|  |  |
| --- | --- |
|  | **Cognizant Academy**  **Over Analysis**  **ASP.Net MVC, Entity Framework, SQL Server Integrated Capability Test**  **Version 0.1** |
|  |
|  |

Table of Contents

[1.0 Introduction 2](#_Toc38027157)

[1.0 Purpose of this document 2](#_Toc38027158)

[2.0 Definitions & Acronyms 3](#_Toc38027159)

[3.0 Project Overview 3](#_Toc38027160)

[4.0 Scope 3](#_Toc38027161)

[5.0 Hardware and Software Requirment 3](#_Toc38027162)

[2.0 System diagram 4](#_Toc38027163)

[3.0 Design for Displaying Over Analysis Details (HomePage) 4](#_Toc38027164)

[1.0 Requirement flow 4](#_Toc38027165)

[2.0 Technical guidelines 5](#_Toc38027166)

[Component Specification – Model 6](#_Toc38027167)

[4.0 Design for Adding Over Analysis Details to the database 7](#_Toc38027168)

[1.0 Requirement flow 7](#_Toc38027169)

[2.0 Add Over Analysis 9](#_Toc38027170)

[3.0 Technical guidelines 11](#_Toc38027171)

[5.0 Standards and Guidelines 13](#_Toc38027173)

[1.0 Controller & View 13](#_Toc38027174)

[6.0 Design constraints 14](#_Toc38027175)

[7.0 Code submission Instructions 14](#_Toc38027176)

[8.0 Evaluation Areas 14](#_Toc38027177)

# Introduction

## Purpose of this document

The purpose of this document is to define the server side implementation of the Over Analysis application.

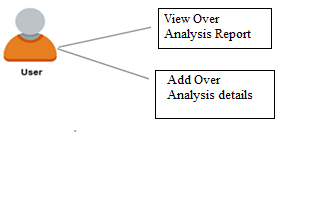
## Definitions & Acronyms

|  |  |
| --- | --- |
| Definition / Acronym | Description |
| ASP.NET MVC | ASP.Net MVC is a Web development framework built on top of ASP.Net with certain changes in the internal workings of web page rendering |

## Project Overview

The project involves analysing over details. The user will enter the over analysis details such as runs taken for each ball and select ball type of either [No Ball] / [Wide Ball] / [Right Ball]. These over analysis details are added in ‘OverAnalysis’ table of the ‘OverAnalysisDB’ database. The Over analysis details of Total Runs and Extras taken in an over which are retrieved from the database and displayed on the view page using group by ‘Overs’ query.

**Use Case Diagram – OverAnalysis**



## Scope

1. Creation of ASP.Net MVC web application for Over Analysis application.

## Hardware and Software Requirment

1. Hardware Requirement:
   1. Developer PC with 8GB RAM
2. Software Requirement
   1. IE or Chrome
   2. .Net Framework 4.5
   3. Visual Studio Professional Edition 2015
   4. SQL Server enterprise edition 2014

# System diagram

Save in database

Validate Over Analysis details

Enter Over Analysis details

Display Over Analysis details

# Design for Displaying Over Analysis Details (HomePage)

## Requirement flow

**Steps Explanation**

1. Application user launches the application.
2. The page is displayed with Over Analysis details retrieved from the database. This is the default homepage*.*

## Technical guidelines

1. Create a controller named ‘OverAnalysisController’.
2. Create an ‘OverAnalysisReport’ action with no arguments in the ‘OverAnalysisController’.

Controller Specification for OverAnalysisReport Action with no argument

|  |  |  |  |
| --- | --- | --- | --- |
| **Class** | **Action Name** | **ReturnType** | **HTTP Method** |
| OverAnalysisController | OverAnalysisReport | ActionResult | GET |

1. In Over Analysis Controller, inside the ‘OverAnalysisReport’ action, use Entity Framework to retrieve the over analysis details from the ‘OverAnalysis’ table in the database and display the total runs and extras for each overs using group by ‘Overs’ column from the table.
2. Create ‘OverAnalysisContext’ class which inherits DbContext class. Include namespace “System.Data.Entity”.
3. Create a constructor of the OverAnalysisContext. Specify the name of the database connection string element to be “OverAnalysisDB” (connection string name in Web.config).
   1. Use “OverAnalysisDB” as ‘Context’ name in Entity Framework. Sample config entry is provided below.

