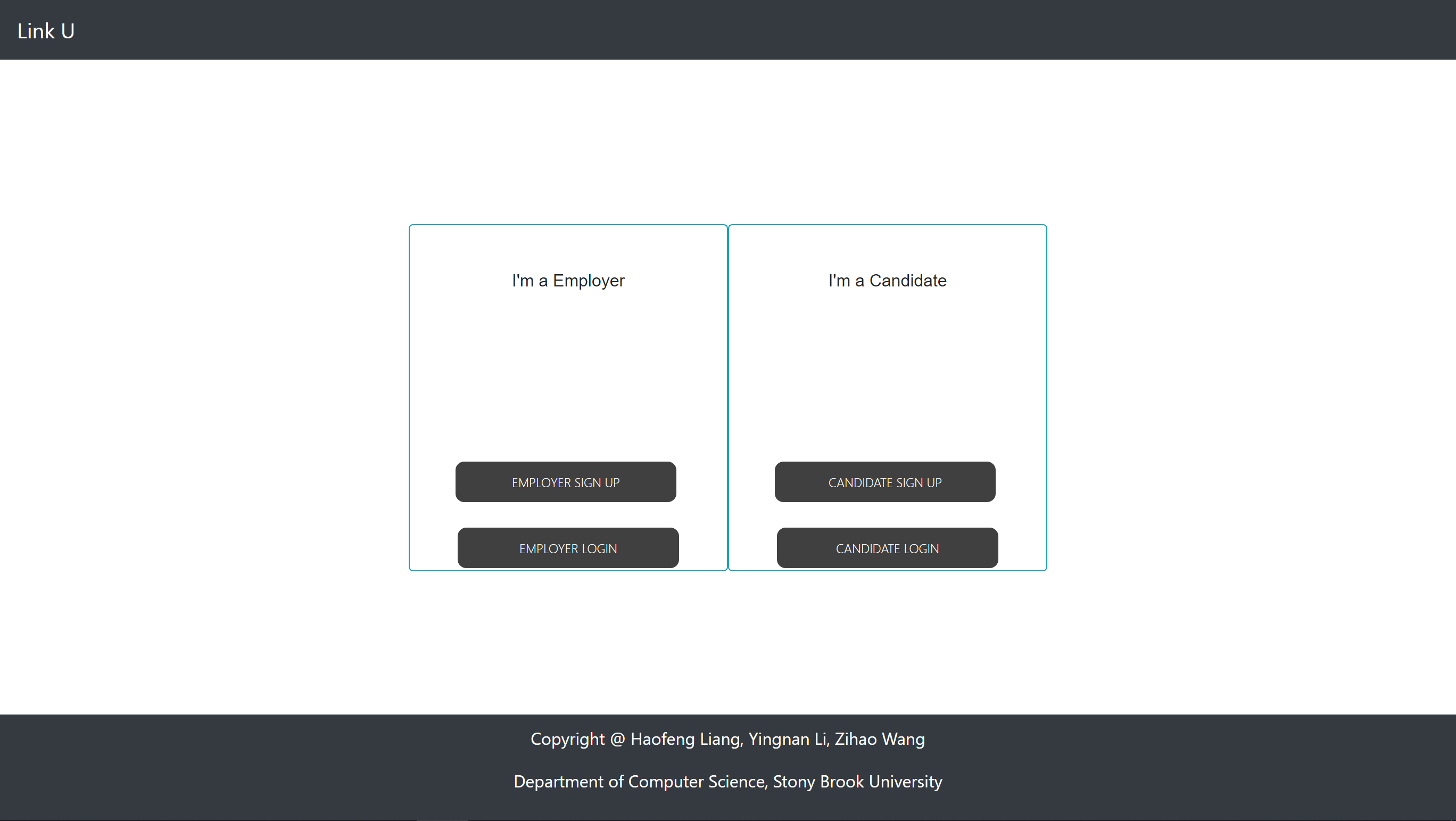
**This is documentation of front end of Opotunity development:**

**By: Yingnan Li, Zhihao Wang, Haofeng Liang**

Write at the beginning:

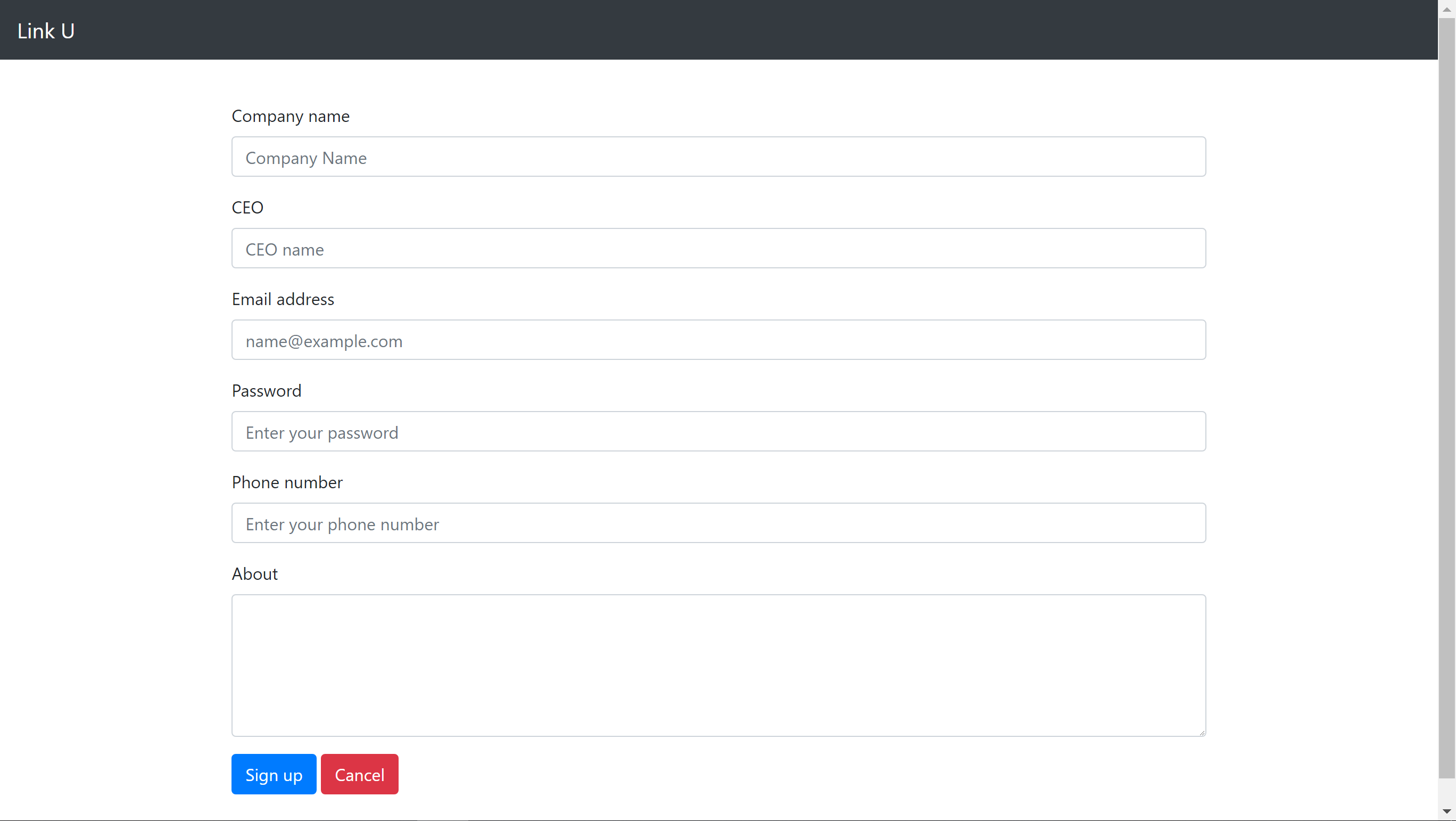
We build up these pages with JSP, a combination of Java and HTML. Also, we mainly use CSS from Bootstrap along with a bit of our costume style.

Front Page:



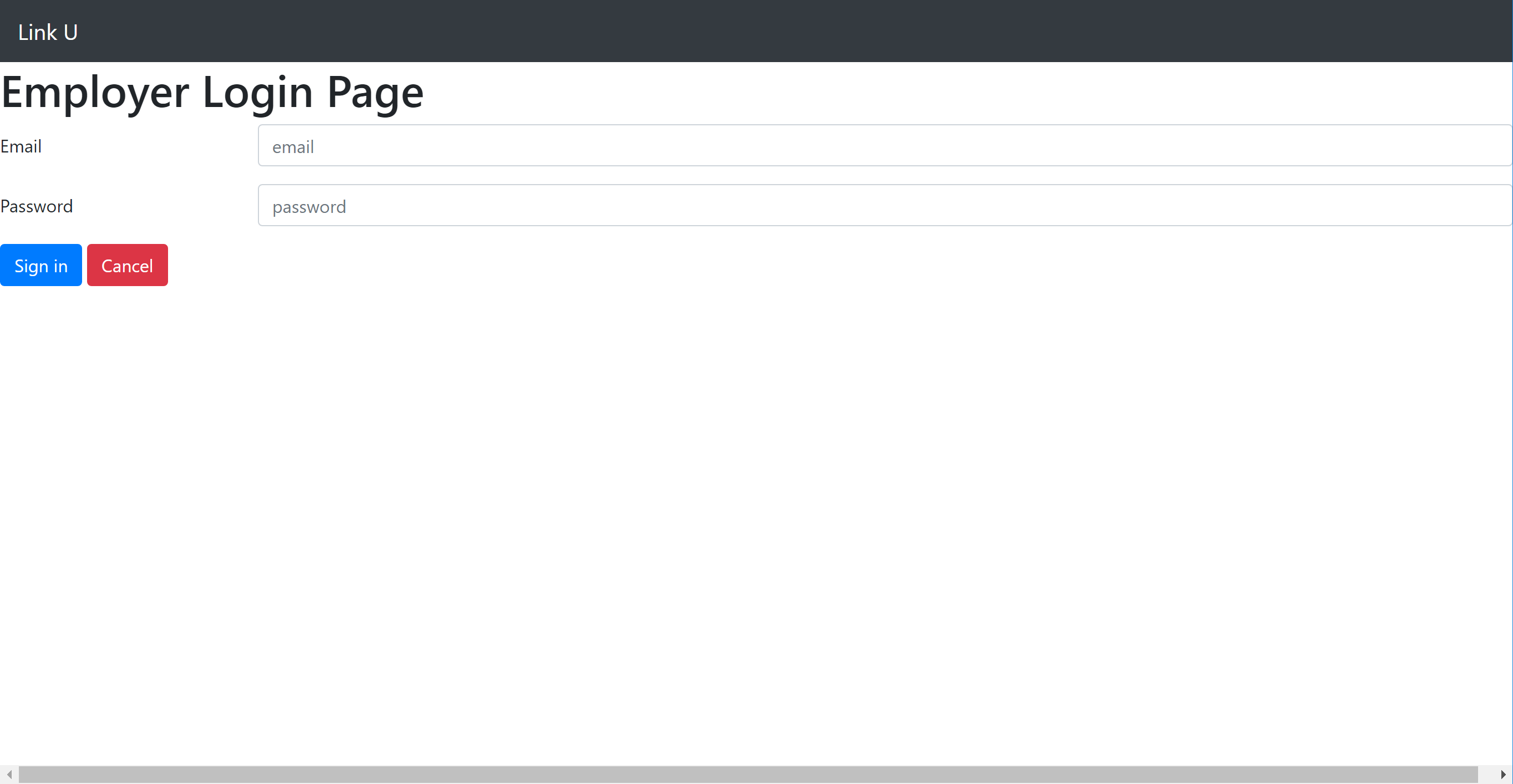
We have left and right pane that design for two type of users, employers and candidates to login or signup if they don’t have an account ready. A nav bar on top along with our brand name, a login button may occurs if the user did log in.

