**User Documentation**

**Team 9 – ResumYay**

Project setup

Since the project is run locally, until we set up the database and website online and purchase the domain the project will have to be set up and installed locally. Thus the steps to install the program for the user is the same as the steps to install the program as a developer. This is planed to be worked on and fixed in the future so the user to access the program would just have to load the webpage linked with ResumYay.

Bracken Brinkel, Tarek Fouad, Tommy Pham, and Marina Alsip

**Project setup and installation (only for windows)**

**Frontend Installation**

To develop Angular applications, you need to install Node.js and npm (Node Package Manager). Follow these steps to install Node.js, npm, and Angular CLI.

**Install Node.js and npm**

**On Windows**

1. Download Node.js:
   * Go to the [Node.js download page](https://nodejs.org/) and download the Windows Installer (.msi) for the LTS (Long Term Support) version.
2. Run the Installer:
   * Double-click the downloaded .msi file and follow the prompts. Ensure that the option to install npm is selected.
3. Verify Installation:
   * Open Command Prompt and run:
   * “ node -v ”
   * “ npm -v ”
   * Confirm that Node.js and npm are installed correctly by seeing a version upon running those commands.
4. If you get an error saying "cannot be loaded because running scripts is disabled on this system": - Type env in the windows bar. - Select the Environment Variables in the bottom corner. - Click on "Path" - Find the path for npm. For example "C:\Users\SomeUser\AppData\Roaming\npm" - Close env.

**Angular Installation**

With Node.js and npm installed, you can now install Angular CLI.

1. Install Angular CLI:
   * Open a terminal or Command Prompt and run:

“ npm install -g @angular/cli ”

1. Verify Angular CLI Installation:
   * Check the version of Angular CLI:

“ ng version ”

1. If you get this error "Could not find the implementation for builder @angular-devkit/build-angular:dev-server on ng serve command [duplicate]":
   * In VS code go to the ResumeBuilderUI.2 folder
   * Run this command in VS Code:

“ npm install ”

1. Reset Visual Studio Code

**Database Installation**

We will be using SQL Server 2022 and to view the data in the database the application being used SQL Server Management Studio 20.

1. Download SQL Server 2022:
   * Go to the [SQL Server download page](https://www.microsoft.com/en-us/sql-server/sql-server-downloads) and download the SQL Server 2022 Developer.
   * After installation a page will prompt the users to connect the server using the connect now button.
   * A command prompt window will pop up after hitting the connect button. Then you can close the installer.
2. Download SQL Server Management Studio 20:
   * Go to the [SQL Server Managment Studio 20 download page](https://learn.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver16) and download the SQL Server 2022 Developer. You can download SMSS from the SQL Server 2022 Installer.
3. Using SQL Server:
   * Open up SQL Server Management Studio 20.
   * Uses the icon with an outlet.
   * Look at Server name and copy that name. (for later use)
   * Open the C# project in visual studio 2022, it will be located in the ResumeBuilder(Team9) folder.
   * In the solution explorer on the right side of the project, find appsettings.json and look for this code on line 10. (The line bellow is the line that you are modifying)

"Data Source=DESKTOP-HB3HD5S;TrustServerCertificate=True;Initial Catalog = ResumeBuilderDb; Integrated Security = true;"

* + Replace Data Source with the Server name you copied from the previous step. ("Source=DESKTOP-HB3HD5S;")
  + In Package Manager Console, update the database with the command bellow.

“ update-database ”

* If you can not find the Package manager Console, use

tools->Nu Get Package Manager->Package Manager Consol.

* + Now the database is set up and ready to be used.
  + Now you can go back to SQL server management studio, make sure to press the check box next to the trust server certificate, then press connect at the bottom left of the connect to server window.

**Run the Angular Application**

1. Start the Development Server:
   * Open the ResumeBuilderUI.2 folder in VS code.
   * Run the following command in the terminal to start the Angular development server: (Make sure the terminal path is the ResumeBuilderUI.2)

“ ng serve -o ”

* If you get an error that states you can not run scripts on the current system. You will have to give your computer permission to runs scripts by running this command in your PowerShell in administrator mode.

“ Set-ExecutionPolicy -ExecutionPolicy Unrestricted ”

* If you get this error :"Error: Could not find the '@angular-devkit/build-angular:dev-server' builder's node package."   
  look at the solution in part 2 step 3.

1. Access the Application:
   * Open your browser and navigate to http://localhost:4200 to view your Angular application.

**Run the Angular Application**

To start running the database make sure to run RsumeBuilder(Team 9) project by pressing the HTTPS button near the top where you would normally see your debug compiler button.

Once the back end is running a swagerUI page with open in your default browser. (make sure that the settings on that browser are allowing localhost pages to be launched).

The front end will start running when you type in the terminal "ng serve -o" (we did this in part 4 step 1).

