SHEFFIELD HALLAM UNIVERSITY

Faculty of Arts, Computing, Engineering and Sciences

Student Name: **PHINA ONYINYE ANYALECHI** Student Number: **26006046**

Course Title: **MSC. COMPUTER AND NETWORK ENGINEERING**

Module Title: **OBJECT ORIENTED METHODS**

Tutor: **DR. ALEX SHENFIELD** Module Code: **16-7213**

Assessment Title: **OBJECT ORIENTED DESIGN IN C#**

Assessment ID: **2CWK50** Date Submitted: **MAY, 2016**

REQUIREMENT SPECIFICATIONS

To undertake a full object oriented design for a CD cataloguing system and implement the final design in C#. The system should hold a collection of entries for audio CDs containing:

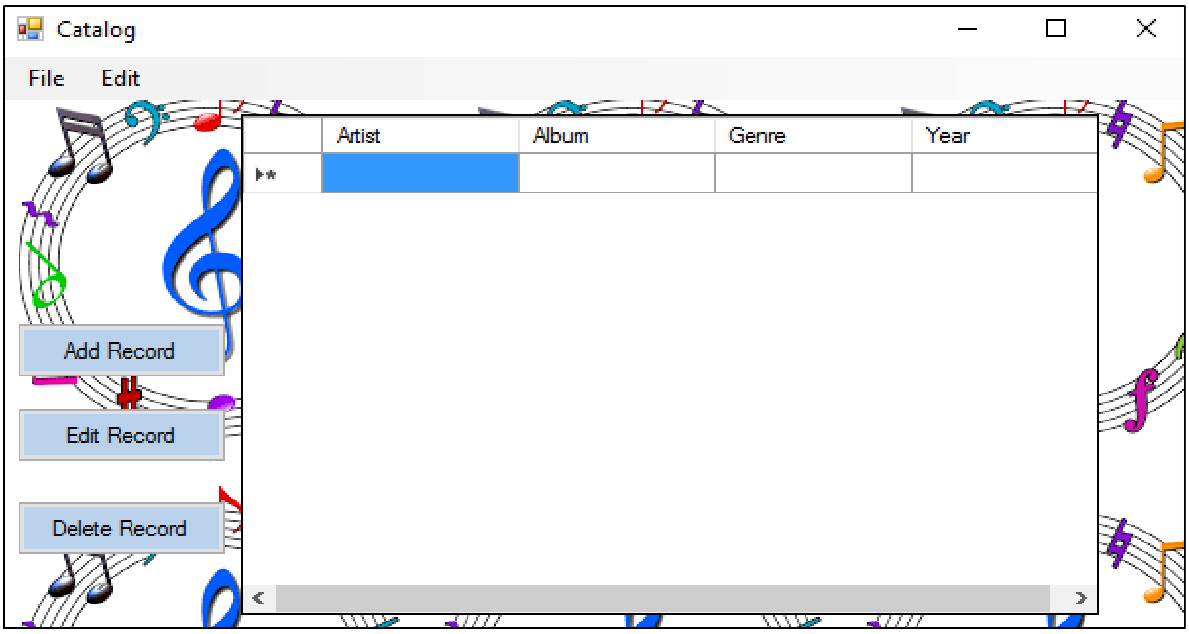
- Artist name
- Album title
- Genre
- Year of recording

It should also add new CD to system, edit an existing record, delete a record, sort the entries in the CD cataloguing system alphabetically by artist, by album, by year of recording, and by genre. It should also create a new catalogue, open a file containing an existing catalogue, close catalogue currently opened, and to save current catalogue to a file (using the standard file menu options). The file menu should also have the exit application option.

The initial requirements assume that a user should be able to work with a single catalogue at a time, therefore if user chooses to create a new catalogue then the current one has to be closed. The application should also keep track of whether changes have been made since the last save and prompt user to save changes when closing the application (or opening a new catalogue).

An example user interface design was given and has been adopted as shown below.

USER INTERFACE

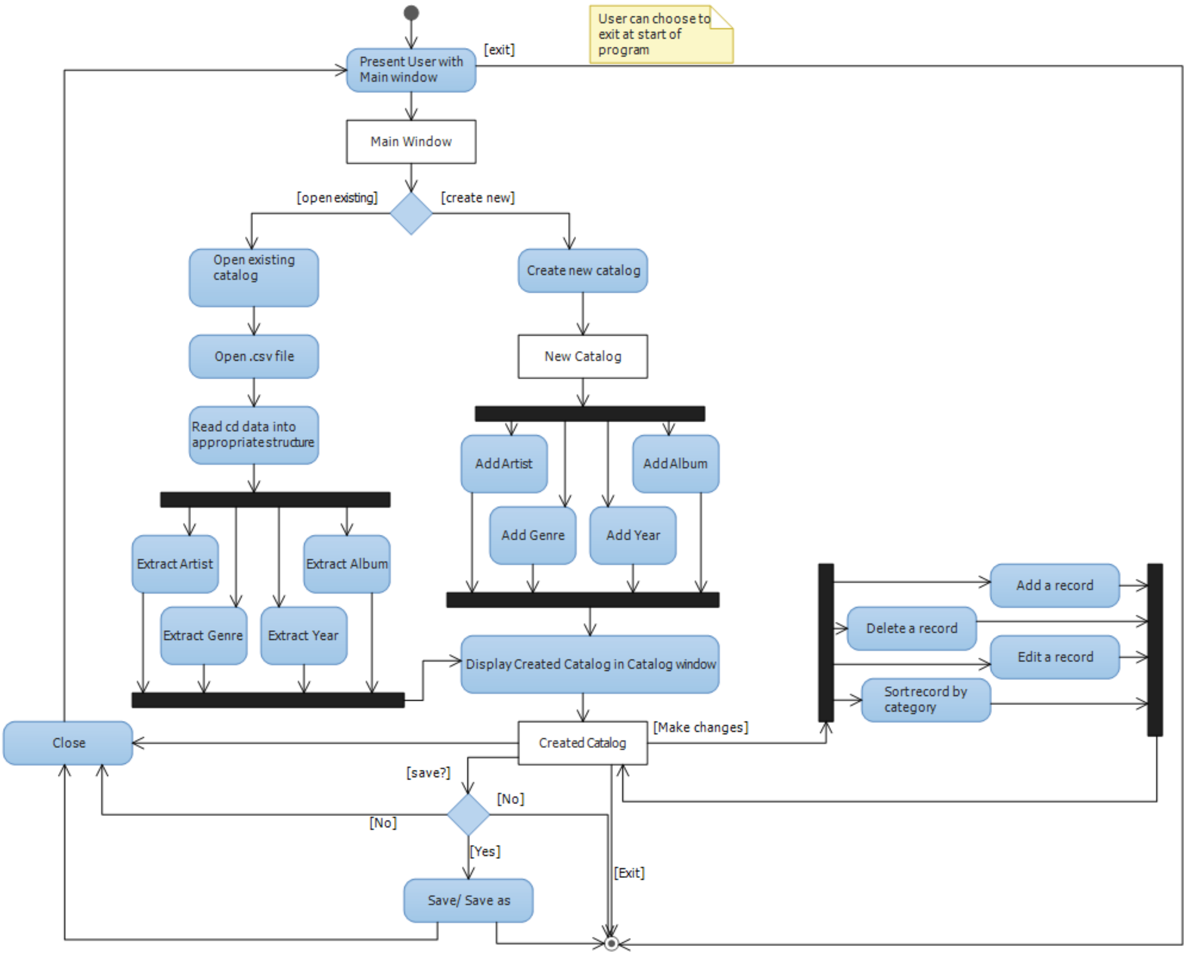


REQUIREMENTS ANALYSIS

Based on the user requirements specifications, this report contains the UML activity diagram, UML use cases diagram, UML sequence diagrams, and UML class diagrams discovered during analysis.

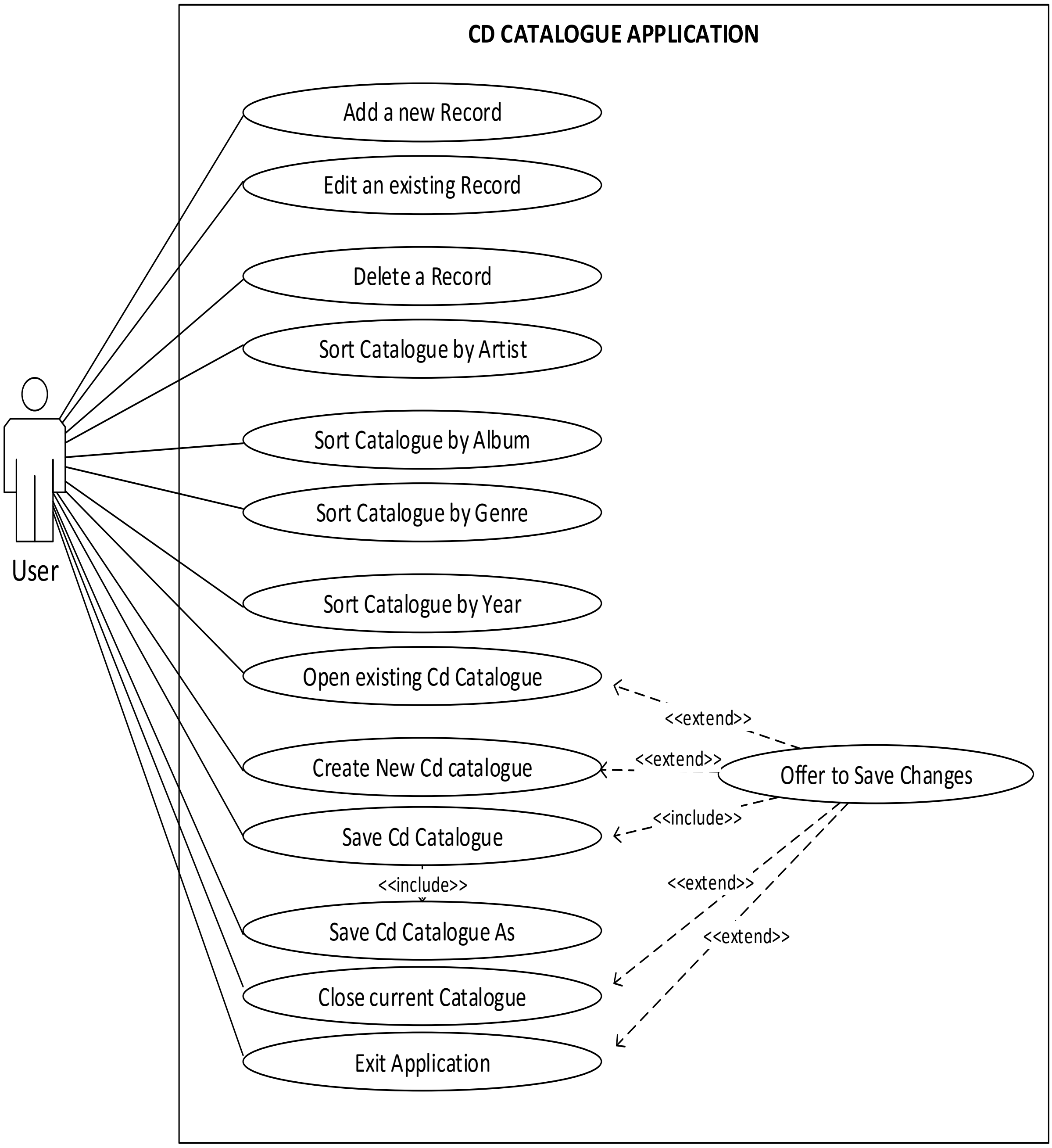
UML ACTIVITY DIAGRAM FOR THE CD CATALOGUE

The activity diagram below shows the behaviour of the CD Cataloguing system. It shows the workflows of the activities step-by-step, describing parallel and conditional activities.



