**Hire Talent. Final Report**

Ricardo Luna, Cindy Torres, Erick G. Guzmán y Mario A. Ruiz

Tecnológico de Monterrey Campus Chihuahua

Chihuahua, México

a01561680@itesm.mx, a01566153@itesm.mx, a01566171@itesm.mx, a01187827@itesm.mx

**Abstract**

This report gathers all the activities performed during Hire Talent project’s development. Everything involved in the project includes using Scrum, XP, Jira, GitHub, and XCode. Scrum and XP for the methodology to carry on the project, Jira for the management of the project, which means that it was used sprints, backlogs, sprint boards and Burndown charts. The team simulated a real case scenario of the development of an app for iPhone, made in XCode, where its main purpose is to connect students and employers for internships. All the process of the development with agile and brief descriptions with screenshots of the functional prototype created will be found in this document.

**Keywords:** Final Project, SCRUM, Swift, Hire Talent, XP, software engineering

1. **Introduction**

SCRUM is a framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value. SCRUM is a simple framework for effective team collaboration on complex products. This project was required to use SCRUM so a lot of terms mentioned in the present document refers to tools commonly used in this framework.

The required application was developed in the Swift programming language and basically it’s an app that matches students with interested enterprises based on their career and skill set. This project is not only about the app that the end users will be able to use, but rather, a combination of all the code, database, reports and documents generated during the software development lifecycle.

1. **Project Scope Definition**

The main objective of the project was to develop a mobile application implementing SCRUM, so in this way the student could understand and use the specified development platform for creating mobile applications, and appreciate and use the specified framework for the software development. Most relevant details about the project are shown below:

* It is required to create a Product Backlog based on user stories.
* Each user story should contain a description, acceptance criteria, story points and priorities.
* UX mockups should be created.
* MVC pattern should be employed.
* A REST Web Service or a Remote Access Database should be used.
* SCRUM should be used for the planning and reporting of the project.
* Some XP practices for the development of the application should be applied.

1. **Solved Problem with the App**

Hire Talent is a mobile application that matches students with interested employers based on their career and skill set, this idea was born because it is believed that there can exist an easier way to communicate students with potential employers. A lot of universities have their own and exclusive platform, and that narrows the number of employers or companies available for their students. This app pretends to centralize all the internship (or even job) offers to make them available for all students signed up in the platform, thus giving a wider range of options to both students and employers.

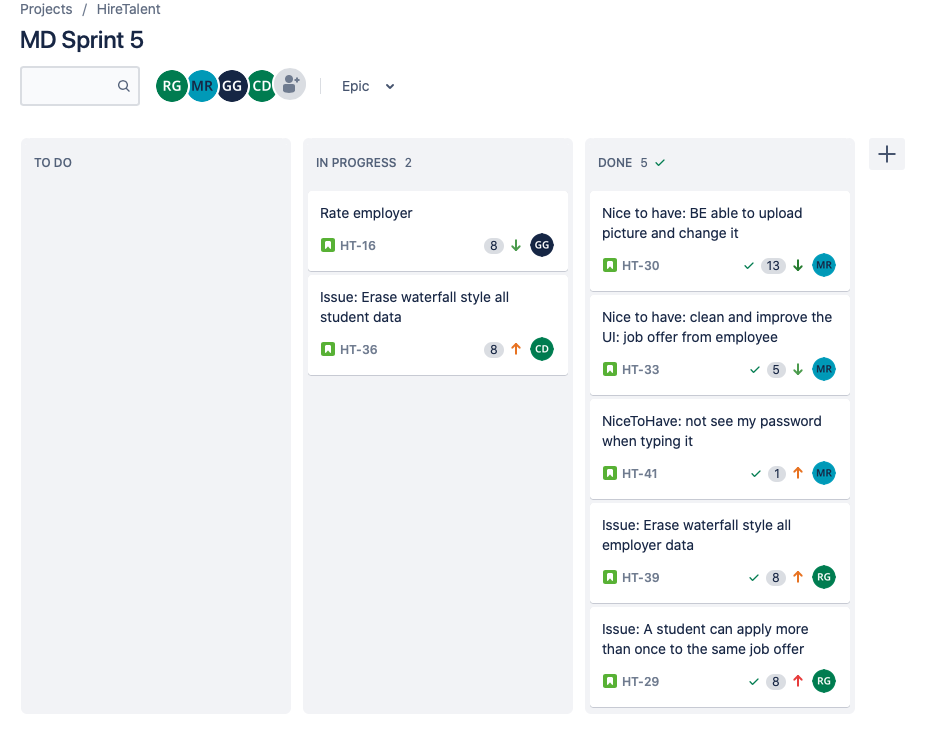
The following list describes the most important considerations that the team had during the app’s development:

* A user can be a student or a employer (not both).
* An employer can be associated with only one enterprise.
* A student should define their major and their set of skills.
* Both student and employer have in their profiles the most basic information. They should be able to edit it whenever they want.
* The job offers are directly associated with the employer, not with the company.
* Employer shall be able to add, edit, close, delete, and see their job offers.
* Employer can see a list of interested students for each offer.
* Employer shall be able to choose and notify the students when they are interested in them.
* Student shall be able to view the job offers that are opened and apply to them.
* Student shall be able to filter job offers based on their major.
* Student shall be able to view the employers who were interested in contacting them (and the related job offer).
* Student shall be able to rate the Employer.
* Both the employer and the student shall be able to delete their accounts and all their information in the system.

1. **Description of the work performed**

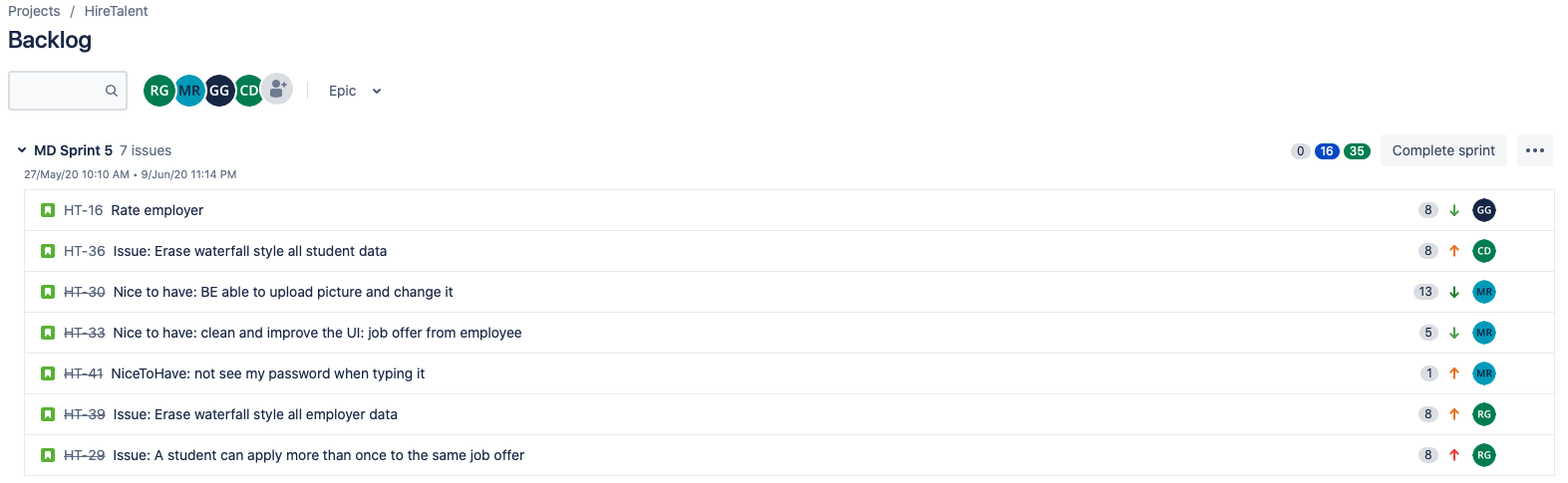
First of all, the team had to get more familiar with the SCRUM framework and despite there was certain knowledge about it, some elements were not well understood at all. Since SCRUM was the driving force behind the organization for the team, some of its recommended practices and tools had to be implemented for this project. The most relevant of them are listed below:

* **Stand up meetings:** The team, in accordance to SCRUM, organized daily a morning meeting from 3 to 10 minutes in length to talk about the individual progress of each member, any roadblocks affecting someone’s progress and what each member of the dev team will work on that day.
* **Product backlog:** The team added all the user stories needed to be completed in order to finish the project.
* **Sprint board:** This tool was used to monitor the status of the user stories during the sprint. There were 3 different possible status: to do, in progress, and done.



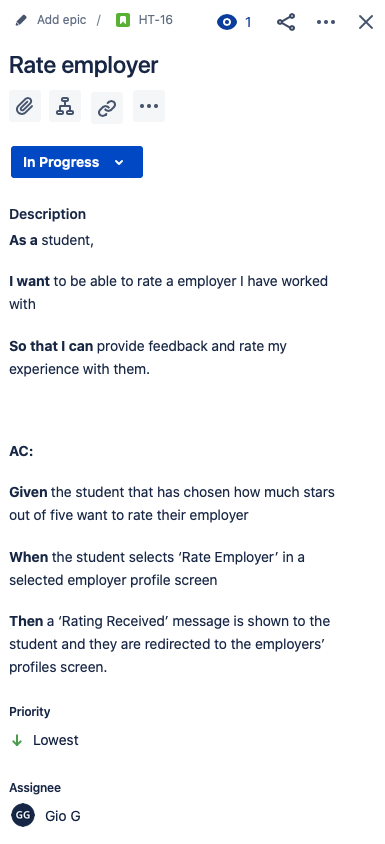
**Figure 1.** Sprint Board Example.

* **Sprint backlog:** For each sprint, the team added all the user stories to be completed in order to deliver a product demo of the highest possible value.



**Figure 2.** Sprint Backlog Example.

* **User stories:** This concept is referring to the most basic unit of a software project. They contain a description, acceptance criteria, name, story points, priority and other information relevant to each specific requirement. There was done a user story for each component of the software product.



**Figure 3.** User Story Example.

* **Burndown chart:** This toolshows the amount of work that has been completed in an epic or sprint, and the total work remaining. This was a useful and powerful tool to track the progress and efficiency that the dev team had in each sprint.