Employer Register Page:



A user may register their account as an employer. he/she must provide a company name, a CEO name, a email address, password and phone number. About is optional provided. After entering all information, user may click on “Sign Up” to register the account, or “Cancel” to go back to the front page.

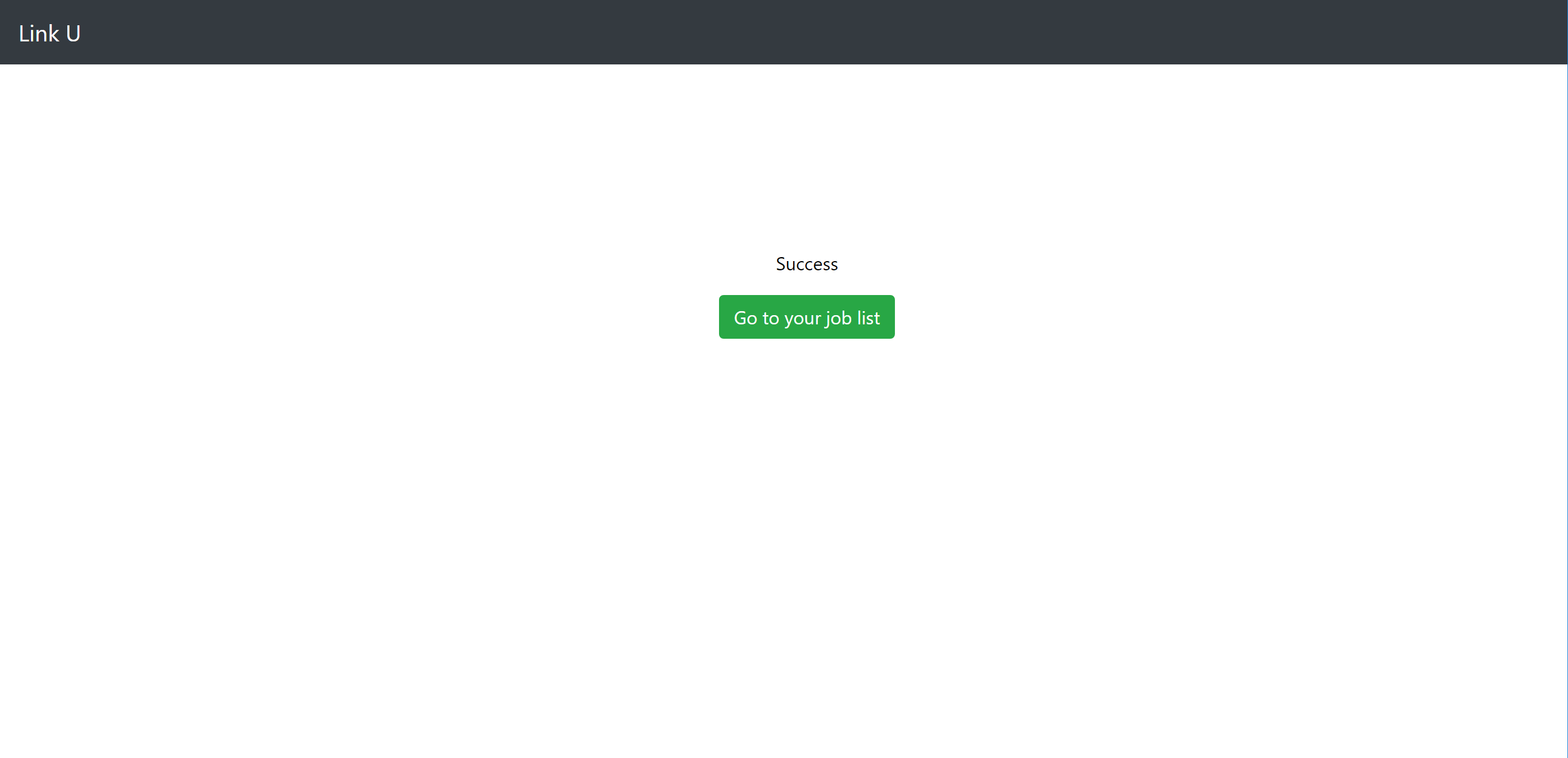
Here, we used a form that contains fields with css provided by Bootstrap for the form. “email address” is implemented as email address input, so the browser can make sure if it is in right format. Password is then implemented as password input, which can hide the input from displaying so provide security to the user. About is then a text area since there might be multiple lines info that user wants to provide. All others are text fields. Two button are both implemented as submit type, which calls different pages.



After having an account, user that wants to login as an employer can access to the Employer login page to login with Email and Password as credentials.

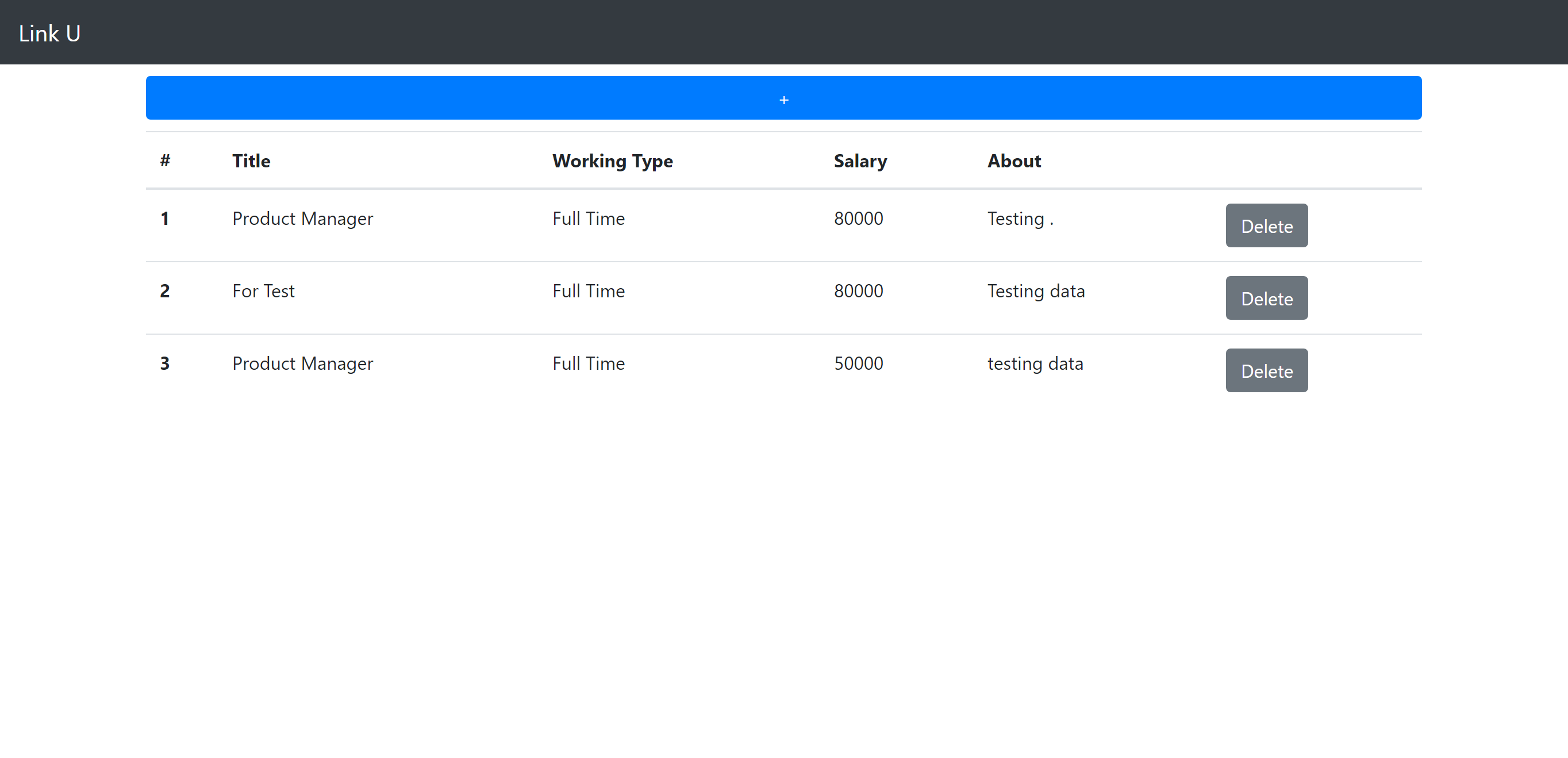
Here, we used a form again. Password will be hide after typed as to secure the privacy. “Cancel” button will then submit nothing, but goes back to front page, where “Sign up” button will submit the credentials to the checking page.

Employer Login Checking Page:



After an employer login, we use this page check if the credentials are valid. If it is, we provide the button that links to the page that shows all jobs that the employer has created. Otherwise, a button that link back to the login form will be provided.

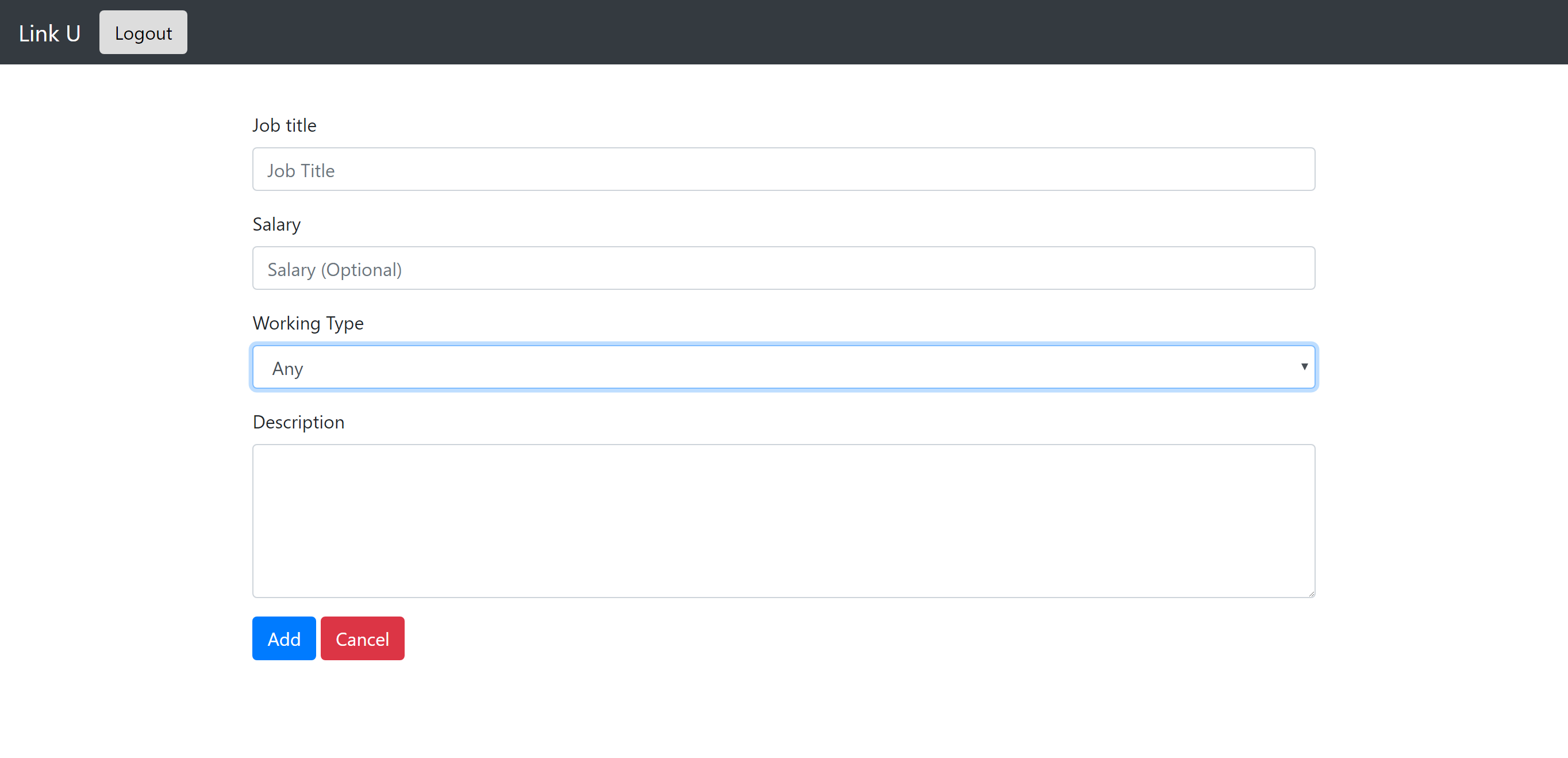
Employer Job List Page:

