**BSHC4 - Cloud Computing**

**Cloud Application Development Project**

**Project Name: nci-cadproject**

**Mark Hayden**

**x14723661**

**Heroku link:**[**https://cadproject.herokuapp.com/doctors/sign\_in**](https://cadproject.herokuapp.com/doctors/sign_in)

**Github repo for project:** [**https://github.com/MarkNCI/shc-pms.git**](https://github.com/MarkNCI/shc-pms.git)

**Github repo for Custom Gem:** [**https://github.com/MarkNCI/calcgem.git**](https://github.com/MarkNCI/calcgem.git)

1. **Introduction**

This project required me to build a Smart Health Care-Patient Management System (SHC-PMS). The required functionality of the system is to allow a doctor to add new patient records, these doctors will also be able to view, edit and delete these records. After each doctor has registered and logged in they can navigate to their profile page and edit their details or delete their account. Each doctor can print out a pdf report of patient details and they can also search patients based on condition and sort them by age. The system will display a pie chart of the conditions of each patient, and the custom gem functionality allows doctors to check the price of a visit based on the patient’s age. The UI design for the application was created using the Materialize gem and adhering to Nielsen’s Heuristics of UI. There is a match between the system and real world applications in term of language, UI design is consistent and provides a degree of user control and freedom. Users are aware of the system status and the site can be used after a minimal amount of training.

**Gems**

Materialize-sass: UI Design

Devise: Doctor Registration/Login

Ransack: Search and Sort

Prawn/Prawn-Table: PDF Report

Chartkick: Pie Chart of Conditions

Calcgem: Custom gem for Price by Age

1. **Development Strategy**

**2.1 Database Design**

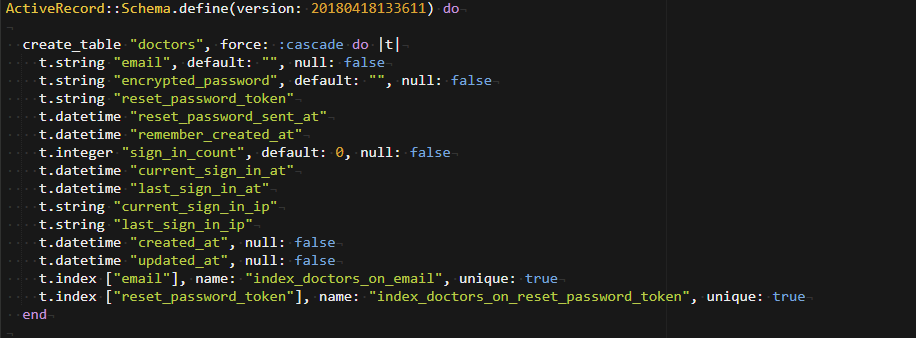
**Doctors Table**

rails generate devise:install

rails g devise:views

rails generate devise doctors

rake db:migrate

****

**Patients Table**

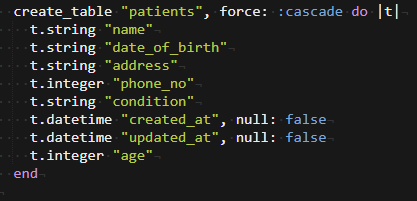
Rails generate controller patients name:string date\_of\_birth:string address:string