Webpage Navigation

This project is a website and has many webpages that will be discussed later in the documentation. This section will list the pages available.

* Home page
* About Page
* Regestration Page\*
* Login Page\*
* Dashboard page\*
* Education page\*
* Work Experience page\*
* Keywords suggestion Computer Science page
* Keywords suggestion Business page
* Skills page\*
* Download page
* Options page\*
* ResumeTemplates Page\*
* Tips and Tricks page
* UCCS strong word suggestion page

The pages with the ‘ \* ‘ at the end are pages that deal directly to the back end or database, while the other pages are mostly type script and deal with data and information in the front end.

Web page fields

* Required Fields: throught the resume there are required fields that the user must input, these will be marked if the user leaves the field empty and tries to move on.

A close up of a box

Description automatically generated

* Non required fields: could be left blank and will not cause any required field notifications.
* Data modification and editing after input: For pages like the jobs and degrees we let the user enter a maximum of 3 jobs and 3 degrees. These would entered similar to any page. When you add the degree or job it will ad it to a list at the bottom of the page then can be deleted or edited by pressing on either button which would add the information to fields that can be modified and saved.

A white rectangular object with a white background

Description automatically generated

* Skills keywords suggestions: Depending on the major we suggest skill keywords that would be relevant to the user (Currently we only have 2 majors Computer Science and Business degrees). Once the user gets to this page the user would select the keywords relevant to them.

A computer keywords screen shot

Description automatically generated

The skills that are selected would appear in the skills section on the next page under skills.

A screenshot of a computer

Description automatically generated

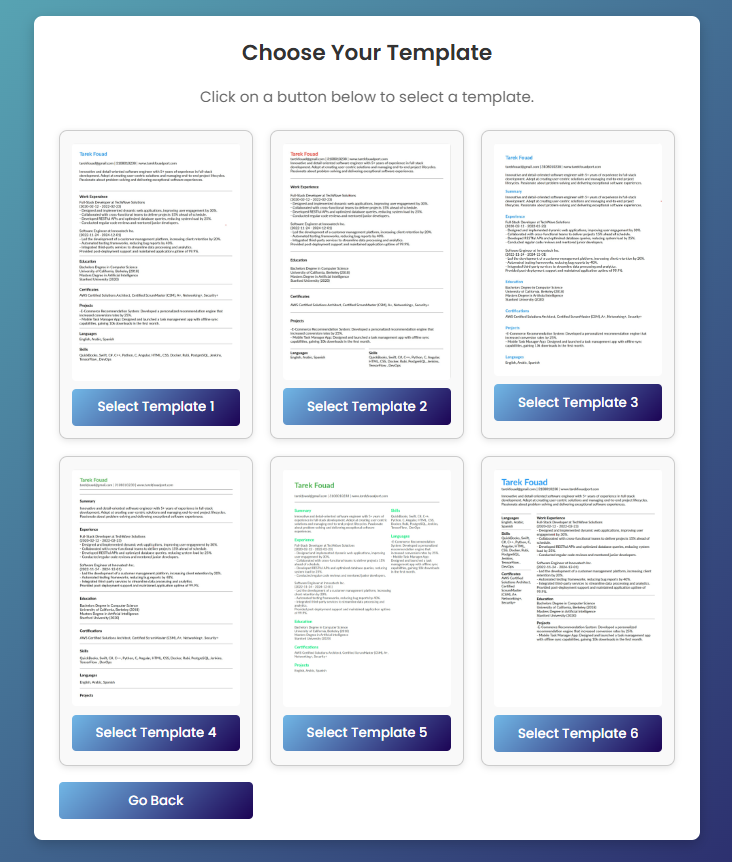
To select the major to get the selected skills keyword page, is on the options page as a drop down at the top.

A screenshot of a computer search box

Description automatically generated

**Resume Template selection and download page**

Once you finish the website pages for the user information the user will be guided to the download page. On the download page the user will press on the Download Resume which will be prompted to the template selection page. The selection page demonstrates all the templates with mock information and how it would look. The user can select the template that matched the style they want the most then it will be downloaded to their default downloads folder for that browser (most browsers will download to the download folder).



**Not logged in users**

The project does support users that are not logged in and don’t want their data to be saved long term in the database. For users that don’t want to create accounts, the user can just press create resume on the home page and this would set them as a temp user where they can start a resume and once they are done and the resumes are created their data is wiped.

A screenshot of a computer

Description automatically generated

**Information pages**

There are some pages that are just information pages that would have some links and information. These pages have not back end integration or usage or much typesctip code. These pages are mainly HTML and CSS.

A screenshot of a web page

Description automatically generated

**About page and contact information**

There is an included page on the main home page that can be accessed and it has the git link for our project, contact information, and the class and project it is used for.

**A screenshot of a computer

Description automatically generated**