

Trey Leigh

Adam Gerard

## CS50 Final Project: Design Documentation

O hai! This was CS50: LINKUPWITH.Me. I am going to give you a little bit of the under the hood of our project. We took a lot of our design from bootstrap. Our design has a navigation bar so that it would be easy to switch pages. The navigation bar and header are at the beginning of all our html/php webpages. A few files are accessible through the navigation bar: index.html (homepage), about.html (about page), recruiters.html, students.html, faq.html, Tryit.html. The first two pages are relatively static. Here, we used the bootstraps template and supplied our own static content regarding what the LINKUPWITH.ME application will look like upon completion and about the aim of the product. Students.html and recruiters.html both have submission forms at the bottom of the page. Upon submission, the content is pasted into recruitersignup.php or studentsignup.php via the `$_POST` global variable. These two files query into a recruiters and students SQL databases respectively (note: we used CS50 query for all query mechanisms during this project).

Next we will discuss faq.html. This page has a form in the shape of a search bar. This form takes the user submission, passes it via `$_POST` to faq.php. In faq.php, `$_POST["question"]` is queries against several keywords found in the sql table "faq3" and places result in an associative array. If the query returns true, then the string is extracted from that array into `$answer`. Answer is then passed into render, which renders answer.php with `$answer` as the string. If the query returns false, `$answer` is again passed into render, but this time as a string saying that the answer to the question is not available at the moment.

Now, we shall discuss Tryit.html. This page has 3 buttons in it that are all of submit type. Each button goes to specific registration page: company registration, recruiter registration, or student registration. Company registration ("register\_company\_company.php") is simply a form with company and desired company code fields. Upon submission, the data is passed into registration\_company.php. If the data queries properly, register\_company.php is rendered. This is the page in which recruiters register. Upon completion of registration, events\_for\_recruiters.php is opened. This file queries through events that this recruiter has and renders recruiting.php while passing an associative array called `$events` into recruiting.php. Recruiting.php is a page with a table using a foreach loop that displays all of the recruiter's upcoming events. Clicking the Add Event button at the bottom of the page opens event\_submission\_form.php. Upon submitting the event details, the data is passed into new\_events.php, which queries the information and redirects to events\_for\_recruiters.php.

If you return to Tryit.html and instead click on students, this will take you to register\_form\_student.php. Upon submitting the information in the fields, the data will be passed into students.php, which will query the information and render student\_events.php. It is important to note that there is also a Login button at the bottom of every register page. Upon being clicked, this opens login\_form.php. The user submission on this page is passed into login.php, which then either redirects the user to students.php or events\_for\_recruiters.php depending on the type of user who has logged in.

Lastly, there is the "find event button". This button is found in student\_events.php. When clicked, it takes the user to find\_event.php, which has a submission form that takes the user to

new\_user\_events.php. The data from the find\_event.php is passed into new\_user\_events.php. This file finds an event that matches the company and the school and then places it in the users\_events SQL table. Lastly, it redirects the user to students.php.

And that is a brief look at how our site operates. These files can be found in the submitted zip. That's all folks!