phone\_no:integer condition:string

rake db:migrate

rails generate migration add\_age\_to\_patients age:integer

rake db:migrate

****

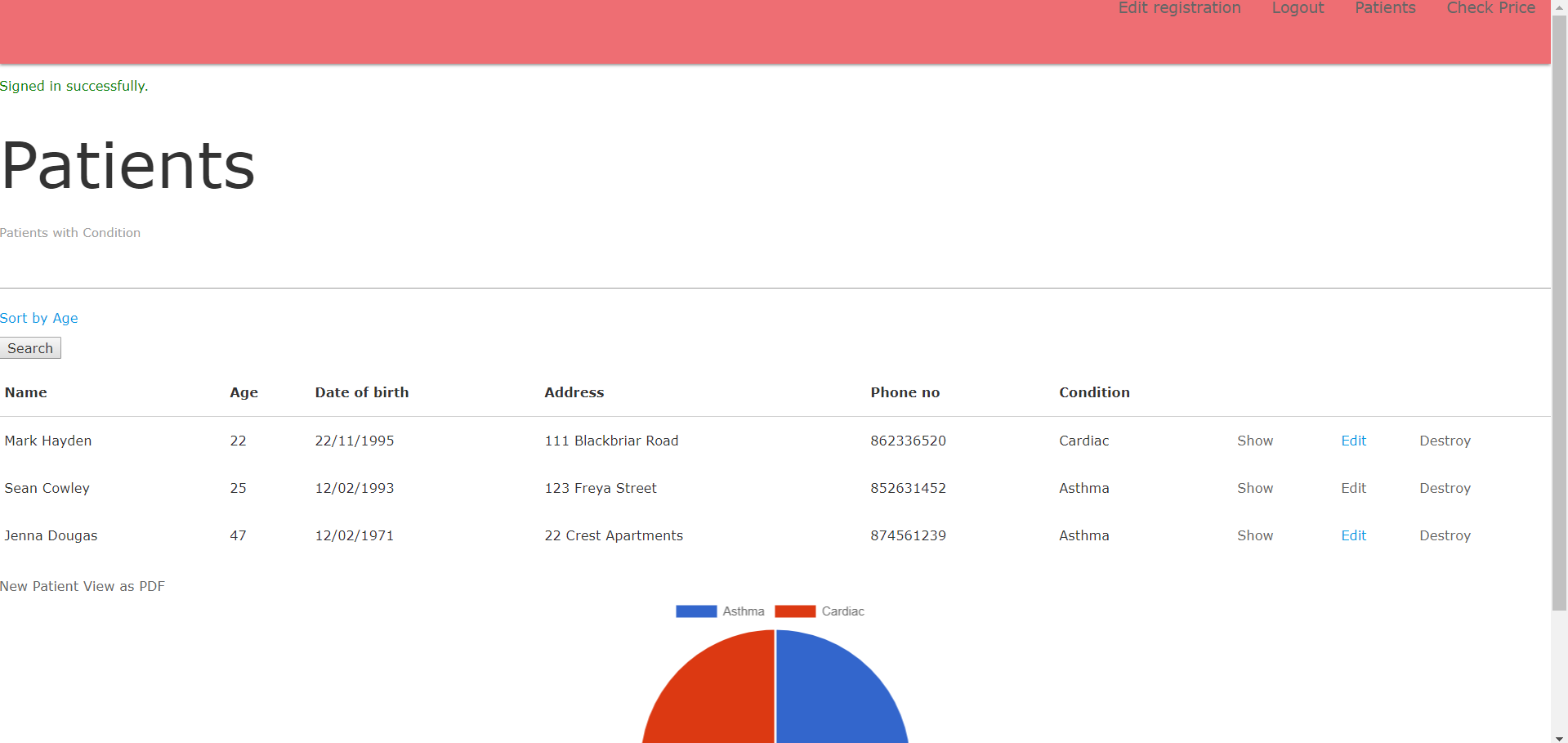
**2.2 Implementation**

**Functionality 1: Add, View, Edit and Delete Patient Records.**

A doctor can add a patient to the system under the headings Name, Age, Date of Birth, Address, Phone Number and Condition. This functionality is implemented using the MVC approach in the following files:

**View All Patient Records**

|  |  |
| --- | --- |
| Model | patient.rb |
| View | Views/patients/index.html.erb |
| Controller | patients\_controller.rb |



**Edit Patient Record**

|  |  |
| --- | --- |
| Model | patient.rb |
| View | Views/patients/edit.html.erb |
| Controller | patients\_controller.rb |