UML USE CASE DIAGRAM FOR THE CD CATALOGUE

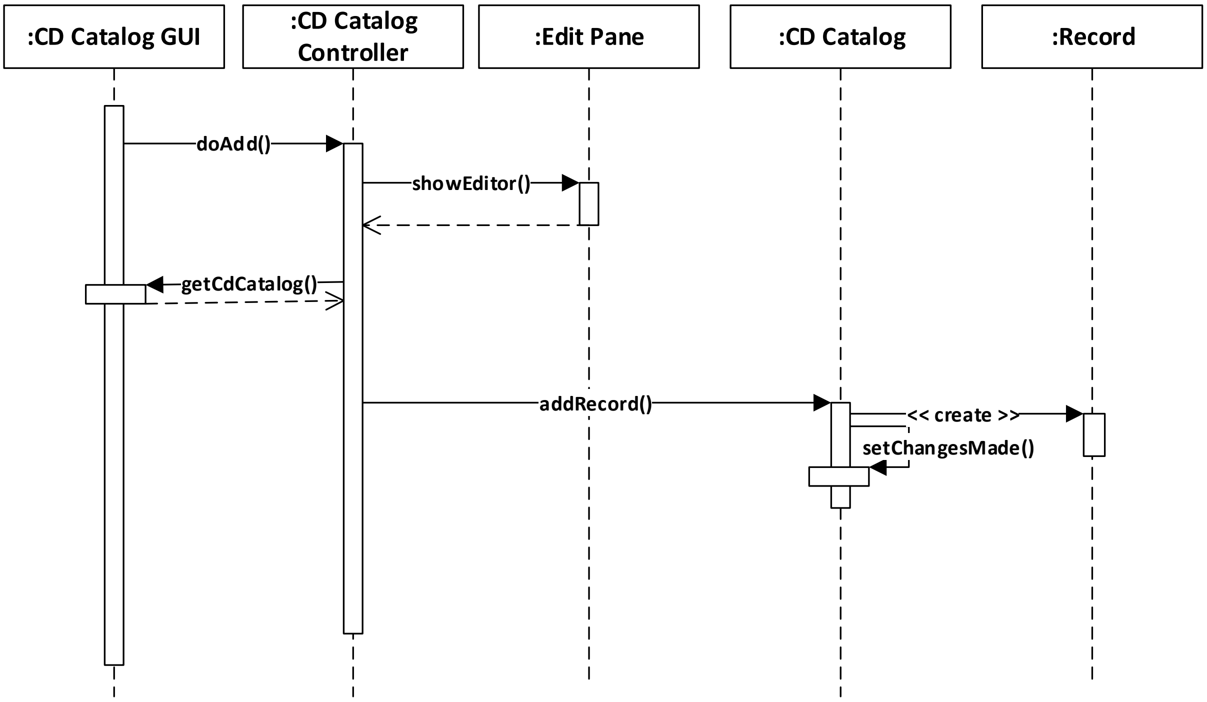
The use cases listed below are listed as a user would think of them, according to the requirement specifications.



INDIVIDUAL USE CASE DIAGRAMS (UML SEQUENCE DIAGRAMS) FOR CD CATALOGUE WITH THE FLOWS OF EVENTS

Add A Record Use Case:

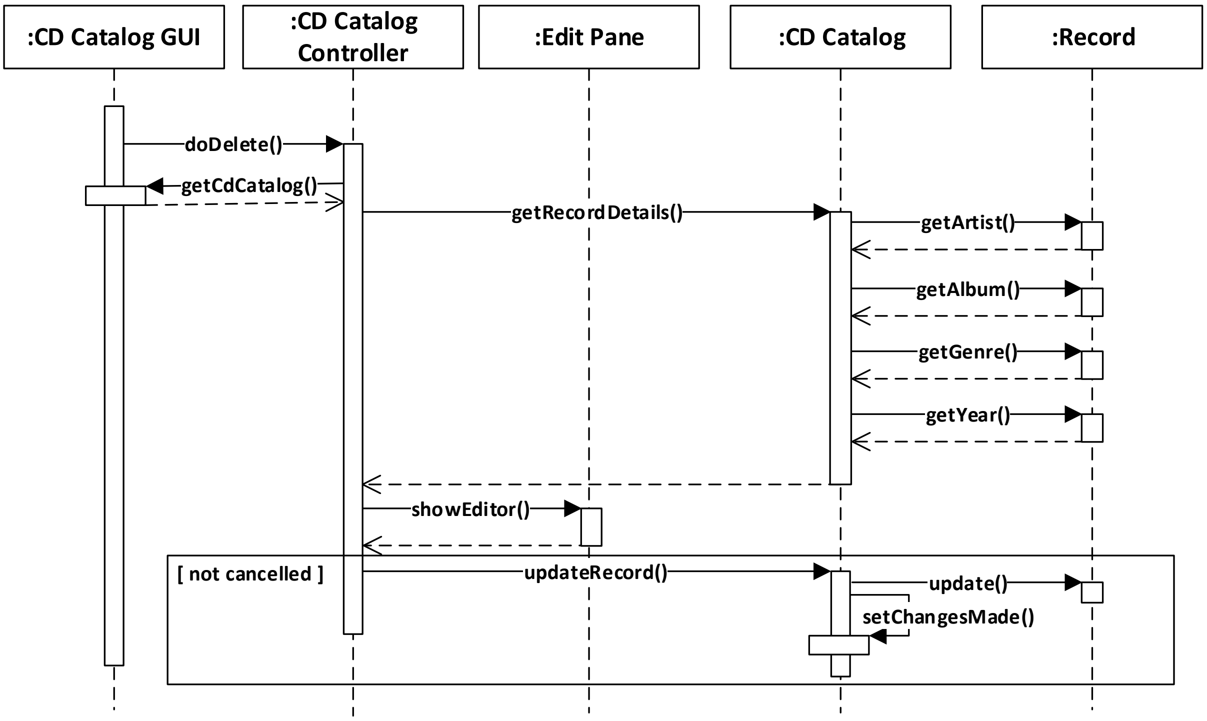
This is initiated when the user clicks the 'Add Record' button in the catalogue window. A dialog box appears, with the title 'Add', containing textboxes for the user to fill in the new cd details. The box is dismissed by clicking either 'Add' or 'Cancel'. If the 'Add' button is clicked, a new cd record is added to the end of the cd catalogue. If the 'Cancel' button is clicked, no changes are made to the cd catalogue.



SEQUENCE DIAGRAM FOR ADD A RECORD

Edit A Record Use Case:

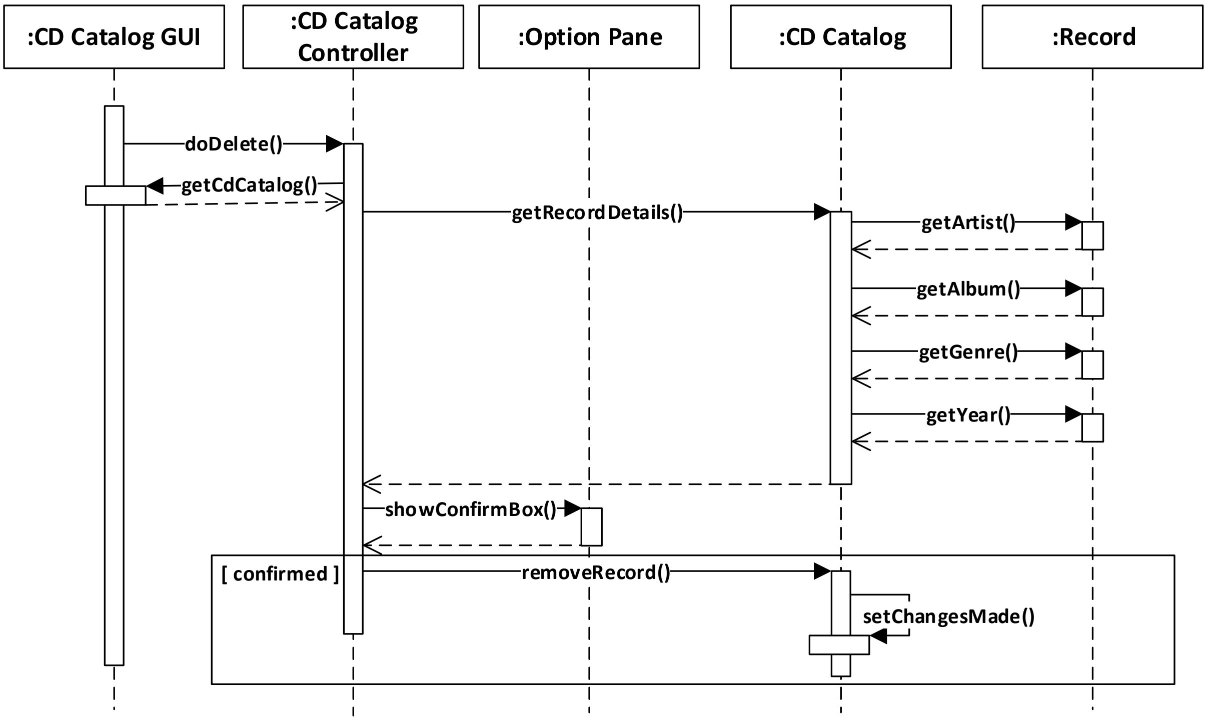
This is initiated when the user highlights a row in the DataGridView containing a cd record and then clicks the 'Edit Record' button in the catalogue window. A dialog box appears, with the title 'Edit', containing textboxes filled with the selected cd record. The user can then edit this record. The box is dismissed by clicking either 'OK' or 'Cancel'. If the 'OK' button is clicked, the selected record is updated to reflect changes made. If the 'Cancel' button is clicked, no changes are made to the cd catalogue.



SEQUENCE DIAGRAM FOR EDIT A RECORD

Delete a Record Use Case:

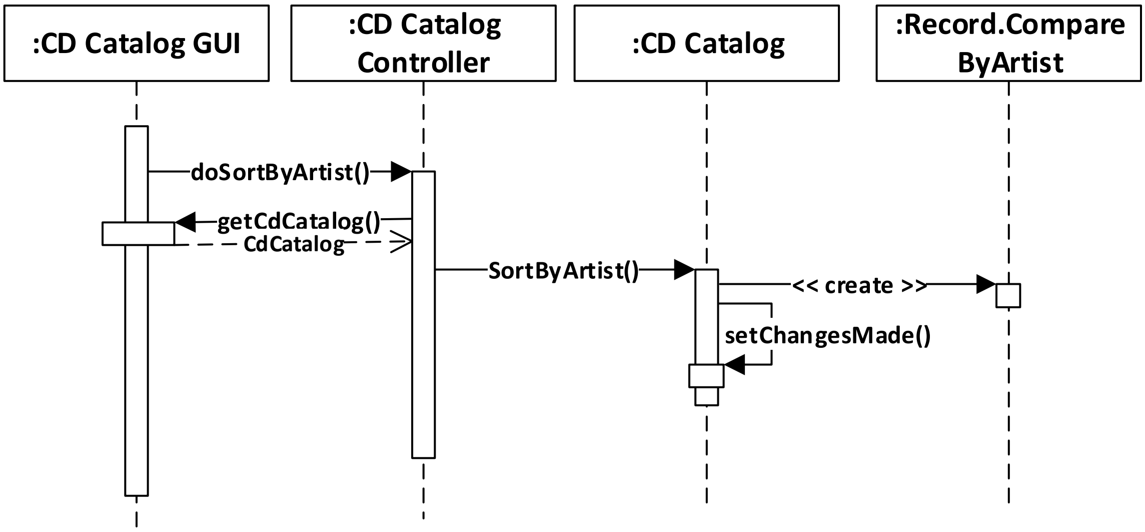
This is initiated when the user highlights a row in the DataGridView containing a cd record and then clicks the 'Delete Record' button in the main window. A dialog box appears asking user to 'Confirm Delete'. The box is dismissed by clicking either 'OK' or 'Cancel'. If the 'OK' button is clicked, the selected record is deleted from the cd catalogue. If the 'Cancel' button is clicked, no changes are made to the cd catalogue.