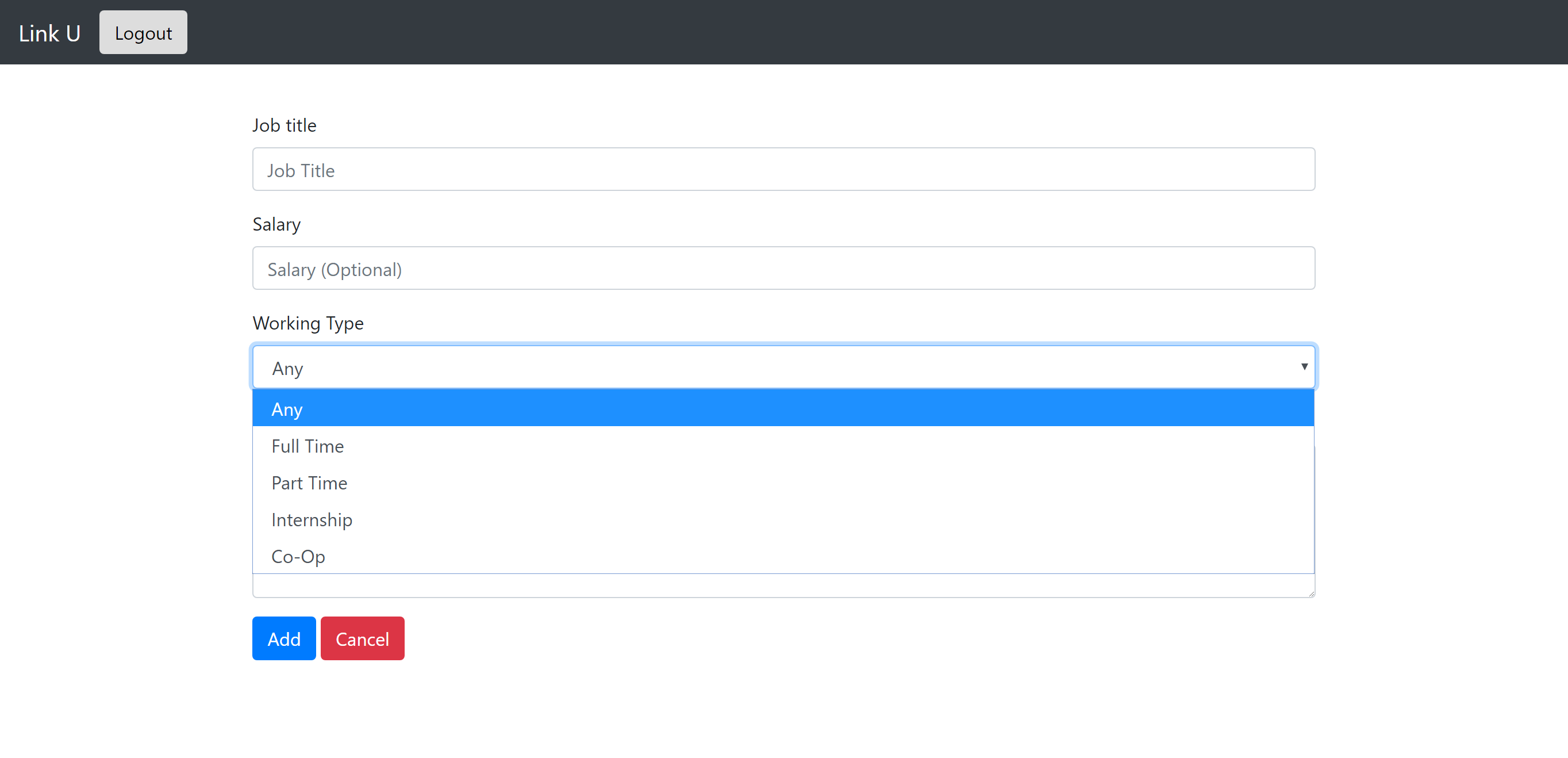


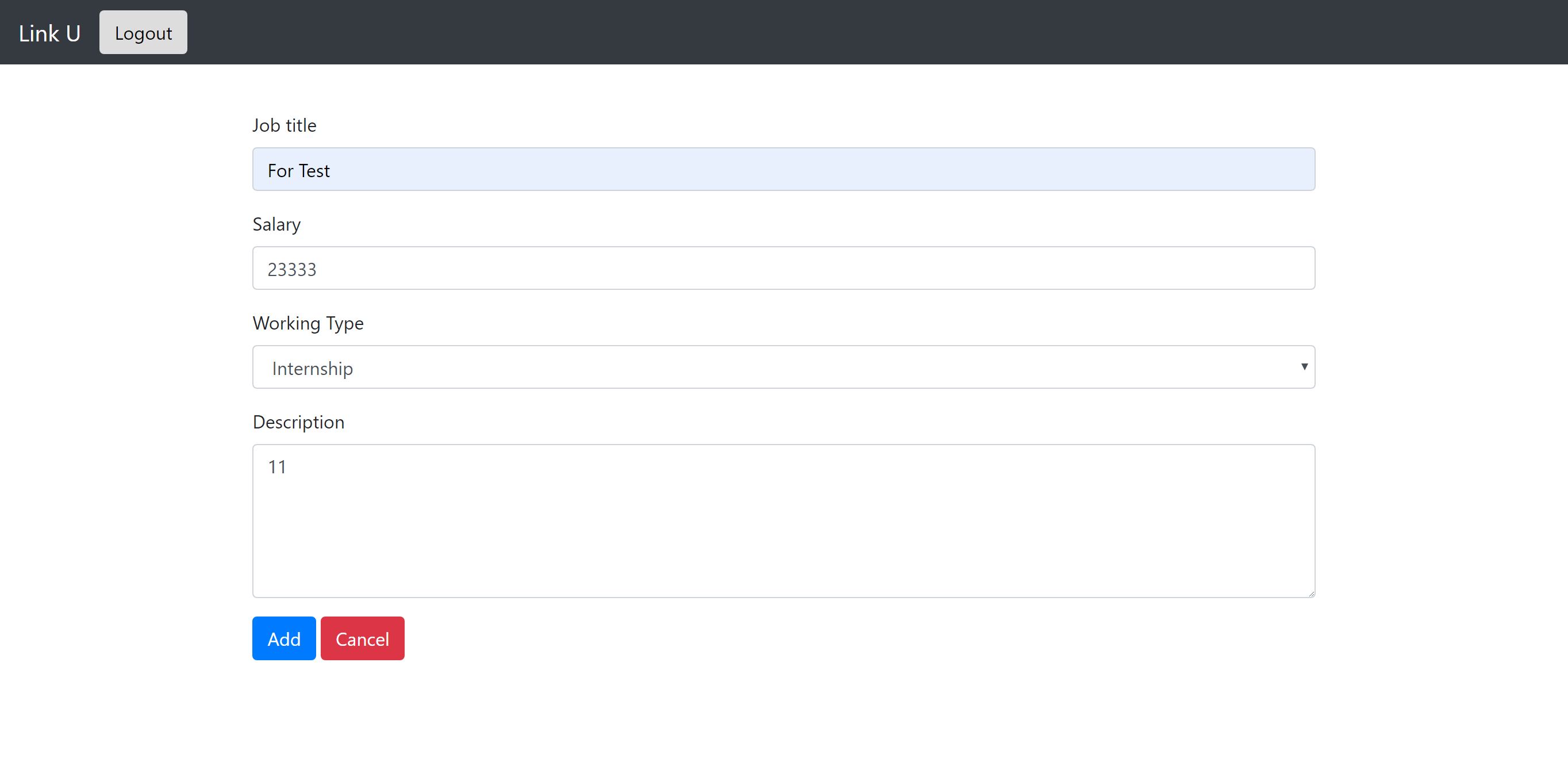
As after user login, a log out button will appear on the nav bar. And this page, only shows to who login as employer. This will show a list of jobs in the system that created by the employer. Along with a “+” button that allows user to go to another page to add a job and “Delete” buttons to delete each row of data.

Here, we implemented a “+” button that pass the user id to add job page, so the page knows who is adding the job. And “Delete” buttons to pass the id of each row data to a function that delete the row from database.

