



Spring 2017
Software Engineering
CSC 3324 - 01

Milestone 4: Software Implementation

Manal Hamdi
Mehdi Laziri
Supervised by: Dr. Bouchaib Falah

Submission Date: Tuesday, May 2, 2017

Table of Contents

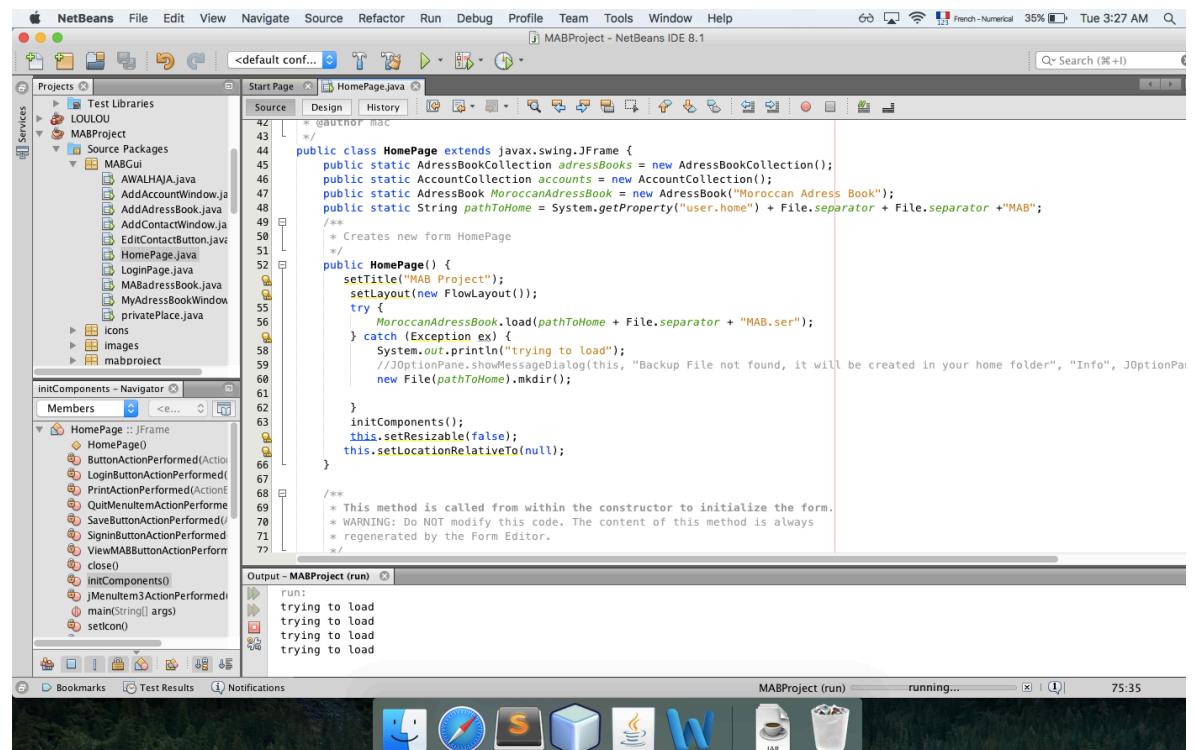
1. Software Implementation.....	2
 1.1. Desktop Application.....	3
 1.2. Web Application.....	4
 1.3. Mobile Application.....	5

1.1. Desktop Application

1.1.1. Tools Used

- To implement the desktop application, we have used java as a language.
- External libraries were added to the application in order to have new features, i.e. export as a pdf.
- IDE used is Netbeans.