SEQUENCE DIAGRAM FOR DELETE A RECORD

Sort Catalogue by Artist Use Case:

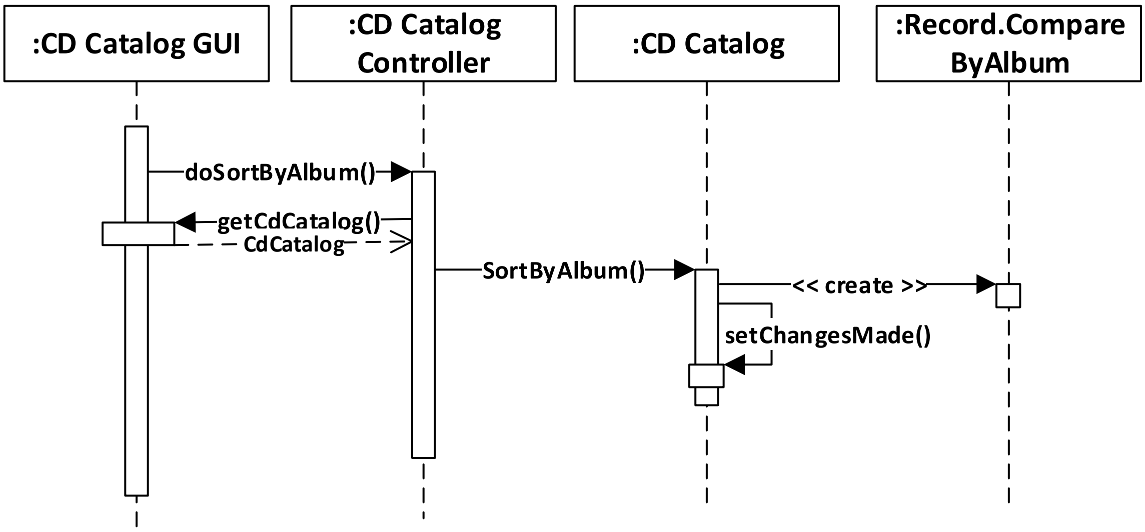
This is initiated when the user chooses 'Sort by Artist' from the Edit Menu. The records in the cd catalogue are sorted in an ascending order by artist.



SEQUENCE DIAGRAM FOR SORT CATALOGUE BY ARTIST

Sort Catalogue by Album Use Case:

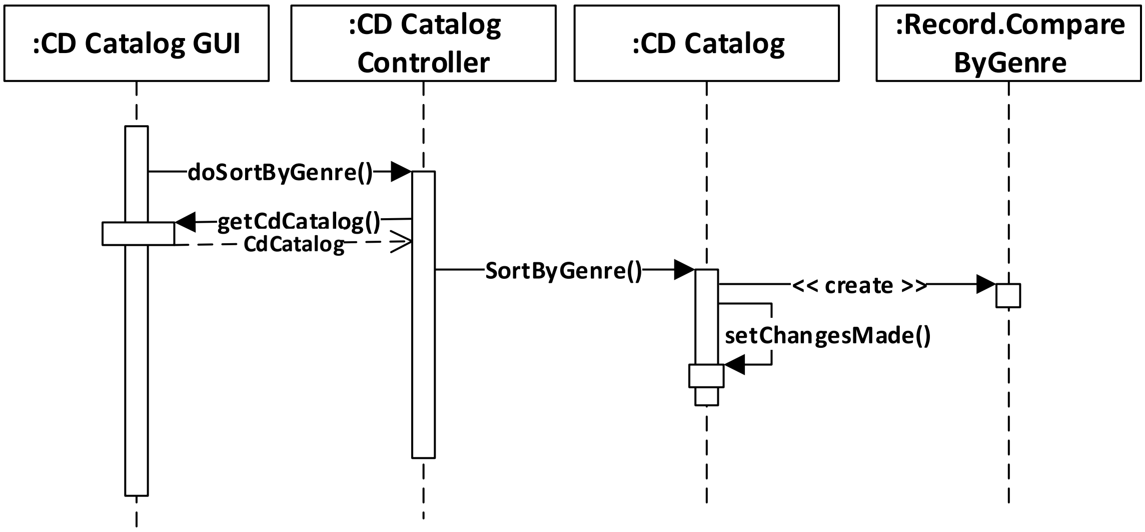
This is initiated when the user chooses 'Sort by Album' from the Edit Menu. The records in the cd catalogue are sorted in an ascending order by album.



SEQUENCE DIAGRAM FOR SORT CATALOGUE BY ALBUM

Sort Catalogue by Genre Use Case:

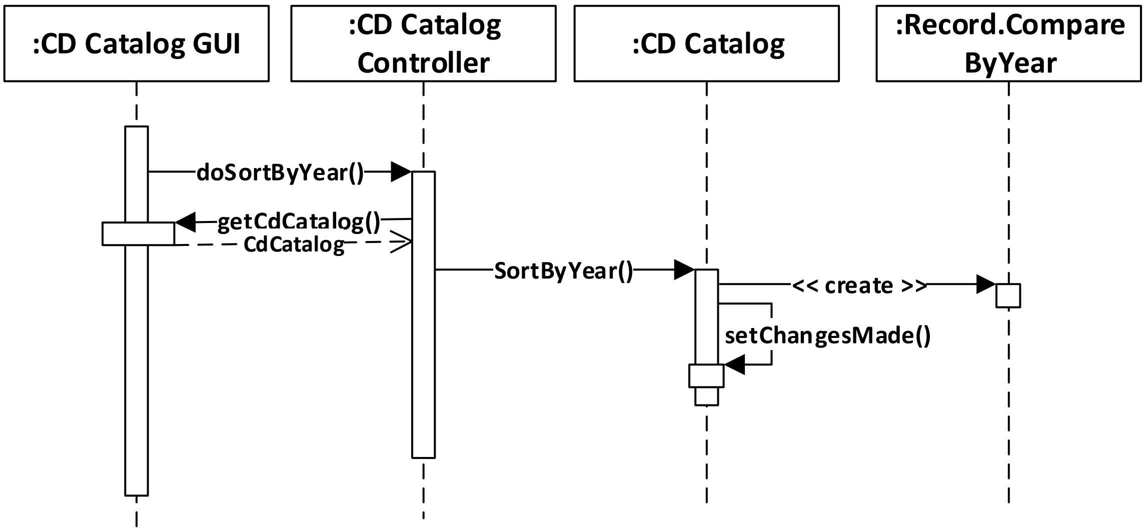
This is initiated when the user chooses 'Sort by Genre' from the Edit Menu. The records in the cd catalogue are sorted in an ascending order by genre.



SEQUENCE DIAGRAM FOR SORT CATALOGUE BY GENRE

Sort Catalogue by Year Use Case:

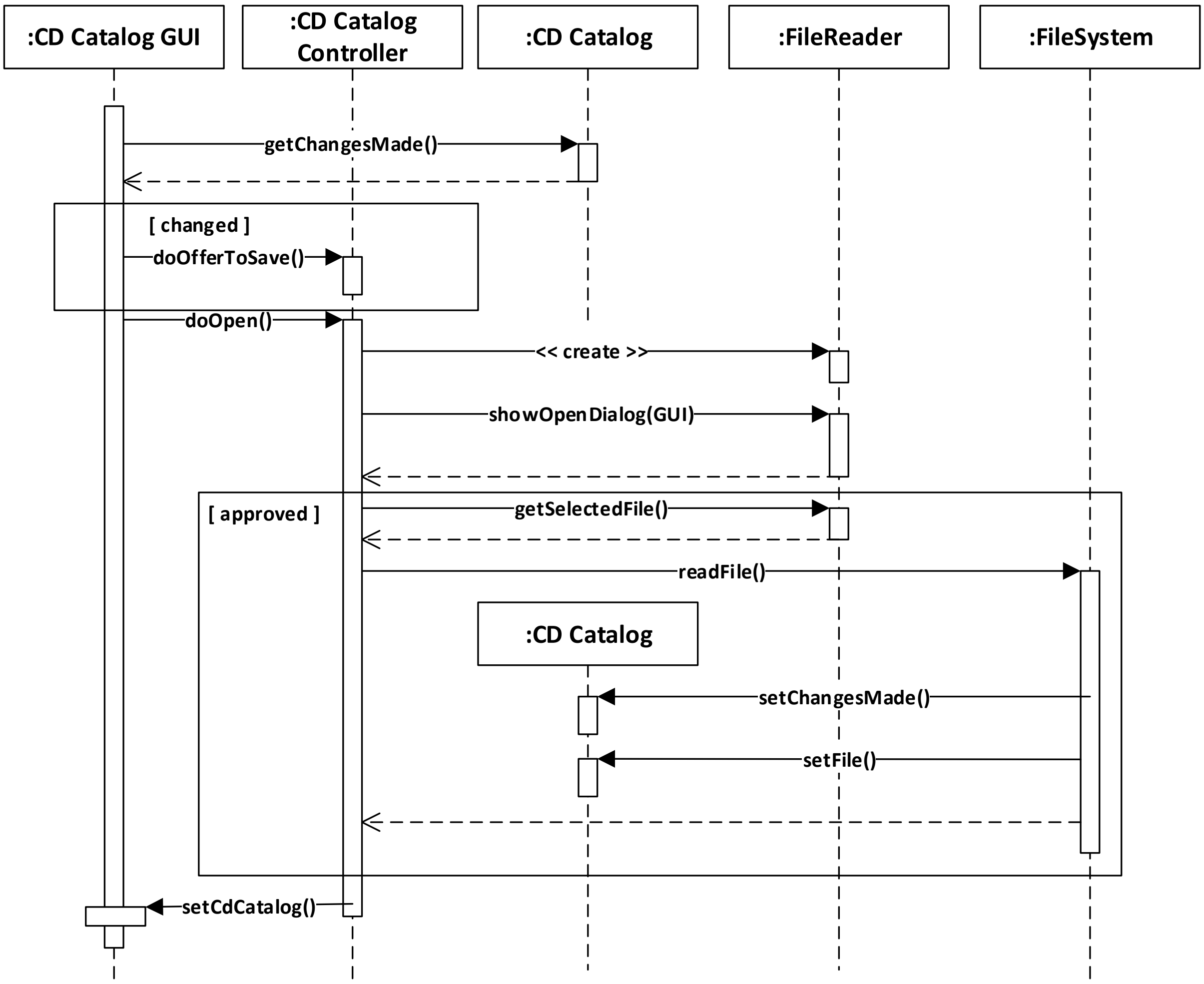
This is initiated when the user chooses 'Sort by Year' from the Edit Menu. The records in the cd catalogue are sorted in an ascending order by year.



SEQUENCE DIAGRAM FOR SORT CATALOGUE BY YEAR

Open an existing catalogue Use Case:

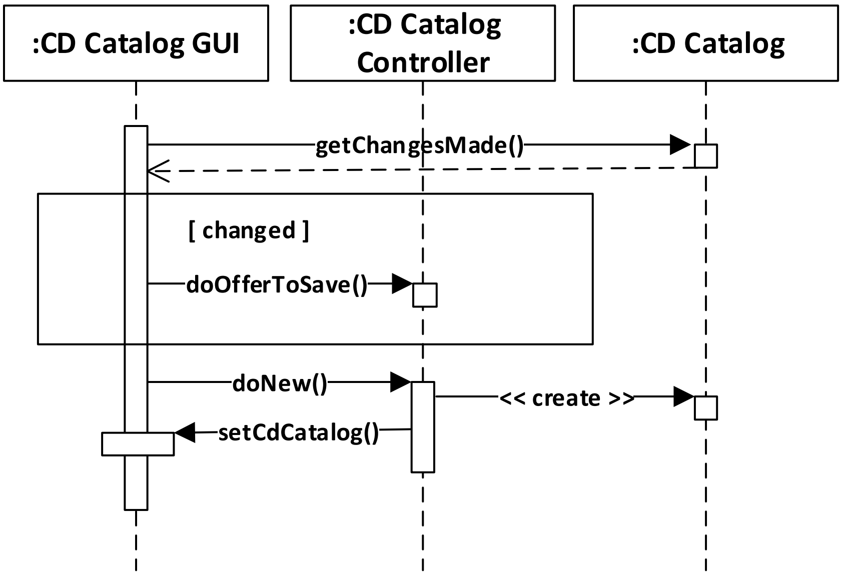
This is initiated when the user chooses 'Open' from the File Menu. If user is working on another catalogue, and changes have been made since the last 'create new…', 'open…','save', or 'save as…' operation, the 'Offer to save' extension is executed. If user cancels the 'save' process, an open file dialog is shown and the user is allowed to choose a file to open. If the user chooses a file and clicks the 'Open' button, the current cd catalogue is replaced by the opened file. The records are read into the DataGridView and displayed on the cd catalogue. If the user clicks the 'Cancel' button, no changes are made to the current catalogue.



