****

Formal application

Postulant: Durón de Luna Oscar Alejandro

Test: *Dev Candidate Test 2*

from 06/01/2023 a las 17:45 hrs.

to 07/01/2023 a las 17:45 hrs.

“Documentation”

Imagen que contiene Logotipo

Descripción generada automáticamente

**Dev Candidate Test 2**

Create a solution that can:

* Persist Employees to a data store
* Validate uniqueness of an employee (uniqueness is defined by RFC)
* Validate RFC Format and length
* Retrieve Employees sorted by born date and filtered optionally by name

Considerations:

* UI can be any Microsoft presentation technology (Web API, [ASP.NET](http://asp.net/) MVC, etc.)
* The components used should be able to be consumed by an alternate presentation layer
* Don’t focus on aesthetics
* Any 3rd party libraries can be used
* Any code that you’ve built for previous applications can be used

public class Employee

{

                public int ID { get; set}

                public string Name { get; set;}

                public string LastName { get; set;}

               public string RFC { get; set;}

                public DateTime BornDate { get; set; }

                public EmployeeStatus Status { get; set; }

}

public enum EmployeeStatus

{

                NotSet,

                Active,

                Inactive,

}

Index

[**PROBLEM STAGE** 4](#_Toc123971803)

[**DESIGN** 4](#_Toc123971804)

[**CREATING THE APPLICATION** 5](#_Toc123971805)

[**CREATING THE REPOSITORY** 7](#_Toc123971806)

**DOCUMENTATION**

# **PROBLEM STAGE**

* Persist Employees to a data store

**Defined by the user**

*ID, Name, LastName, RFC, BornDate, Status (EmployeeStatus)*

* Validate uniqueness of an employee (uniqueness is defined by RFC)
* Validate RFC Format and length
* Retrieve Employees sorted by born date and filtered optionally by name

Sorted by Born date/name

Retrieve: ID, Name, LastName, RFC, BornDate (with **format**), Status (EmployeeStatus with **choices**)

**Side notes about considerations**:

* The components used should be able to be consumed by an alternate presentation layer – *BFF?*
* Any code that you’ve built for previous applications can be used – **There is no previous code**

# **DESIGN**

Tabla

Descripción generada automáticamente

# **CREATING THE APPLICATION**

* InitializeGridView to create the columns inside of the grid (Form1.cs)
* FillingGrid receives a List to fill the grid (Form1.cs)
* SaveEmployee to save the data given by the user (Form1.cs)
* Btn\_save\_Click it triggers different validations and methods, such as SaveEmployee (Form1.cs)
* Btn\_OrderByName\_Click it fills the grid with the list’s data sorted by name (Form1.cs)

***DataEmployees.cs***

**Obligatory Methods:**

* InsertData Insert the data given into a List (DataEmployees)
* ValidateUniquenessRFC Verify that the RFC is not repeated (DataEmployees)
* ValidateRFCFormatAndLenght Verify the RFC format and length (DataEmployees)
* OrderDataByBornDate Order the list by born date (DataEmployees)
* OrderDataByName Order the list by name (DataEmployees)

**Optional methods:**

* ValidString that must verify if a given string has a valid length or is valid (DataEmployees)
* RetrieveNewID Seems like there is not any disposition about the ID, so I made it automatic. (DataEmployees)

**Considerations:**

**How is an RFC conformed?**

Escala de tiempo

Descripción generada automáticamente

El RFC necesita **dos apellidos**, pero según la ley, si el usuario solo cuenta con un apellido entonces se conformará con la **primera y segunda letras del apellido paterno o materno**, según figure en el acta de nacimiento, **más la primera y segunda letras del nombre.**

**Initializing the list**

|  |  |
| --- | --- |
|  | ID = 1,  Name= "Pedro",  LastName = "Rubens",  RFC = "RUPE770628001",  BornDate = DateTime.Parse("28/06/1977"),  Status = EmployeeStatus.Active, |
| Théodore Géricault - Wikipedia, la enciclopedia libre | ID = 2,  Name= "Theodore",  LastName = "Gericault",  RFC = "GETH910926002",  BornDate = DateTime.Parse("26/09/1991"),  Status = EmployeeStatus.Inactive, |
| Paolo Caliari (Paolo Veronese) - Biografia, opere e mostre - Arte.it | ID = 3,  Name= "Paolo",  LastName = "Caliari",  RFC = "PACA880419003",  BornDate = DateTime.Parse("19/04/1988"),  Status = EmployeeStatus.NotSet, |

# **REPOSITORY**