Recruiters everywhere are always looking for talent at the college level to fill internship and entry level roles. Since not all employers have the resources to actively visit schools searching for talent, universities have provided Career Sites that enable employers to post jobs along with contact info, so that students can apply. The problem with these career sites (including FIU SCIS Career Fair) are:

• Employers are not able to virtually interview a student
• Employers are not able to look at student profiles, proactively searching for talent in their own time
• Students do not get notified of new job postings
• Lack of communication between employers and students

The system shall...
- Allow students and employers to register with the system.
- Allow students and employers to take part in a video interview.
- Allow students to upload a resume.
- Allow students to associate skills to their profile.
- Allow students to integrate with their LinkedIn account
- Allow students to apply to open job postings with a cover letter.
- Allow students to reply to an employer’s message.
- Allow employers to post jobs.
- Allow employers to close a job posting.
- Allow employers to associate skills to a job posting.
- Allow employers to search for students based on skills.
- Allow employers to view student profiles.
- Allow employers to send messages to students.
- Allow employers to give students a “virtual handshake” to show interest in the student.
- Allow an administrator to disable an account.
- Allow an administrator to close a job posting.
- Allow an administrator to validate an employer registration.

The system shall allow an administrator to validate an employer registration.

The system shall allow employers to close a job posting.

The system shall allow employers to send messages to students.

The system shall allow employers to give students a “virtual handshake” to show interest in the student.

The system shall allow an administrator to disable an account.

Testing tool: Selenium.
Selenium is a suite of tools to automate web browsers across many platforms. It can run on many browsers, operating systems, and can be controlled by many programming languages and frameworks. Testing was performed using Firefox browser and Java.

### System Architecture

1. **HTTP Request**
2. **YII Router**
3. **Action**
4. **Call**
5. **Controller**
6. **Model**
7. **HTML**
8. **View**

### Object Design

![Object Design Diagram]

### Implementation

### Verification

<table>
<thead>
<tr>
<th>Test Case ID</th>
<th>Purpose</th>
<th>Test Setup</th>
<th>Actual Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>VJD-TC39</td>
<td>Test Notification to Accept Video Interview Functionally</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Screenshots

![Video Interviews Screenshot]

## Acknowledgement

The material presented in this poster is based upon the work supported by Tomer Doar. I am thankful to the help that I received from my group members, Andres Gonzalez, Diego Perez, Emmanuel Corvo and Justin Korah.