SEQUENCE DIAGRAM FOR OPEN EXISTING CD CATALOGUE

Create a New Catalogue Use Case:

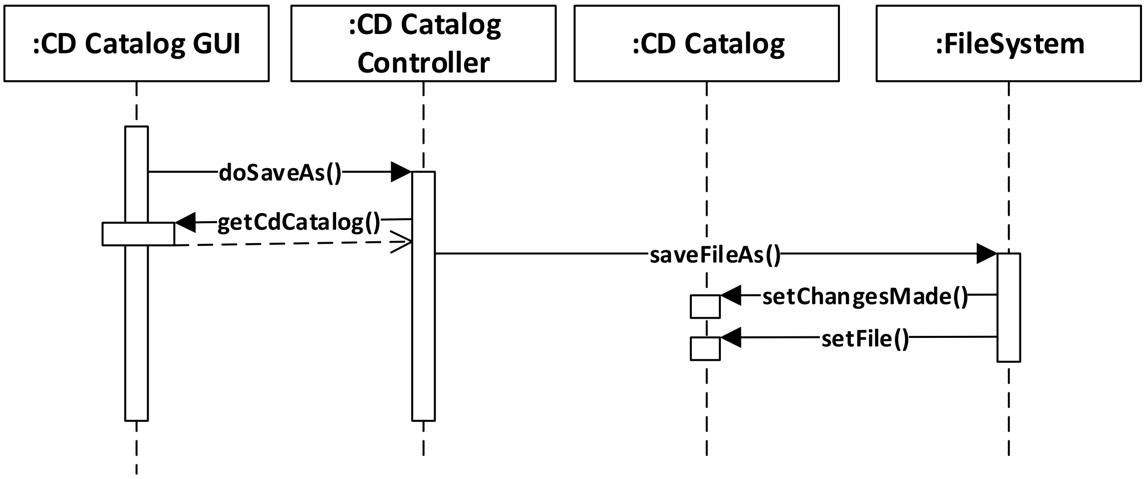
This is initiated when the user chooses 'Create new catalogue' from the File Menu. If user is working on another catalogue, and changes have been made since the last 'create new…', 'open…','save', or 'save as…' operation, the 'Offer to save' extension is executed. If user cancels the 'save' process, a new cd catalogue is created, replacing the current catalogue.



SEQUENCE DIAGRAM FOR CREATE NEW CD CATALOGUE

Save cd catalogue as… Use Case:

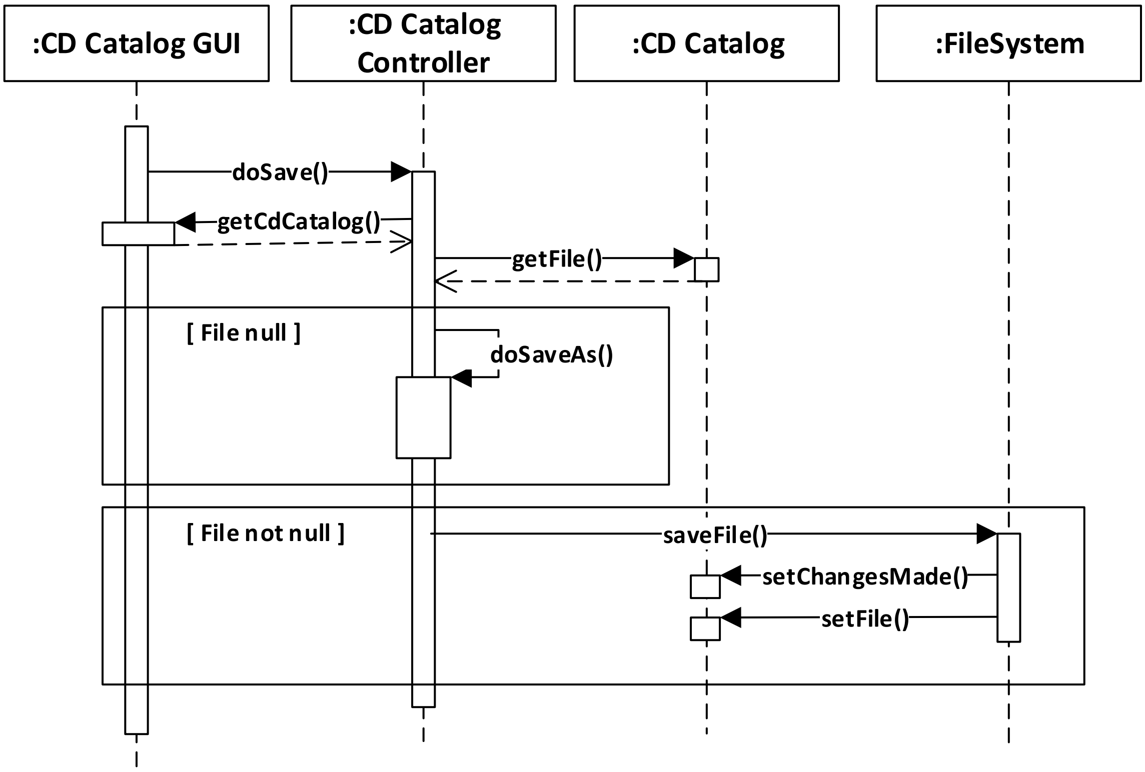
This is initiated when the user chooses 'Save as…' from the File Menu. A save file dialog box is displayed for user to choose file location and name. If the save file dialog box is cancelled, the file does not get saved.



SEQUENCE DIAGRAM FOR SAVE CD CATALOGUE AS

Save cd catalogue Use Case:

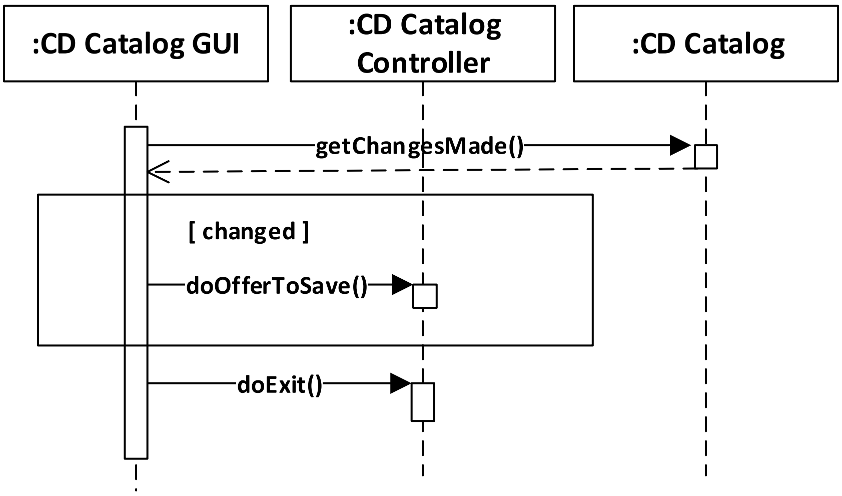
This is initiated when the user chooses 'Save' from the File Menu. If the current catalogue is an existing file, it saves the changes made to the file. Else, the 'Save as…' extension is executed.



SEQUENCE DIAGRAM FOR SAVE CD CATALOGUE

Exit Application Use Case:

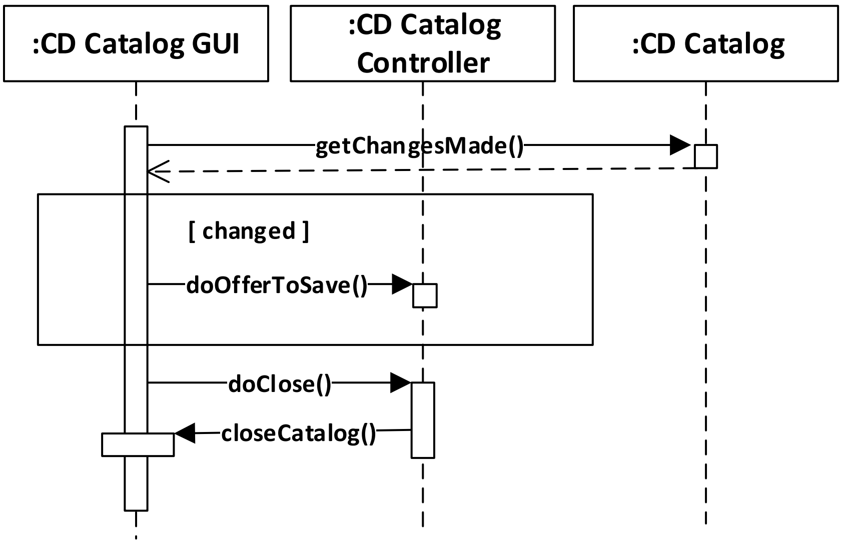
This is initiated when the user chooses 'Exit' from the File Menu, or clicks the close button for the main window. If user made changes to the catalogue since the last 'create new…', 'open…','save', or 'save as…' operation, the 'Offer to save' extension is executed before program is terminated. If user cancels the 'save' process, the programs exits anyways.



SEQUENCE DIAGRAM FOR EXIT APPLICATION

Close current Catalogue Use Case:

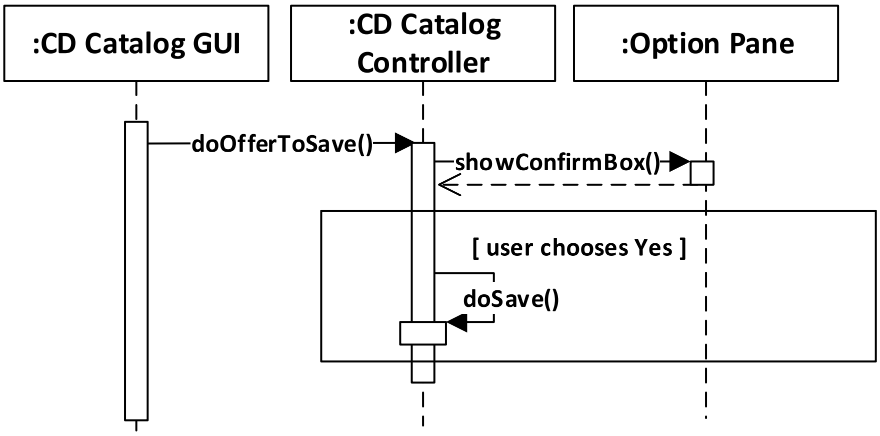
This is initiated when the user chooses 'Close' from the File Menu. If user made changes to the catalogue since the last 'create new…', 'open…','save', or 'save as…' operation, the 'Offer to save' extension is executed before window is closed. If user cancels the 'save' process, the window closes anyways, displaying the main window (start of the program).



SEQUENCE DIAGRAM FOR CLOSE CURRENT CATALOGUE

Offer to save changes Use Case:

This is initiated when the user chooses either the 'create new…', 'open…', 'close', or 'exit' from the File menu. If changes have been made to the current cd catalogue since the last 'create new…', 'open…','save', or 'save as…' operation, a dialogue box is displayed, asking user to 'Save your changes?' with an 'OK' and 'Cancel' button. If user chooses the 'Ok' button, the 'save' extension is executed (which includes the 'save as…' extension for non-existing files). If the user chooses 'Cancel', the original operation is executed anyways, whereas the current catalogue does not save changes made since last operations.



