Online Resume Creator

22nd Annual Information Technology Competition (ITC)

Alyssa Benipayo
Erickson Angeles
Eric Kha
Ryan Trenh
Maria Vargas

Table of Contents

Introduction

The purpose of this project is to develop an Online Resume Creator that can automatically help users create professional resumes. The users only need to pick one of out the four resume templates and fill in the corresponding sections. To save the resume, the Online Resume Creator provides the options to either download the resume or email it.

Our submitted project is a prototype, but we will complete our official Online Resume Creator by April 14th.

Specific Developer Build Softwares

Eclipse IDE

We used the Eclipse IDE to build our Online Resume Creator. Eclipse is a community for individuals and organizations who wish to collaborate on commercially-friendly open source software. Its projects are focused on building an open development platform comprised of extensible frameworks, tools and runtimes for building, deploying and managing software across the lifecycle. For a full description of the the software, visit the official home page: http://www.eclipse.org

Web Design Applications

Regarding the presentation of our web pages, we utilized HTML, CSS, and Bootstrap (for themes).
For full tutorials, click on the the following links:

- HTML: https://www.w3schools.com/html/
- CSS: https://www.w3schools.com/css/
- Bootstrap: https://themes.getbootstrap.com/

Java

The programming language is Java. The following link is the tutorial to learn the language: https://www.tutorialspoint.com/java/index.htm

Java Server and JRE Library

To deploy and run our JSP and Java Servlet, we used Apache Tomcat., which is an open source implementation of the Java Servlet, JavaServer Pages, Java Expression Language and Java WebSocket technologies. For more information, visit Apache Tomcat's website:

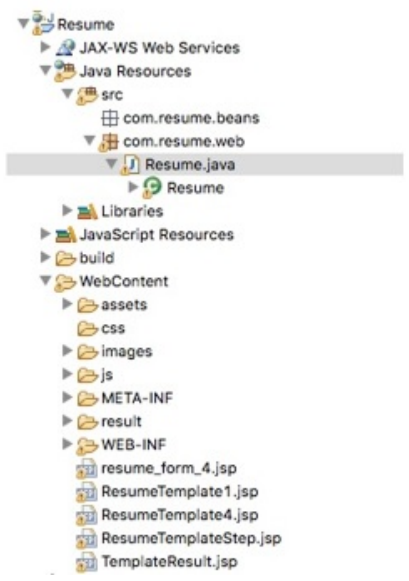
http://tomcat.apache.org/ We also utilized the Java SE Runtime Environment (JRE) library to run Java language. Downloads for the JRE Library is available at http://www.oracle.com/technetwork/java/javase/downloads/jre8-downloads-2133155.html.

Github

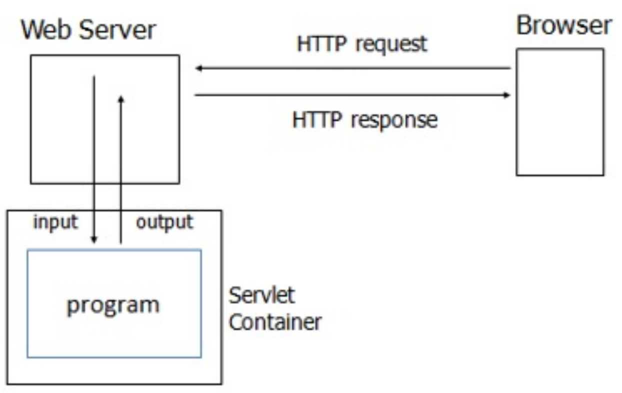
We added our working code on a development platform called Github. Github is a platform where we can host and review code, manage projects, and build software alongside millions of other developers. For more details, visit https://github.com.