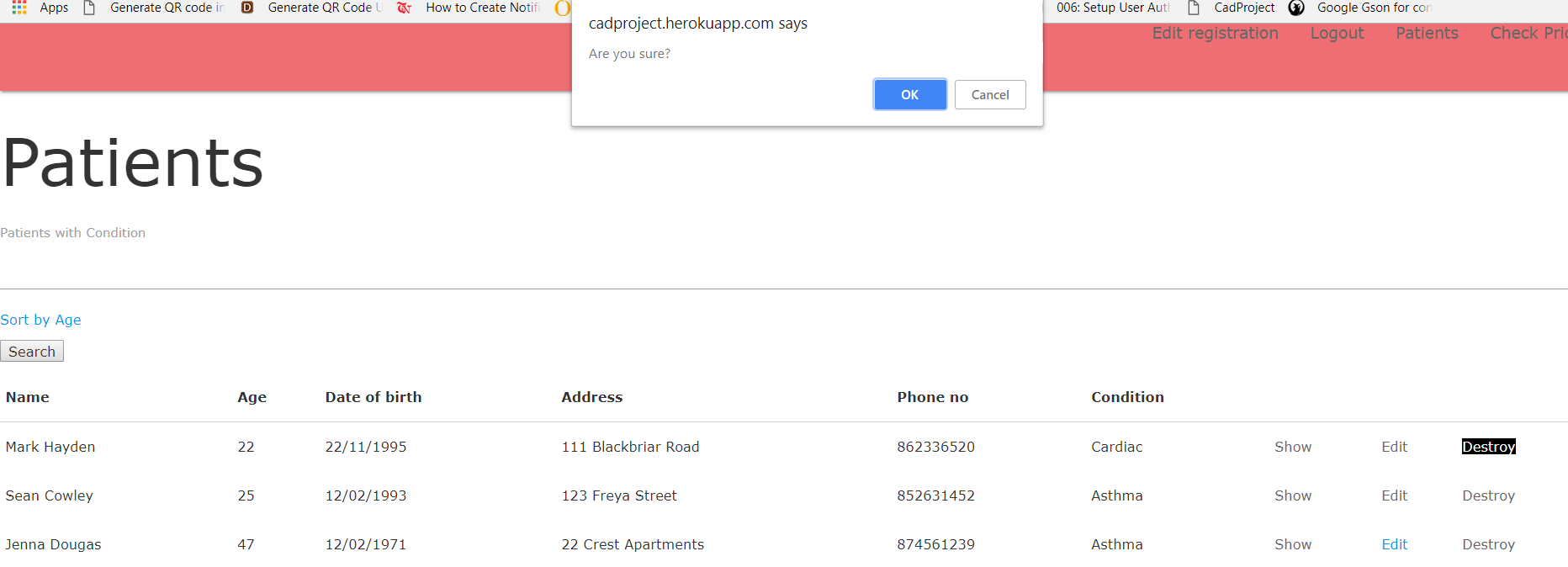
**View Patient Record**

|  |  |
| --- | --- |
| Model | patient.rb |
| View | Views/patients/show.html.erb |
| Controller | patients\_controller.rb |

****

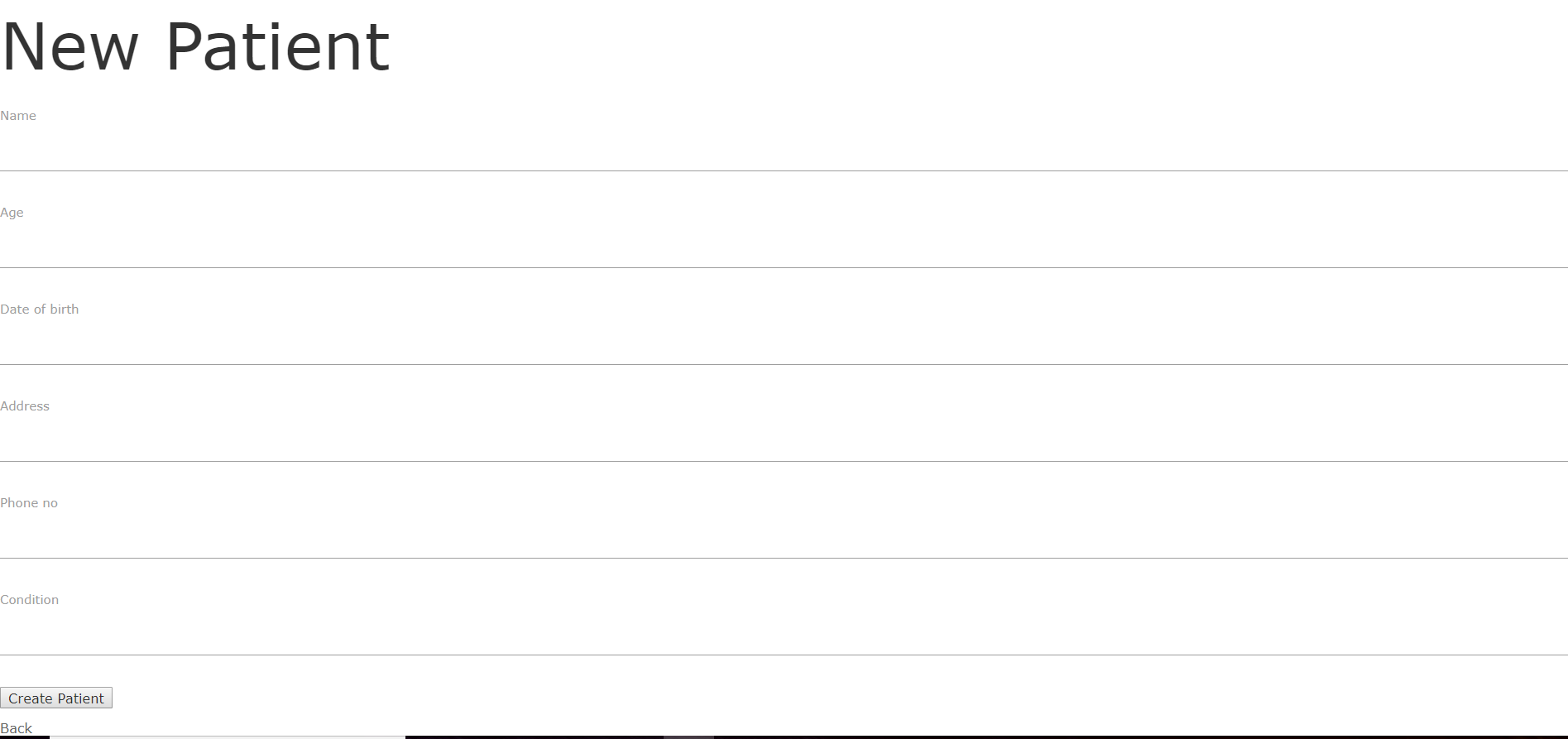
**Delete Patient Record**

|  |  |
| --- | --- |
| Model | patient.rb |
| View | N/A |
| Controller | patients\_controller.rb |



**Add Patient Record**

|  |  |
| --- | --- |
| Model | patient.rb |
| View | Views/patients/new.html.erb |
| Controller | patients\_controller.rb |