SEQUENCE DIAGRAM FOR OFFER TO SAVE CHANGES

UML CLASS DIAGRAM FOR THE CD CATALOGUE

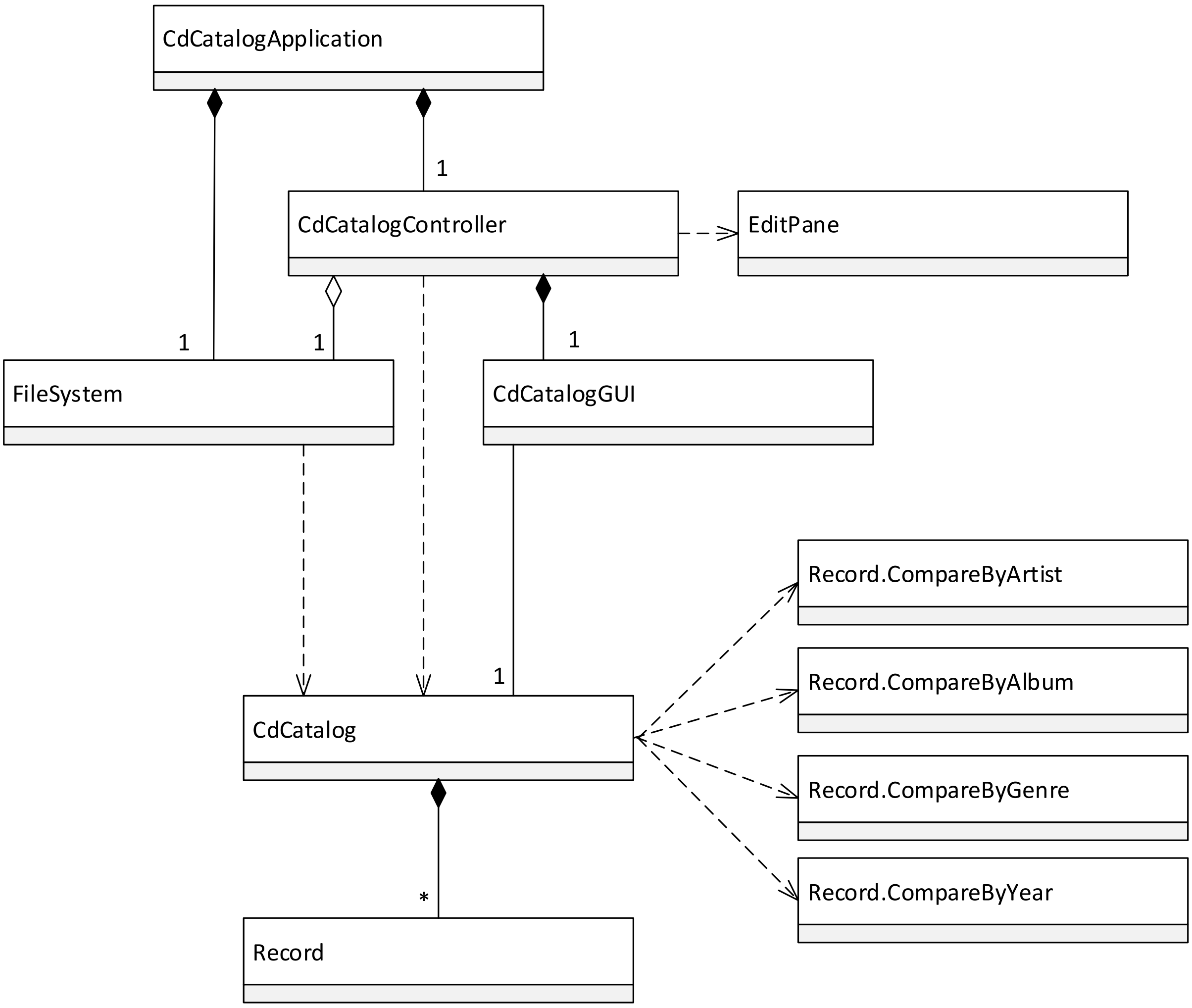
The diagram includes classes discovered during analysis, plus additional classes discovered during design. It includes:

- CdCatalogApplication: main class for the application; responsible for creating the FileSystem and GUI objects and starting up the application.
- EditPane: a utility class for reading multiple values at a single time.
- Record.CompareByArtist: Comparator for comparing two Record objects by artist (used for sorting by artist).
- Record.CompareByAlbum: Comparator for comparing two Record objects by album (used for sorting by album).
- Record.CompareByGenre: Comparator for comparing two Record objects by genre (used for sorting by genre).
- Record.CompareByYear: Comparator for comparing two Record objects by year (used for sorting by year).

The following relationships hold between the objects:

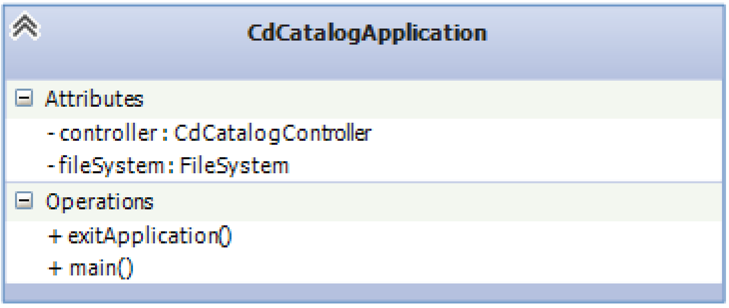
- The main application object is responsible for creating a single file system and a single controller object.
- The file system object is responsible for saving and reloading the cd catalogue.
- The controller object is responsible for creating a single GUI object.
- The controller object is responsible for initially creating the cd catalogue object, but the GUI is responsible for keeping track of its current cd catalogue – having only one at a time.
- The GUI object and the cd catalogue object are related by an observer-observable relationship, so that changes to the cd catalogue is displayed
- The cd catalogue object is responsible for creating and keeping track of person objects, of which there can be many in any given catalogue.
- An EditPane object is used by the controller to allow user to enter multiple items of data about a record.
- A comparator object of the appropriate kind is used by the catalogue object when sorting itself.

Shown below is the class diagram for the CD Cataloguing system.



UML CLASS DIAGRAM OF THE CD CATALOGUE

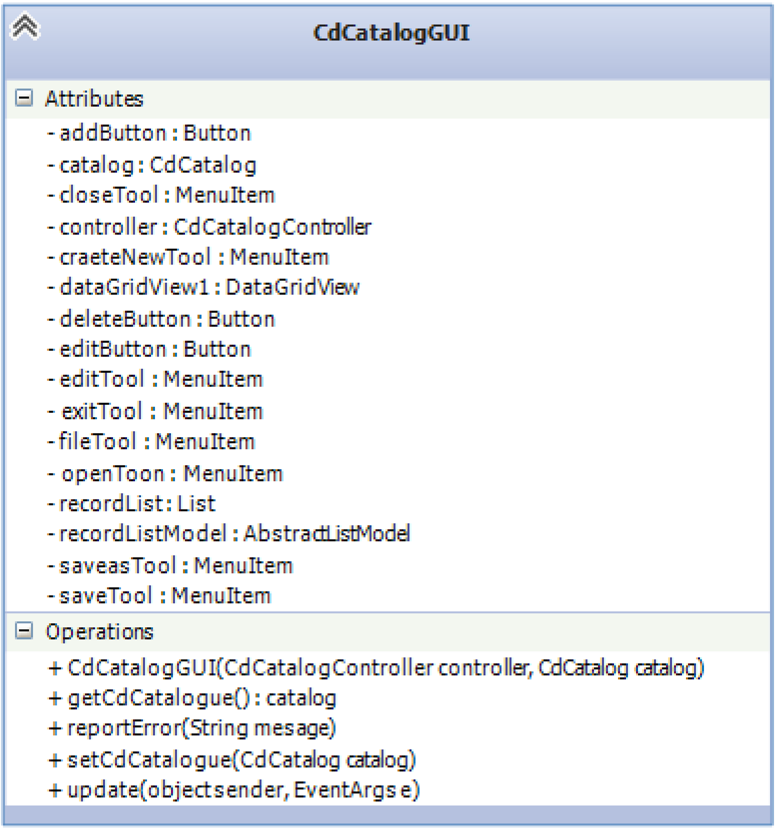
DETAILED CLASS DESIGN DIAGRAMS FOR THE CD CATALOGUE



CdCatalogApplication

Attributes
- controller : CdCatalogController
- fileSystem : FileSystem

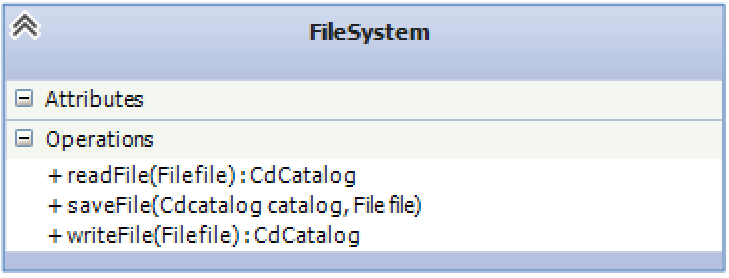
Operations
+ exitApplication()
+ main()



CdCatalogGUI

Attributes
- addButton : Button
- catalog : CdCatalog
- closeTool : MenuItem
- controller : CdCatalogController
- craeteNewTool : MenuItem
- dataGridView1 : DataGridView
- deleteButton : Button
- editButton : Button
- editTool : MenuItem
- exitTool : MenuItem
- fileTool : MenuItem
- openToon : MenuItem
- recordList : List
- recordListModel : AbstractListModel
- saveasTool : MenuItem
- saveTool : MenuItem

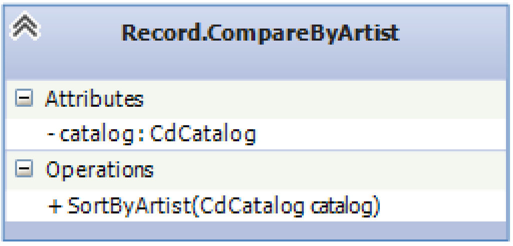
Operations
+ CdCatalogGUI(CdCatalogController controller, CdCatalog catalog)
+ getCdCatalogue() : catalog
+ reportError(String mesage)
+ setCdCatalogue(CdCatalog catalog)
+ update(object sender, EventArgs e)



FileSystem

Attributes

Operations
+ readFile(File file) : CdCatalog
+ saveFile(Cdcatalog catalog, File file)
+ writeFile(File file) : CdCatalog



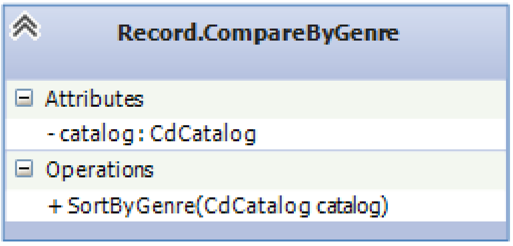
Record.CompareByArtist

Attributes

- catalog : CdCatalog

Operations

+ SortByArtist(CdCatalog catalog)



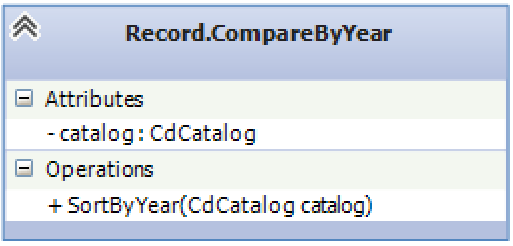
Record.CompareByGenre

Attributes

- catalog : CdCatalog

Operations

+ SortByGenre(CdCatalog catalog)



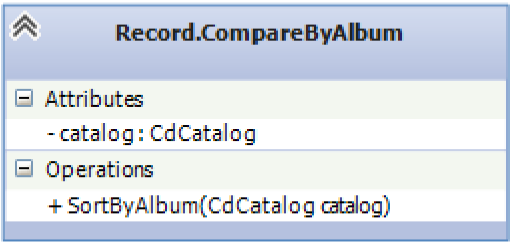
Record.CompareByYear

Attributes

- catalog : CdCatalog

Operations

+ SortByYear(CdCatalog catalog)



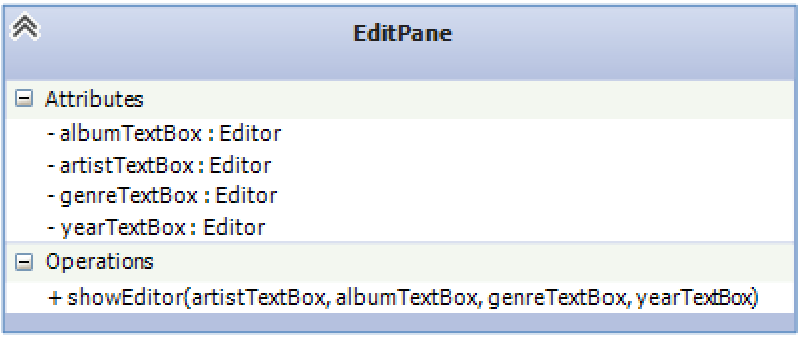
Record.CompareByAlbum

Attributes

- catalog : CdCatalog

Operations

+ SortByAlbum(CdCatalog catalog)



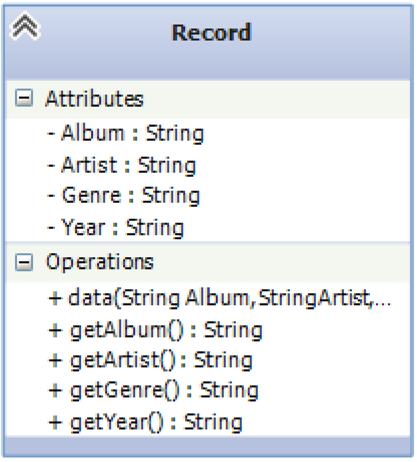
EditPane

Attributes

- albumTextBox : Editor
- artistTextBox : Editor
- genreTextBox : Editor
- yearTextBox : Editor

Operations

+ showEditor(artistTextBox, albumTextBox, genreTextBox, yearTextBox)



