

Gestión Eficiente de Base de Datos de Personas

Danna Alexandra Espinosa Arenas

Cristian Felipe Perafan Chilito

Facultad de Ingeniería y Diseño, Universidad Icesi

09781 : Computación y estructuras discretas I.

Ph. Uram Anibal Sosa Aguirre.

2 may 2022

ÍNDICE:

[FUNCTIONAL REQUIREMENTS](#)

[CLASS DIAGRAM](#)

[TEST CASE DESIGN](#)

FUNCTIONAL REQUIREMENTS

Requirement #1:

Nombre o Identificador	R1. Creation of people		
Resumen	The system must allow the automatic creation of persons with their respective data, for which the databases suggested by the teacher must be implemented (code (auto-generated), name, surname, sex, date of birth, height, nationality and photograph).		
Entradas	Nombre	Tipo	Condición
	Name	String	
	Surname	String	data entered above must be different from null
	Sex	String	data entered above must be different from null
	Date of birth	Date	data entered above must be different from null
	Height	Double	data entered above must be different from null
	Nationality	String	data entered above must be different from null
	Photograph	Png	data entered above must be different from null
Resultado	A person is created with the above data and a unique code is generated to identify him/her.		
Salidas	Persons		

Requirement #2:

Nombre o Identificador	R2. Persistent		
Resumen	The program must allow basic persistent storage functions (Create, Read, Update and Delete) with the system data, especially the data of persons.		
Entradas	Nombre	Tipo	Condición
	Option	String	
Resultado	The requested action is executed, e.g., read databases.		
Salidas			

Requirement #3:

Nombre o Identificador	R3. Updating data person		
Resumen	The program should allow updating a person's data.		
Entradas	Nombre	Tipo	Condición
	Name	String	The previous name of the person must be different from null
	Surname	String	The previous Surname of the person must be different from null
	Sex	String	The previous sex of the person must be different from null
	Date of birth	Date	The previous date of birth of the person must be different from null
	Height	Double	The previous height of the person must be different from null
	Nationality	String	The previous nationality of the person must be different from null
	Photograph	Png	The previous photograph of the person must be different from null
Resultado	The person's data is maintained		
Salidas			

Requirement #4:

Nombre o Identificador	R4.Deleting data person		
Resumen	The program should allow deleting a person's data.		
Entradas	Nombre	Tipo	Condición
	Code	String	This person must exist in order to be eliminated
Resultado	The person is removed from the application		
Salidas			

Requirement #5:

Nombre o Identificador	R5 Searching by First Name		
Resumen	The program must allow a person to be searched by first name and display the list of found persons		
Entradas	Nombre	Tipo	Condición
	First Name		There must have been at least one person in the application.
Resultado	The list of search results by first name is displayed.		
Salidas	list of persons found	ArrayList	

Requirement #6:

Nombre o Identificador	R6 Searching by Last Name		
Resumen	The program must allow a person to be searched by last name and display the list of found persons		
Entradas	Nombre	Tipo	Condición
	Last name	String	There must have been at least one person in the application.
Resultado	The list of search results by last name is displayed.		
Salidas	list of person found	ArrayList	

Requirement #7:

Nombre o Identificador	R7.Searching by Full Name		
Resumen	The program must allow a person to be searched by full name and display the list of found persons		
Entradas	Nombre	Tipo	Condición
	Full name	String	There must have been at least one person in the application.
Resultado	The list of search results by full name is displayed.		
Salidas	list of persons found	ArrayList	

Requirement #8:

Nombre o Identificador	R8. Searching by Code		
Resumen	The program must allow a person to be searched by code and display the list of found persons		
Entradas	Nombre	Tipo	Condición
	Code	String	There must have been at least one person in the application.
Resultado	The list of search results by code is displayed.		
Salidas	List of persons found	ArrayList	

CLASS DIAGRAM

TEST CASE DESIGN

Configuración de los Escenarios

Nombre	Clase	Escenario
		-

* El nombre de los escenarios puede ser setupStage1, setupStage2, etc.

* La clase es la clase de testing correspondiente al modelo donde acontece el escenario. Por ejemplo si usted está probando User, clase será UserTest.

* El escenario es la descripción de las condiciones iniciales del escenario.

Diseño de Casos de Prueba

Objetivo de la Prueba:				
Clase	Método	Escenario	Valores de Entrada	Resultado

* En el objetivo de la prueba debe escribir una descripción sobre qué es lo que específicamente está probando del modelo del programa.

* La clase es la clase del modelo que está siendo puesto a prueba.

* El método es específicamente el método de la clase que está siendo puesto a prueba.

* El escenario se refiere al nombre del escenario que usted definió. Todos los casos de prueba corresponden a escenarios.

* Los valores de entrada son valores que entran al método puesto a prueba.

* El resultado es lo que se espera que suceda luego de ejecutar el método.