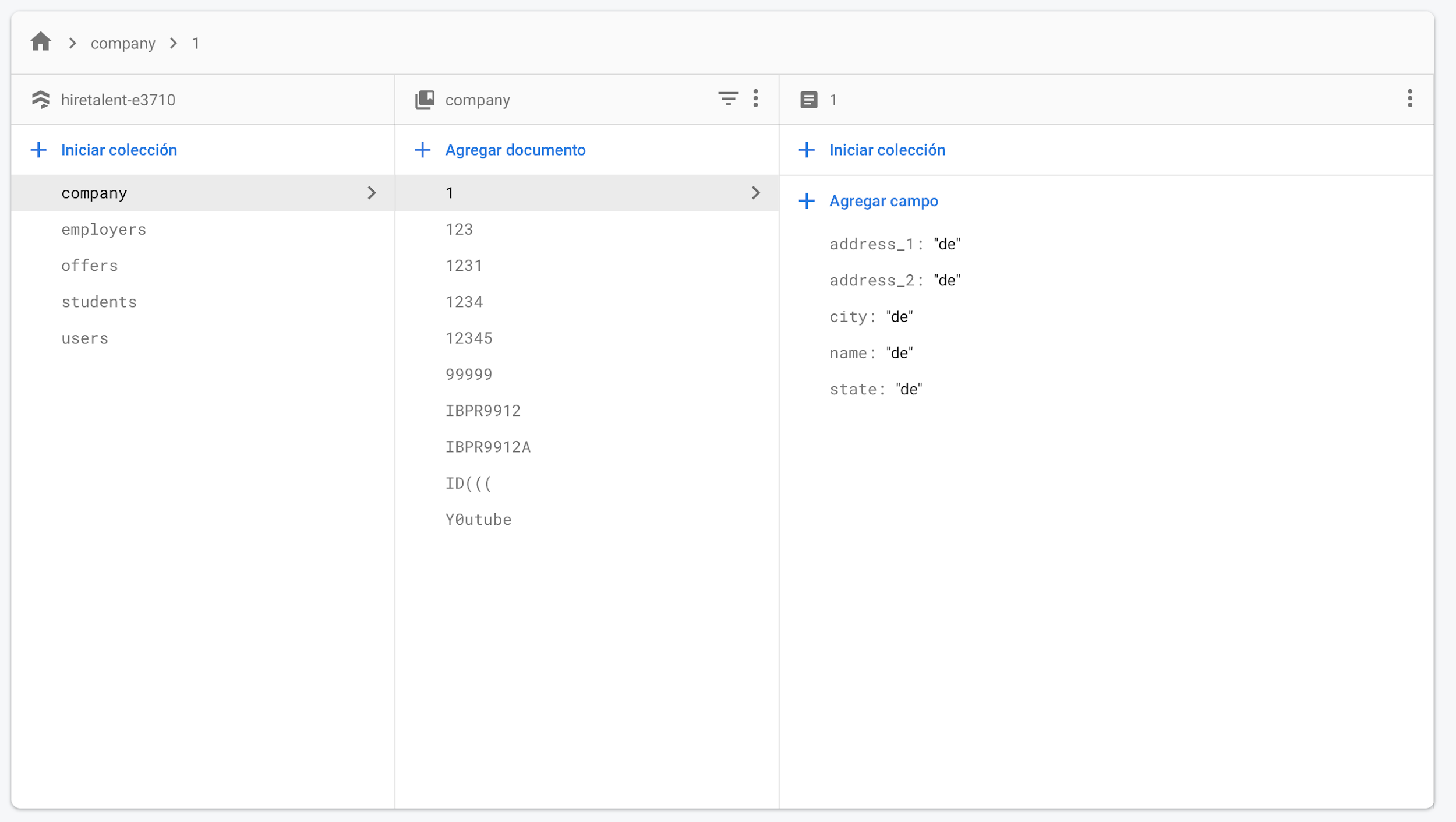


**Figure 4.** Burndown Example.

If there is any doubt about the concept “sprint”, here is a definition that should clarify it: a sprint is a short, time-boxed period when a SCRUM team works to complete a set amount of work. Sprints are at the very heart of SCRUM and agile methodologies, and getting sprints right will help your agile team ship better software with fewer headaches [1].

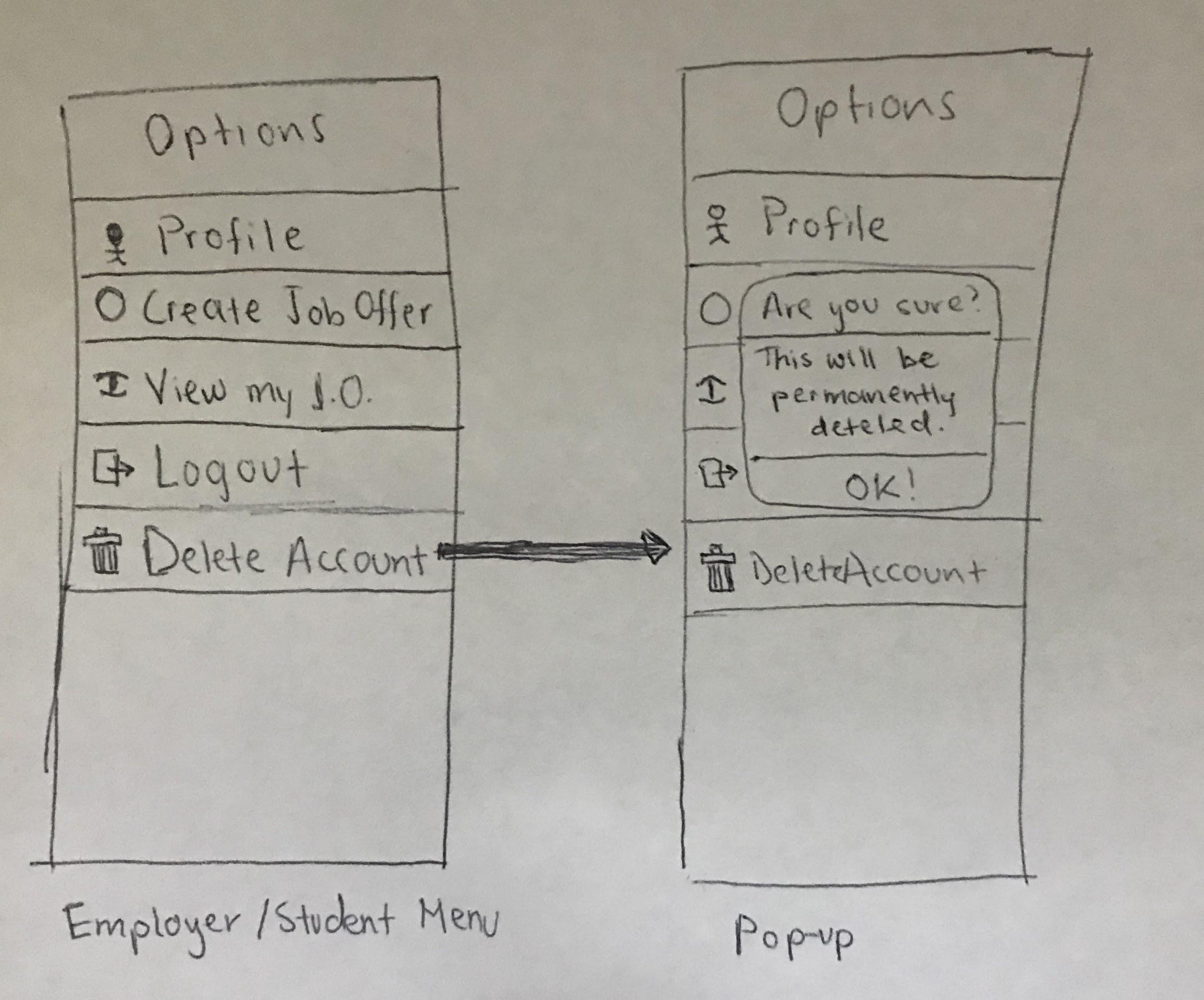
Before starting to build the mobile app, it was required to define the programming language and control management system. The programming language would be a very important and transcendental decision because it would represent the technology to be implemented in all the project. Nevertheless, there wasn't any doubt about it and it was accorded to use Swift. Swift is a powerful and intuitive programming language for macOS, iOS, watchOS and tvOS but in this project it would be used just for iOS. Taking a decision about the control management system to be used in the project neither was a difficult resolution, the team decided to make use of git and GitHub thanks to the previous experience had with them.

After the definition of the programming language, the team had to decide the database to be used, and after some proposals of the team members it was agreed that it would be Firestore. That was a great decision because there were already interest about learning Firebase technologies, but the team members had not had the opportunity and time to do that. Additionally, to this choice, it was necessary to use Cocoapods to manage the dependencies for the mobile app, and once that all of them were included and some other components in Firebase were configured, the team had already built successfully the connection between the mobile app and a remote access database.



**Figure 5.** Firestore Database.

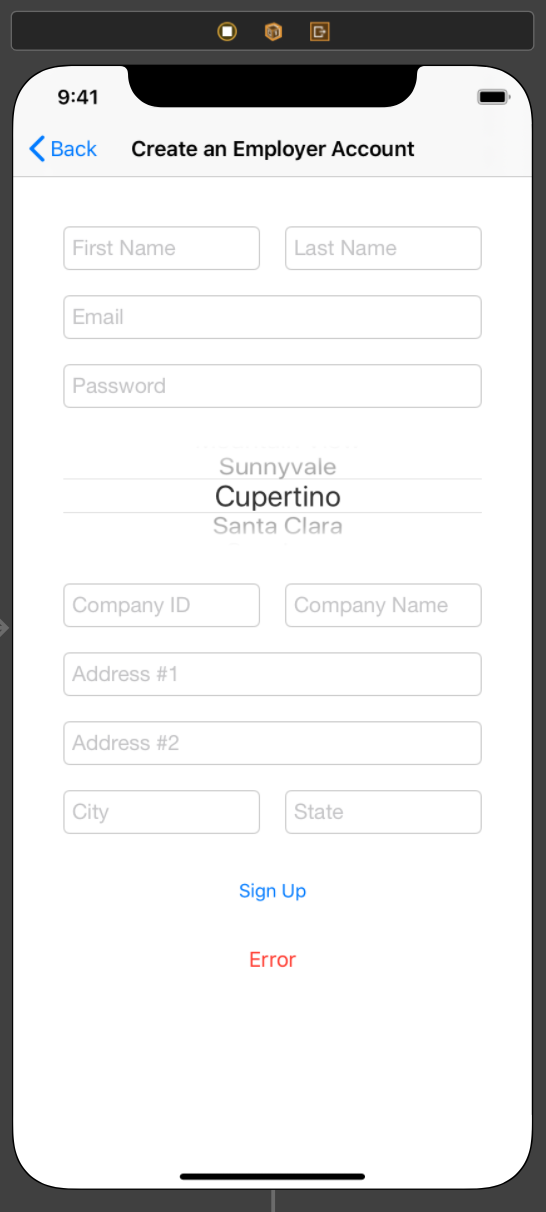
During the software development process the UX mockups were an important factor because they were useful to define high-profile visual design drafts used to represent the structure of information, visualize the content and demonstrate the basic functionalities in a static way. The most of the times the UX mockups were done in paper and actually that was a good idea because any desired change could be done in less than 5 minutes and without affecting some already done work.



**Figure 6.** UX Mockup Example.

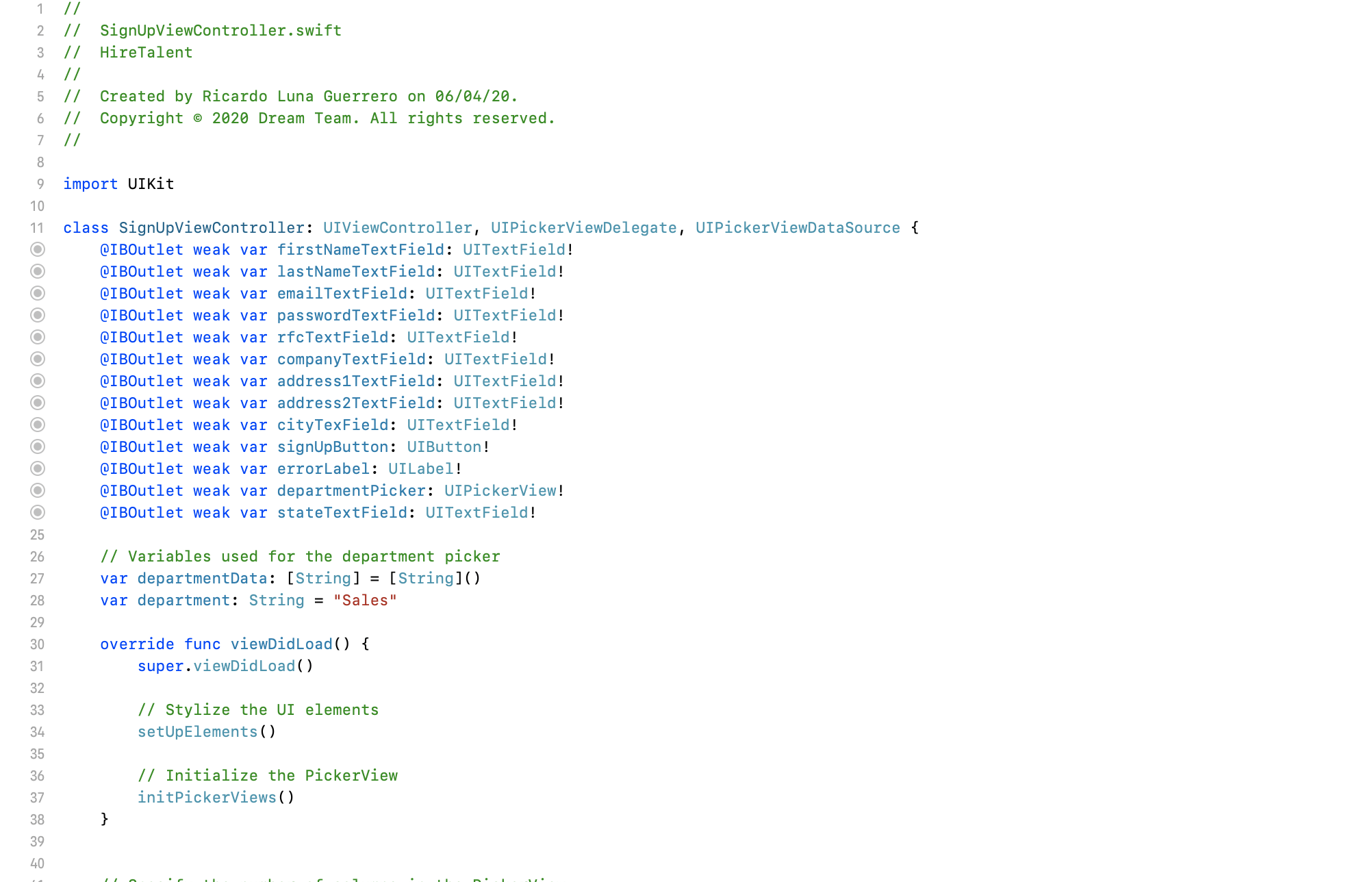
In the course of the code’s construction, the MVC pattern was very helpful. MVC is short for Model, View, and Controller. MVC is a popular way of organizing your code. The big idea behind MVC is that each section of your code has a purpose, and those purposes are different. Some of your code holds the data of your app, some of your code makes your app look nice, and some of your code controls how your app functions [2].