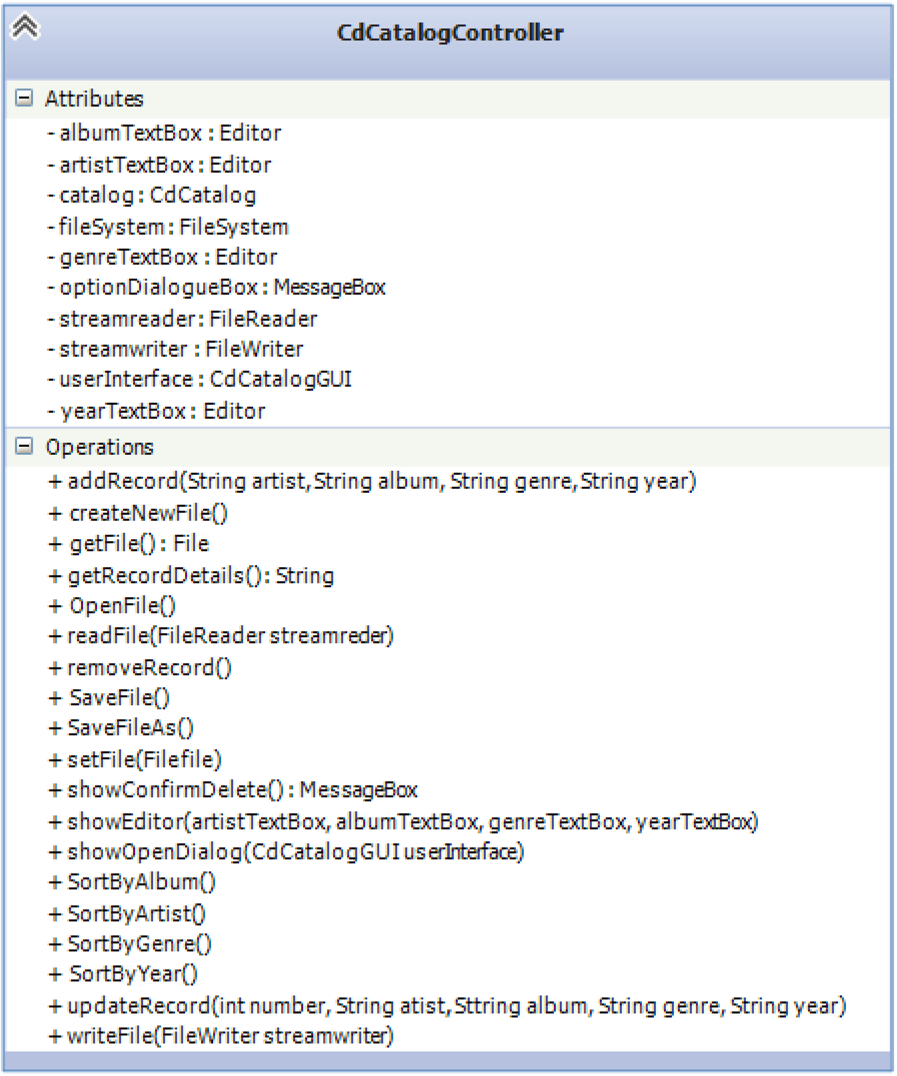
Record

Attributes

- Album : String
- Artist : String
- Genre : String
- Year : String

Operations

+ data(String Album, StringArtist,...
+ getAlbum() : String
+ getArtist() : String
+ getGenre() : String
+ getYear() : String



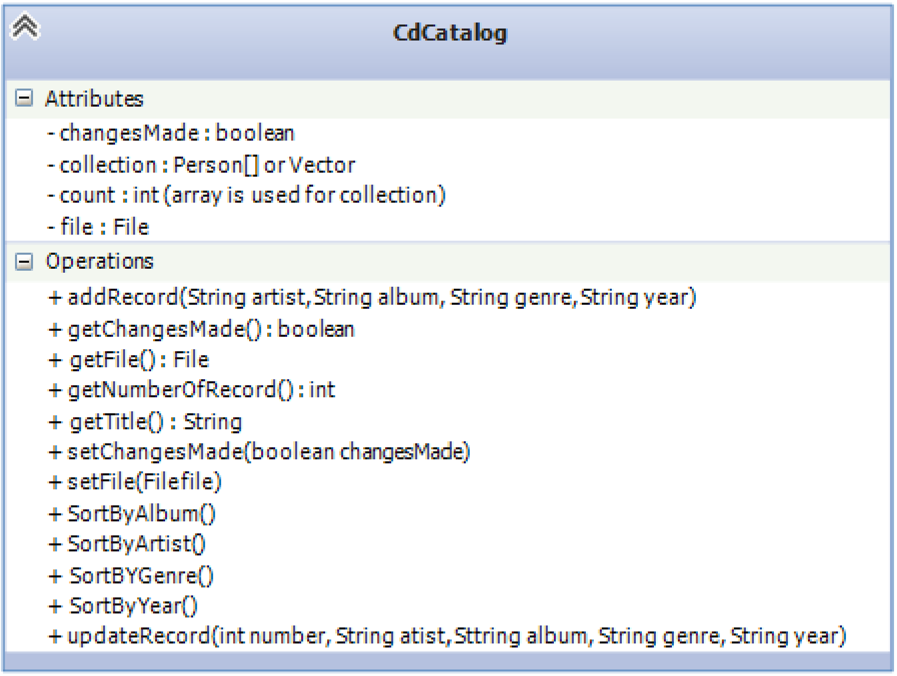
CdCatalogController

Attributes

- albumTextBox : Editor
- artistTextBox : Editor
- catalog : CdCatalog
- fileSystem : FileSystem
- genreTextBox : Editor
- optionDialogueBox : MessageBox
- streamreader : FileReader
- streamwriter : FileWriter
- userInterface : CdCatalogGUI
- yearTextBox : Editor

Operations

+ addRecord(String artist, String album, String genre, String year)
+ createNewFile()
+ getFile() : File
+ getRecordDetails(): String
+ OpenFile()
+ readFile(FileReader streamreder)
+ removeRecord()
+ SaveFile()
+ SaveFileAs()
+ setFile(Filefile)
+ showConfirmDelete() : MessageBox
+ showEditor(artistTextBox, albumTextBox, genreTextBox, yearTextBox)
+ showOpenDialog(CdCatalogGUI userInterface)
+ SortByAlbum()
+ SortByArtist()
+ SortByGenre()
+ SortByYear()
+ updateRecord(int number, String atist, Sttring album, String genre, String year)
+ writeFile(FileWriter streamwriter)



```
CdCatalog
Attributes
    - changesMade : boolean
    - collection : Person[] or Vector
    - count : int (array is used for collection)
    - file : File
Operations
    + addRecord(String artist, String album, String genre, String year)
    + getChangesMade() : boolean
    + getFile() : File
    + getNumberOfRecord() : int
    + getTitle() : String
    + setChangesMade(boolean changesMade)
    + setFile(Filefile)
    + SortByAlbum()
    + SortByArtist()
    + SortBYGenre()
    + SortByYear()
    + updateRecord(int number, String atist, Sttring album, String genre, String year)
```

TEST CASES AND RESULTS OF THE TESTING PROCEDURE.

CASE 1: Main window staying always open

Test case: Opening existing file in the main window

Result: Success

Test case: Creating new catalogue from main window

Result: Success

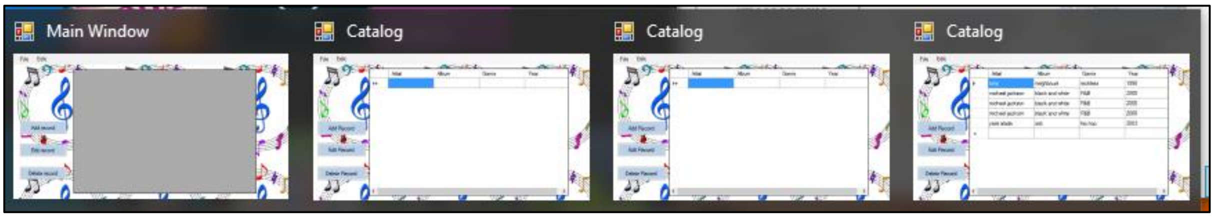
Test case: Creating another new catalogue from main window

Result: Success

Test case: Opening another existing file from main window

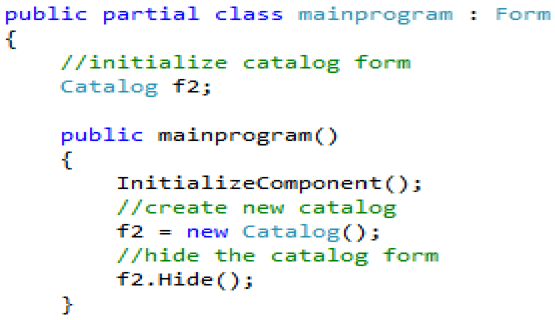
Result: Success

Problem encountered: Catalogue opens more than one catalogue window from the main window. This is shown in Picture 1.



Picture 1: multiple opened windows

Solution: Added condition to main window such that only one catalogue is created at a time as shown in Picture 2. Problem solved.



```
public partial class mainprogram : Form
{
    //initialize catalog form
    Catalog f2;

    public mainprogram()
    {
        InitializeComponent();
        //create new catalog
        f2 = new Catalog();
        //hide the catalog form
        f2.Hide();
    }
```

Picture 2: initialized components for the main window

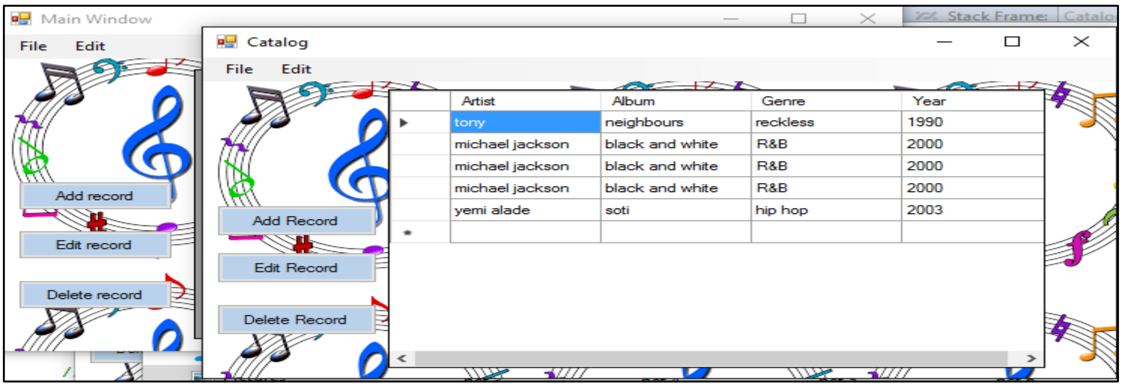
Test case: Since main window allows only one catalogue to be created at a time, creating a new catalogue from main window, after opening an existing file

Result: New catalogue is not created, existing catalogue remains unchanged.

Test case: Whereas creating a new catalogue from the already opened catalogue

Result: Success

Problem: The inability to create a new catalogue from main window as shown in Picture 3.



	Artist	Album	Genre	Year
▶	tony	neighbours	reckless	1990
	michael jackson	black and white	R&B	2000
	michael jackson	black and white	R&B	2000
	michael jackson	black and white	R&B	2000
	yemi alade	soti	hip hop	2003
*				

Picture 3: inability to create a new catalogue from main window

Test case: After a catalogue has been created or opened from main window and is closed, creating a new catalogue from main window

Result: Success

Test case: After a catalogue has been created or opened from main window and another catalogue is opened from the catalogue window, this closes the initial catalogue and opens a new one. Creating a new catalogue from main window, after this process…

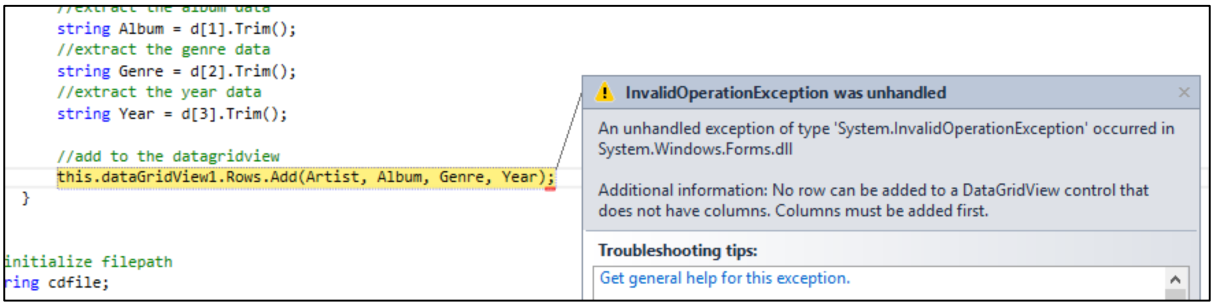
Result: Another catalogue is created, without closing the already opened catalogue

Problem: Multiple catalogue windows just as shown in Picture 1.