1.1.2. Code

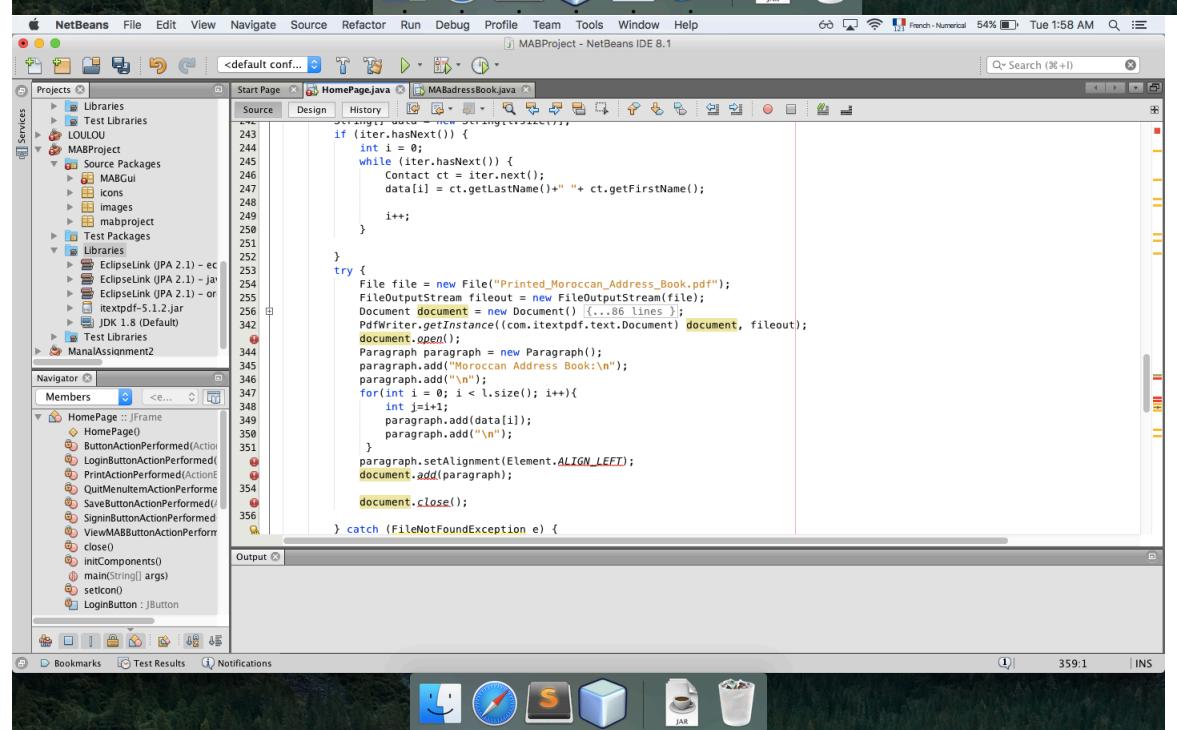


```
NetBeans 8.1 - MABProject - Start Page - HomePage.java
```

```
42 * @author Mac
43 */
44 public class HomePage extends javax.swing.JFrame {
45     public static AdressBookCollection adressBooks = new AdressBookCollection();
46     public static AccountCollection accounts = new AccountCollection();
47     public static AdressBook MoroccanAdressBook = new AdressBook("Moroccan Adress Book");
48     public static String pathToHome = System.getProperty("user.home") + File.separator + File.separator + "MAB";
49     /**
50      * Creates new form HomePage
51     */
52     public HomePage() {
53         setTitle("MAB Project");
54         setLayout(new FlowLayout());
55         try {
56             MoroccanAdressBook.load(pathToHome + File.separator + "MAB.ser");
57         } catch (Exception ex) {
58             System.out.println("trying to load");
59             //JOptionPane.showMessageDialog(this, "Backup File not found, it will be created in your home folder", "Info", JOptionPane.DEFAULT_OPTION);
60             new File(pathToHome).mkdir();
61         }
62         initComponents();
63         this.setResizable(false);
64         this.setLocationRelativeTo(null);
65     }
66
67     /**
68      * This method is called from within the constructor to initialize the form.
69      * WARNING: Do NOT modify this code. The content of this method is always
70      * regenerated by the Form Editor.
71     */
72 }
```