Add Job Page:

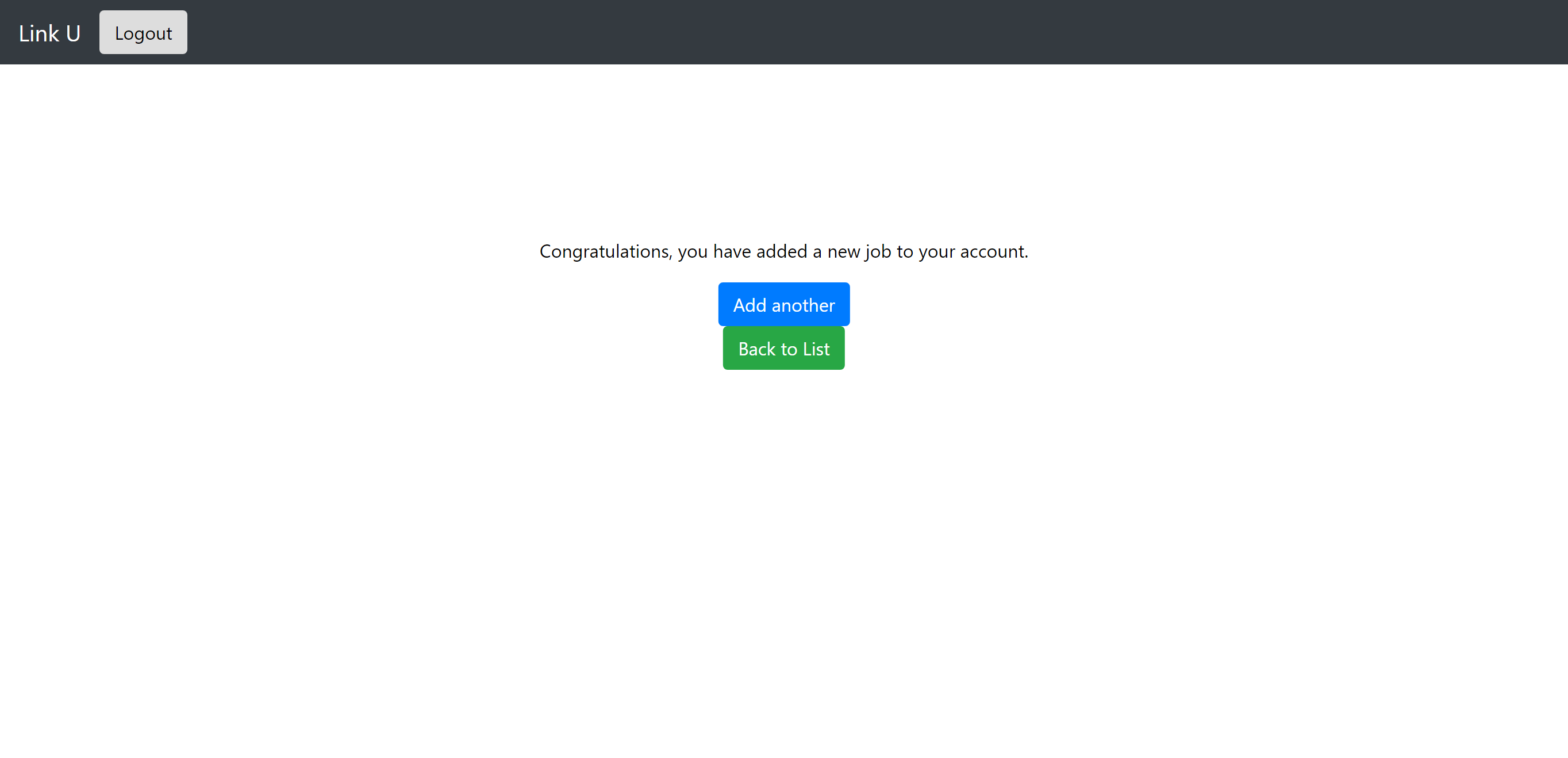




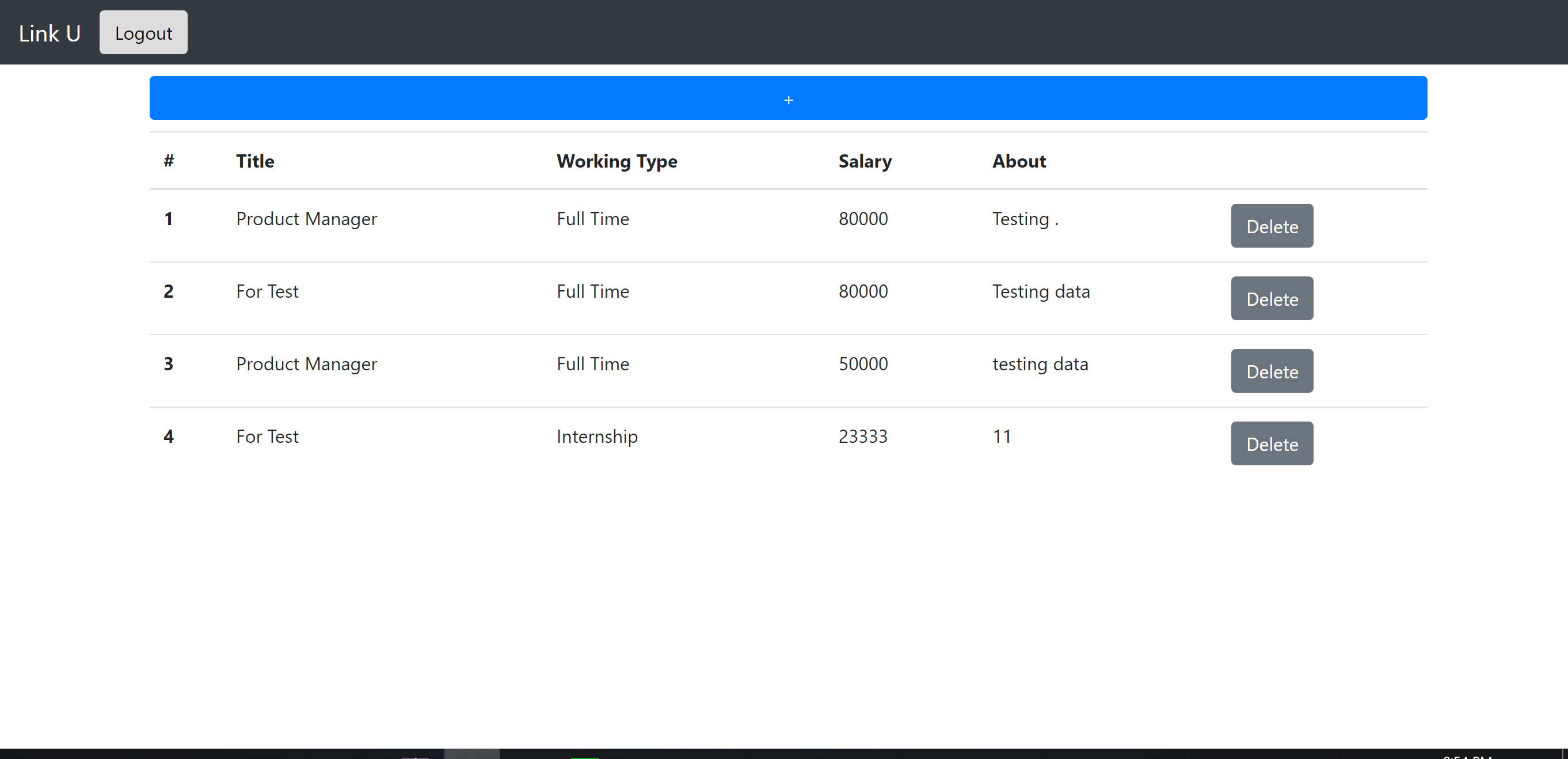


If an employer clicks on the “+” button on its job list page, this page will then show up. Text field for both Jon title and Salary. A select list for working type, user may select one of {Any, Full Time, Part Time, Internship, Co-Op} as the value. And a text area for Description. After click on “Add” button, all info will pass to a page that add a row into database, “Cancel” will then bring the user back to list.

Job added confirmation page:

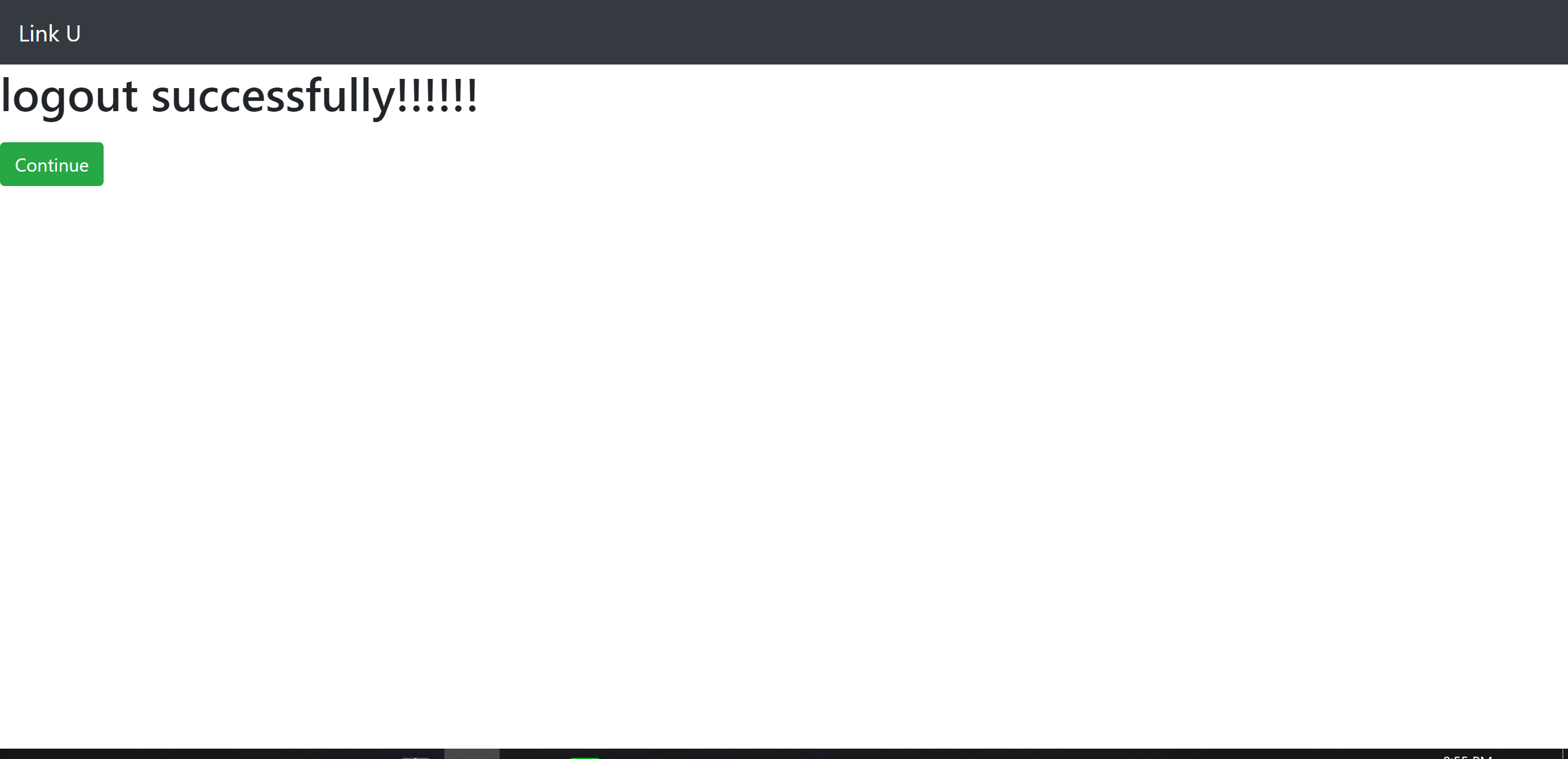


If a job is added successfully, this page will then show up. With confirmation text along with a button that bring user back to add job page to add another job, or another button that brings user back to job list page.



