HND Computing:

Software Development

Graded unit 2

H48W35

Portfolio Of Evidence

Brodie Harkins

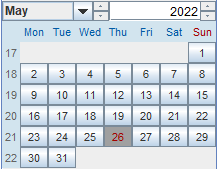
26/05/2022

Table Of Contents

1. [Importing JAR Calendar file](#JAR)
2. [Design Ideas](#DesignIdeas)
3. [Research Into Development](#Research)
4. [Lessons Learnt During Development](#LessonsLearnt)

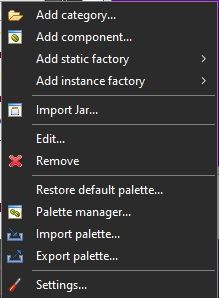
Importing a JAR Calendar library

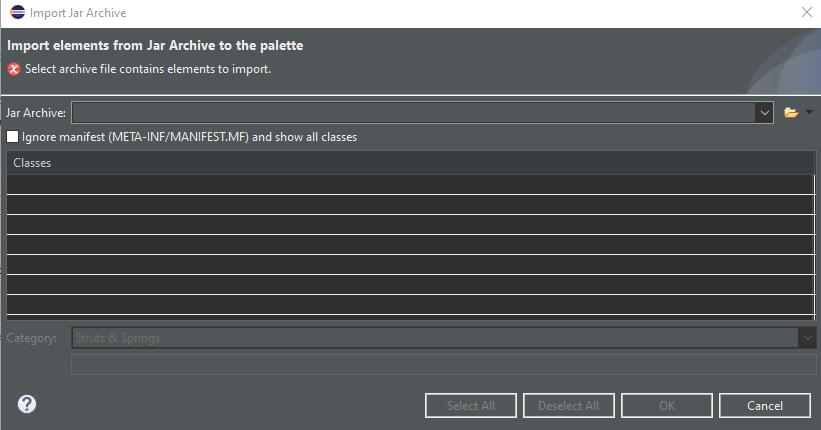
In order to search for a calendar to import into the application that will look like this:



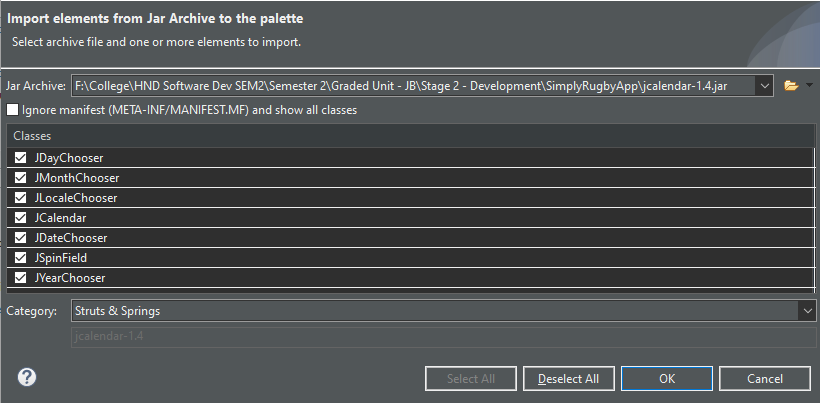
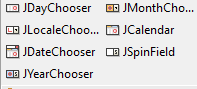
You can find the link to this file here: <http://www.java2s.com/Code/Jar/j/Downloadjcalendar14jar.htm>

Once you have downloaded this file. Extract the JAR file to the location of your project.

Open up your eclipse project and select your design view. Once you have opened your design view, right click on the component Palette, and select ***Import Jar***.

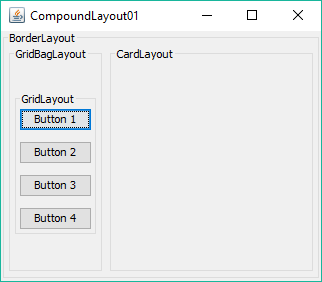
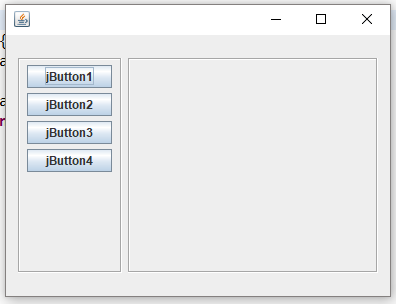
Once you have selected to import your JAR file. You will be greeted with this screen

When on this screen, select the folder next to the drop-down box. Select ***FileSystem***. Browse your files and look for where you have saved your JAR file.

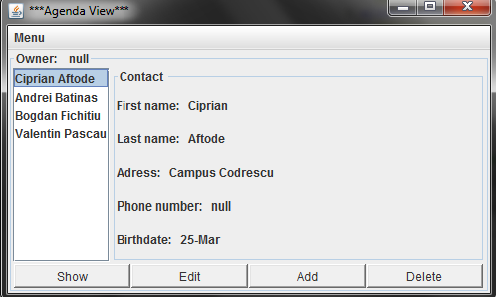
As you can see above, the JAR Calendar file has been imported. You can select all elements of the Calendar components that you want to import for your needs. In this example we will import all elements of the JAR File.

