customers."

- This sentence describes the system that users will interact with, which is the application domain.

"A major design issue is how to provide access to the accounts when the customer cannot establish an online connection."

- This describes a users problem which is part of the application domain.

"One proposal is that accounts are made available on the mobile computer, even if the server is not up. In this case,the accounts show the amounts from the last connected session."

- These mentions of accounts, relate to the modeling space of the system, which is part of the solution domain.

2.4 Exercise 4

A passenger aircraft is composed of several millions of individual parts and requires thousands of persons to assemble. A four-lane highway bridge is another example of complexity. The first version of Word for Windows, a word processor released by Microsoft in November 1989, required 55 person-years, resulted into 249,000 lines of source code, and was delivered 4 years late. Aircraft and highway bridges are usually delivered on time and below budget, whereas software is often not. Discuss what are, in your opinion, the differences between developing an aircraft, a bridge, and a word processor, which would cause this situation.

Unlike physical constructions such as airplanes and bridges, software entities are always almost always unique. Since the solution being developed is often unprecedented it can be difficult to anticipate time and money required. Whereas for a bridge or an aircraft the requirements for creating the product is more predictable. For example you need certain materials, you have an exact blueprint for their construction and a lot of examples of previous similar work. When it comes to software you mainly have an overview of what you need and not necessarily an exact way of getting where you want to be.

Other differences are in regards to making changes to the product. When creating software such as Word it is quite easy to change and update, whereas a bridge is almost set in stone once it is complete. An aircraft is somewhere in between where it has both software and hardware. The flight computer systems are easier to update unlike the physical manifestation of the aircraft.

2.5 Exercise 5

Specify which of these statements are functional requirements and which are nonfunctional requirements:

1. *"The TicketDistributor must enable a traveler to buy weekly passes."*
2. *"The TicketDistributor must be written in Java."*
3. *"The TicketDistributor must be easy to use."*
4. *"The TicketDistributor must always be available."*
5. *"The TicketDistributor must provide a phone number to call when it fails."*

1. Functional
2. Non-Functional
3. Non-Functional
4. Functional
5. Non-Functional

2.6 Exercise 6

What is the purpose of modeling?

A model can be seen as an abstraction of a system. In that way when you make a model, you ignore irrelevant details, and only focus on the important parts of the system.

When making software systems the process of developing is very complex. To make it less complex and easier to understand for everyone involved in the development of a software solution, you build a model of the application domain and solution domain.

First when modeling the application domain, the model is used to represent the real world concepts it deals with. This model can then be transformed into a solution domain model, with is more detailed and makes it possible for a developer to have an idea of the code which needs to be written for solving the problem.