For this project the view component is composed by all the storyboards’ elements.

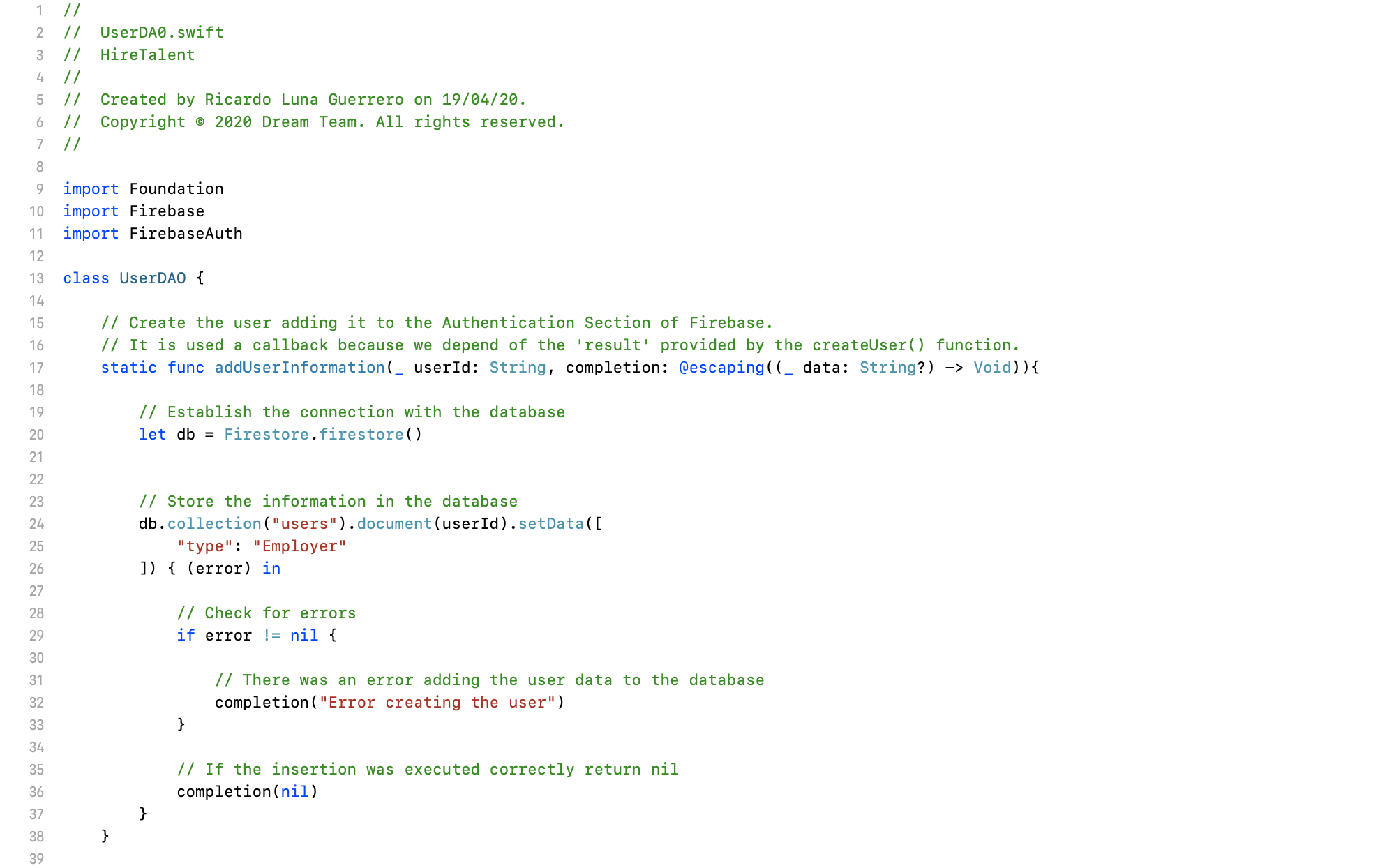


**Figure 7.** View Component Example.

The controller component groups all the view controllers of the application.

 **Figure 8.** Controller Component Example.

The model component is represented by all the existent DAOs, this is the only one that have access to the database.



**Figure 9.** Model Component Example.

Extreme Programming (XP) is an agile software development methodology that aims to produce higher quality software, and higher quality of life for the development team. XP is the most specific of the agile methodologies regarding appropriate engineering practices for software development [3]. The principal XP values applied in this project were:

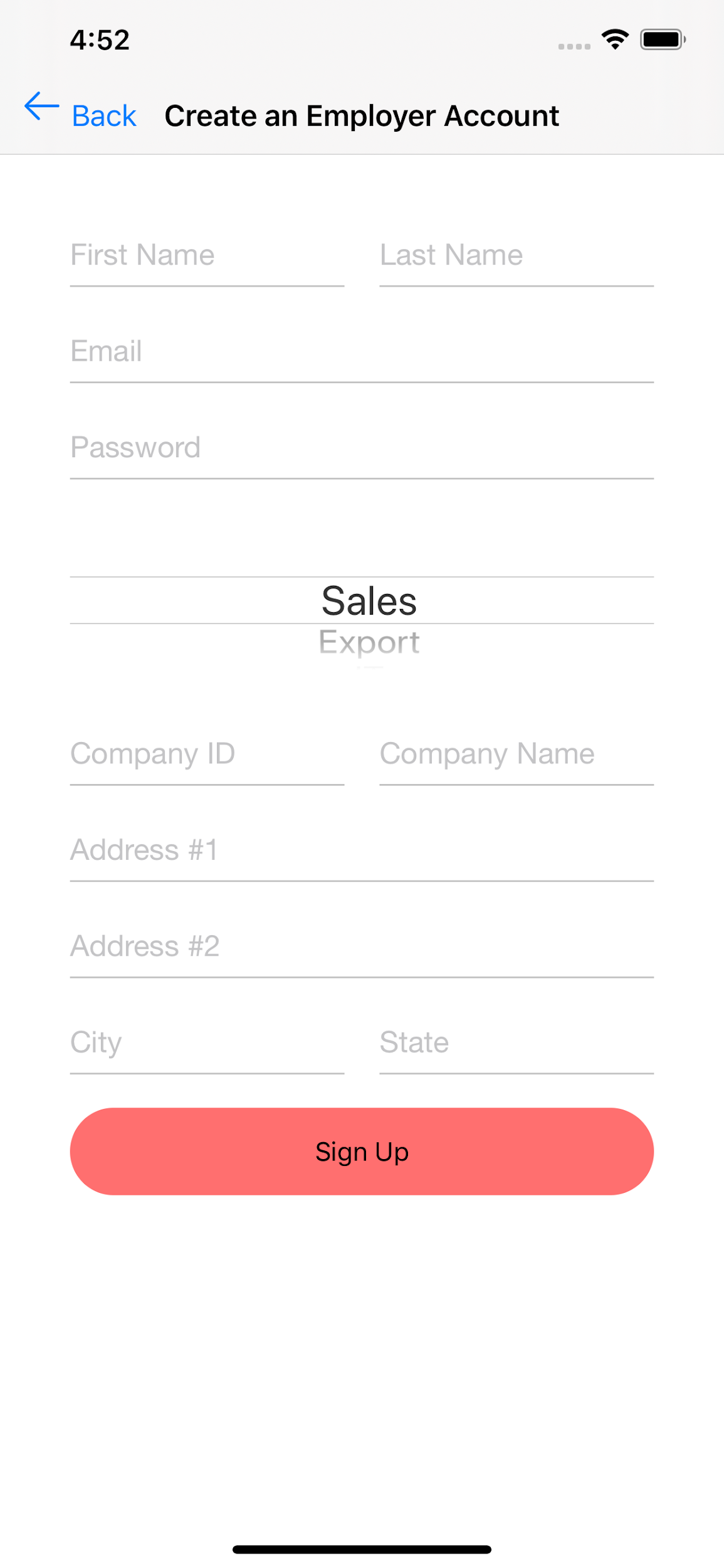
* **Communication:** Team members well understood that communication to transfer knowledge from one member to everyone else on the team is an essential value when software is being developed.
* **Feedback:** The team recognized how important feedback is and its incredible good value as a tool for identifying and giving advices to the rest of the team about their practices.
* **Respect:** Team members respected each other in order to communicate with the others, provide and accept feedback, and work together to identify simple solutions.

Besides the values, the core of XP is its interconnected set of software development practices. Small releases were the most useful XP practice in this project because the delivery of the software via frequent releases of live functionality created concrete value to the customer and made the team feel much more comfortable with the distribution of the activities among the time.

Undoubtedly the small releases are a very powerful tool to help customer to gain confidence in the progress of the project.