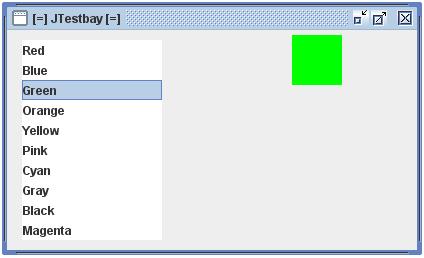
Once you have imported your JAR file. You’ll find in your palette, the date components that can now be placed into your application.

Design Ideas

During the course of planning & developing the project. I researched into some design ideas on how this application might look. Below, I have put together a collection of images to illustrate the ideas and thought process behind my final design decision.







From these ideas I liked the idea of having a side bar on the left side of the screen which would manipulate the middle of the page based on what the user was wanting to do within the program utilizing, lists, buttons & panels.

Research into Development of App

Udemy Tutorial:

Within this tutorial of Java. It will go over the basics of constructing an Object Orientated Application.

John Percell (N/A) Java Tutorial for Complete Beginners

URL: <https://www.udemy.com/course/java-tutorial/>

[Last Accessed: 23rd March 2022]

iLearn:

This website has various examples and tutorials covering implementation of Java concepts and utilizing swing to create GUI Applications.

iLearn (2022) Advanced Object Orientated Design

URL: [**Advanced OOP 2021/2022 (JB)**](https://ilearn.fife.ac.uk/course/view.php?id=10194)

[Last Accessed: 26th May 2022]

JavaPoint:

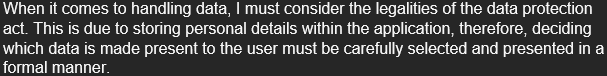
This website contains all the implementation a documentation of each component within the Java swing. This has helped explain what each component does within the framework of Swing.

JavaPoint(2022) Java Swing Tutorial

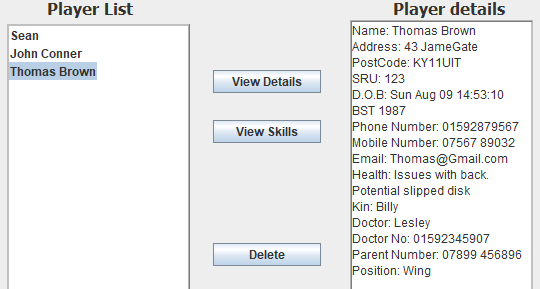
URL: <https://www.javatpoint.com/java-swing>

[Last Accessed: 12th April 2022]

Professional/Ethical Issues During Development

During the development of the solution plan. I had made the design decision to allow admin users to add player and juniors to the club and coach users can only view those player details. This does cross into the data protection act as you are dealing with personal information of players. As mentioned in the solution plan, I also covered the legalities of storing users personal data within the system, As stated below.

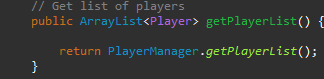
I ultimately decided when the implementation of the SimplyRugby app. I would use a text box which would only display the information and it cannot be edited as a coach. The function of editing a players personal information is only dedicated to the admin user of the club.

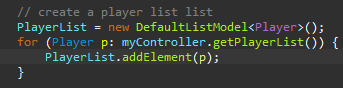
A snippet of how I done this is shown below.

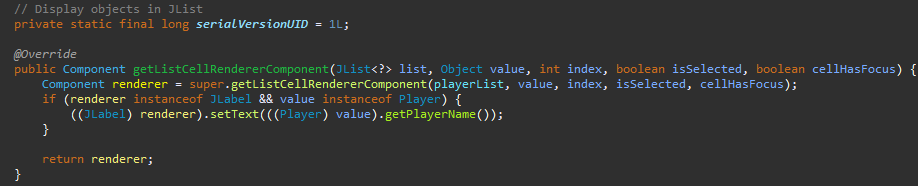
Lessons Learnt During/Before project

Before the development of our SimplyRugby application I learnt how to implement a JList that contains objects in Java Swing. Originally, I had implemented data in the JLists as Strings. The problem arose when I was attempting to view and edit a player skill. I fixed this issue by changing how the JList gets the player data. This is done by creating a JPanel and adding a JList into the JPanel.

This will allow you to design your list, to make it more appealing to the user. Upon creation of the JList I created a getter within the controller to obtain player objects from the ArrayList of Players.



Once it has gathered the objects, it will pass them into a DefaultListModel or known as a DLM. This will then pass it into a cell renderer which will create the object within the JList. DLM is show below.

The Cell Renderer is shown below.

This will pass in the list and obtain the name of the object that is in the list of Player objects.

Thus, the final output will look like this:

