Annotated Bibliography

Carri Martin

Regis University

**Current product research**

Rapla . (n.d.). Rapla - Introduction. Retrieved from <https://http://rapla.org/>

Rapla is a resource and event planning system that features a graphical calendar interface. It is a fully customizable and configurable calendar that includes common features and functions.

I found this during my product search to see what kinds of open source calendar options were currently available. This product seems to be a nice option that will work with or without database support and has a nice graphical interface.

Berger, M. (n.d.). BORG Calendar. Retrieved from <https://mikeberger.github.io/borg_calendar/>

The BORG Calendar is a combination calendar and task tracking system with functionality that is similar to Microsoft Outlook. It was designed to be a small self-contained Java application that can run locally from a flash drive.

This appears to be a good product with nice graphical interfaces for its calendar functions as well as it's task tracking functions that utilizes Java Swing.

DHTMLX. (n.d.). How to Start. Retrieved from <https://docs.dhtmlx.com/scheduler/how_to_start.html>

This article is one of the support documents posted on the DHTMLX website. It discusses how to get started using their Javascript calendar component. This component provides a very nice graphical event calendar that supports a wide range of views and features.

I examined this article to determine if I could utilize a free version for our practicum project. Unfortunately, utilizing this component turned out to be well beyond my current java skill set.

**Database research**

Nadal, J. (2015, January). Connect to Mysql database with java (Netbeans). Retrieved from <https://www.youtube.com/watch?v=e3gnhsGqNmI>

This is a short tutorial on how to connect a MySQL database into a java application using the Netbeans IDE. It provides a one class example of how to call on that database connection and how to insert new values into it.

I found this video very helpful for figuring out how to set up a very basic database connection within Netbeans and how to test it.

Giunipero, T. (n.d.). Creating a Simple Web Application Using a MySQL Database. Retrieved from <https://netbeans.org/kb/docs/web/mysql-webapp.html>

This is article provides a short tutorial on how to create a simple web application that uses a MySQL database for storage and data retrieval. It covers everything from the initial project creation to setting up JavaServer Pages to displaying dynamic logic and information from the database.

I found this tutorial to be a very useful source of information on how to not only start a web application project but also how to set up JSPs to display the information my project needs. It opened with a very interesting discussion of two-tier architecture where a client communicates directly with a server which applied directly to the way my own project is being developed.

Stackoverflow. (2015, October). Glassfish Admin Console throws java.lang.IllegalStateException when creating JDBC Pool. Retrieved from <https://stackoverflow.com/questions/33048435/glassfish-admin-console-throws-java-lang-illegalstateexception-when-creating-jdb>

This forum question was originally posted almost three years ago on a known bug in the Glassfish Admin Console that persists to this day. This bug prevents the creation of a JDBC connection pool which, in turn, prevents the project from reaching the database.

I found this source to be invaluable in my efforts to figure out why my project couldn't reach my database. It provided me with the workaround that effectively saved the day and allowed me to progress with my project development. After looking through the various solutions in this post I finally went with the option to modify an existing sample pool in the Glassfish Admin Console.

Tutorialspoint. (n.d.). JDBC - Statements, PreparedStatement and CallableStatement. Retrieved from <https://www.tutorialspoint.com/jdbc/jdbc-statements.htm>

This tutorial discusses the concept behind JDBC prepared statements and how they are used to define the properties that enable an application to send SQL commands to the database as well as receive data back. Prepared statements are generally used in situations where SQL statements are needed multiple times and when they must accept varying input parameters at runtime.

I found this source to be an excellent reference for how prepared statements are used as well as how to implement them. Because my project works so closely with the database I needed to master this skill very quickly in order to proceed smoothly with the development of various use cases.

Stackoverflow. (2009, August). jQuery UI DatePicker - Change Date Format. Retrieved from <https://stackoverflow.com/questions/1328025/jquery-ui-datepicker-change-date-format>

This particular source is a forum question that addresses an issue I encountered using the jQuery Datepicker widget. Without modification this widget standardizes date entry in the format of MM/DD/YYYY.

I found the solutions provided in this source to be extremely helpful in my quest to figure out how I could change the date format produced by the datepicker. The MySQL database required dates to be formatted YYYY-MM-DD for successful entry insertion.

**Design research**

Oracle. (n.d.). Class Calendar. Retrieved from <https://docs.oracle.com/javase/7/docs/api/java/util/Calendar.html>

This is an Oracle document describing the Java Calendar class which is an abstract class. It is useful for such things as determining the date one week in advance or for determining which day of the week a specific date falls on.

I found this useful for understanding the capabilities of Java's calendar class as it might pertain to my practicum project.