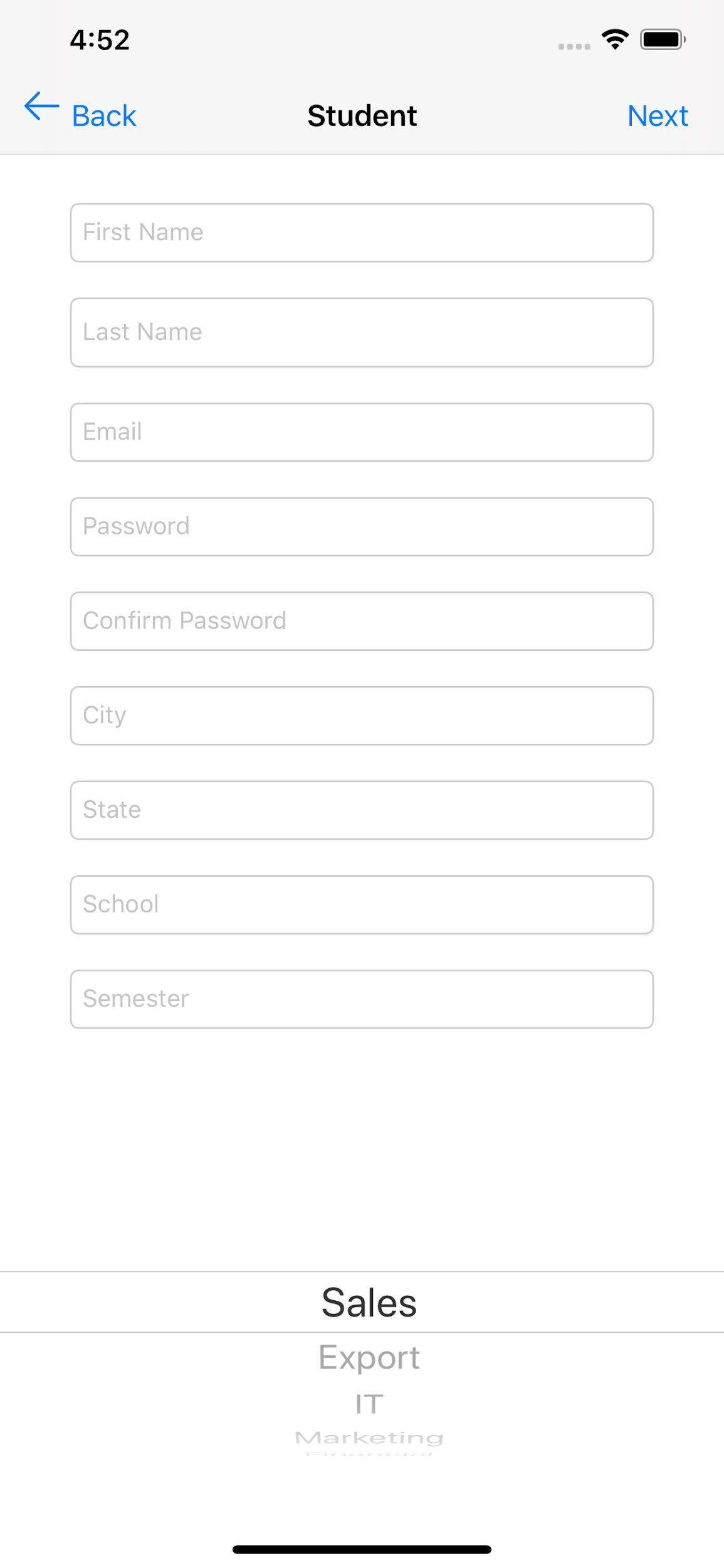
The team had the opportunity to develop the app prototype with all the major functionalities described in the Solved Problem section. In the next pages there are many brief descriptions and screenshots that demo the app’s prototype functionality. There were not included all the functionalities in order to do not make this report too extensive.

In figure 10 there’s the view that let’s an employer create their account.



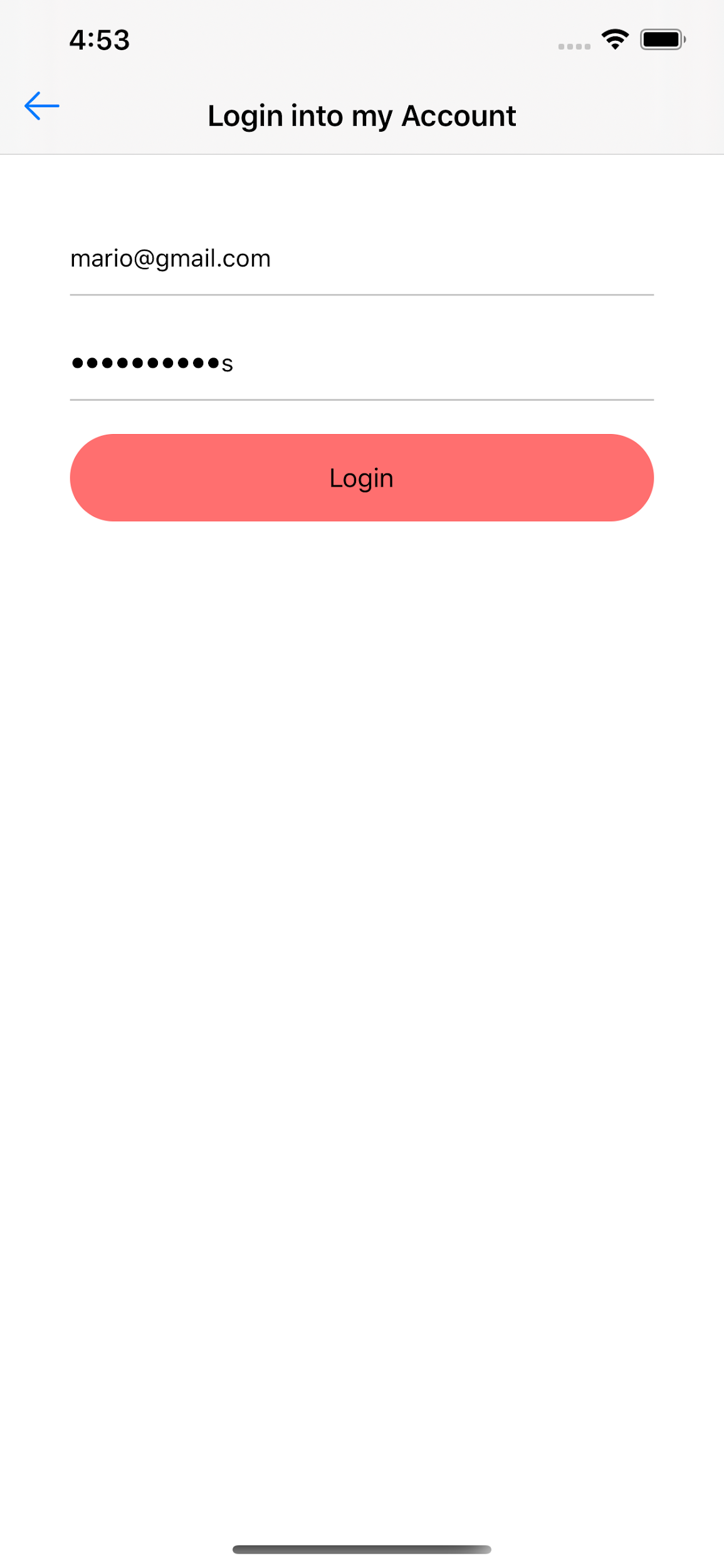
**Figure 10.** Create an Employer Account.

In figure 11 there’s the view to create a Student account. When they select next, they are prompted to add their resume.



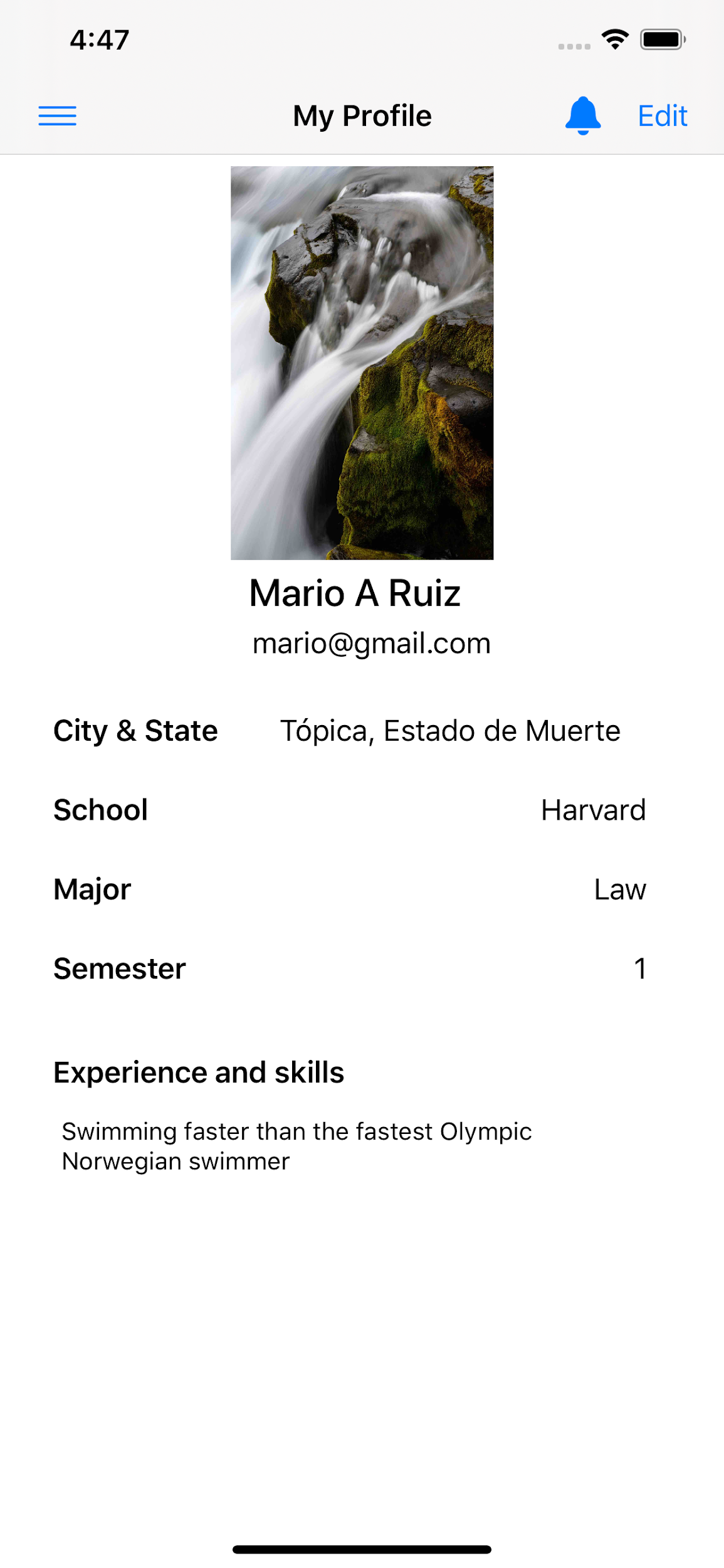
**Figure 11.** Create a Student Account.

In figure 12 there’s the view where the user, either Employer or Student, can log in to their account.



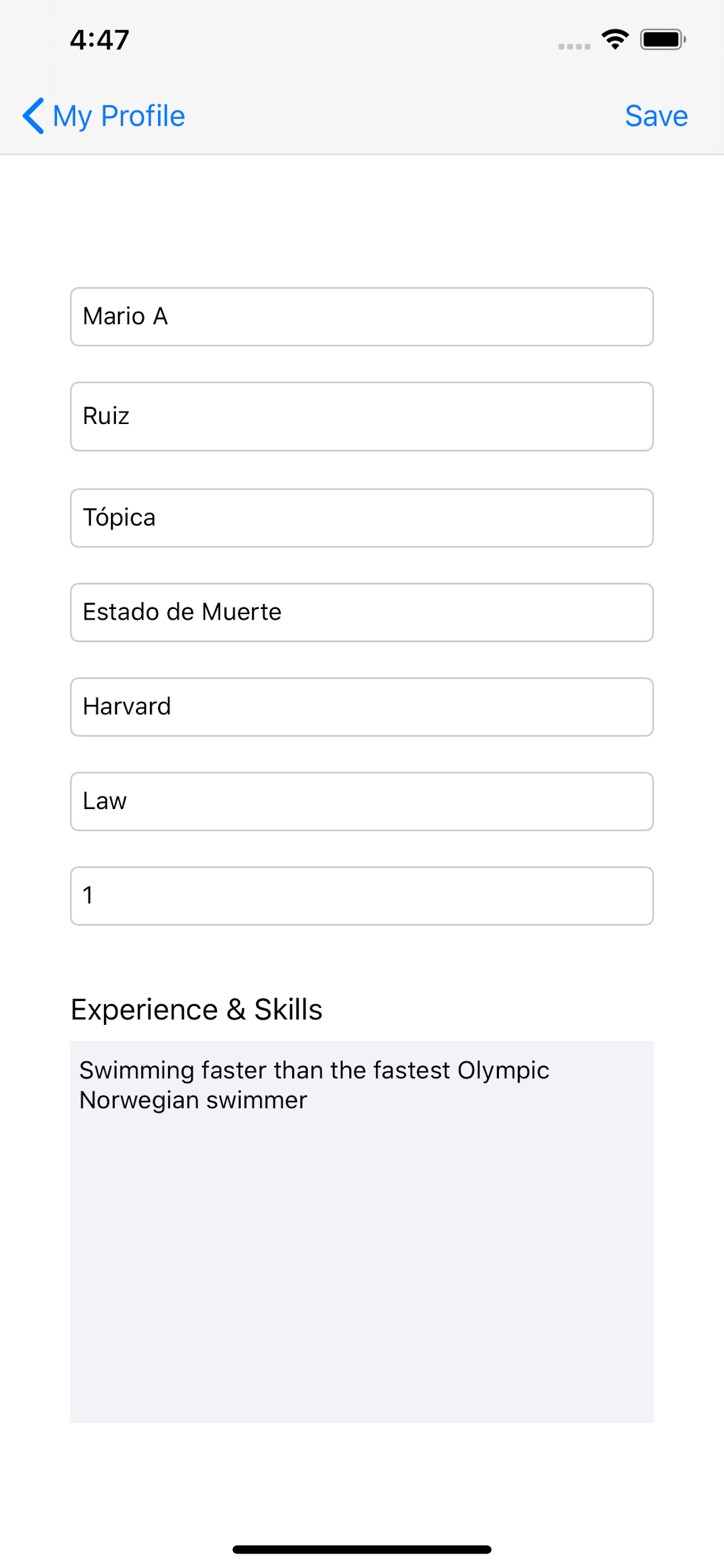
**Figure 12.** Log in to an Account.

