# Introduction

ESC GymApp is a Windows Forms application written in C# using Visual Studio 2017. We were tasked to produce a piece of software using the contextual scenario below and follow all professional computing standards and good practice procedures. The finished solution consists of the frontend (GUI) and the backend (code).

# Scenario

You are asked to use your software design, development, testing and evaluation understanding and skills to produce a program that meets the client’s requirements.

East Surrey College gym has commissioned you as a software developer to write a program that will assess a gym member’s requirement to maintain their current weight accurately.

You need to create a program that will give the learners gym member information about:

• Their current basal metabolic rate (BMR)

• Their current body mass index (BMI)

• Their target BMI

• The number of kilocalories to maintain their current weight.

# Scope

## In Scope

## The program calculates and displays a gym member's basal metabolic rate (BMR)

## Calculates and display the gym member's body mass index (BMI)

## The number of kilocalories to maintain their current weight.

## The BMR calculation is given to 2 decimal places.

## The BMI calculation is given to 1 decimal place.

## The kilocalorie requirement output is shown rounded to a whole number.

## The program needs a "login" screen to access the application.

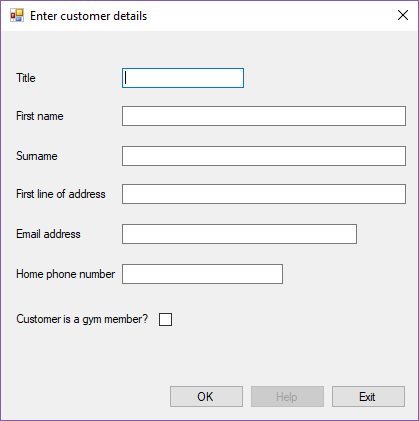
## A screen to collect customer details, if they're an existing gym member, name, address, email and telephone.

## Non-Functional Requirements

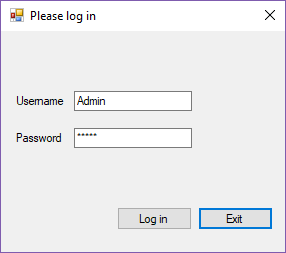
## *Must be Windows Form application written in C# & Java*

# Form design

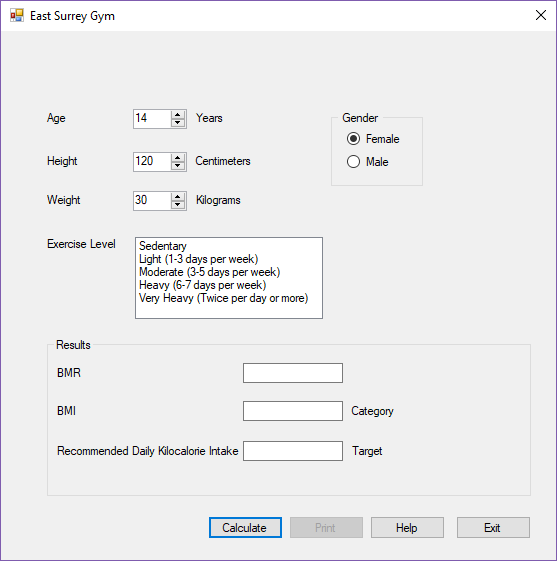
## Customer Details Form



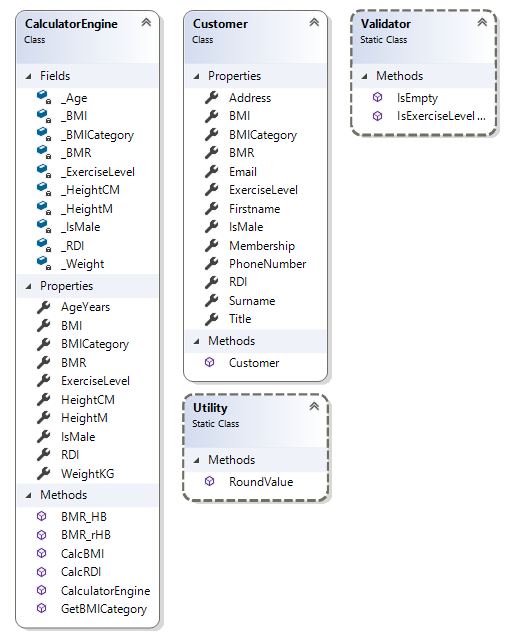
## Login Form



## Gym Form

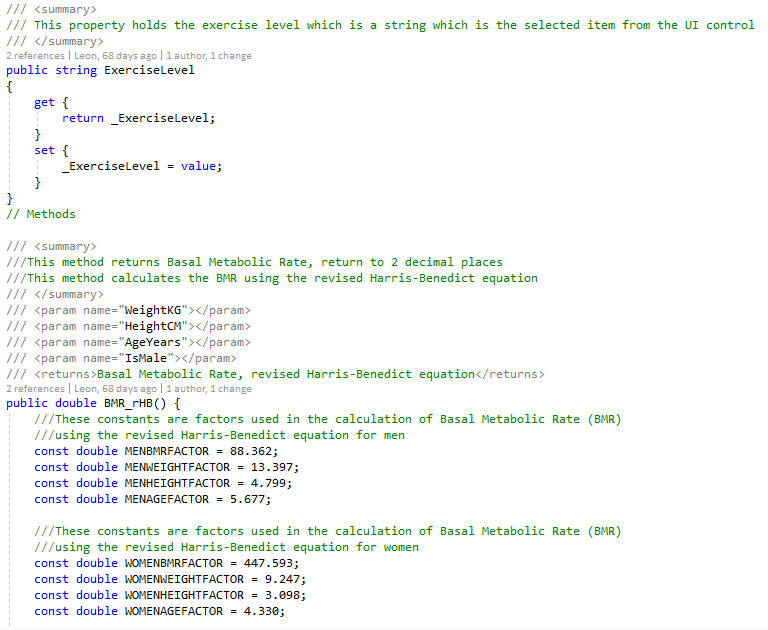


# Class Diagram

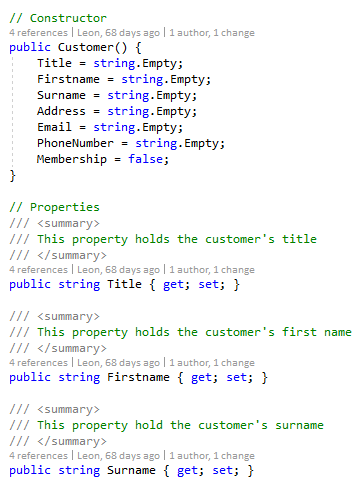


# Classes

## CalculatorEngine.cs



## Customer.cs



## EnteringDetails.cs

## ESCGym.cs

## Login.cs

## SummaryValues.cs

## Utility.cs

## Validator.cs

# Properties

## Adding properties

## CalculatorEngine properties

## Customer properties

# Methods

## RoundValue : double

## GetBMICategory : string

## BMR\_HB : double

# Testing

## NUnit - Unit Tests

## CalculatorEngineTest.cs

## CustomerTest.cs

## UtilityTest.cs

## ValidatorTest.cs

# Review

## Richard’s Review

## What went well

### Development

### Database Schema

### QA Testing

## What didn't go well

### Project Management

### Development

### QA Testing