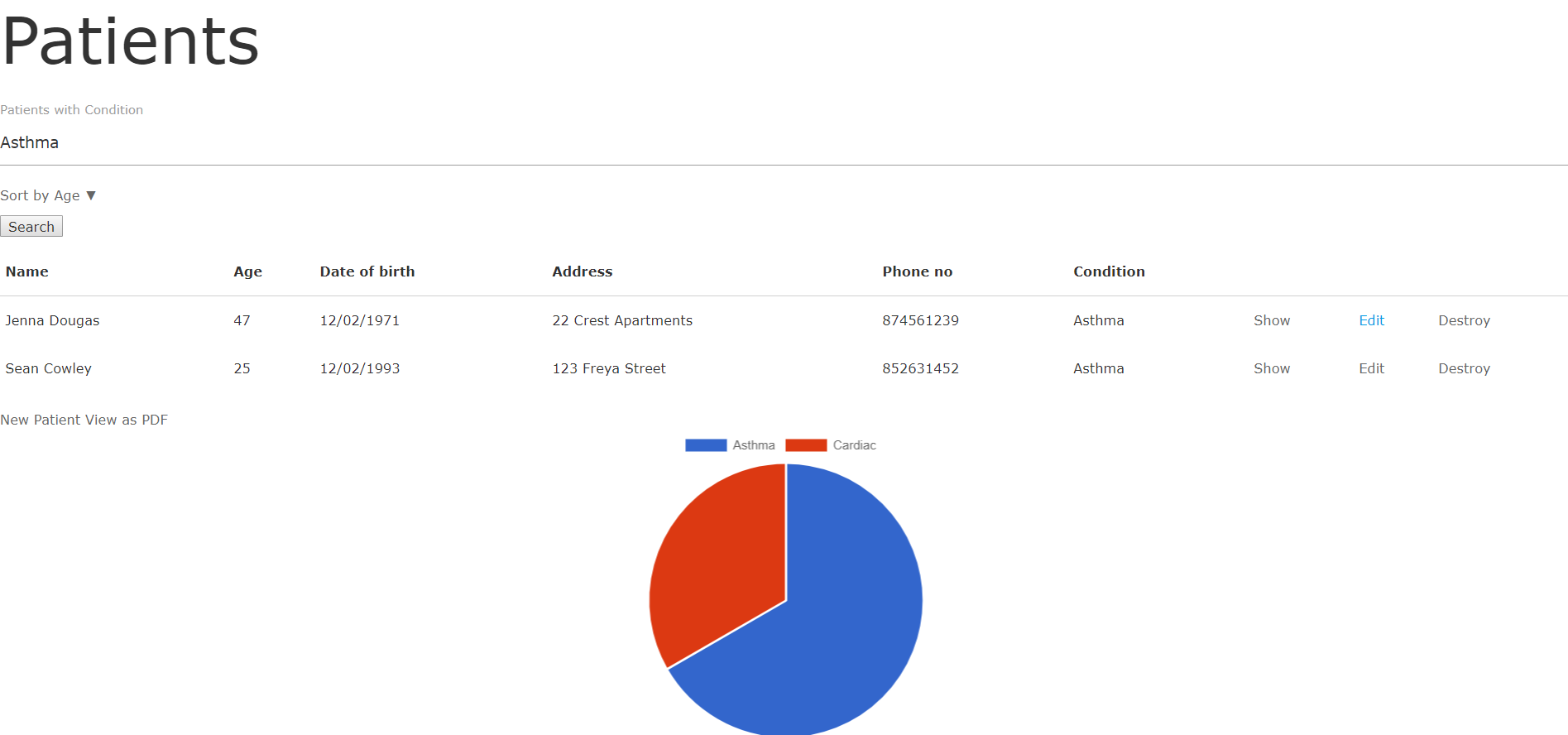
****

**Functionality 2: Search and Sort**

Doctors can search patient records by Condition, for example Asthma, and can sort the results by Age. This is done using the Ransack gem.

**Search/Sort Patient Records**

|  |  |
| --- | --- |
| Model | patient.rb |
| View | Views/patients/index.html.erb |
| Controller | patients\_controller.rb |

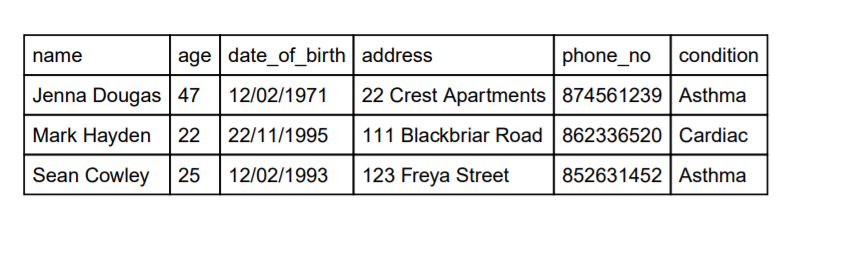


**Functionality 3: Reporting**

Allows doctors to print out a pdf report of all patient records. This is done using the gems Prawn and Prawn-Table.