Test case: After a catalogue has been created or opened from main window and is closed, opening another existing file from main window

Result: Error is shown

Problem: After the initial window is closed, opening an existing catalogue file results in an error as shown in Picture 4.



```
      //extract the album data
      string Album = d[1].Trim();
      //extract the genre data
      string Genre = d[2].Trim();
      //extract the year data
      string Year = d[3].Trim();

      //add to the datagridview
      this.dataGridView1.Rows.Add(Artist, Album, Genre, Year);
  }

initialize filepath
ring cdfile;
```

InvalidOperationException was unhandled

An unhandled exception of type 'System.InvalidOperationException' occurred in System.Windows.Forms.dll

Additional information: No row can be added to a DataGridView control that does not have columns. Columns must be added first.

Troubleshooting tips:

Get general help for this exception.

Picture 4: error message shown.

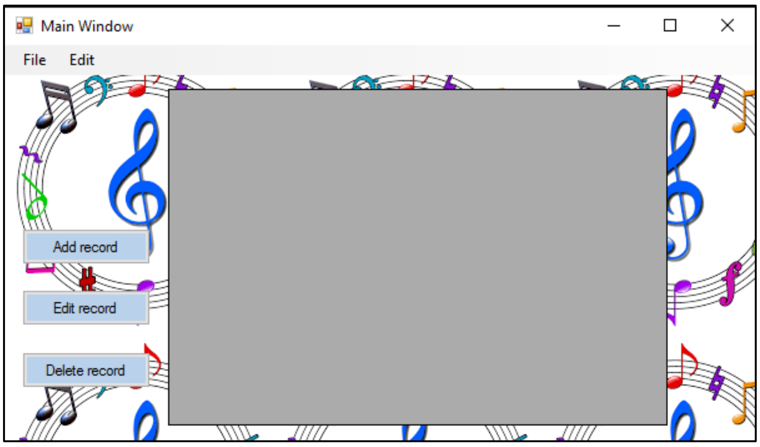
Overall Case Solution: Hide main window, after opening an existing file or creating a new catalogue. To be revealed only when current catalogue is closed. Hence, user is shown only one catalogue window at a time.

CASE 2: Working with a single window at a time

Test case: Opening the main window

Result: Success

Image: shown in Picture 5



Picture 5: opening the main window

Test case: Opening an existing catalogue file from main window

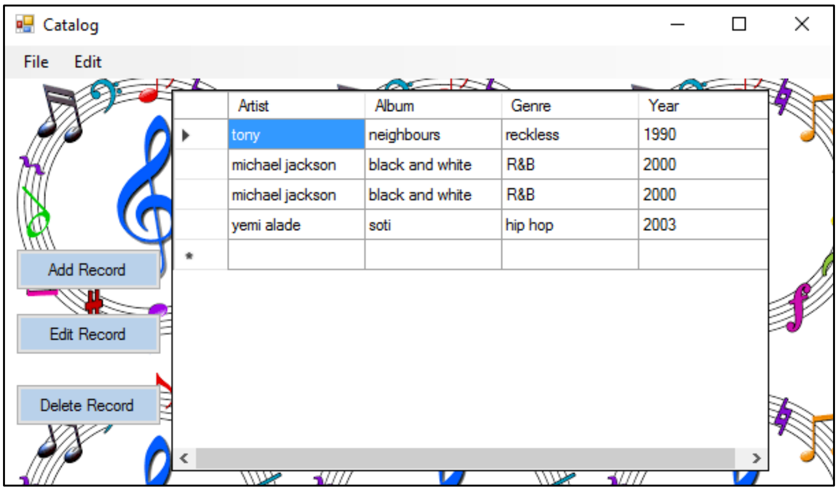
Result: Success

Image: shown in Picture 6

Test case: Opening an existing catalogue file from catalogue window

Result: Success

Image: shown in Picture 6



	Artist	Album	Genre	Year
▶	tony	neighbours	reckless	1990
	michael jackson	black and white	R&B	2000
	michael jackson	black and white	R&B	2000
	yemi alade	soti	hip hop	2003
*				

Picture 6: opening existing catalogue

Test case: Creating a new catalogue from the main window

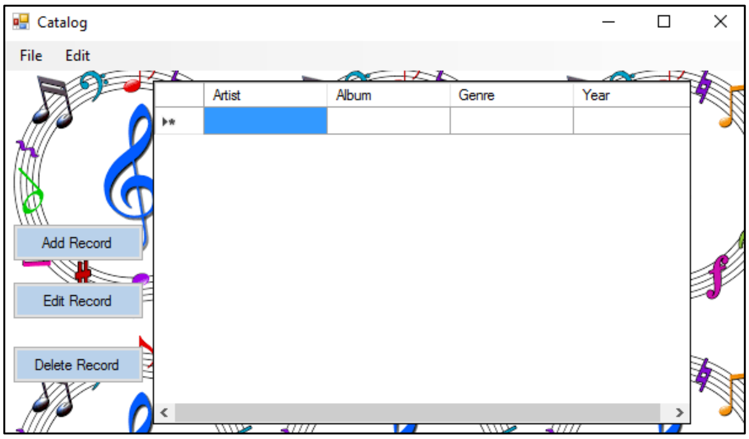
Result: Success

Image: shown in Picture 7

Test case: Creating a new catalogue from the catalogue window

Result: Success

Image: shown in Picture 7



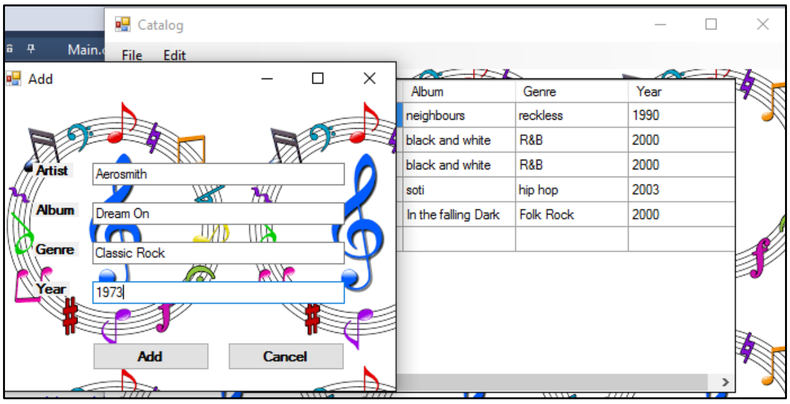
	Artist	Album	Genre	Year
▸*				

Picture 7: creating new catalogue

Test case: Adding record to the catalogue

Result: Success

Image: shown in Picture 8



Artist: Aerosmith

Album: Dream On

Genre: Classic Rock

Year: 1973

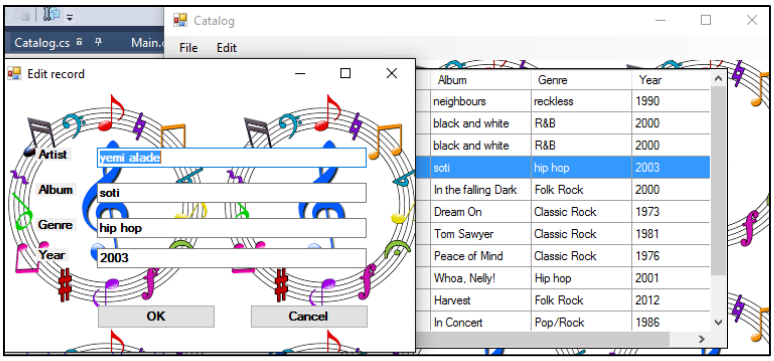
Album	Genre	Year
neighbours	reckless	1990
black and white	R&B	2000
black and white	R&B	2000
soti	hip hop	2003
In the falling Dark	Folk Rock	2000

Picture 8: adding a record

Test case: Editing selected record from the catalogue

Result: Success

Image: shown in Picture 9



Artist: yemi alade

Album: soti

Genre: hip hop

Year: 2003

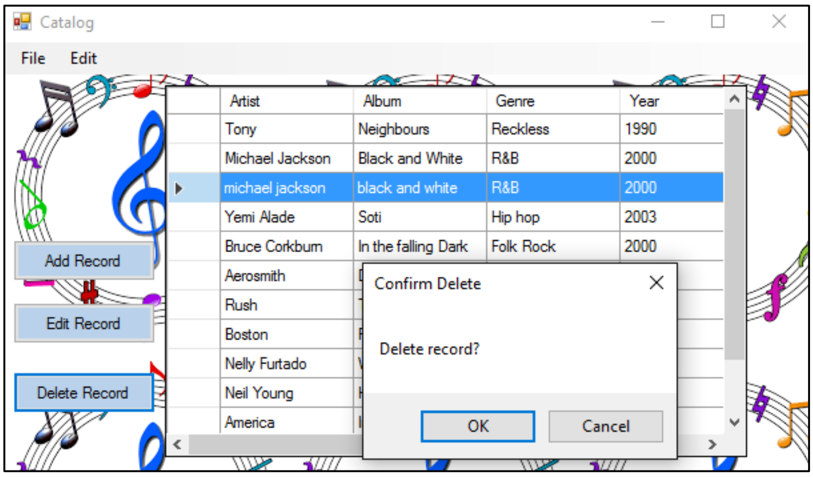
Album	Genre	Year
neighbours	reckless	1990
black and white	R&B	2000
black and white	R&B	2000
soti	hip hop	2003
In the falling Dark	Folk Rock	2000
Dream On	Classic Rock	1973
Tom Sawyer	Classic Rock	1981
Peace of Mind	Classic Rock	1976
Whoa, Nelly!	Hip hop	2001
Harvest	Folk Rock	2012
In Concert	Pop/Rock	1986

Picture 9: editing a selected record

Test case: Deleting record from the catalogue

Result: Success

Image: shown in Picture 10



Artist	Album	Genre	Year
Tony	Neighbours	Reckless	1990
Michael Jackson	Black and White	R&B	2000
michael jackson	black and white	R&B	2000
Yemi Alade	Soti	Hip hop	2003
Bruce Corkbum	In the falling Dark	Folk Rock	2000
Aerosmith			
Rush			
Boston			
Nelly Furtado			
Neil Young			
America			

Confirm Delete

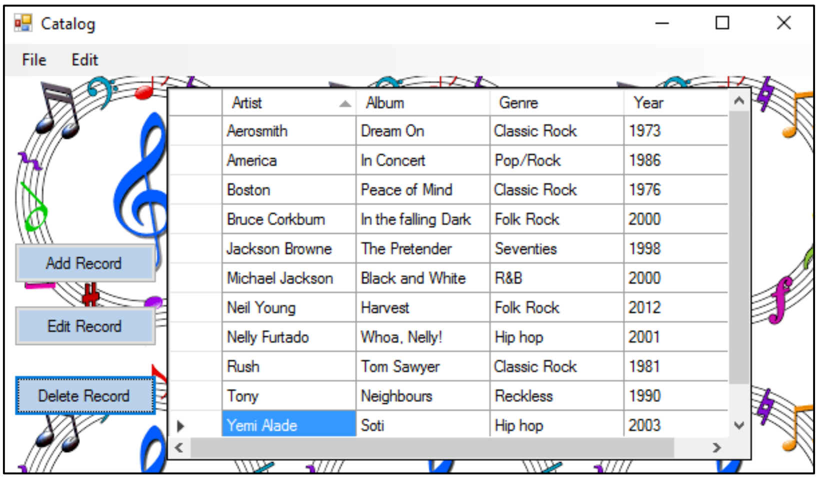
Delete record?