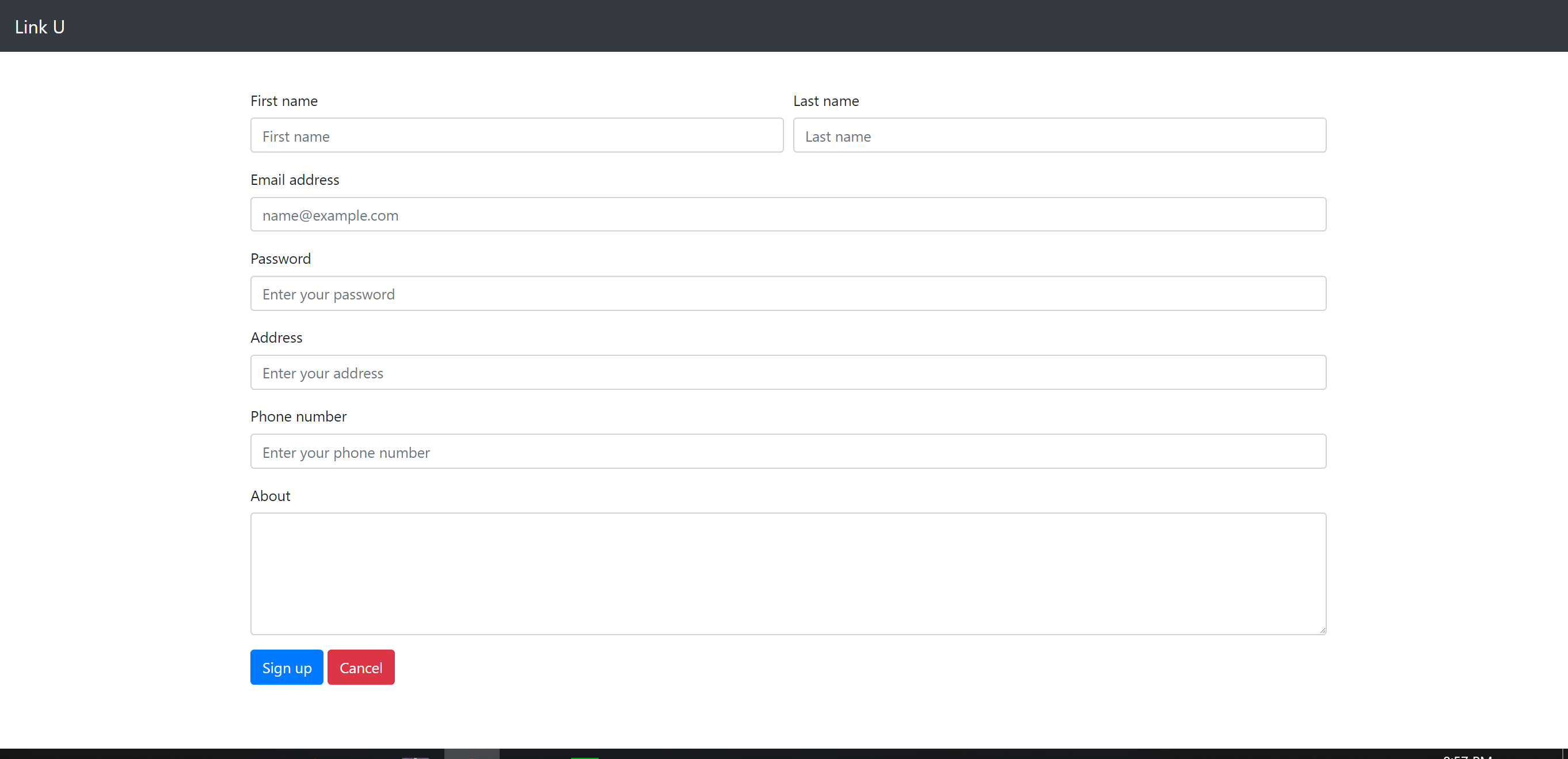
If everything doing successfully, a job will added to the list and shown to the user.

Log out confirmation:



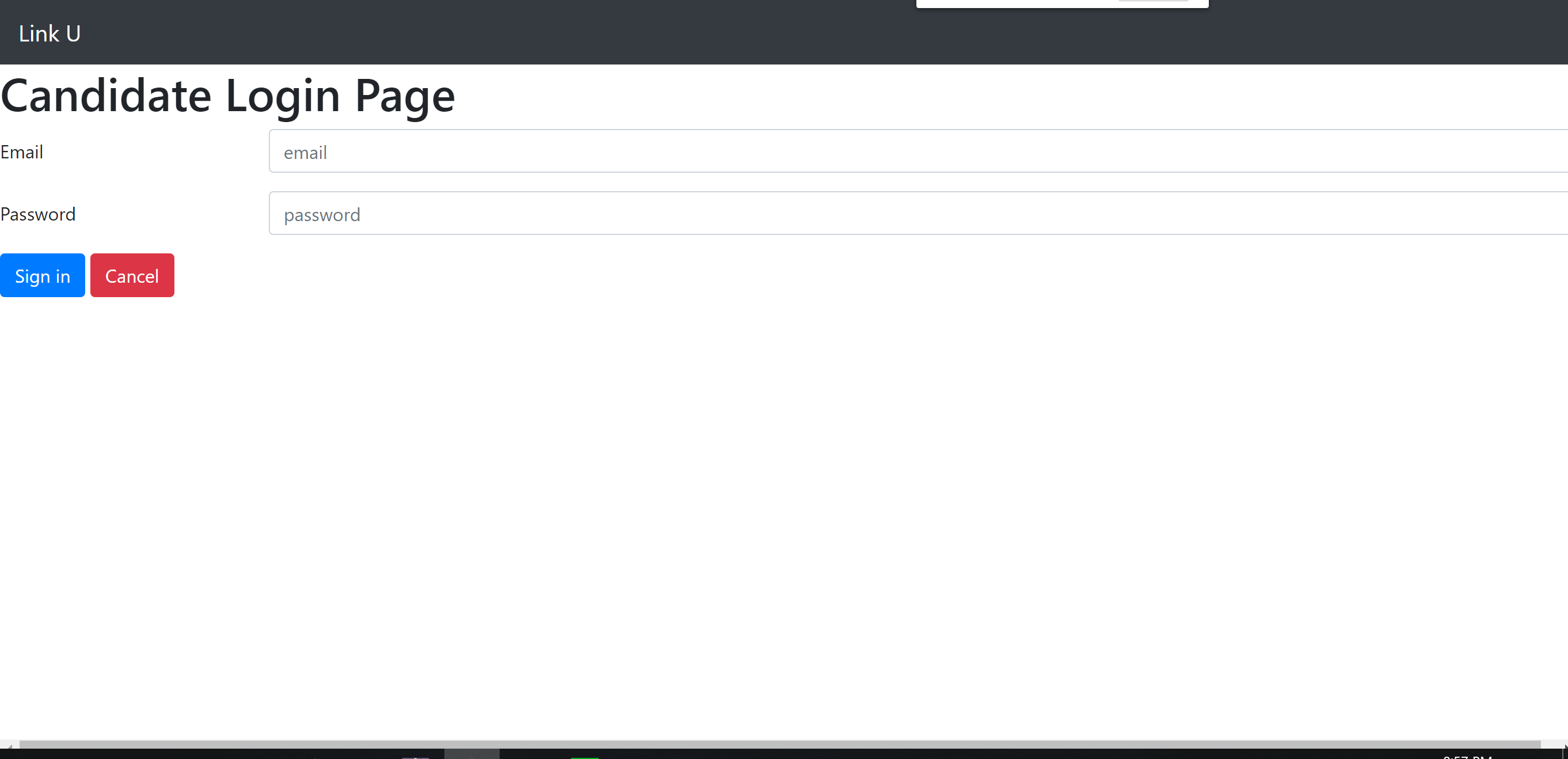
As when the employer finish with the program, he/she may click on log out button appears on previous pages. If successfully doing so, a confirm page will then print out, along with a button that links back to front page.

Candidate register page:



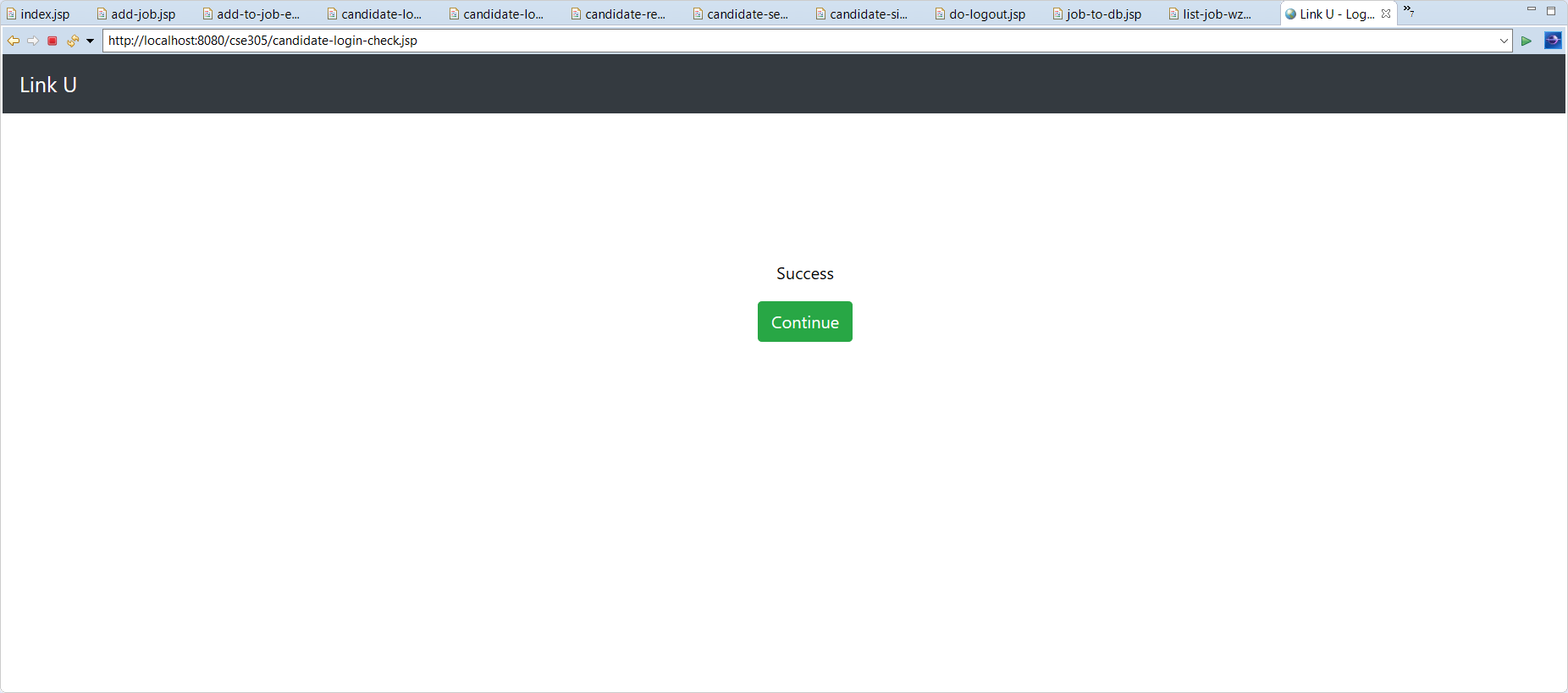
Like the implementation of Employer register page, except the field names are different.