After the student logs in, they are taken to their home view that displays their profile. Figure 13 shows the profile, the menu at the top left, and the notifications and edit buttons at the top right of the screen view.



**Figure 13.** Student Profile.

If the Student wants to edit their information, they can do it by pressing the edit button at the top right corner. Then, the edit Student profile view is displayed as seen in the figure 14.



**Figure 14.** Edit Student Profile.

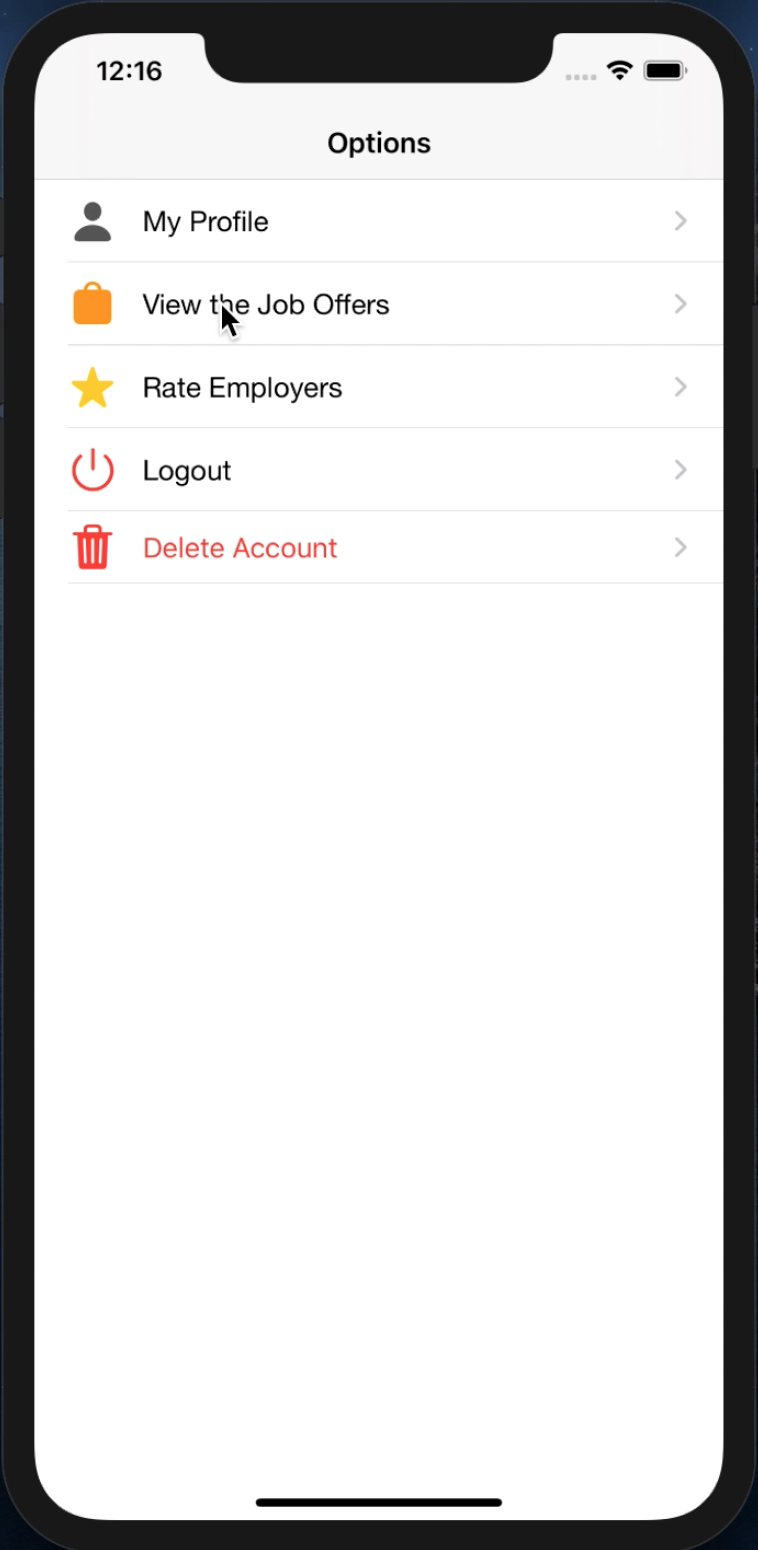
Both the Student and the Employer are able to change their profile picture. They just have to tap on it and select a photo from their photo library as seen in Figure 15. It takes a while to upload to the server, but they are notified to wait.

When they first create their profiles, they have a default picture, later on they can change it.



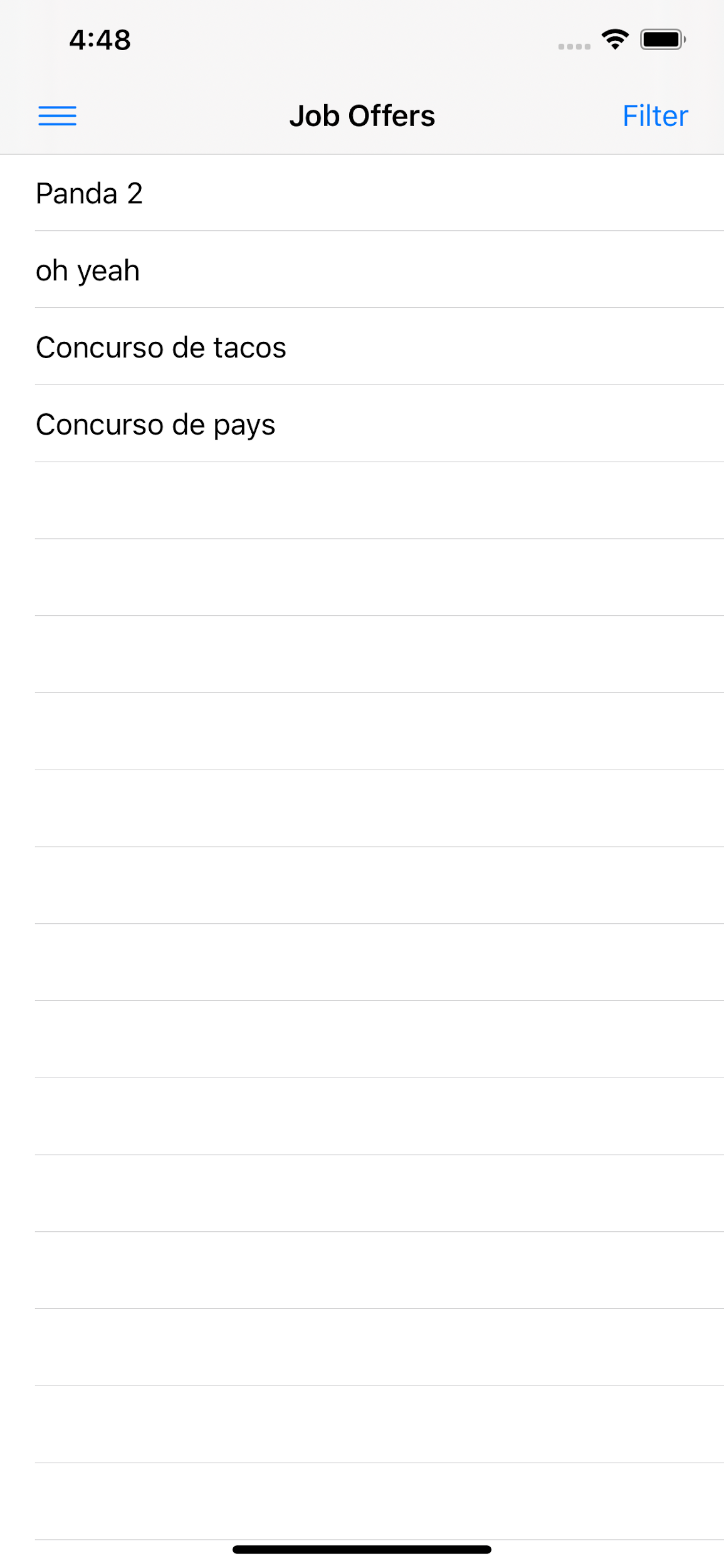
**Figure 15.** Change Profile Picture.

The Student navigates through the menu at the left corner with a view such as the Figure 16. Here they can choose to rate an employer, view open job offers, log out, or delete their account.



**Figure 16.** Student’s Menu.

If the student chooses to see the available job offers, a list of offers is displayed as seen in Figure 17. Here they are able to filter them to their major, and in an improved prototype, city or any other handy filtering capabilities.