<connectionStrings>

<add name="OverAnalysisDB"

connectionString="Data Source=(LocalDb)\MSSQLLocalDB;Initial Catalog= OverAnalysisDB; Integrated Security= true" providerName="System.Data.SqlClient"/>

</connectionStrings>

1. Declare a property ‘OverAnalysis’ of type DbSet<OverAnalysis> in the OverAnalysisContext class.
2. The Code First approach is used to implement the entityframework.
3. Retrieve all the Over analysis details from the database.
4. Create the OverAnalysis model with Id, Overs, BallType, and RunsTaken as property.

# Component Specification – Model

|  |  |  |  |
| --- | --- | --- | --- |
| **Class** | **DataType** | **Property** | **Data Annotation** |
| OverAnalysis | int | Id | Key |
|  |  |  |
| int | Overs |  |
| string | BallType |  |
| int | RunsTaken | Range(1,7,ErrorMessage = “Runs Taken should be between 1 and 7”) |

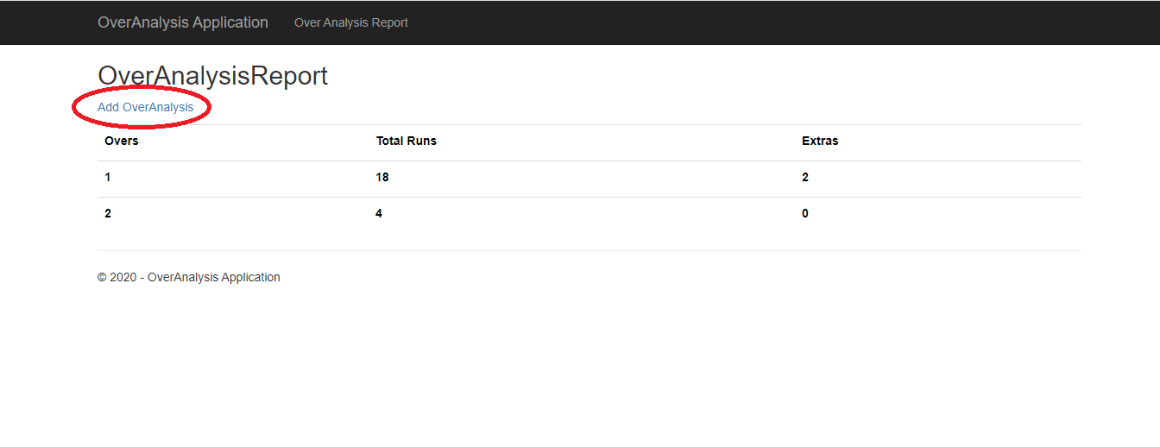
Note:

* + Use ‘**HTMLHelper’** to list Over analysis details.

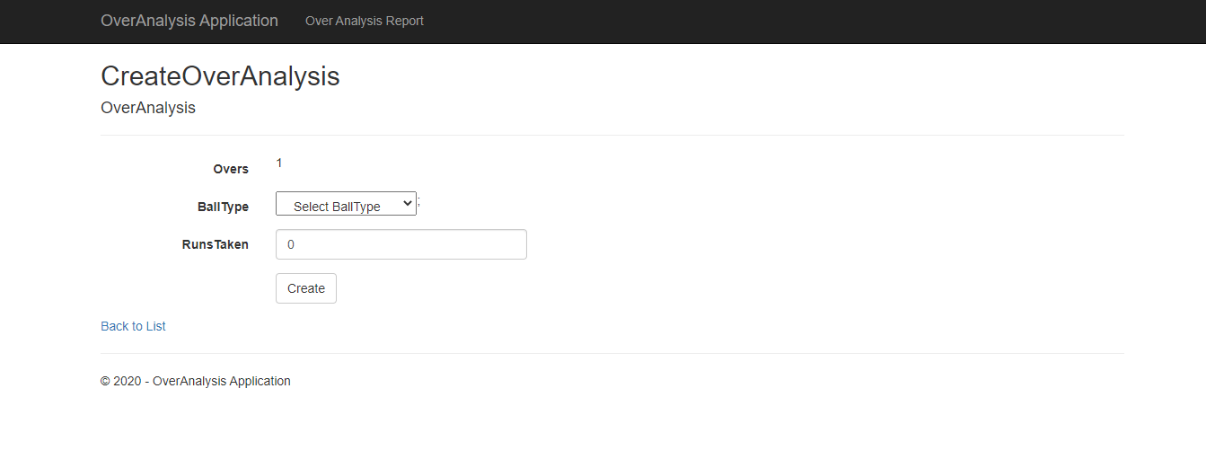
# Design for Adding Over Analysis details to the database

## Requirement flow

**Steps Explanation**

1. Create a link with text “Add OverAnalysis” in the Over Analysis Report page.
2. 
3. On clicking this ink, the page should navigate to the ‘Add OverAnalysis’ page.
4. The user must be able to enter the over analysis details in the displayed form.