Build Performance

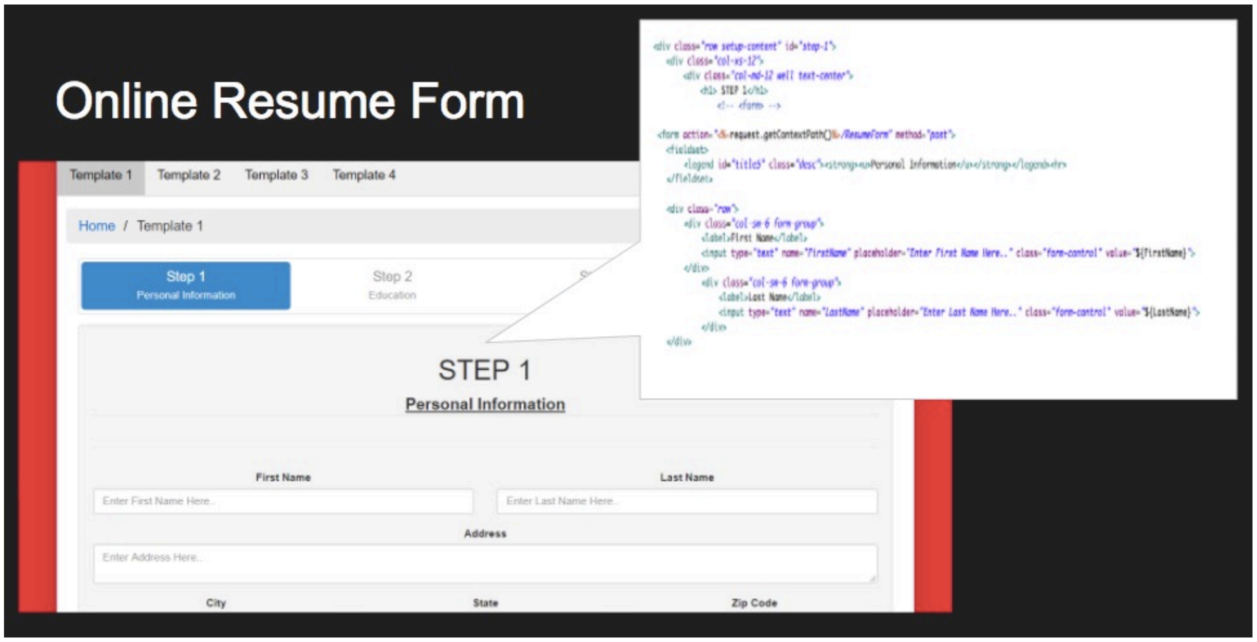
Current Online Resume Creator Architecture



```
▼ Resume
  ▶ JAX-WS Web Services
  ▼ Java Resources
    ▼ src
        com.resume.beans
      ▼ com.resume.web
        ▼ Resume.java
          ▶ Resume
    ▶ Libraries
  ▶ JavaScript Resources
  ▶ build
  ▼ WebContent
    ▶ assets
      css
    ▶ images
    ▶ js
    ▶ META-INF
    ▶ result
    ▶ WEB-INF
      resume_form_4.jsp
      ResumeTemplate1.jsp
      ResumeTemplate4.jsp
      ResumeTemplateStep.jsp
      TemplateResult.jsp
```