Picture 10: deleting a record

<u>Test case</u>: Sorting record by artist

Result: Success

Image: shown in Picture 11



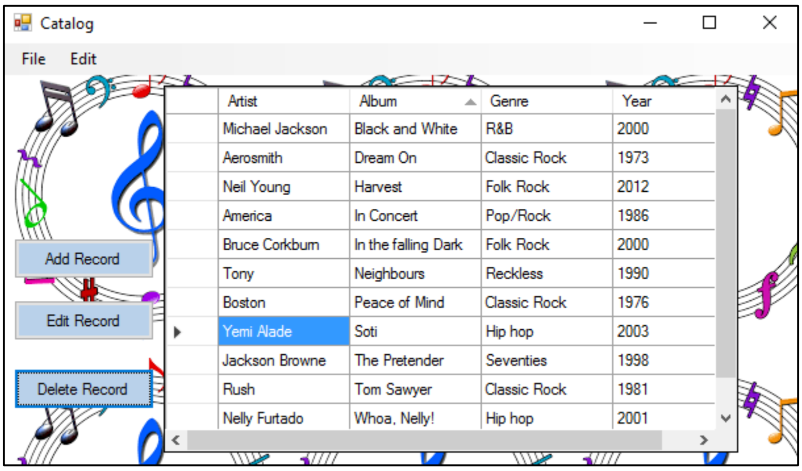
	Artist	Album	Genre	Year
	Aerosmith	Dream On	Classic Rock	1973
	America	In Concert	Pop/Rock	1986
	Boston	Peace of Mind	Classic Rock	1976
	Bruce Corkbum	In the falling Dark	Folk Rock	2000
	Jackson Browne	The Pretender	Seventies	1998
	Michael Jackson	Black and White	R&B	2000
	Neil Young	Harvest	Folk Rock	2012
	Nelly Furtado	Whoa, Nelly!	Hip hop	2001
	Rush	Tom Sawyer	Classic Rock	1981
	Tony	Neighbours	Reckless	1990
▶	Yemi Alade	Soti	Hip hop	2003

Picture 11: sorting by artist

<u>Test case</u>: Sorting record by album

Result: Success

Image: shown in Picture 12



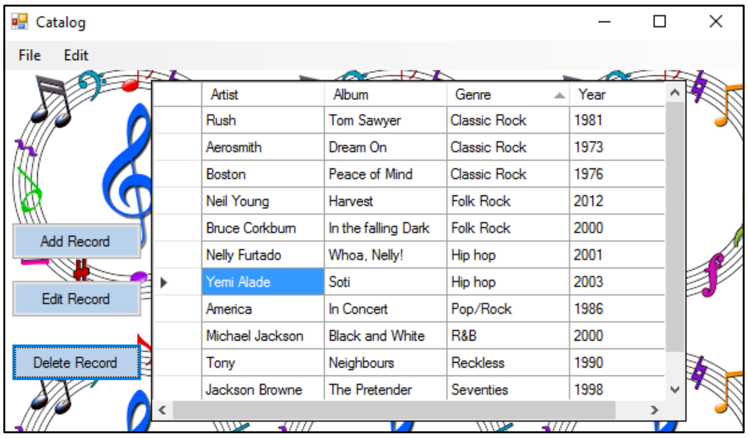
	Artist	Album	Genre	Year
	Michael Jackson	Black and White	R&B	2000
	Aerosmith	Dream On	Classic Rock	1973
	Neil Young	Harvest	Folk Rock	2012
	America	In Concert	Pop/Rock	1986
	Bruce Corkbum	In the falling Dark	Folk Rock	2000
	Tony	Neighbours	Reckless	1990
	Boston	Peace of Mind	Classic Rock	1976
▶	Yemi Alade	Soti	Hip hop	2003
	Jackson Browne	The Pretender	Seventies	1998
	Rush	Tom Sawyer	Classic Rock	1981
	Nelly Furtado	Whoa, Nelly!	Hip hop	2001

Picture 12: sorting by album

<u>Test case</u>: Sorting record by genre

Result: Success

Image: shown in Picture 13



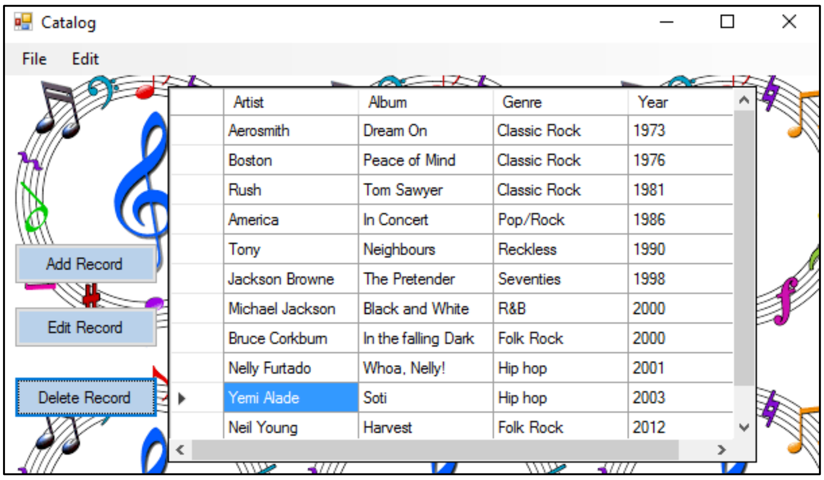
Artist	Album	Genre	Year
Rush	Tom Sawyer	Classic Rock	1981
Aerosmith	Dream On	Classic Rock	1973
Boston	Peace of Mind	Classic Rock	1976
Neil Young	Harvest	Folk Rock	2012
Bruce Corkbum	In the falling Dark	Folk Rock	2000
Nelly Furtado	Whoa, Nelly!	Hip hop	2001
Yemi Alade	Soti	Hip hop	2003
America	In Concert	Pop/Rock	1986
Michael Jackson	Black and White	R&B	2000
Tony	Neighbours	Reckless	1990
Jackson Browne	The Pretender	Seventies	1998

Picture 13: sorting by genre

<u>Test case</u>: Sorting record by year

Result: Success

Image: shown in Picture 14



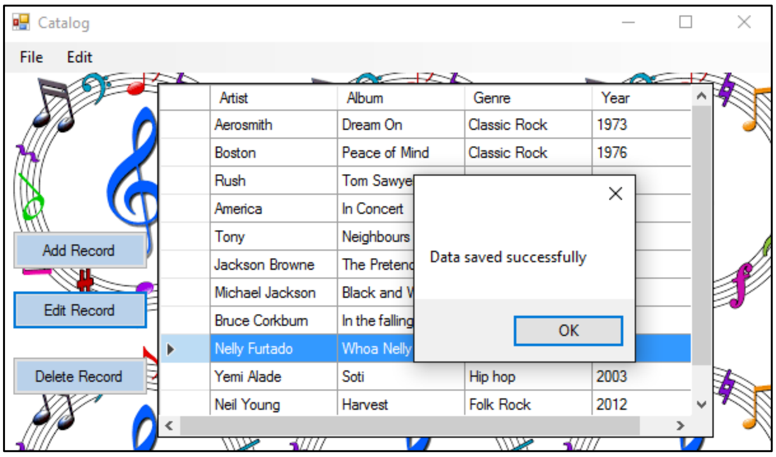
Artist	Album	Genre	Year
Aerosmith	Dream On	Classic Rock	1973
Boston	Peace of Mind	Classic Rock	1976
Rush	Tom Sawyer	Classic Rock	1981
America	In Concert	Pop/Rock	1986
Tony	Neighbours	Reckless	1990
Jackson Browne	The Pretender	Seventies	1998
Michael Jackson	Black and White	R&B	2000
Bruce Corkbum	In the falling Dark	Folk Rock	2000
Nelly Furtado	Whoa, Nelly!	Hip hop	2001
Yemi Alade	Soti	Hip hop	2003
Neil Young	Harvest	Folk Rock	2012

Picture 14: sorting by year

Test case: Saving an existing catalogue file to the filesystem

Result: Success

Image: shown in Picture 15



Artist	Album	Genre	Year
Aerosmith	Dream On	Classic Rock	1973
Boston	Peace of Mind	Classic Rock	1976
Rush	Tom Sawye		
America	In Concert		
Tony	Neighbours		
Jackson Browne	The Pretenc		
Michael Jackson	Black and V		
Bruce Corkbum	In the falling		
Nelly Furtado	Whoa Nelly		
Yemi Alade	Soti	Hip hop	2003
Neil Young	Harvest	Folk Rock	2012

Data saved successfully

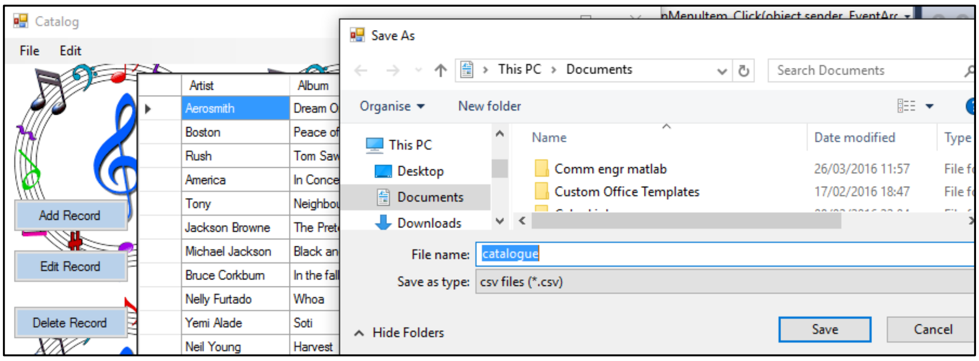
OK

Picture 15: saving an existing file

Test case: Saving catalogue as a new file to the filesystem

Result: Success

Image: shown in Picture 16



Artist	Album
Aerosmith	Dream O
Boston	Peace of
Rush	Tom Saw
America	In Conce
Tony	Neighbou
Jackson Browne	The Pret
Michael Jackson	Black an
Bruce Corkbum	In the fall
Nelly Furtado	Whoa
Yemi Alade	Soti
Neil Young	Harvest

Name	Date modified	Type
Comm engr matlab	26/03/2016 11:57	File f
Custom Office Templates	17/02/2016 18:47	File f

File name: catalogue

Save as type: csv files (*.csv)

Picture 16: saving as new file

Test case: Closing the catalogue window

Result: Success

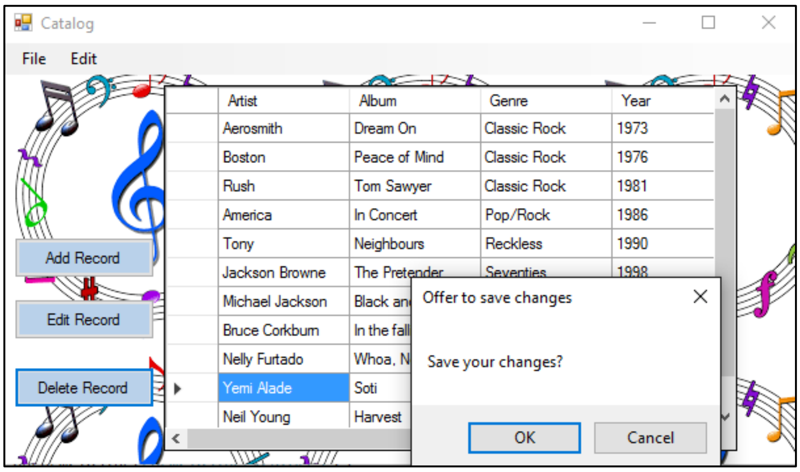
<u>Test case</u>: Exiting the program

Result: Success

<u>Test case</u>: Prompting user with the offer to save option

Result: Success

Image: shown in Picture 17



Artist	Album	Genre	Year
Aerosmith	Dream On	Classic Rock	1973
Boston	Peace of Mind	Classic Rock	1976
Rush	Tom Sawyer	Classic Rock	1981
America	In Concert	Pop/Rock	1986
Tony	Neighbours	Reckless	1990
Jackson Browne	The Pretender	Seventies	1998
Michael Jackson	Black an		
Bruce Corkbum	In the fall		
Nelly Furtado	Whoa, N		
Yemi Alade	Soti		
Neil Young	Harvest		

Offer to save changes

Save your changes?

Picture 17: offer to save option

CASE 3: Performance Testing

- Displaying multiple catalogue windows? => No
- Textboxes accepts alphanumeric characters? => Yes
- Close button exiting application? => Yes
- Keeping track of changes made (bool property)? => Yes
- Saving catalogue in proper locations? => Yes
- Easy to use? => Yes
- Accepts entries from the catalogue to edit pane? => Yes
- Working with one catalogue window at a time? => Yes
- Closing current catalogue when user opens another one? => Yes
- Confirmation delete prompt? => Yes
- Prompt user if saved? => Yes