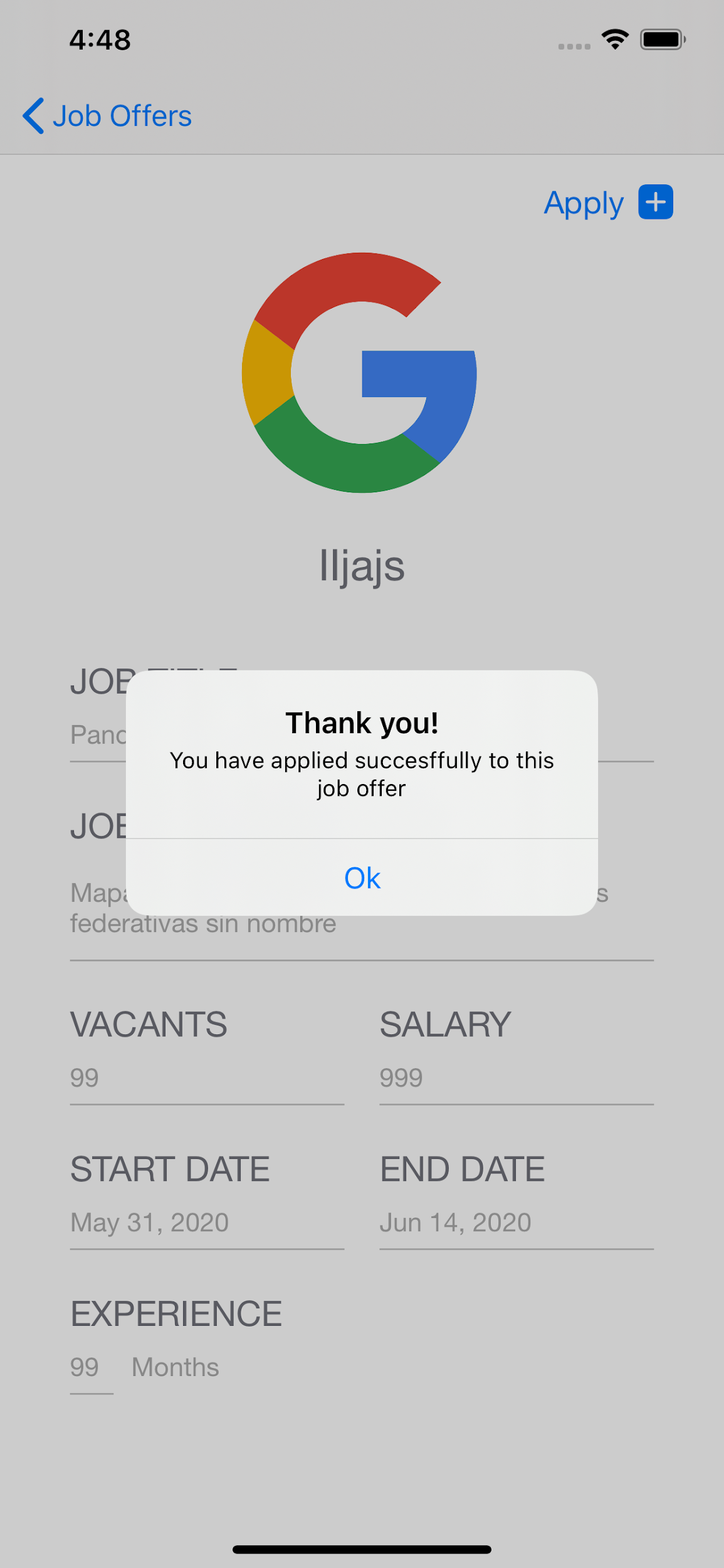
**Figure 17.** Student’s view with open job offers.

The student is able to see details of each job offer as seen in Figure 18. The image is a placeholder for the company logo. In this prototype the functionality to set a company profile picture and change it was not implemented.



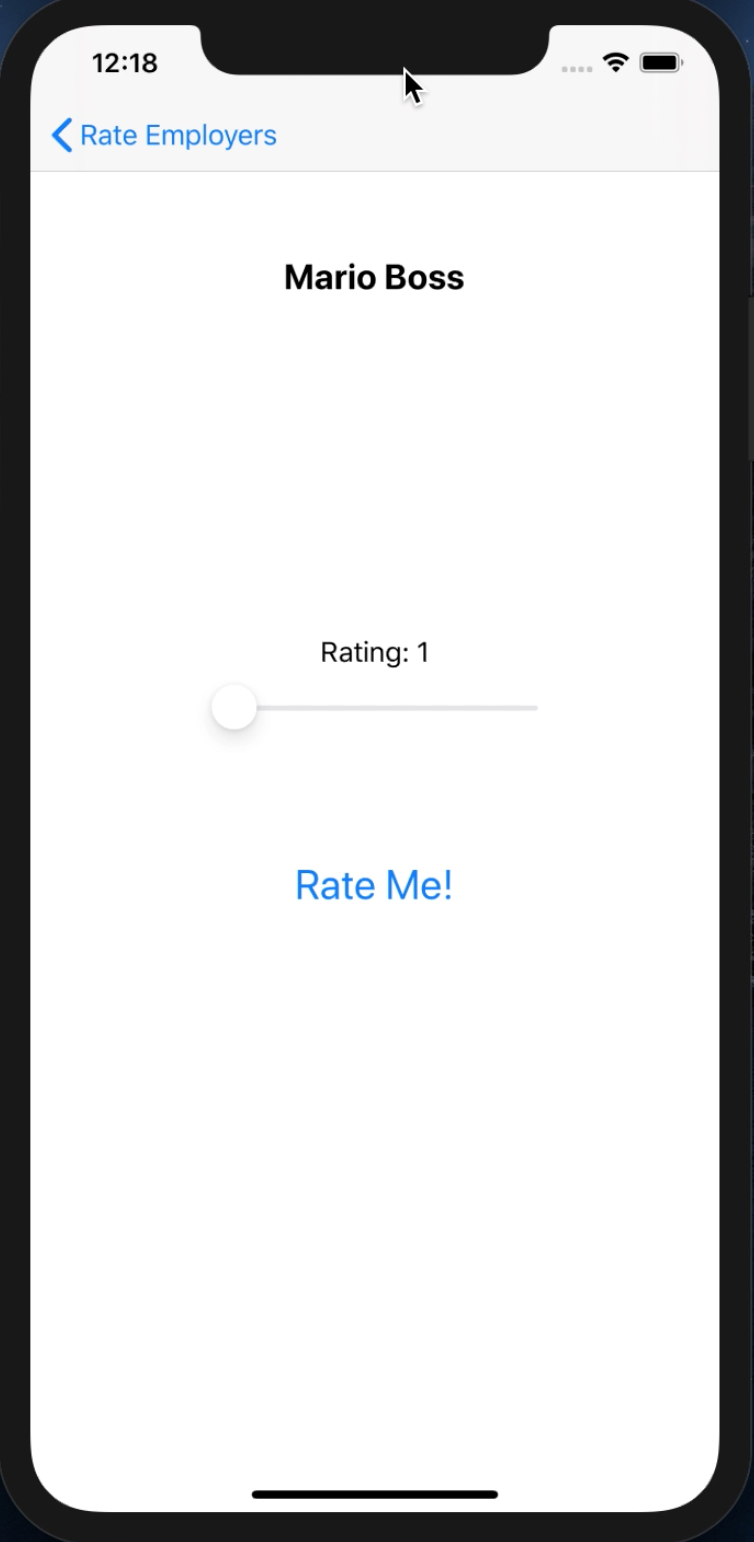
**Figure 18.** Job Offer Details.

They can choose to apply by selecting the button at the top right corner, then a view tells them if it was possible to add them to the list of interested students in the offer as seen in the Figure 19.



**Figure 19.** Successfully Applied to Job Offer.

Finally, speaking of the Student’s actions, they can rate the employer as seen in Figure 20.



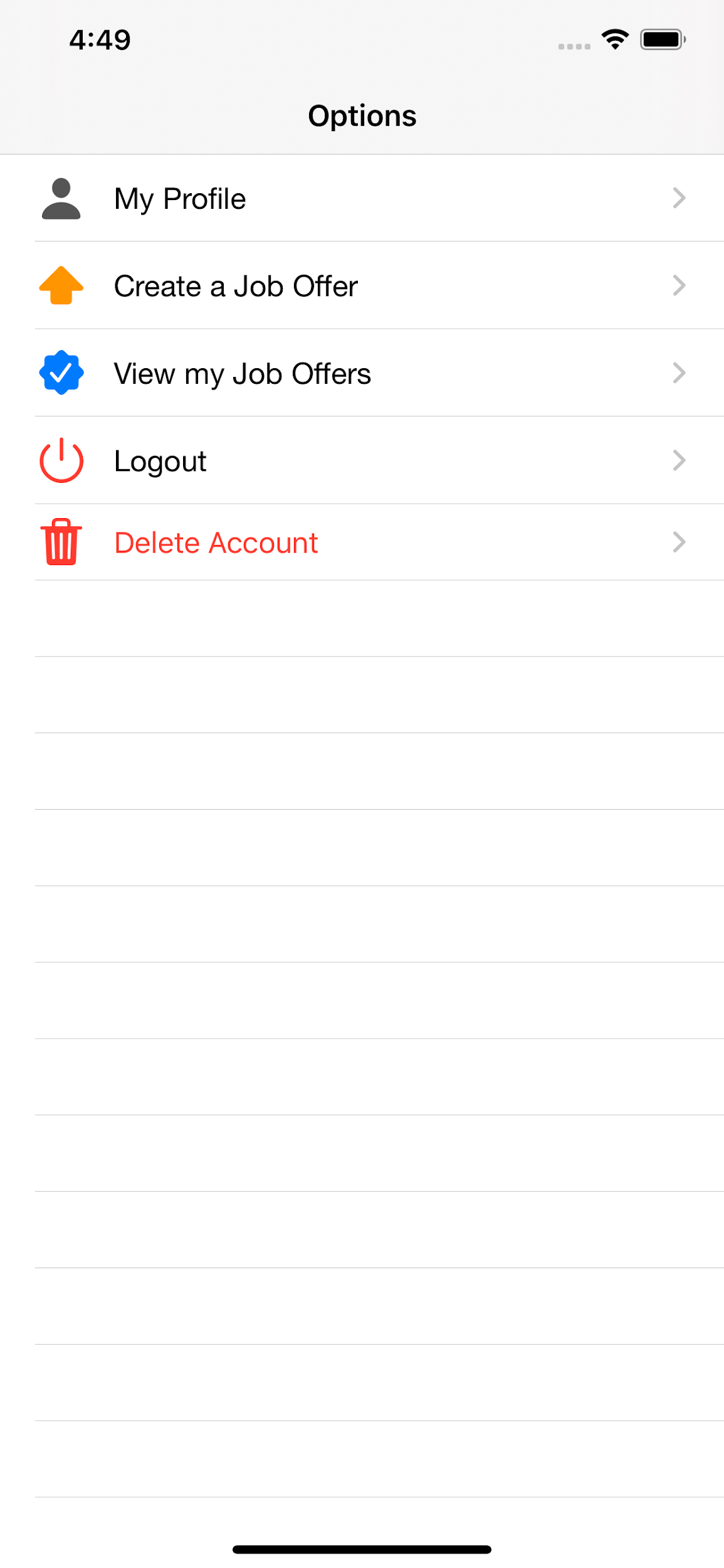
**Figure 20.** Rate an Employer.

Switching over to the Employer. Figure 21 shows their profile, which is the first view they see when they log in. Here, they just have a menu option at the top left corner, and their information. By the way, the functionality to edit their profile has not been implemented for this prototype.



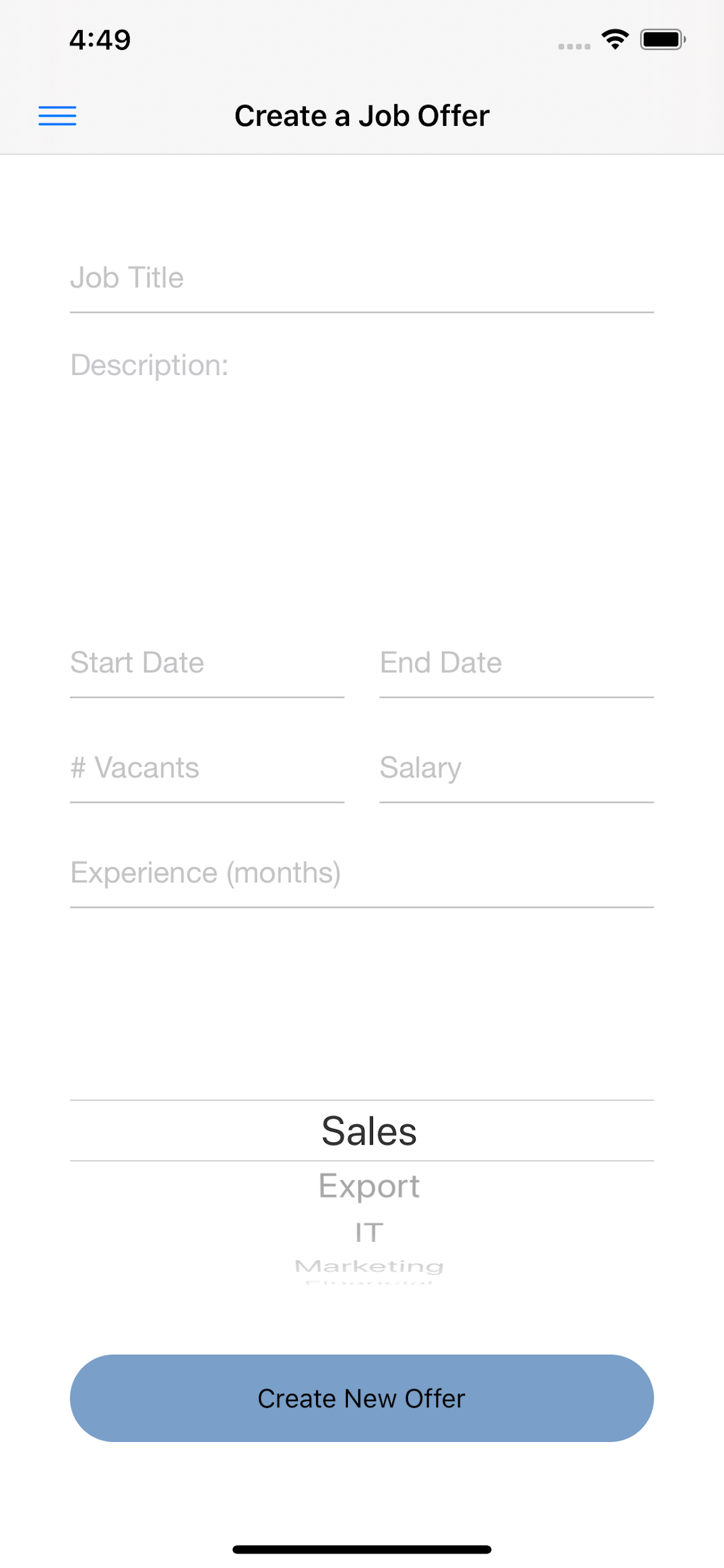
**Figure 21.** Employer Profile.