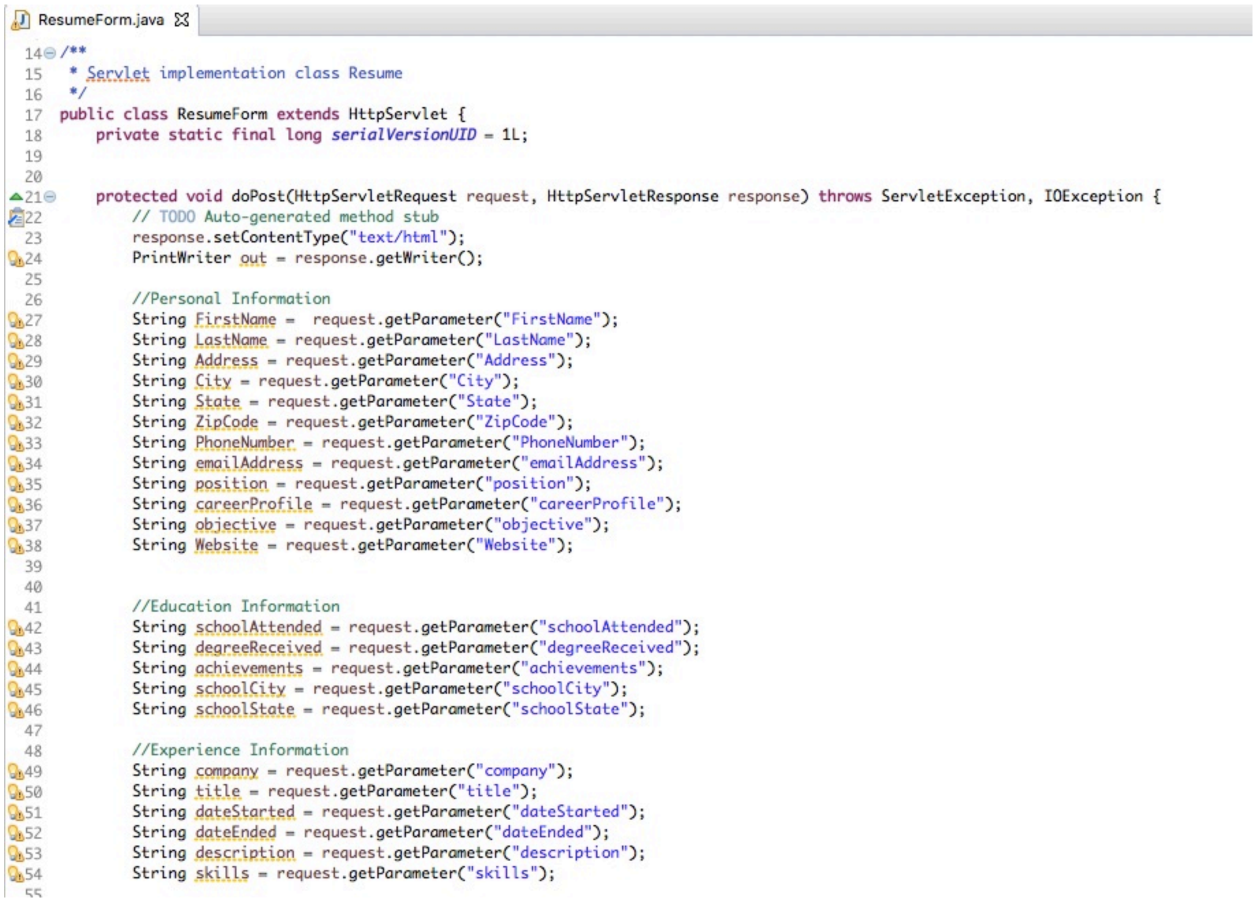


1. The request comes to Filter via web.xml. No validation at the moment.
2. User input information on Resume form.



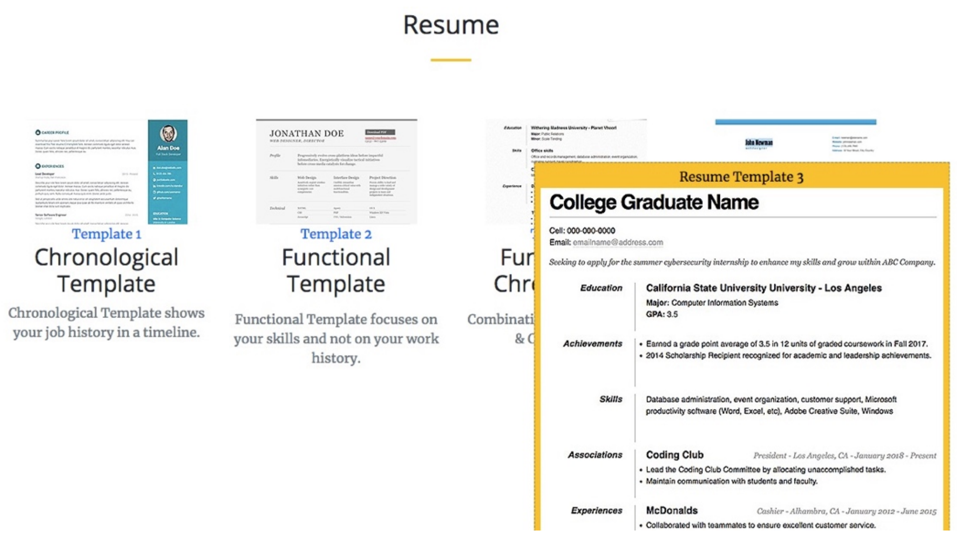
```html
<div class="row setup-content" id="step-1">
  <div class="col-xs-12">
      <div class="col-md-12 well text-center">
          <h1> STEP 1</h1>
              <!-- <form> -->

 <form action="<%=request.getContextPath()%>/ResumeForm" method="post">
  <fieldset>
      <legend id="title5" class="desc"><strong><u>Personal Information</u></strong></legend><hr>
  </fieldset>

  <div class="row">
      <div class="col-sm-6 form-group">
          <label>First Name</label>
          <input type="text" name="FirstName" placeholder="Enter First Name Here.." class="form-control" value="${FirstName}">
      </div>
          <div class="col-sm-6 form-group">
              <label>Last Name</label>
              <input type="text" name="LastName" placeholder="Enter Last Name Here.." class="form-control" value="${LastName}">
          </div>
  </div>
```