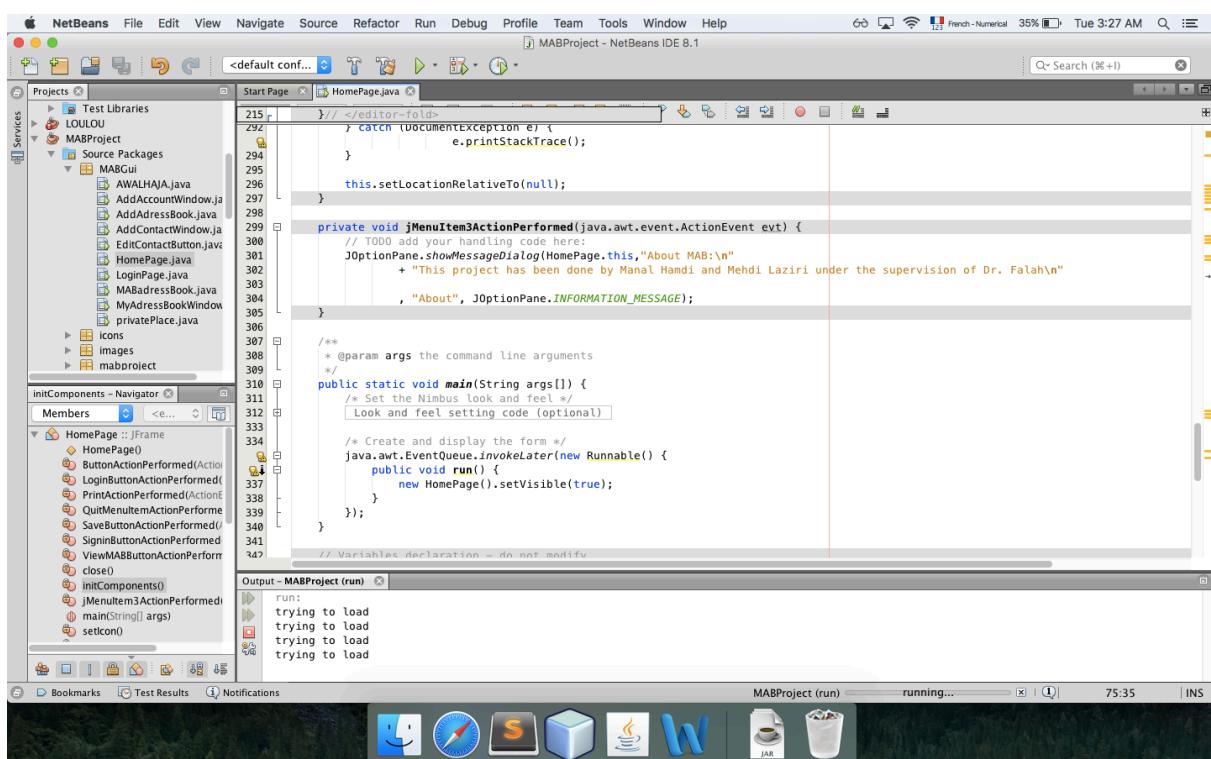
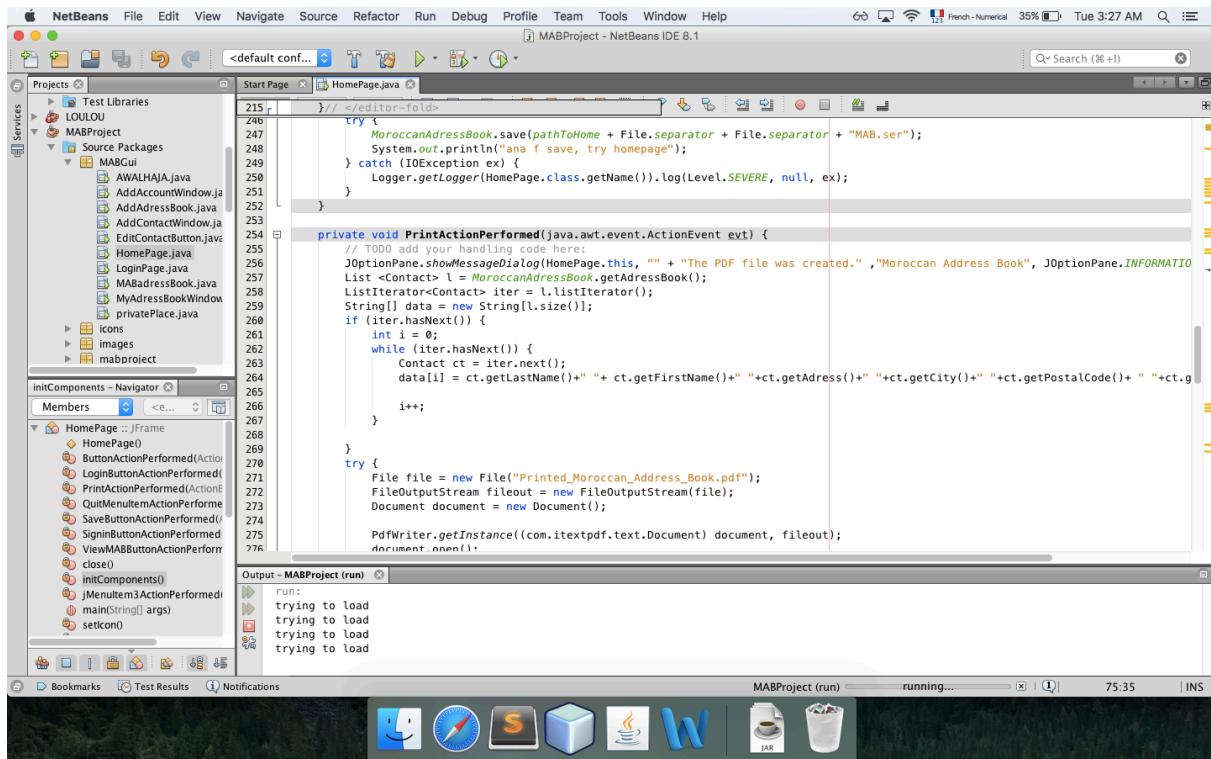
Output - MABProject (run) :

```
RUN:
trying to load
```