Just like the Student, the Employer has a navigation menu view, as seen in Figure 22. Here, they can create a job offer, view their job offers, log out, or delete their account.



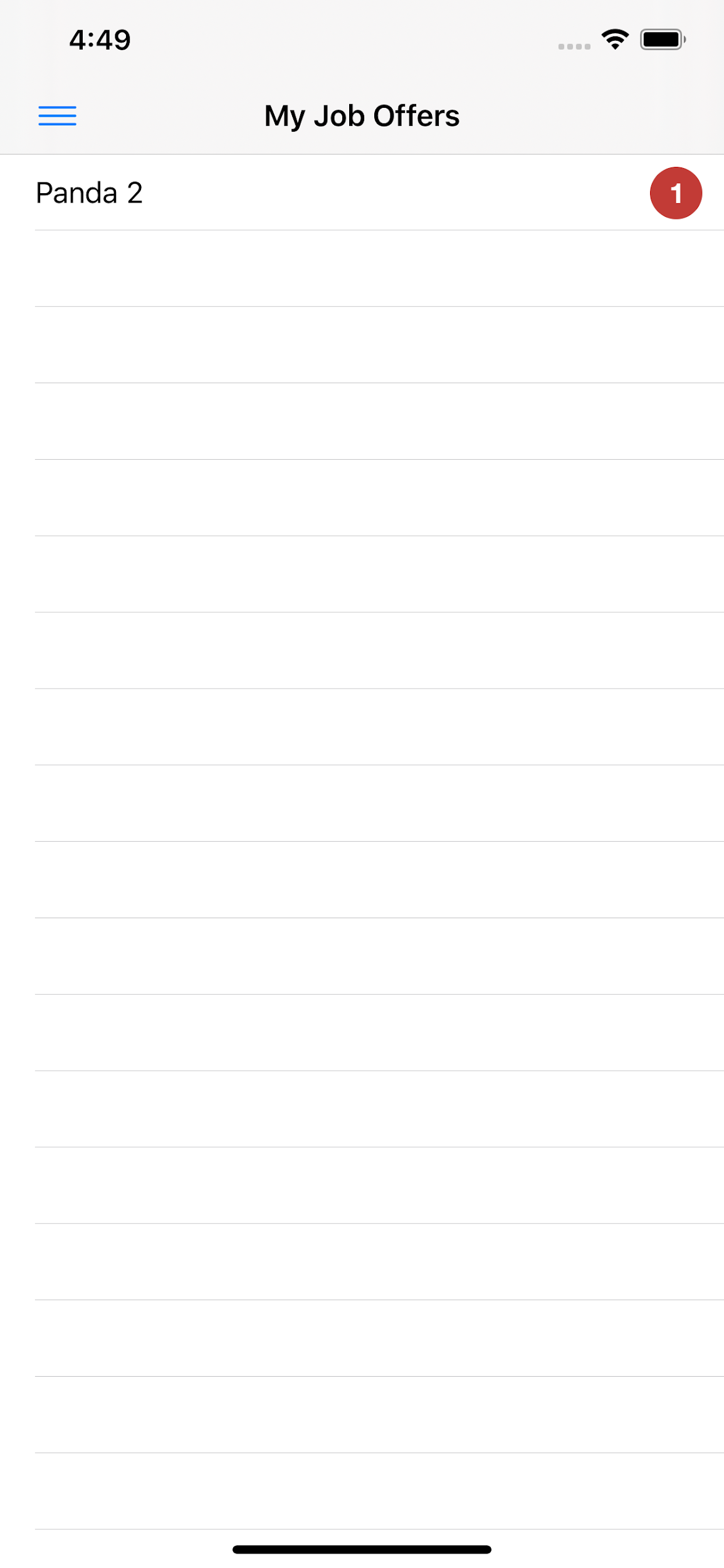
**Figure 22.** Employer Navigation View.

When the Employer selects the option to create a job offer, the view shown in Figure 23 is displayed.



**Figure 23.** Create a Job Offer

Or, if the Employer wants to see their job offers, they see a view like Figure 24. The red circle with a number tells how many students have applied to it.

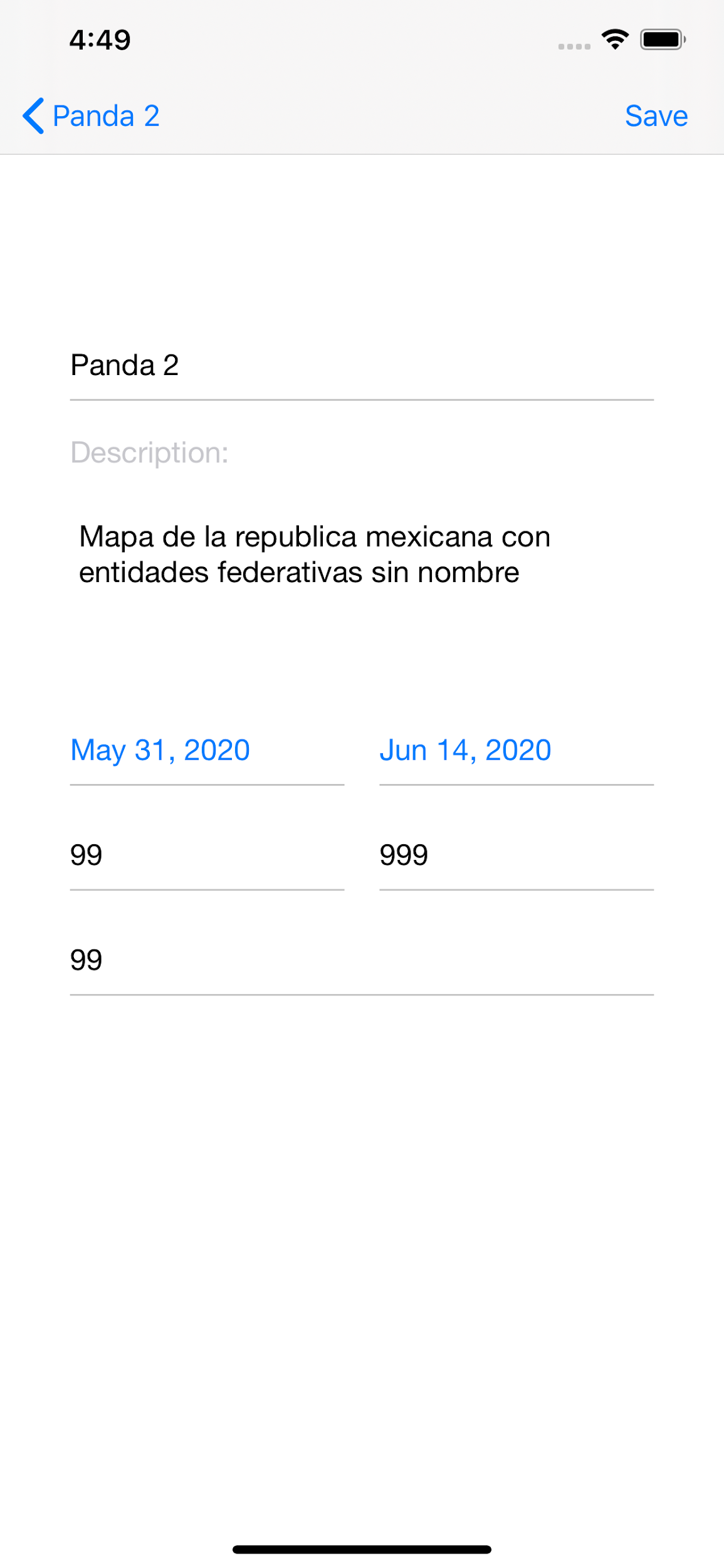


**Figure 24.** Employer’s job offers.

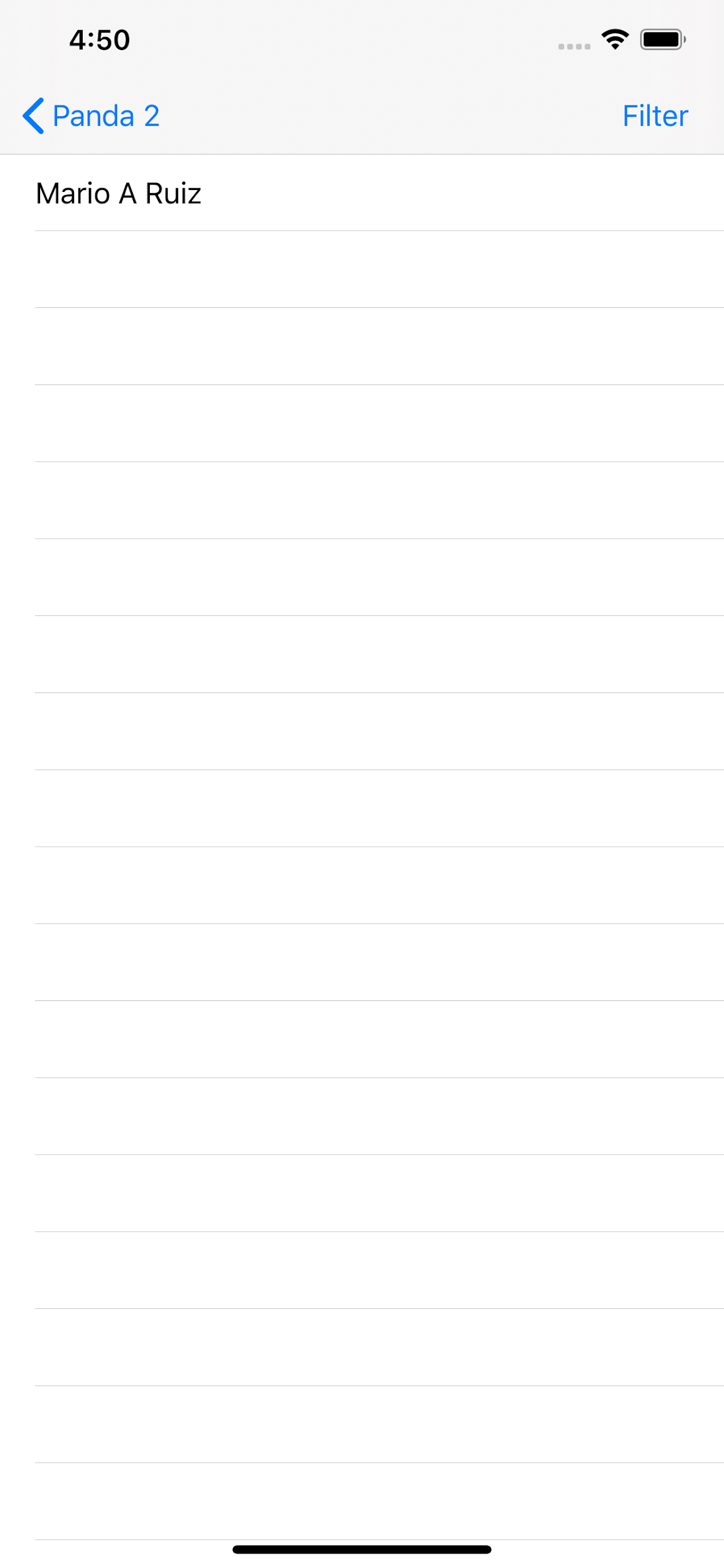
Then, the Employer can see the details of the job offer (figure 25), edit it (figure 26), see interested students (figure 27), or close it.