3. Servlet collects the information from the user input



```java
/**
 * Servlet implementation class Resume
 */
public class ResumeForm extends HttpServlet {
    private static final long serialVersionUID = 1L;


    protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        // TODO Auto-generated method stub
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();

        //Personal Information
        String FirstName =  request.getParameter("FirstName");
        String LastName = request.getParameter("LastName");
        String Address = request.getParameter("Address");
        String City = request.getParameter("City");
        String State = request.getParameter("State");
        String ZipCode = request.getParameter("ZipCode");
        String PhoneNumber = request.getParameter("PhoneNumber");
        String emailAddress = request.getParameter("emailAddress");
        String position = request.getParameter("position");
        String careerProfile = request.getParameter("careerProfile");
        String objective = request.getParameter("objective");
        String Website = request.getParameter("Website");


        //Education Information
        String schoolAttended = request.getParameter("schoolAttended");
        String degreeReceived = request.getParameter("degreeReceived");
        String achievements = request.getParameter("achievements");
        String schoolCity = request.getParameter("schoolCity");
        String schoolState = request.getParameter("schoolState");

        //Experience Information
        String company = request.getParameter("company");
        String title = request.getParameter("title");
        String dateStarted = request.getParameter("dateStarted");
        String dateEnded = request.getParameter("dateEnded");
        String description = request.getParameter("description");
        String skills = request.getParameter("skills");
```

4. Result will be posted on a new jsp page.- to collect information from the servlet, we use the most basic scripting functions <% ... %>, which encloses a JSP scriptlet. A scriptlet is a fragment of Java code that is run when the user requests the page.



Resume

Template 1
Chronological Template
Chronological Template shows your job history in a timeline.

Template 2
Functional Template
Functional Template focuses on your skills and not on your work history.

Fur
Chr
Combinati
& C

Resume Template 3

College Graduate Name

Cell: 000-000-0000
Email: emailname@address.com

Seeking to apply for the summer cybersecurity internship to enhance my skills and grow within ABC Company.

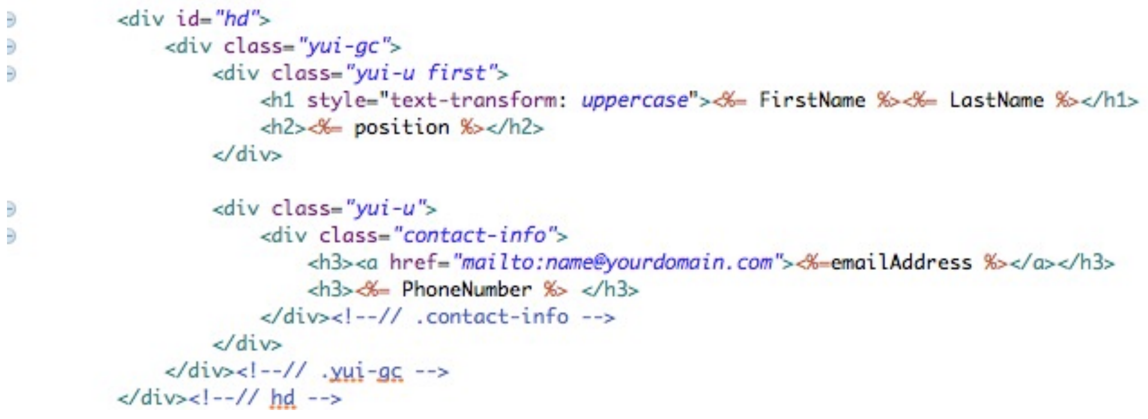
Education — **California State University University - Los Angeles**
Major: Computer Information Systems
GPA: 3.5

Achievements
- Earned a grade point average of 3.5 in 12 units of graded coursework in Fall 2017.
- 2014 Scholarship Recipient recognized for academic and leadership achievements.