## Add Over Analysis



UI Controls for Over Analysis Updation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data Element** | **Control type** | **Default Values** | **Editable Field** | **Mandatory/Not Mandatory** |
| Ball Type | DropDownList | No | Yes | Mandatory |
| Runs Taken | Textbox | No | Yes | Mandatory |

Validate the form to ensure there are no empty fields.

1. On Submit, the over analysis details must be added to the ‘OverAnalysis’ table in the database. And it redirects to “OverAnalysisReport” view page.

## Technical guidelines

* Steps for adding Over Analysis – Points 1 to 3
* Steps for saving the details to database - Points 4 to 7

1. Create an **‘CreateOverAnalysis’** action with no arguments. This ‘**CreateOverAnalysis**’ action return type must be‘ActionResult’

Controller Specification for **CreateOverAnalysis** Actionwith no argument

|  |  |  |  |
| --- | --- | --- | --- |
| **Class** | **Action Name** | **ReturnType** | **HTTP**  **Method** |
| OverAnalysisController | CreateOverAnalysis | ActionResult | GET |

1. The ‘Add OverAnalysis’ link is mapped to this action. This action must return ‘CreateOverAnalysis’ View containing the form to enter the Over Analysis details.

Note:

* Use ‘**HTMLHelper’** to create a form.
* Using HTMLHelper will automatically create ‘id’ for all the form element for the input box. These ‘id’ is required for auto-evaluation.

Like,

id="Overs", id="BallType", id=”RunsTaken”, id="Submit"

id of ‘submit’ button must be ‘Submit’

1. Create an **‘CreateOverAnalysis’** action (HttpPost) with **“OverAnalysis”** model as arguments in the ‘OverAnalysisController’. This ‘CreateOverAnalysis’ action return type must be ‘ActionResult’.

Controller Specification for **Create Action** with argument

|  |  |  |  |
| --- | --- | --- | --- |
| **Class** | **Action Name** | **ReturnType** | **HTTP**  **Method** |
| OverAnalysisController | CreateOverAnalysis( OverAnalysis overanalysis) | ActionResult | POST |

1. This ‘CreateOverAnalysis’ action is implemented as POST to get the values of Over analysis details posted from the form.
2. Use **Entity framework** to add the details to the database.
3. Use the ‘OverAnalysisContext’ to connect to the database.
4. Inside the ‘CreateOverAnalysis’ action,
   1. Create an instance of ‘OverAnalysisContext’.
   2. Add the data in OverAnalysis object to its entity.

# Standards and Guidelines

## Controller & View

1. Action methods should have a meaningful name
2. Remove unused Action methods
3. There should not be any hard coded values in code. It has to be referenced from Web.config file
4. Database connection string should be set in the ConnectionStrings section of Web.config and NOT in the AppSettings
5. Meaningful names should be given to the controls created in View

# Design constraints

Required packages are already supplied with ‘Using’ statement. So do not try to add packages using Nuget packages.

# Code submission Instructions

* + 1. Do not change the code skeleton given, as your code will be auto-evaluated.
    2. Your last submitted solution will be considered for detailed evaluation.
    3. Make sure to submit the solution before the specified time limit. You will not be allowed to submit the solution once the mention time for the assessment is over.

# Evaluation Areas

|  |  |
| --- | --- |
| 1 | Launched application lands in the Home page |
| 2 | Home Page UI contains the over analysis details |
| 3 | Link ‘Add OverAnalysis’ navigates to CreateOverAnalysis page |
| 4 | CreateOverAnalysis Page contains required form elements |
| 5 | Submit the over analysis details |
| 6 | Display the over analysis details |
| 8 | Form validation for empty fields |
| 9 | Implementation of OverAnalysisContext |