**Course: Object-Oriented Programming Fundamentals in C#**

1. -Develop an application using the topics that seen in the course.  
   **Note**: Some topics may not apply.
2. -Implements IBaseData and ILogger interfaces on all the entities of the project.
3. -The Log method of the entities must log all the attributes of the object.
4. -Develop repositories for all entities in the project.
5. -Develop at least one unit test for one entity repository, the test must have a RetrieveValid and RetrieveNotFound methods, is desirable to get the data directly from database choosing the implementation of you choose to call the database.
6. -Commit the solution to Git repository in the folder with your initials.

* **Identifying classes from requirements**
  + Analyze the business problem
  + Start with the nouns
  + Define appropriate members
  + Consider time
  + Abstraction
  + Encapsulation
* **Building entity classes**
  + Layering the application
  + Building the business logic layer component
  + Building a class: Properties
  + Using Snippets
  + Testing the class
  + Testing the class: Valid values
  + Testing the class: Invalid values
  + Working with Objects
  + Objects are reference types
  + Static modifier
* **Building entity classes – Methods**
  + Building a class: Methods
  + Testing the methods
  + Terms
  + Building a class: constructors
  + Building the remaining classes
* **Separation of responsibilities**
  + Separation of concerns
  + Revisiting the class diagrams
  + Building the repository class
  + Testing a repository class
  + Building the remaining repositories
* **Establishing relationships**
  + Defining the relationships
  + Types of relationships
  + Collaboration
  + Composition: References
  + Population the referenced objects
  + Testing a composition relationship
  + Composition: Ids
  + Inheritance
* **Leveraging reuse through inheritance**
  + Secrets of reuse
  + The .NET Object class
  + Overriding base class functionality
  + Polymorphism
  + Building a base class
  + Demo: Building a base class
  + Preparing overridable base class members
* **Building reusable components**
  + Scenario
  + Building a reusable component
  + Testing the reusable component
  + Using the reusable component
  + Static classes
  + Extension methods
  + Static method vs Extension method
* **Understanding interfaces**
  + Class interface
  + Interface Metaphors
  + Setting up the demo
  + Defining an Interface
  + Implementing an Interface
  + Interface-based Polymorphism

**Exercise**

* **Identifying classes from requirements**

**Application Name**: Phone Book

**--Requirements**:

* + Build a web application that displays the employee data required to contact him.
  + Displays the following employee data:
    - Employee number
    - Full name
    - Job
    - Cost Center
    - Department
    - Cell phone number
    - Short phone number
    - Extension phone number
    - Shift
  + The application must have a search engine to filter the information:
    - A general search engine that allows to search the entire data set.
    - Each column must have its filter.
  + The application must have an administration menu to:
    - Add an employee.
    - Edit an employee.
    - Remove an employee.
  + The application must have a user catalog to:
    - Add a new user.
    - Edit a user.
    - Remove a user.
  + The user types are:
    - Administrator: Access to all features of the application.
    - Common user: Access only to see the contact employee data.
  + The application must have a configurable logo image.
  + The application must have the option to export the contact employee data to Excel, PDF and print.

**--Classes**

**Employee**

* + Employee number
  + First name
  + Last name
  + Second name
  + Email
  + Cell phone number
  + Short number
  + Extension number
  + Status
  + Shift
  + Job
  + Department

**Department**

* + Description
  + Cost center
  + Status
  + Manager

**Job**

* + Description
  + Status
  + Department

**Shift**

* + Description
  + Status

**User**

* + Username
  + Password
  + Status
  + First name
  + Last name
  + Second last name
  + Email
  + User type
  + Shift
  + Department

**User Type**

* + Description
  + Status