Candidate login page:



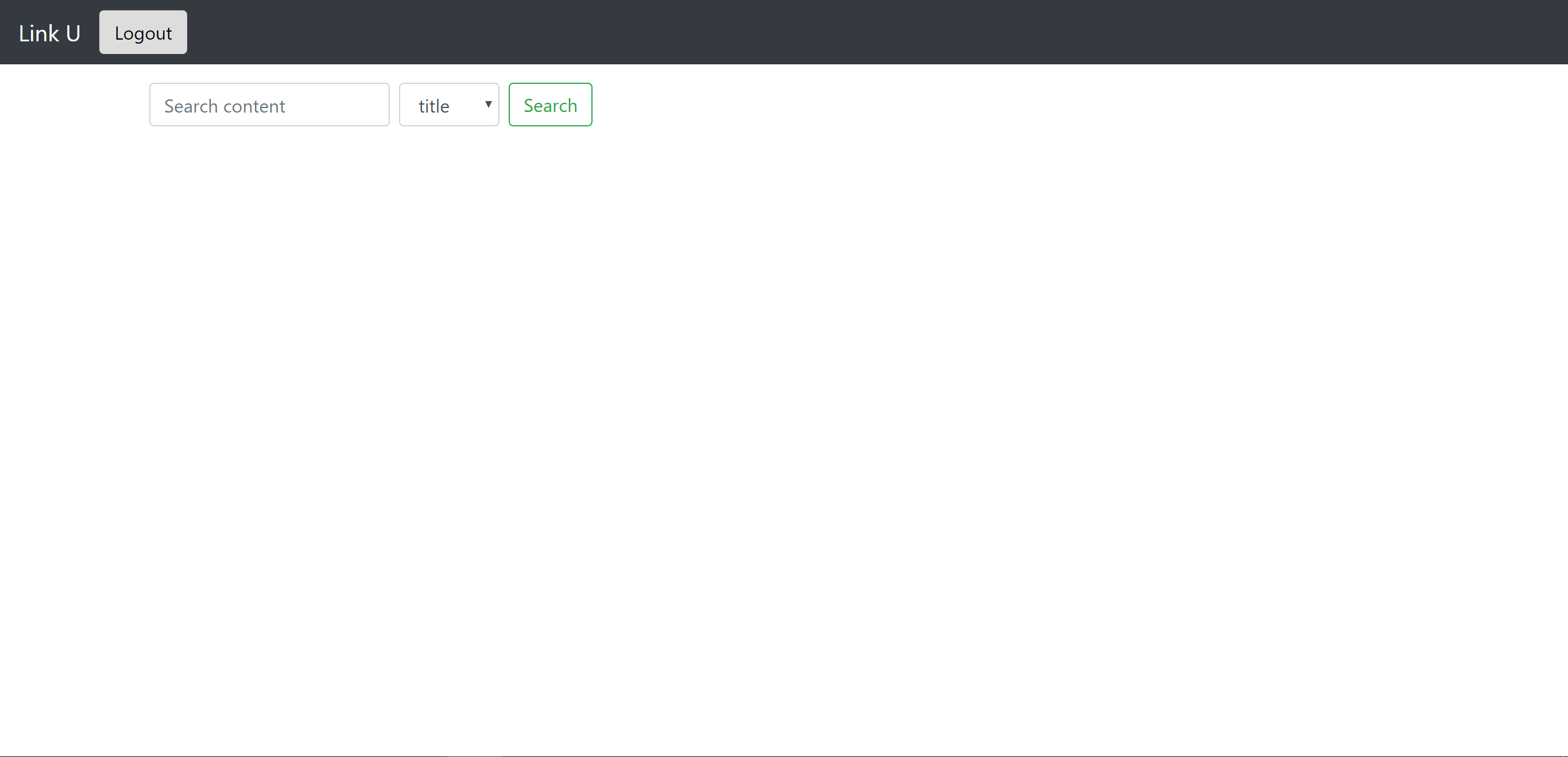
Similar to the implementation of Employer login page, except this will direct to a different page to check the credential with a different table.

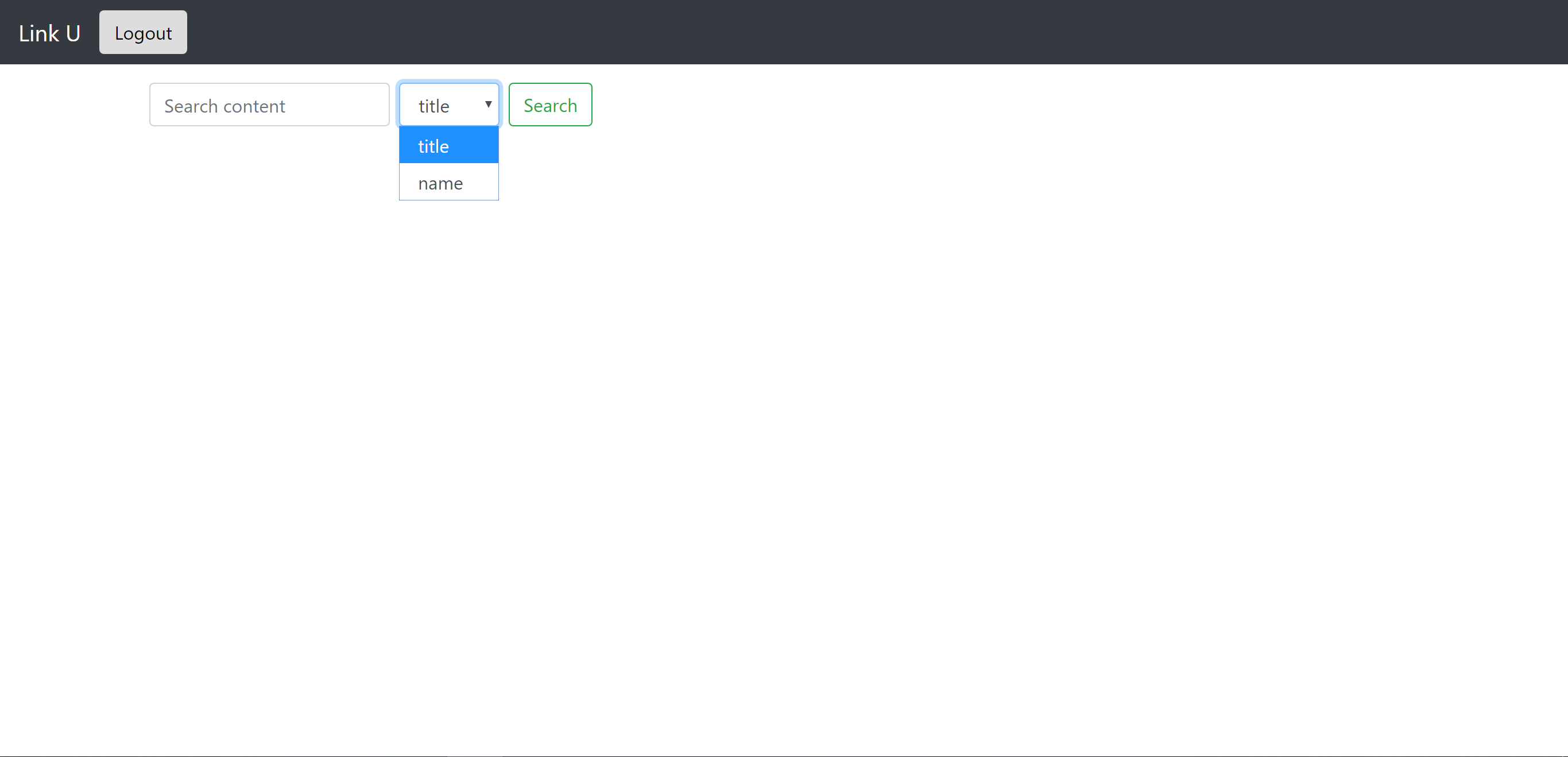
Login confirmation page (Candidate):