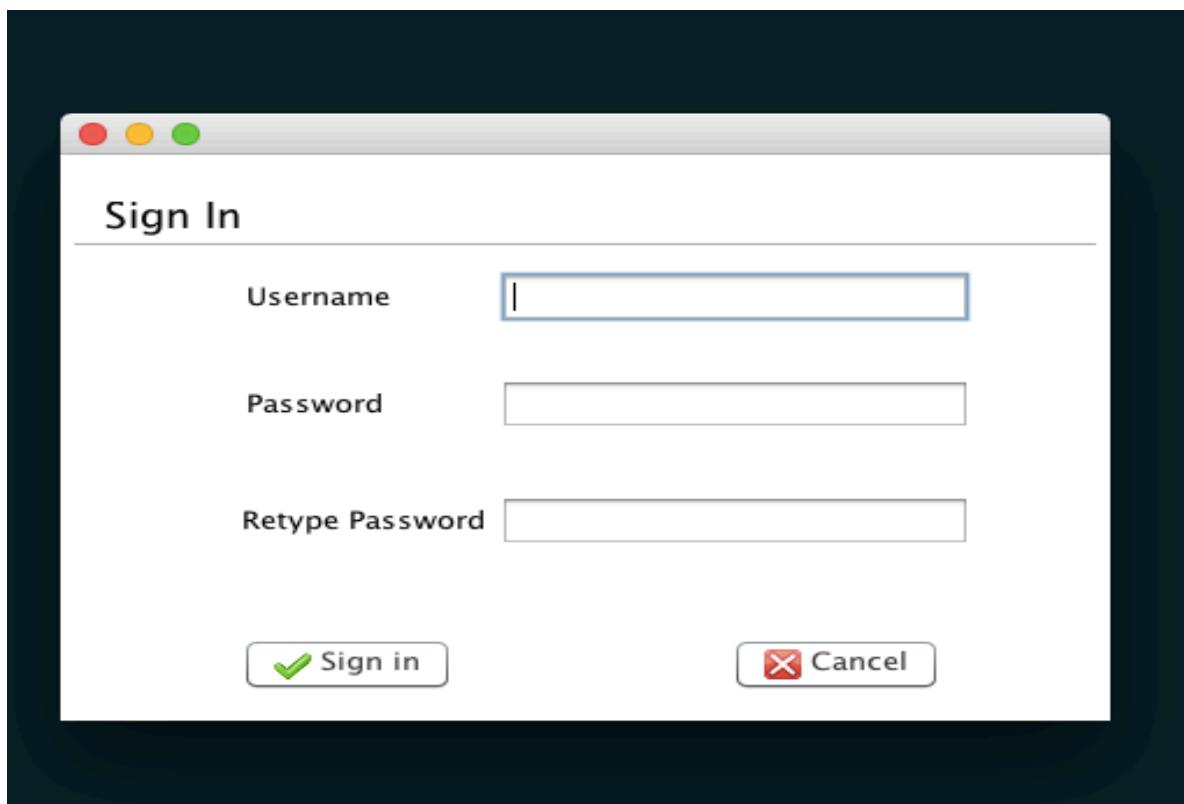


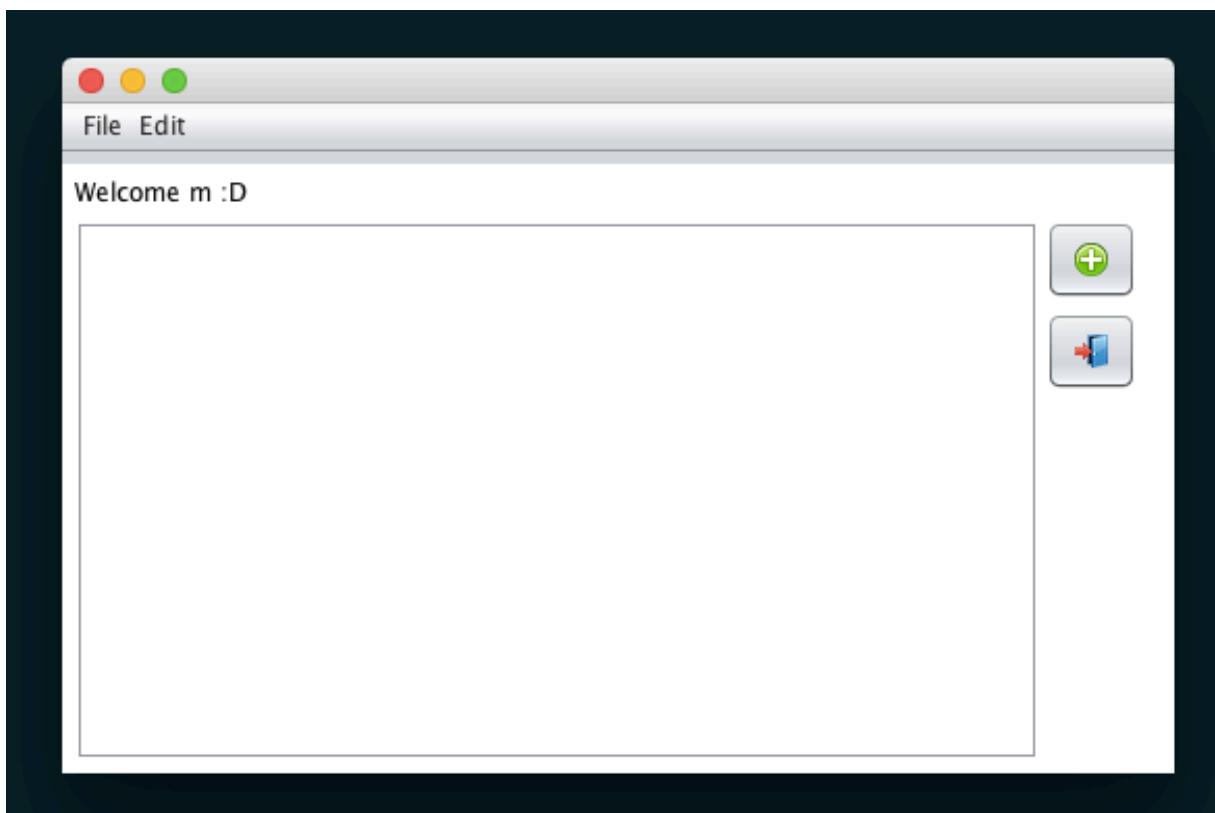
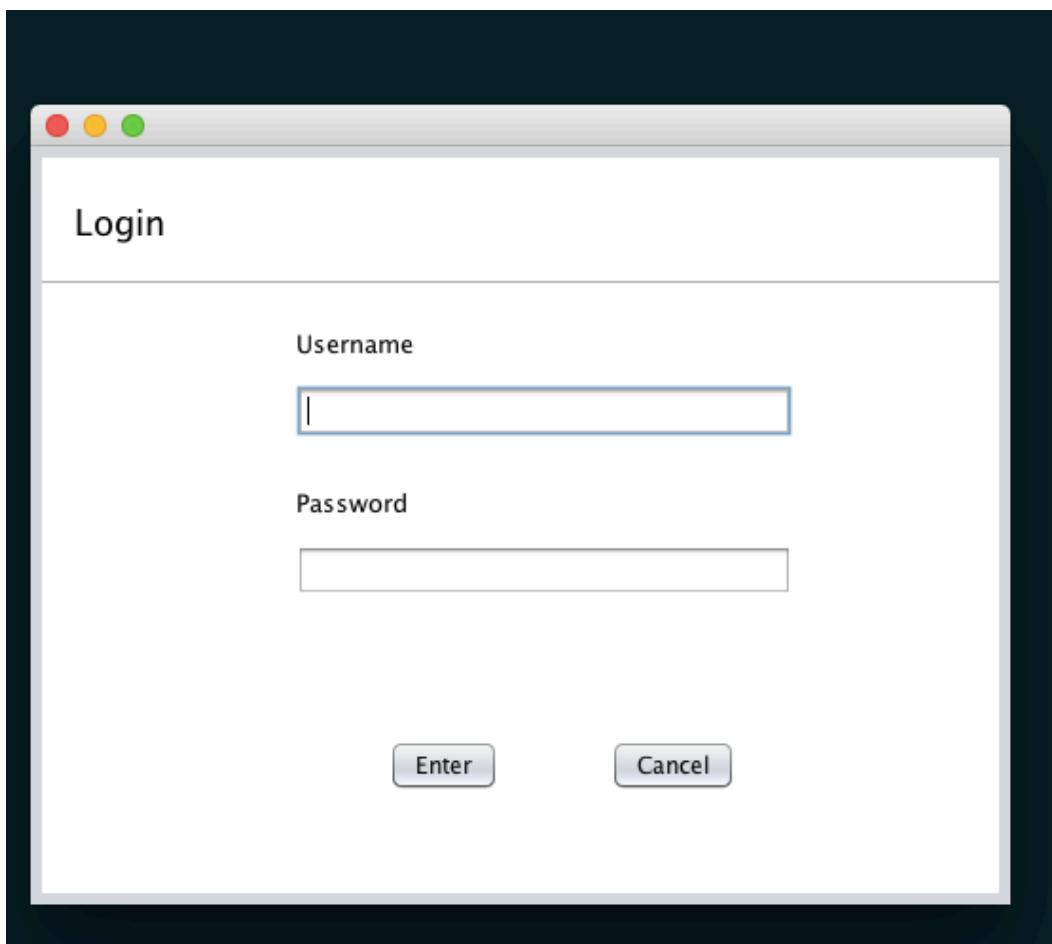
```
NetBeans 8.1 - MABProject - Start Page - MABaddressBook.java
```

```
243     if (iter.hasNext()) {
244         int i = 0;
245         while (iter.hasNext()) {
246             Contact ct = iter.next();
247             data[i] = ct.getLastName() + " " + ct.getFirstName();
248             i++;
249         }
250     }
251 }
252 try {
253     File file = new File("Printed_Moroccan_Address_Book.pdf");
254     FileOutputStream fileout = new FileOutputStream(file);
255     Document document = new Document(PageSize.A4);
256     PdfWriter.getInstance(document, fileout);
257     document.open();
258     Paragraph paragraph = new Paragraph();
259     paragraph.add("Moroccan Address Book:\n");
260     for(int i = 0; i < data.length; i++){
261         int j=i+1;
262         paragraph.add(data[i]);
263         paragraph.add("\n");
264     }
265     paragraph.setAlignment(Element.ALIGN_LEFT);
266     document.add(paragraph);
267
268     document.close();
269 } catch (FileNotFoundException e) {
```