**Reporting**

|  |  |
| --- | --- |
| Model | app/pdfs/patient\_pdf |
| View | Views/patients/index.html.erb |
| Controller | patients\_controller.rb |

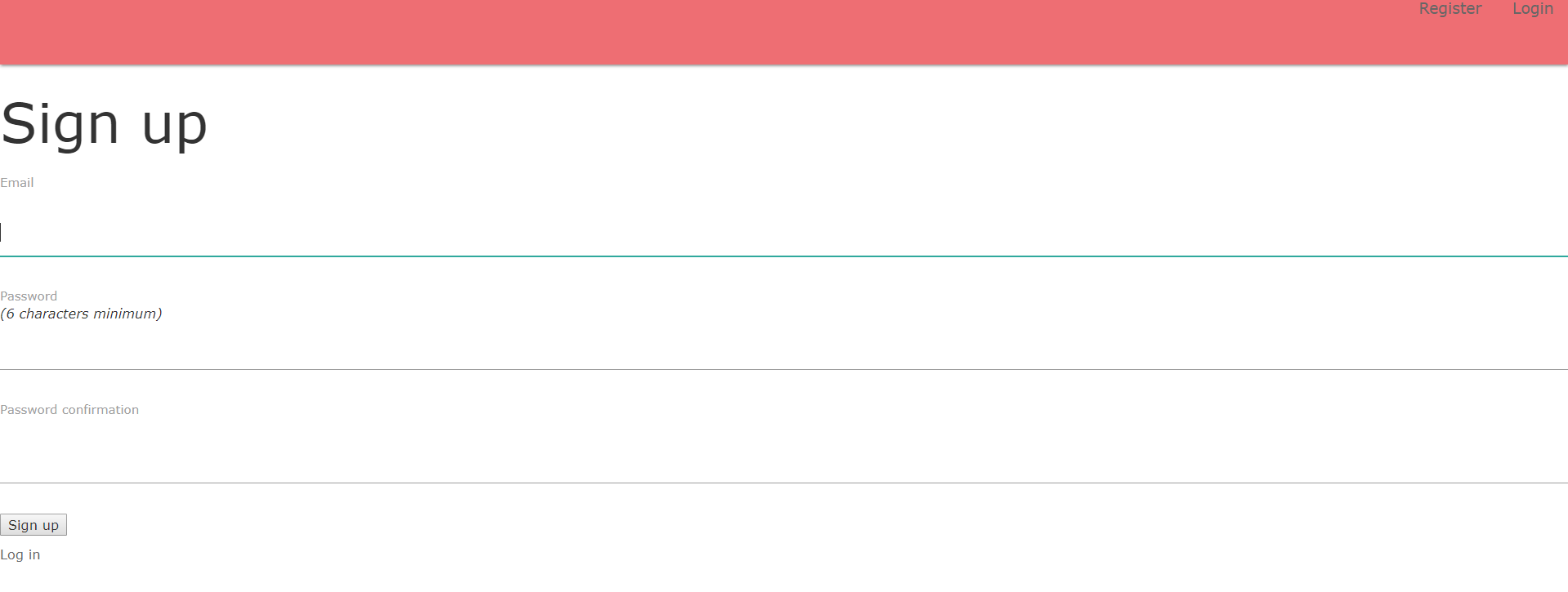


**Functionality 4: Allow Doctor Login and Registration**

Allows Doctors to sign up to the system, log in and out and change their details. Doctors can also delete their accounts if they so choose. This is done using the Devise gem.