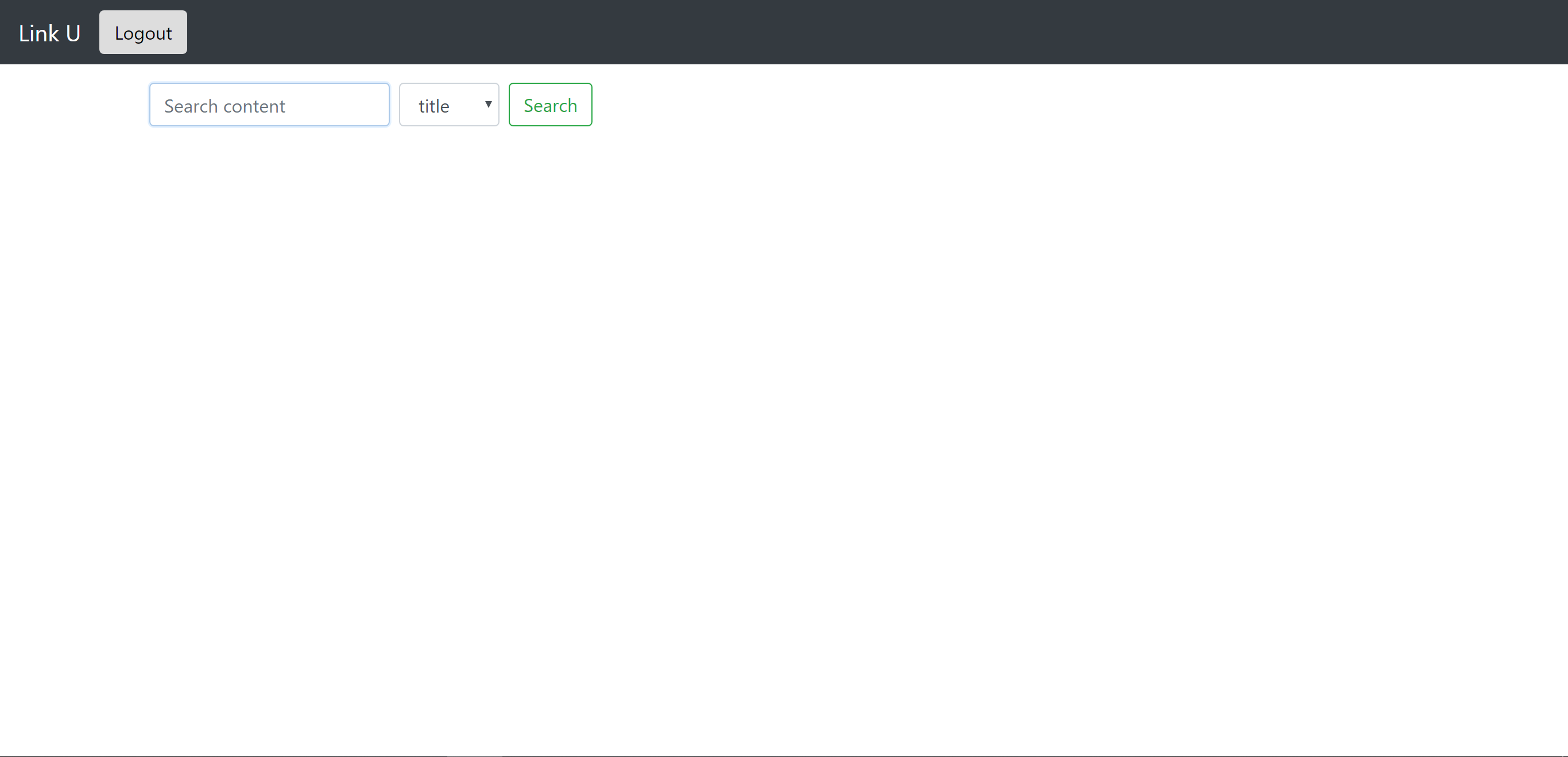


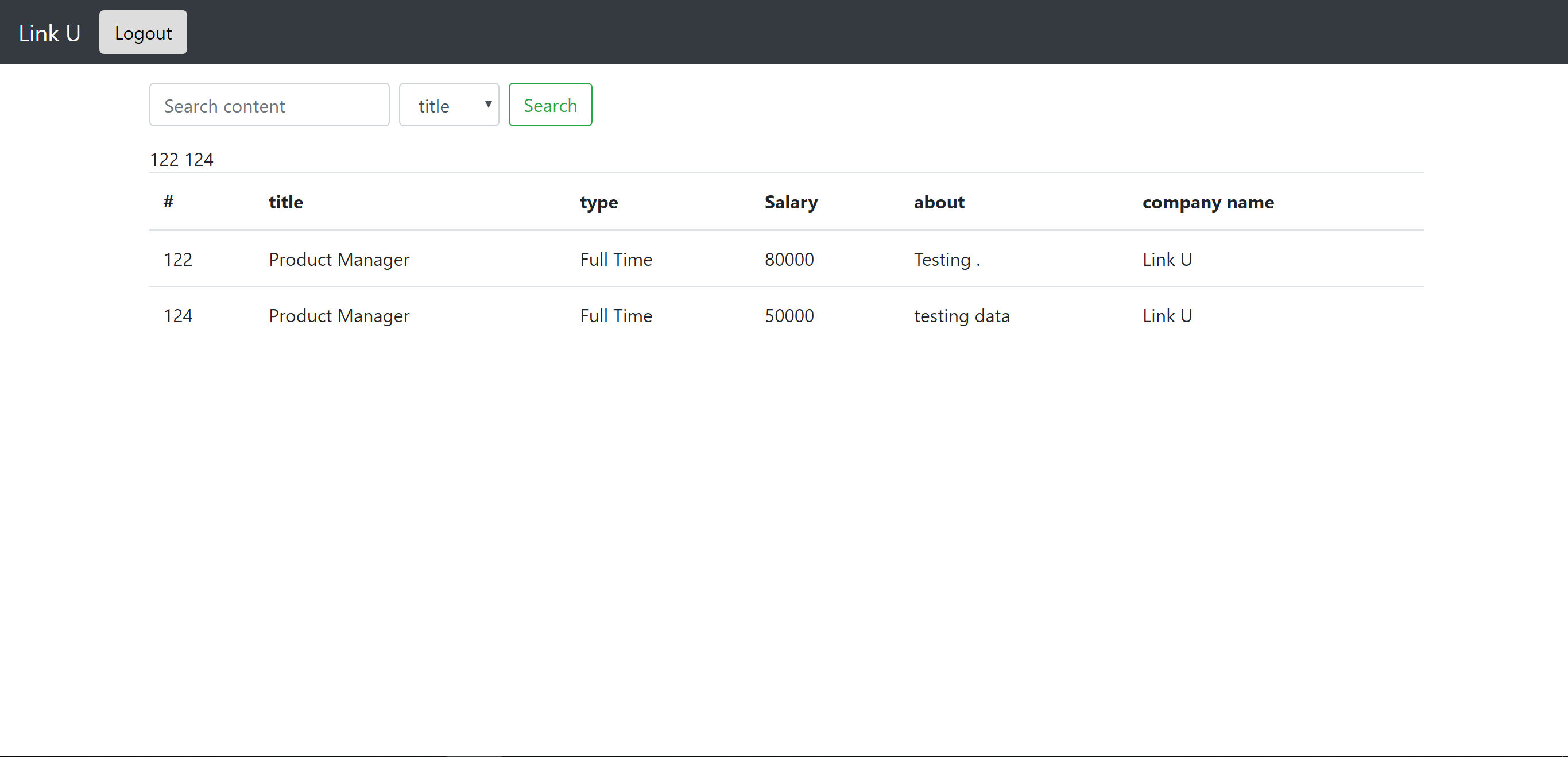
This will check the credentials to see if they are valid. If so, a button that links to a job searching page will be provide along with confirmation text. Other implementations are similar to Employer login confirmation page.

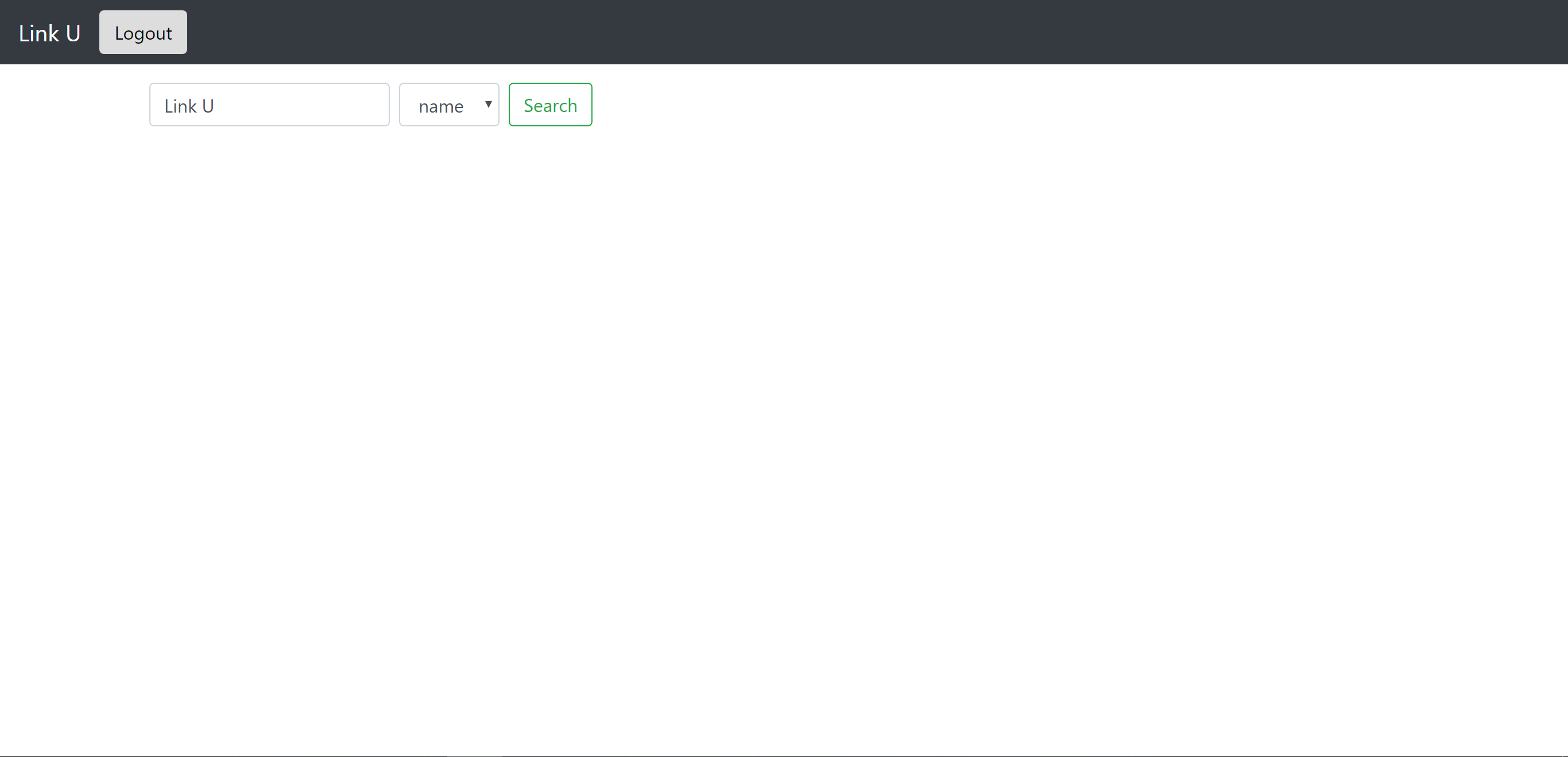
Job Search Page:

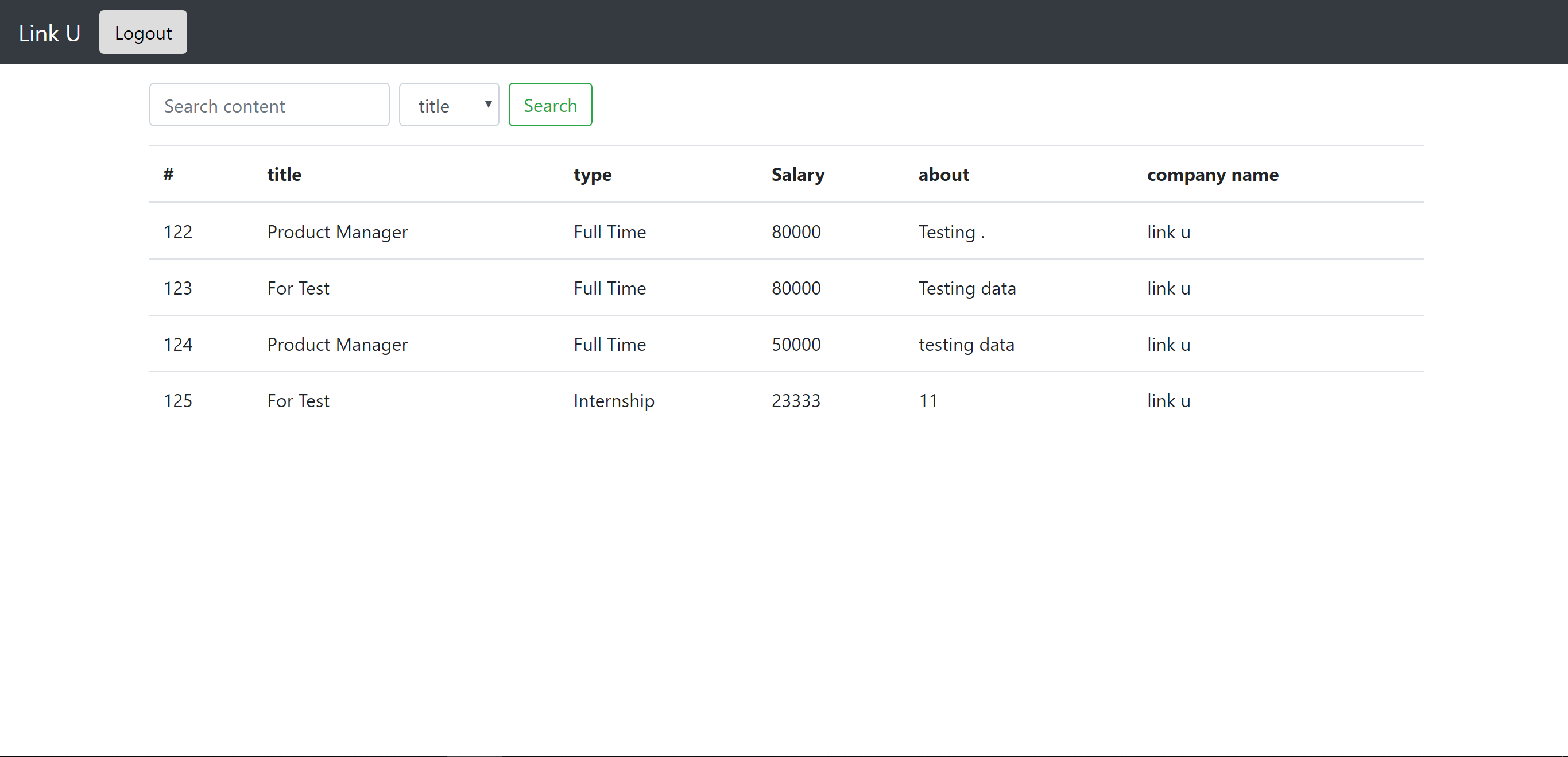








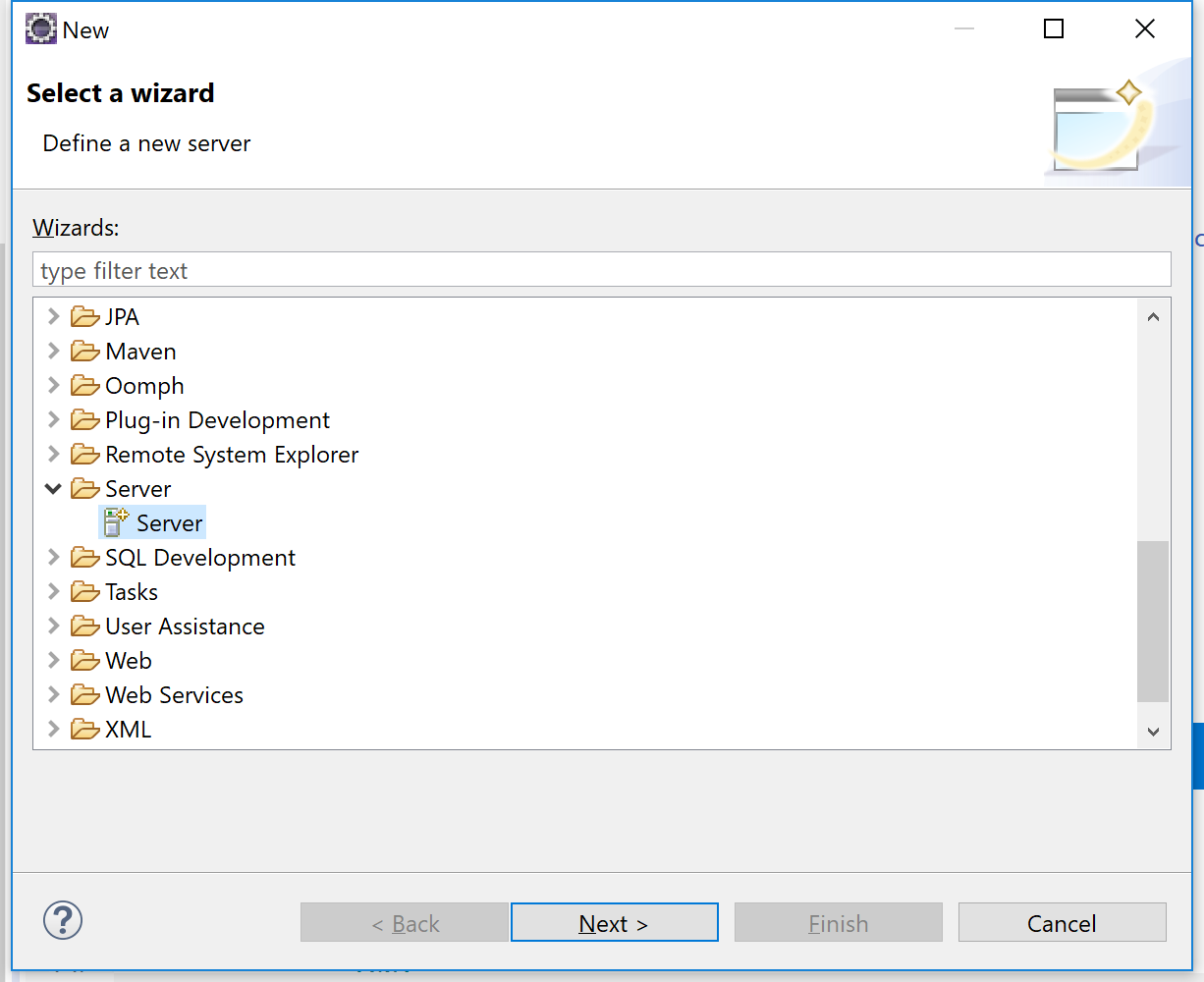




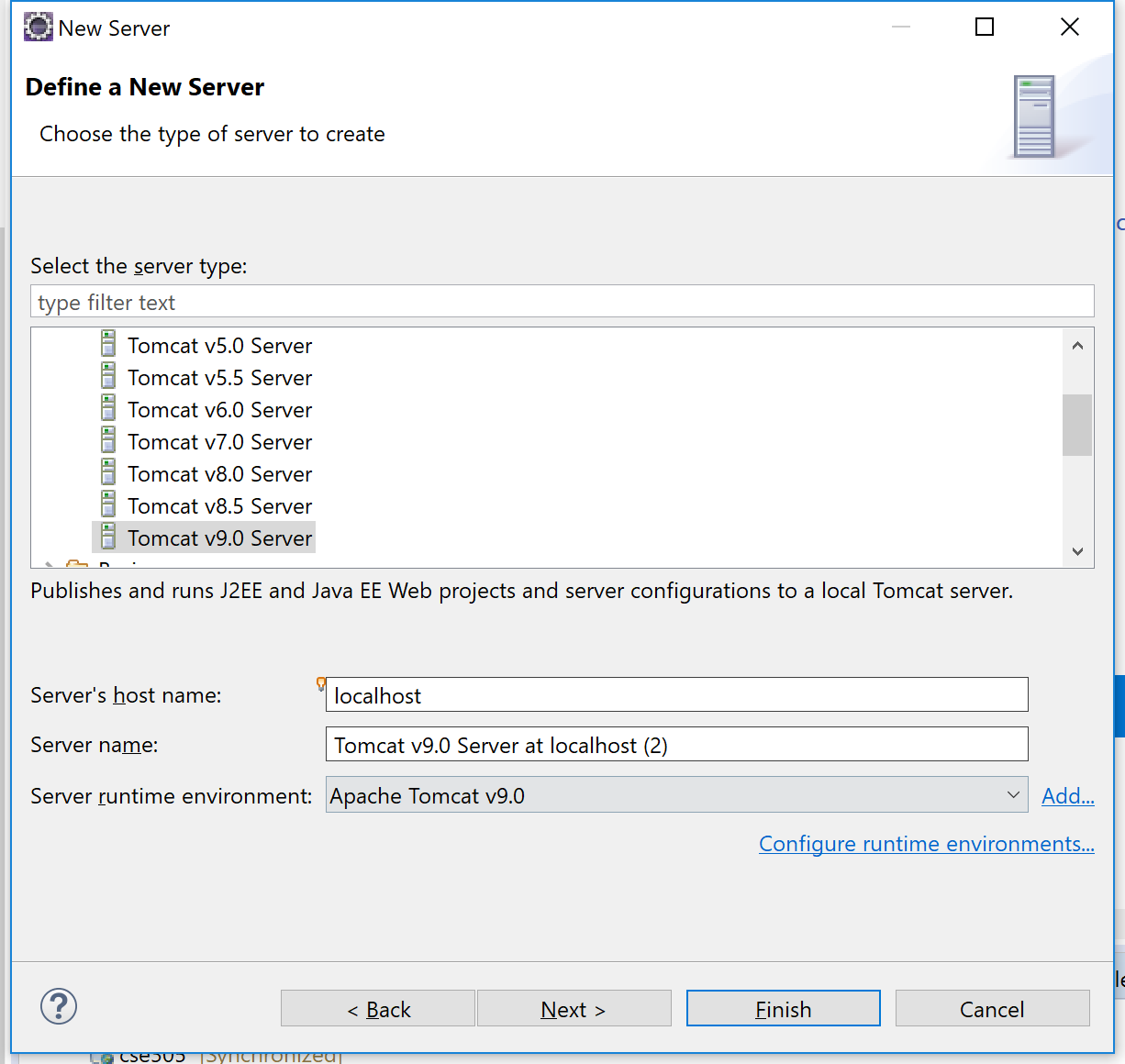
Providing a text field to get the search text from user, along with a selection list that allowing user to pick search with “title” (Job title) or “name” (Company Name) if valid information is provided. A list of job that with the provide title or company name will be shown with a table.

**Install Apache Tomcat on Eclipse EE**

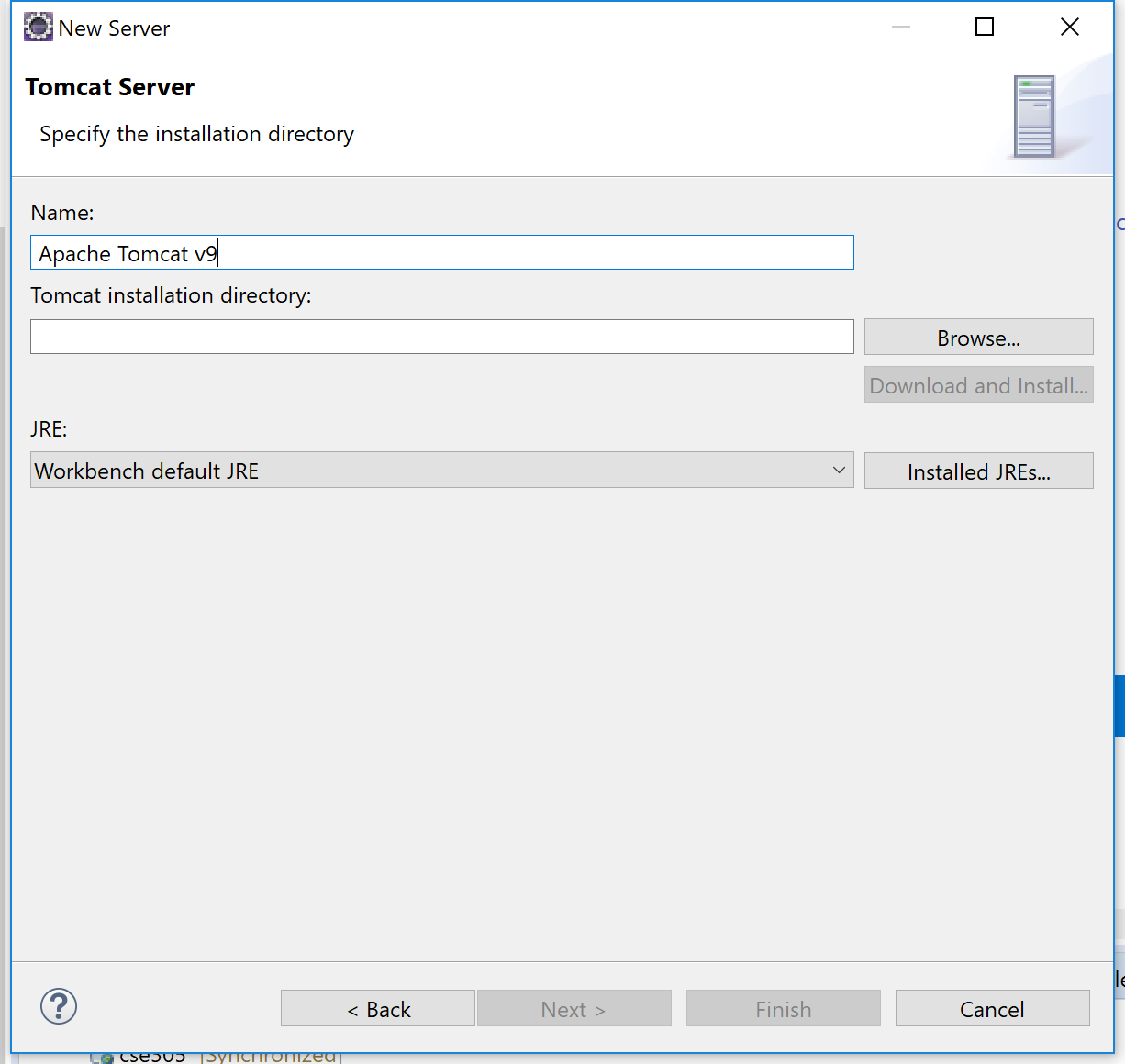
1. From the Eclipse main menu choose **File > New > Other...**
2. Select **Server > Server**.
3. Click **Next**.



1. Select **Tomcat vx.x Server**
2. Click **Next**.



1. Browse to the folder of your Tomcat installation.



1. Select **Finish**.