Test of skills and knowledge

Mandatory assignment

Bike rental system

Case Description

Zealand - Sjællands erhvervsakdemi wants to support the green transition. They have therefore acquired bicycles, which can be rented by the students. Thus, Zealand wants to help reduce CO_2 emissions and meet the UN's world goals. To facilitate the administration of the rental of bicycles, Zealand needs a bicycle rental system (BikeRentalSystem).

By using the system, it should be possible to manage the rental of bicycles for the students. In the system, it must be possible to create and manage bicycles that can be rented (CRUD bicycles). For each bike, the year of purchase, model, frame color and Chassis number must be registered.
It must also be possible to manage bicycle rentals information. In connection with the rental, the start date and end date must be registered, just as the mileage start and end must be registered.
Each tenant must be registered by name, address, telephone number and student number (unique). It must be possible to manage tenants (CRUD tenants) in the system.

The first draft of the domain model looks like this:

Fig.1 First draft of the domain model

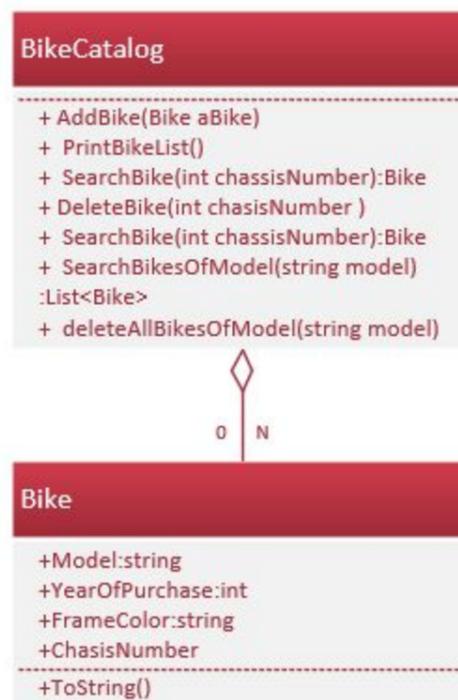The first draft of the design class diagram looks like this:

Fig.2 First draft of the design class diagram.

Assignment 1. User stories (SWD)

Write at least 3 user stories with acceptance criteria for the BikeRentalSystem.

Assignment 2. Domain model (SWD)

Expand the first version of the domain model in fig. 1. You must model the central parts of the BikeRentalSystem. It should contain a maximum of 6 conceptual classes.
You should use a digital drawing tool like Visio or DrawIO. You are free to choose your drawing tool.

Assignment 3. Design class diagram (SWD)

Expand the first version of the design class diagram in fig. 2. Draw a new design class diagram based on the domain model you have created in assignment 2..
You should use a digital drawing tool like Visio or DrawIO. You are free to choose your drawing tool.

Assignment 4. Create console application and Implement the class Bike (SWC)

Create a new console application (.net core) for the **BikeRentalSystem**. In the project you should implement a class Bike following the design class diagram.

- It should be possible to get and set the values for the properties **Model**, **YearOfPurchase** and **FrameColor**.
- It should be possible to get the **ChassisNumber**, but it should only be possible to set the ChassisNumber through the constructor.
- Implement a constructor that can initialize all the instance fields in the object.
- Implement a ToString method in the class. It should print out information about all the instance fields in the object.

 Create 3 instances of class Bike in the Main method of the class Program and print out each of them on the Console.

Assignment 5. Implement the class BikeCatalog (SWC)

In the BikeCatalog the collection of Bike objects should be maintained in a data structure of type List or type Dictionary. There should be only one bike object with the same chassis number in the BikeCatalog.

- Create a constructor for the BikeCatalog
  - Use the constructor to initialize the collection. (ret bullets)

- public void AddBike(Bike aBike)
  - This method adds a single Bike object to the collection.
  - In the Main method of the class Program:
    - Create an instance of the BikeCatalog and add the 3 objects you created in assignment 4.

- public void PrintBikeList()
  - This method should write all the Bike objects in the collection to the console window.
  - Each Bike object should be printed on a separate line
  - In the end the method should print out the number of objects in the collection and the current date and time.
  - Test the method in the Main method of the class Program.

- public Bike SearchBike(int chassisNumber)
  - This method returns a Bike object which has a matching ChassisNumber in the collection of objects.
  - If no such object exists the method returns null.
  - Test the method in the Main method of the class Program.

Assignment 6. Exceptions (SWC)

All the bikes in Zealand's BikesSystem should have a unique Chassis number. If the administration tries to add two bikes with the same ChassisNumber it is considered an error.
- Change the implementation of the BikeCatalog, so that it throws an ArgumentException if you try to add a Bike with a ChassisNumber that already is registered in the BikeCatalog.
- Write code to catch/handle the ArgumentException in the Main method of the class Program
- Try to add two new Bike objects with the same ChassisNumber and write out an error message in the console window.

Assignment 7. Implement inheritance (SWC & SWD)

Zealand also wants to rent out electric bikes (EBike). An EBike **is-a** Bike. The EBike should have a **Motor** property eg. 750 W.

- Implement a class EBike.
  - It should have a property **Motor**, a constructor and a ToString method.
  - It should be possible to initialize the properties through the constructors.
  - The ToString methods should return a string with information about the current object.
- Update the design class diagram with the EBike class (SWD).
- Test your implementation in the Main method:

- ○ Create 2 instances of the type EBike.
- ○ Add the objects to the BikeCatalog.
- ○ Print the Bike objects in the BikeCatalog using PrintBikeList() in theBikeCatalog.
- ● <u>Explain </u>in your own words the concept "inheritance".

Assignment 8. Extend the BikeCatalog (SWC)

Implement the following methods:

- ● *public void DeleteBike(int chasisNumber)*
  - ■ This method deletes a bike object from the collection of bikes, specifically the object which has a matching chassis number. If no such object exists no object is deleted.
- ● *public List<Bike> SearchBikesOfModel(string model)*
  - ■ The methods search for all bikes of a certain model in the collection of bikes
  - ■ The method returns a List of all the bikes object with a certain model.
  - ■ If no bikes exist in the collection of the certain model it returns an empty List.
- ● *public void deleteAllBikesOfModel(string model)*
  - ■ The methods deletes all bikes of a certain model in the collection of bikes

Assignment 9. (SWC)

Implement the classes from your design class diagram that you made in assignment 3.