Skills — Database administration, event organization, customer support, Microsoft productivity software (Word, Excel, etc), Adobe Creative Suite, Windows

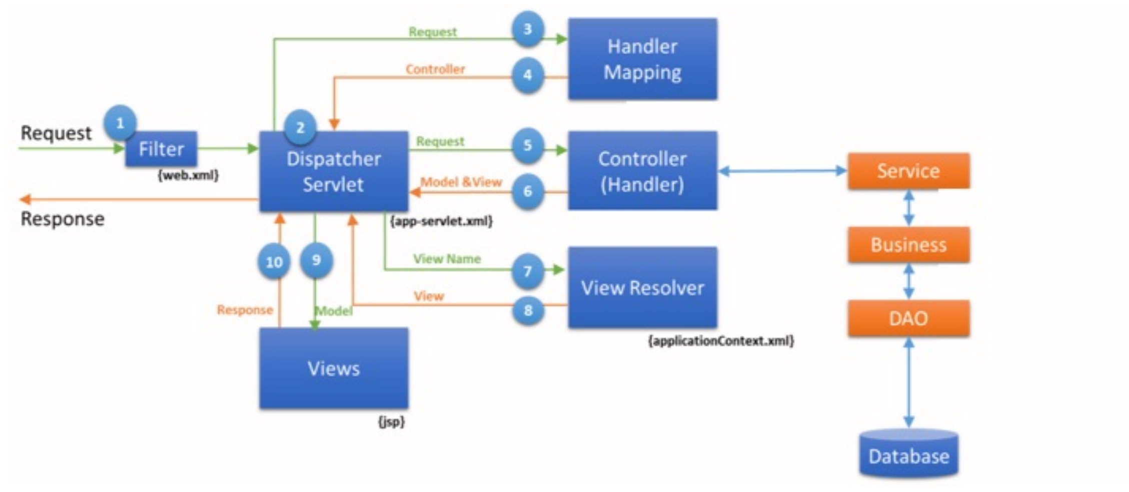
Associations — **Coding Club** *President - Los Angeles, CA - January 2018 - Present*
- Lead the Coding Club Committee by allocating unaccomplished tasks.
- Maintain communication with students and faculty.

Experiences — **McDonalds** *Cashier - Alhambra, CA - January 2012 - June 2015*
- Collaborated with teammates to ensure excellent customer service.



```
<div id="hd">
    <div class="yui-gc">
        <div class="yui-u first">
            <h1 style="text-transform: uppercase"><%= FirstName %><%= LastName %></h1>
            <h2><%= position %></h2>
        </div>

        <div class="yui-u">
            <div class="contact-info">
                <h3><a href="mailto:name@yourdomain.com"><%=emailAddress %></a></h3>
                <h3><%= PhoneNumber %> </h3>
            </div><!--// .contact-info -->
        </div>
    </div><!--// .yui-gc -->
</div><!--// hd -->
```

Future Online Resume Creator Architecture



1. The request comes to Filter via web.xml. All pre-processing, security validations, sessions checks are completed and request is forwarded to DispatcherServlet defined in web.xml
2. It automatically loads the spring configuration file whose name format is <servlet-name>-servlet.xml,where <servlet-name> is configured in web.xml file.
3. The DispatchServlet consults Handler Mapping to decide controller class to delegate.
4. The Handler Mapping decides one controller among many controllers based on URL path.
5. The DispatchServlet delegates request to the corresponding Controller.
6. The Controller returns ModelAndView, which contains logical view and model data, back to DispatchServlet.
7. The DispatcherServlet consults a view resolver to resolve response view name including its location.
8. The ViewResolver returns physical response view page name.
9. Finally, the DispatchServlet does view navigation to response page.

Authors

Alyssa Benipayo
Erickson Angeles
Eric Kha
Ryan Trenh
Maria Vargas

Faculty Advisor

Shilpa Balan