1.1.3. GUI





1.2. Web Application

1.2.1. Tools Used

- We have used many **Ruby on Rails** to implement the back end functionalities of our desktop application. We have used the **Bootstrap Framework** to take care of all front end matters.
- We have used **Heroku** to host our web application online and to have an online Database which would facilitate data transfer between the Mobile, Desktop and Web Application.
- We have used **Electron** to export our Web Application as a desktop app(Coded mainly using JavaScript, HTML and CSS) in order to give the user the opportunity to access our application on his/her preferred operating system.
- We followed the MVC model

1.2.2. Snippets of Code

```
<%= bootstrap_devise_error_messages! %>
<div class="panel panel-default devise-bs">
  <div class="panel-heading">
    <h4><%= t('.sign_up', default: 'Sign up') %></h4>
  </div>
  <div class="panel-body">
    <%= form_for(resource, as: resource_name, url: registration_path(resource_name),
html: { role: 'form' }) do |f| %>
      <div class="form-group">
        <%= f.label :email %>
        <%= f.email_field :email, autofocus: true, class: 'form-control' %>
      </div>
      <div class="form-group">
        <%= f.label :password %>
        <%= f.password_field :password, class: 'form-control' %>
      </div>
      <div class="form-group">
        <%= f.label :password_confirmation %>
        <%= f.password_field :password_confirmation, class: 'form-control' %>
      </div>
      <%= f.submit t('.sign_up', default: 'Sign up'), class: 'btn btn-primary' %>
    <% end %>
  </div>
</div>
<%= render 'devise/shared/links' %>
```

```

1  <% if controller_name != 'sessions' %>
2    <%= link_to t('.sign_in', default: 'Sign in'), new_session_path(resource_name) %><br />
3  <% end %>
4
5  <% if devise_mapping.registerable? && controller_name != 'registrations' %>
6    <%= link_to t('.sign_up', default: 'Sign up'), new_registration_path(resource_name) %><br />
7  <% end %>
8
9  <% if devise_mapping.recoverable? && controller_name != 'passwords' %>
10   <%= link_to t('.forgot_your_password', default: 'Forgot your password?'), new_password_path(resource_name) %><br />
11 <% end %>
12
13 <% if devise_mapping.confirmable? && controller_name != 'confirmations' %>
14   <%= link_to t('.didn_t_receive_confirmation_instructions', default: "Didn't receive confirmation instructions?"), new_confirmation_path(resource_name) %><br />
15 <% end %>
16
17 <% if devise_mapping.lockable? && resource_class.unlock_strategy_enabled?(:email) && controller_name != 'unlocks' %>
18   <%= link_to t('.didn_t_receive_unlock_instructions', default: "Didn't receive unlock instructions?"), new_unlock_path(resource_name) %><br />
19 <% end %>
20
21 <% if devise_mapping.omniauthable? %>
22   <% resource_class.omniauth_providers.each do |provider| %>
23     <%= link_to t('.sign_in_with_provider', provider: provider.to_s.titleize, default: "Sign in with #{provider.to_s.titleize}"), omniauth_authorize_path(resource_name, provider) %><br />
24   <% end %>
25 <% end %>
26
27 <%= bootstrap_devise_error_messages! %>
28 <div class="panel panel-default devise-bs">
  <div class="panel-heading">
    <h4><%= t('.title', resource: resource_class.model_name.human, default: "Edit #{resource_name.to_s.humanize}") %></h4>
  </div>
  <div class="panel-body">
    <%= form_for(resource, as: resource_name, url: registration_path(resource_name), html: { method: :put }) do |f| %>