Sparky GlassFish. (2011, January). Java EE 6 Development with NetBeans and GlassFish. Retrieved from <https://www.youtube.com/watch?v=UBNaiVWwAZw>

This tutorial starts out with a very simple Java EE 6 web application example that uses servlets and discusses how to set up Enterprise Java Beans and JavaServer Faces pages.

I found this video to be an excellent refresher for how to start Java EE web applications and use various components. It was also very helpful in determining what basis to use for my practicum project by swaying me towards Enterprise Java and the use of a browser-based user interface presentation.

Stackoverflow. (2011, February). Show JDBC ResultSet in HTML in JSP page using MVC and DAO pattern. Retrieved from <https://stackoverflow.com/questions/5003142/show-jdbc-resultset-in-html-in-jsp-page-using-mvc-and-dao-pattern>

Although this was a forum question the responses it received from the Stackoverflow community discussed how to implement a Model, View, Controller pattern using JavaServer Pages and data provided from a database.

The discussion in this post provided an excellent overview of how the MVC design pattern is being used by professionals in field. I found the example code they provided very helpful for the development of my project.

jQuery. (n.d.). Datepicker. Retrieved from <http://jqueryui.com/datepicker/>

This page provided the source code to utilize JQuery's popup datepicker widget. It also provided API documentation.

I needed this functionality for my project because I wanted a way to standardize the way dates were entered into form fields. This enables me to bypass logic on the back end that I would have needed to handle all of the various ways dates are entered when manually typed.

CodeJava. (2018, April). How to create dynamic drop down list in JSP from database. Retrieved from <http://www.codejava.net/java-ee/jsp/how-to-create-dynamic-drop-down-list-in-jsp-from-database>

This article provided an excellent tutorial on how create a dynamic dropdown list in a JSP using tag library c:ForEach tags on the front end and Servlet/DAO classes on the back end. It provided an explanation at every stage of the process discussing why each step was taken and how it worked.

This tutorial was instrumental in the development of my own dynamic dropdowns not only for its discussion of how to utilize c:ForEach tag but also how to handle the servlet and DAO coding as well.

Stackoverflow. (2011, March). How to get the current date/time in Java. Retrieved from <https://stackoverflow.com/questions/5175728/how-to-get-the-current-date-time-in-java>

This was a Stack overflow forum question that discussed how to get the current date/time for a Java application. In this thread, forum members debated various options ranging from using the Calendar class to Joda time objects.

I found this forum posting not only useful for providing a way that I could obtain the current date for my project but also interesting for the discussion it provided on the options. How to code this simple function appears to matter of preference depending on whether one wishes to take time zones into account.

Tittel, E., & Minnick , C. (n.d.). How to Use Password Fields and Hidden Fields in Your HTML5 Form. Retrieved from <https://www.dummies.com/web-design-development/html5/how-to-use-password-fields-and-hidden-fields-in-your-html5-form/>

This article provided a tutorial on how to use password fields and hidden fields in an HTML5 form. The authors also discussed how a password field conceals a user's password so that it may not be read as the user types it in.

I found this article to be very useful while working with my HTML login page. As a simple form of security, concealing a user's password from casual onlookers is a basic necessity for modern web-based applications.

Stackoverflow. (2016, August). Adding bootstrap template to jsp. Retrieved from <https://stackoverflow.com/questions/38859248/adding-bootstrap-template-to-jsp>

This forum question discussed how to add a bootstrap template to a JSP. This allows the JSP to draw on the template for the overall style displayed on the page.

I found the answers to this forum question very useful. They helped me to figure out the correct way to direct my JSPs to a bootstrap template so that I could utilize its style formatting to beautify otherwise plain pages.

w3schools.com. (n.d.). Bootstrap 3 Tutorial. Retrieved from <https://www.w3schools.com/bootstrap/default.asp>

This tutorial from w3schools.com provides instruction on a popular HTML, CSS, and Javascript framework known as bootstrap. This framework provides formatting and color themes/templates for HTML web pages.

This tutorial provided an excellent resource to help me learn more about bootstrap as a template for beautifying my JSP pages. I found it's examples very easy to follow and utilize within my project.

Bibeault, B. (n.d.). Scriptless JSP Pages: The Front Man. Retrieved from <https://javaranch.com/journal/200603/frontman.html>

This article discusses the highly debated subject of scriptless JSP pages. It begins with a historical review outlining the script filled beginnings of JSP development before moving on to the discussion revolving around why they should be kept scriptless. The author also offers a discussion on various design patterns before concluding with the Front Controller pattern.

I found this article to be an interesting argument for scriptless JSP pages. It provided a very compelling stance for why scripts must not be allowed within JSP page code.