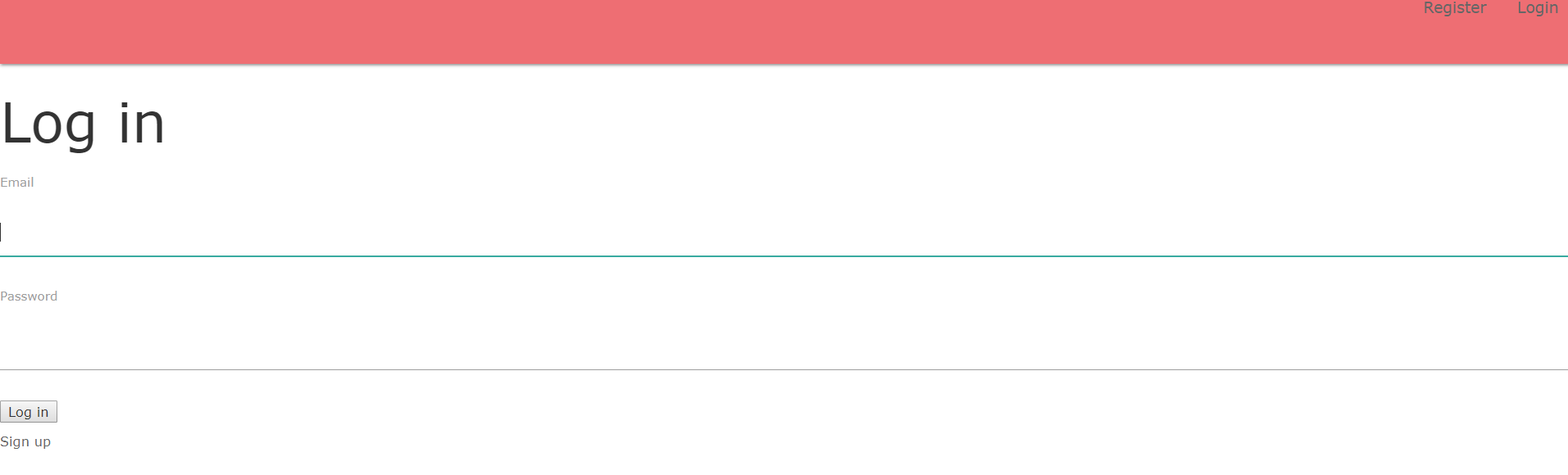
**Sign up**

|  |  |
| --- | --- |
| Model | doctor.rb |
| View | Views/devise/new.html.erb |
| Controller | application\_controller.rb |



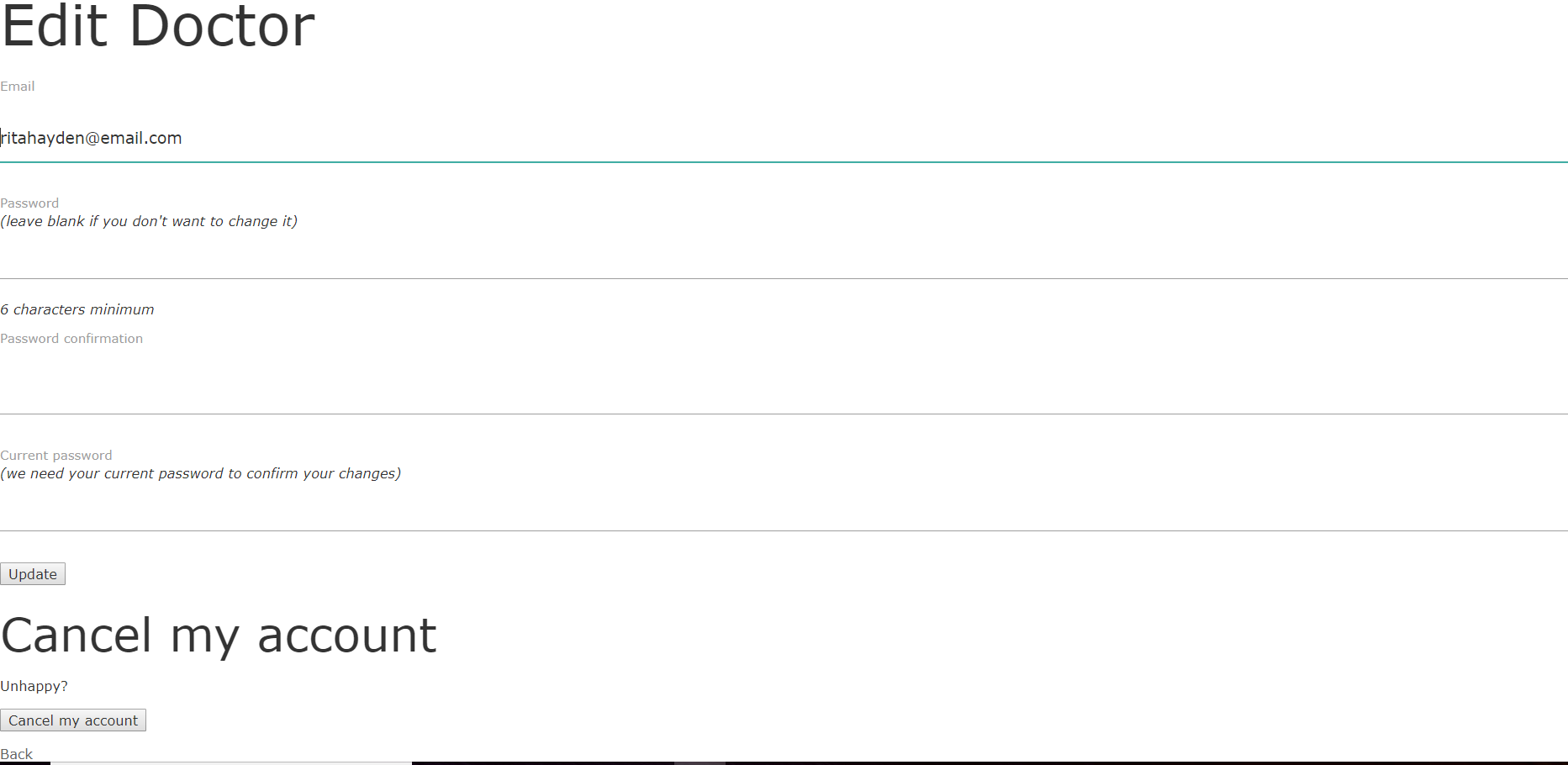
**Log in**

|  |  |
| --- | --- |
| Model | doctor.rb |
| View | Views/devise/sessions/new.html.erb |
| Controller | application\_controller.rb |



**Edit Doctor**

|  |  |
| --- | --- |
| Model | doctor.rb |
| View | Views/devise/registrations/edit.html.erb |
| Controller | patients\_controller.rb |