**Figure 25.** Job Offer Details

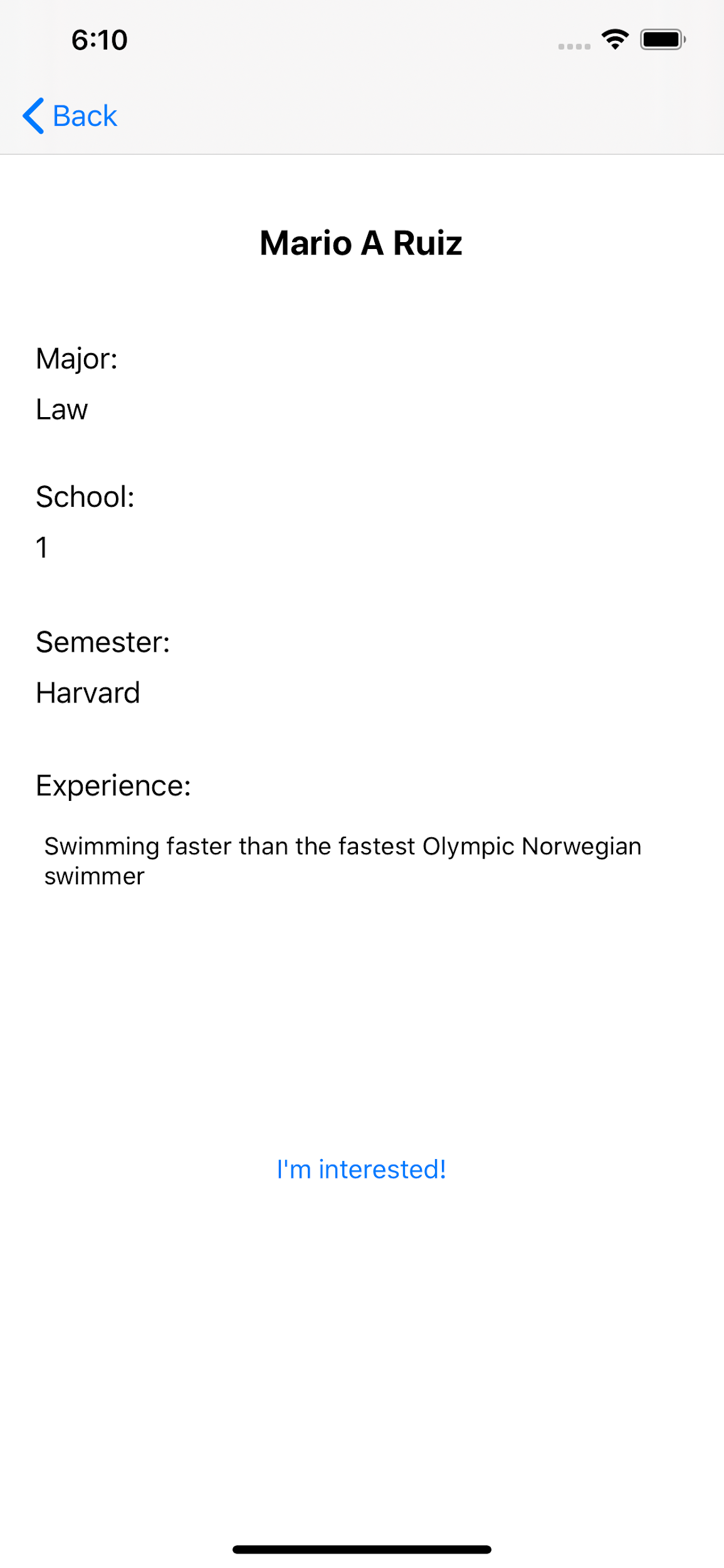


**Figure 26.** Edit Job Offer Details.

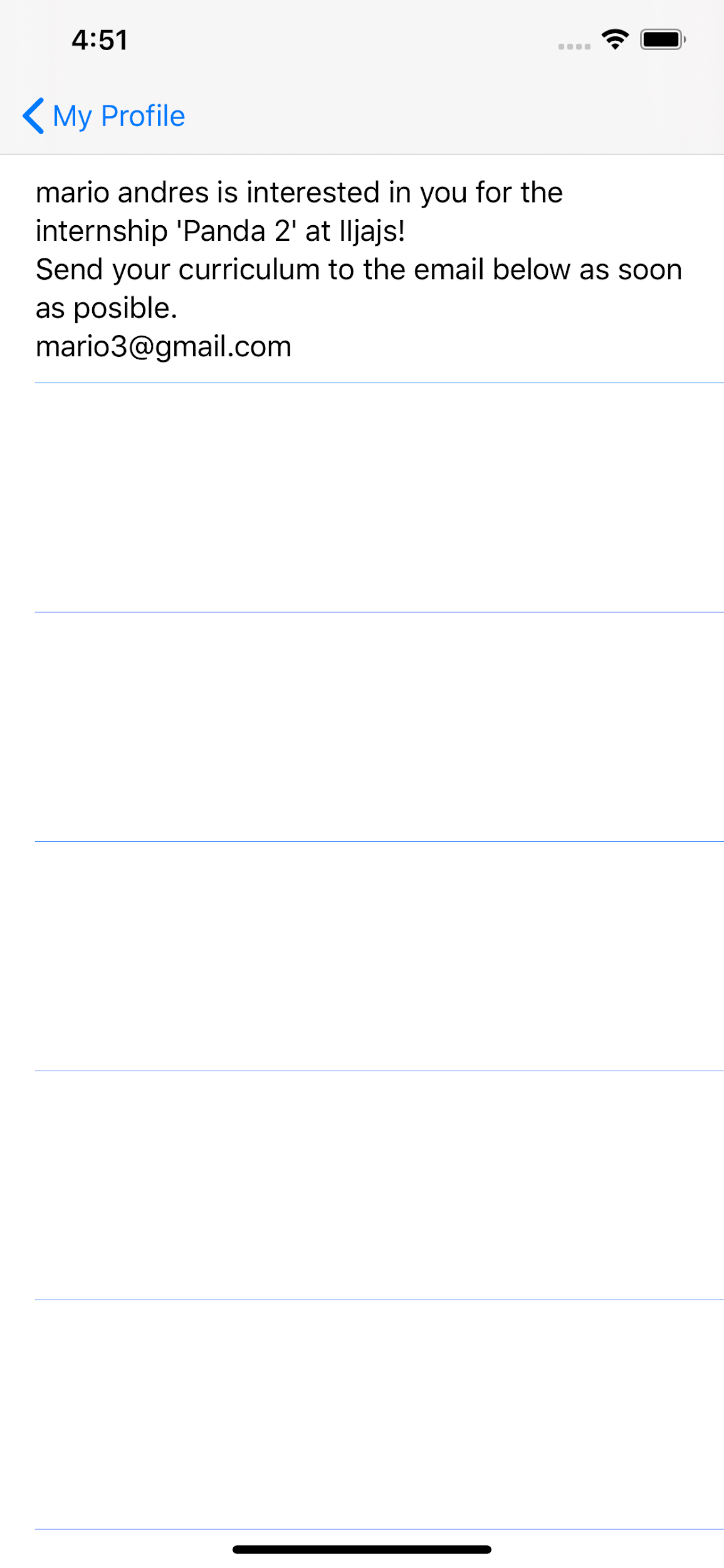


**Figure 27.** Interested Students.

Finally, the employer can view the student’s profile with their information and curriculum, like in Figure 28. If they are interested in contacting them they press the virtual button with that label, and the student is notified. Remember that the notifications button was at the top right of the Student’s profile. turns red when there’s a new notification. Then, when they tap on the button they see the view in Figure 29.



**Figure 28.** Student’s Profile from Employer’s Perspective.



**Figure 29.** Student’s notification.

1. **Lessons Learned**

**Ricardo**: I learned a lot of things in this project. First of all, I could finally program on Swift and I felt much more comfortable with it than with other similar programming languages, I could understand better how is that SCRUM really works, I could participate in a small project implementing it, I was able to apply some XP values and good practices, I was able to use Firebase, and finally I could make a better management of my time thanks to the small releases required every 2 weeks.

**Cindy**: I learned to use swift and how to apply it in a project, in addition to the SCRUM methodology. I found it quite useful because this is how many companies work on their projects, and I also consider it helped us to keep better track of the work done. Also, I learned about Firebase and I found it to be very practical.

**Giovanni**: I learned that mobile development can get pretty messy real fast. This is true the more functionalities and views that are added to the project. I have basic knowledge on the usage of git and GitHub but before this team project I hadn't used it much collaboratively, so this further improved my abilities with GitHub as a team member.

**Mario**: I learned from this project about SCRUM better, which I had never used it because I haven’t had any job in IT. Learned anything between standup meetings, backlogs, git and demos. On the other hand, I learned about Firebase, and that was very cool. I had never worked with servers or web services before. It made me feel like a powerful wizard.

1. **Conclusions**

**Ricardo**: I conclude that this was one of my favorite projects of my career because it had a lot of different activities to be performed, and maybe at the beginning I wasn’t convinced at all but as the project progressed I was more and more excited. I really appreciate to have projects like this in which we can work along all the semester, integrate a lot of topics and discover many more that can be very useful in our professional life. I really changed my mind about many subjects and I expect to still doing it in the future with projects like this.

**Cindy**: In conclusion, I like how the SCRUM methodology works, dividing the project in little deliveries, with the daily stand up, especially now in quarantine I think it was a really essential step to have finished the project.

**Giovanni**: I conclude that using SCRUM for software development is pretty fascinating as it accomplishes what is trying to do: give more independence to the development teams and its individual members. It forces you to stick to your word because you are the one saying what you will accomplish for each sprint and that’s a pretty good incentive. Mobile development isn’t what I expected but it was an enjoyable experience.

**Mario**: I conclude that it is interesting how when you divide a big problem into small user stories it’s less overwhelming. And… As the teams works on them and complete them, the whole app is building up, and once you are very advanced in the project and you look back, it’s just mesmerizing.

1. **Acknowledgment**

We’d like to acknowledge the Engineer Carlos Ventura Molina because he had the great idea of an app that link employers and students. Also, we want to acknowledge Firebase because it is a free tool which provided us a flexible and online database (and also it can scale up in order to cover the demand and has a web browser making it really convenient and easy to use). And finally, we want to acknowledge Jira because it made a lot easier to manage the sprints and the backlogs.

1. **References**
2. Atlassian Agile Coach, “What are sprints?”, https://www.atlassian.com/agile/scrum/sprints, visited on 6/6/2020
3. Codecademy. “MVC: Model, View, Controller”, https://www.codecademy.com/articles/mvc, visited on 6/6/2020
4. AgileAlliance, “What is Extreme Programming (XP)?”, https://www.agilealliance.org/glossary/xp,visited on 6/6/2020