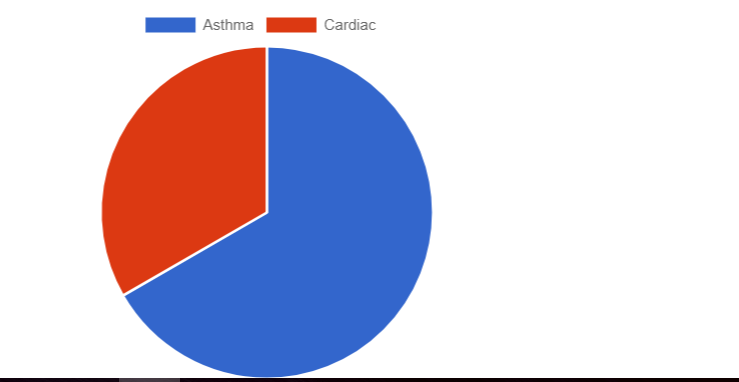


**Functionality 5: Display Conditions on Chart**

When patients are added to the system their conditions will populate a pie chart that will be displayed below their records. This is done using the Chartkick gem.

**Display Patient Conditions**

|  |  |
| --- | --- |
| Model | N/A |
| View | Views/patients/index.html.erb |
| Controller | patients\_controller.rb |

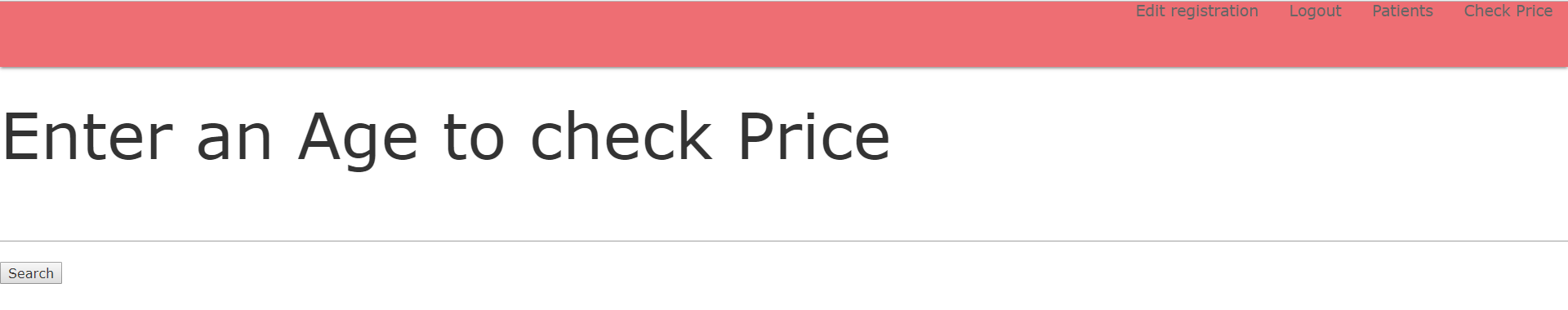


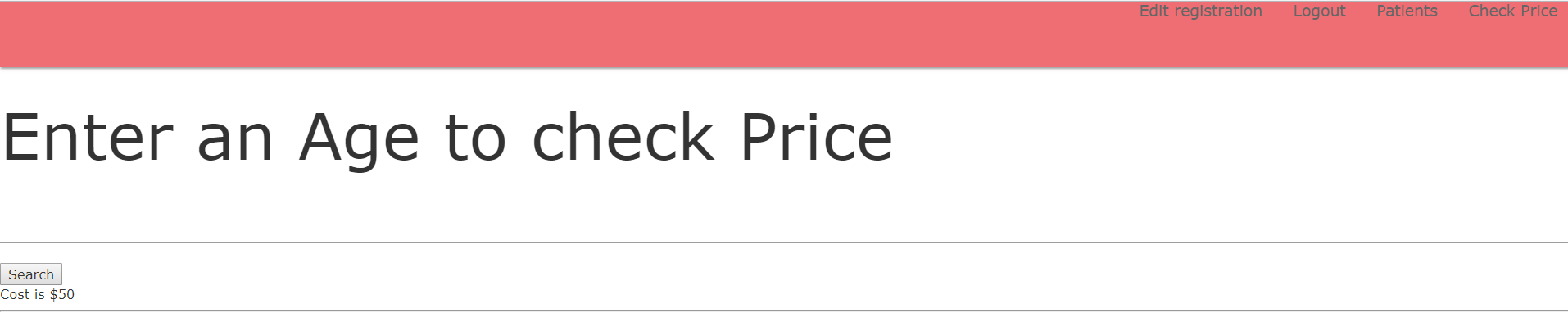
**Functionality 6: Check Price by Age (Custom Gem)**

Doctors can check the price of a patient’s appointment by entering in their age and seeing the result. This is done using calcgem, a custom gem of my own creation. It is deployed on Github as per the 3rd link on the cover page and is connected to the app via a Github link in the Gemfile.

**Check Price By Age**

|  |  |
| --- | --- |
| Model | N/A |
| View | Views/patients/checkprice.html.erb  Views/patients/cost.html.erb |
| Controller | patients\_controller.rb |

****

****

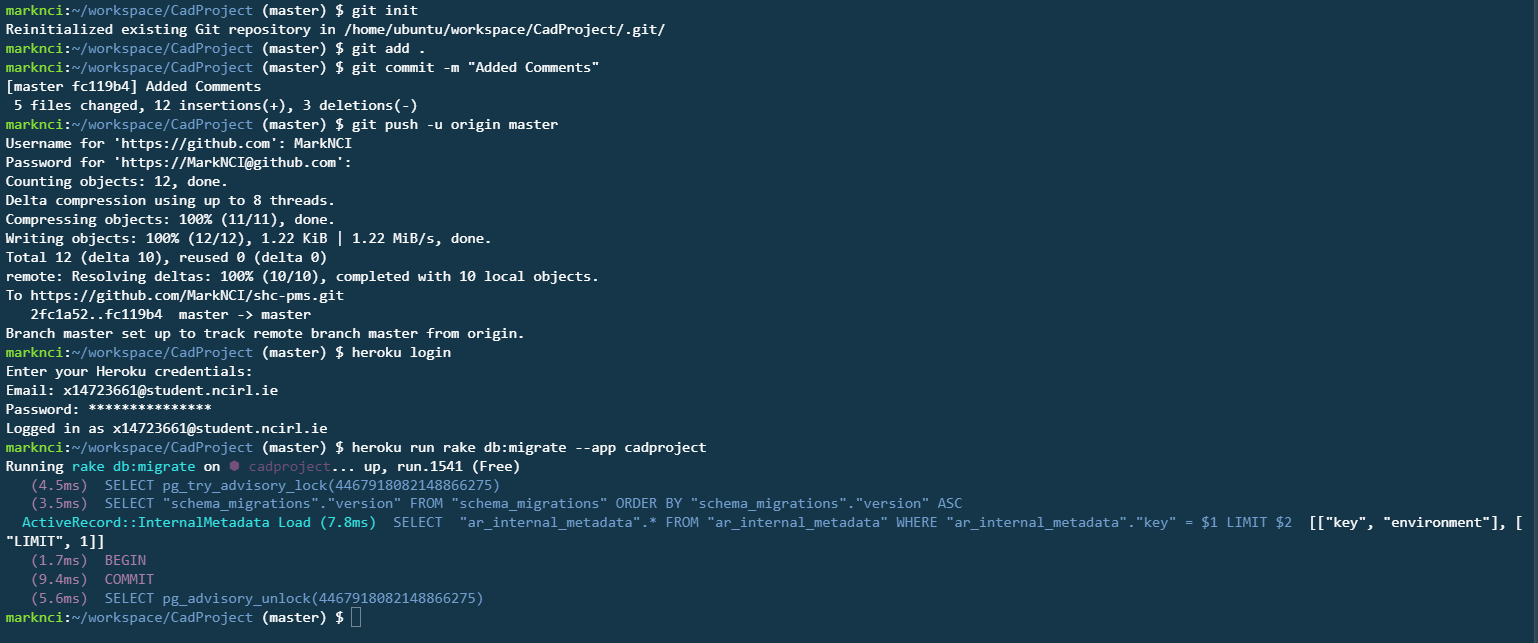
1. **Deployment**

The application was developed using Cloud9 and is deployed on Heroku via Github using this link: <https://cadproject.herokuapp.com/doctors/sign_in>

**Login Details**

Username: ritahayden@email.com Password: Password

Screenshot of Github/Heroku deployment



1. **References**
2. Chartkick. 2018. Chartkick - Create beautiful JavaScript charts with one line of Ruby. [ONLINE] Available at: <https://www.chartkick.com/>. [Accessed 22 April 2018].
3. http://plataformatec.com.br/. 2018. Github - plataformatec/devise: Flexible authentication solution for Rails with Warden.. [ONLINE] Available at: <https://github.com/plataformatec/devise>. [Accessed 22 April 2018].
4. https://github.com/activerecord-hackery/ransack. 2018. Github - activerecord-hackery/ransack: Object-based searching.. [ONLINE] Available at: <https://github.com/activerecord-hackery/ransack>. [Accessed 22 April 2018].
5. Michael Hartl. 2016. *Learn Ruby on Rails with the Best Free Online Tutorial | Ruby on Rails Tutorial with Michael Hartl*. [ONLINE] Available at: <https://www.railstutorial.org/book>. [Accessed 22 April 2018].
6. Melvin Ch'ng. 2016. *Ruby on Rails Tutorial*. [ONLINE] Available at: <http://melvinchng.github.io/rails/>. [Accessed 22 April 2018].
7. NourZa Channel. (2016). *Ruby on Rails convert Ruby html to pdf file gem 'prawn'*. [Online Video]. 13 April 2016. Available from: <https://www.youtube.com/watch?v=BW5zwqj37Lo>. [Accessed: 22 April 2018].