      <div class="form-group">
        <%= f.label :email %>
        <%= f.email_field :email, autofocus: true, class: 'form-control' %>
      </div>
      <div class="form-group">
        <%= f.label :password %> <i>(<%= t('.leave_blank_if_you_don_t_want_to_change_it', default: "leave blank if you don't want to change it") %>)</i>
        <%= f.password_field :password, autocomplete: "off", class: 'form-control' %>
      </div>
      <div class="form-group">
        <%= f.label :password_confirmation %>
        <%= f.password_field :password_confirmation, class: 'form-control' %>
      </div>
      <div class="form-group">
        <%= f.label :current_password %> <i>(<%= t('.we_need_your_current_password_to_confirm_your_changes', default: "we need your current password to confirm your changes") %>)</i>
        <%= f.password_field :current_password, class: 'form-control' %>
      </div>
      <%= f.submit t('.update', default: 'Update'), class: 'btn btn-primary' %>
    <% end %>
  </div>
</div>
<p><%= t('.unhappy', default: 'Unhappy') %>? <%= link_to t('.cancel_my_account', default: 'Cancel my account'), registration_path(resource_name), data: { confirm: t('.are_you_sure', default: "Are you sure?") }, method: :delete %>.</p>
<%= link_to t('.back', default: 'Back'), :back %>
```

```

<%= bootstrap_devise_error_messages! %>


<div class="panel-heading">
    <h4><%= t('.title', resource: resource_class.model_name.human, default: "Edit #{resource_name.to_s.humanize}") %></h4>
  </div>
  <div class="panel-body">
    <%= form_for(resource, as: resource_name, url: registration_path(resource_name), html: { method: :put }) do |f| %>
      <div class="form-group">
        <%= f.label :email %>
        <%= f.email_field :email, autofocus: true, class: 'form-control' %>
      </div>
      <div class="form-group">
        <%= f.label :password %> <i>(<%= t('.leave_blank_if_you_don_t_want_to_change_it', default: "leave blank if you don't want to change it") %>)</i>
        <%= f.password_field :password, autocomplete: "off", class: 'form-control' %>
      </div>
      <div class="form-group">
        <%= f.label :password_confirmation %>
        <%= f.password_field :password_confirmation, class: 'form-control' %>
      </div>
      <div class="form-group">
        <%= f.label :current_password %> <i>(<%= t('.we_need_your_current_password_to_confirm_your_changes', default: 'we need your current password to confirm your changes') %>)</i>
        <%= f.password_field :current_password, class: 'form-control' %>
      </div>
      <%= f.submit t('.update', default: 'Update'), class: 'btn btn-primary' %>
    <% end %>
  </div>
</div>

<p><%= t('.unhappy', default: 'Unhappy') %>? <%= link_to t('.cancel_my_account', default: 'Cancel my account'), registration_path(resource_name), data: { confirm: t('.are_you_sure', default: "Are you sure?") }, method: :delete %>.</p>

<%= link_to t('.back', default: 'Back'), :back %>


```

```

1 class User < ApplicationRecord
2   # Include default devise modules. Others available are:
3   # :confirmable, :lockable, :timeoutable and :omniauthable
4   has_many :address_books
5   has_many :contacts
6   devise :database_authenticatable, :registerable,
7   :recoverable, :rememberable, :trackable, :validatable
8 end

```

```

1 class Contact < ApplicationRecord
2   belongs_to :address_book
3   belongs_to :user
4 end

```

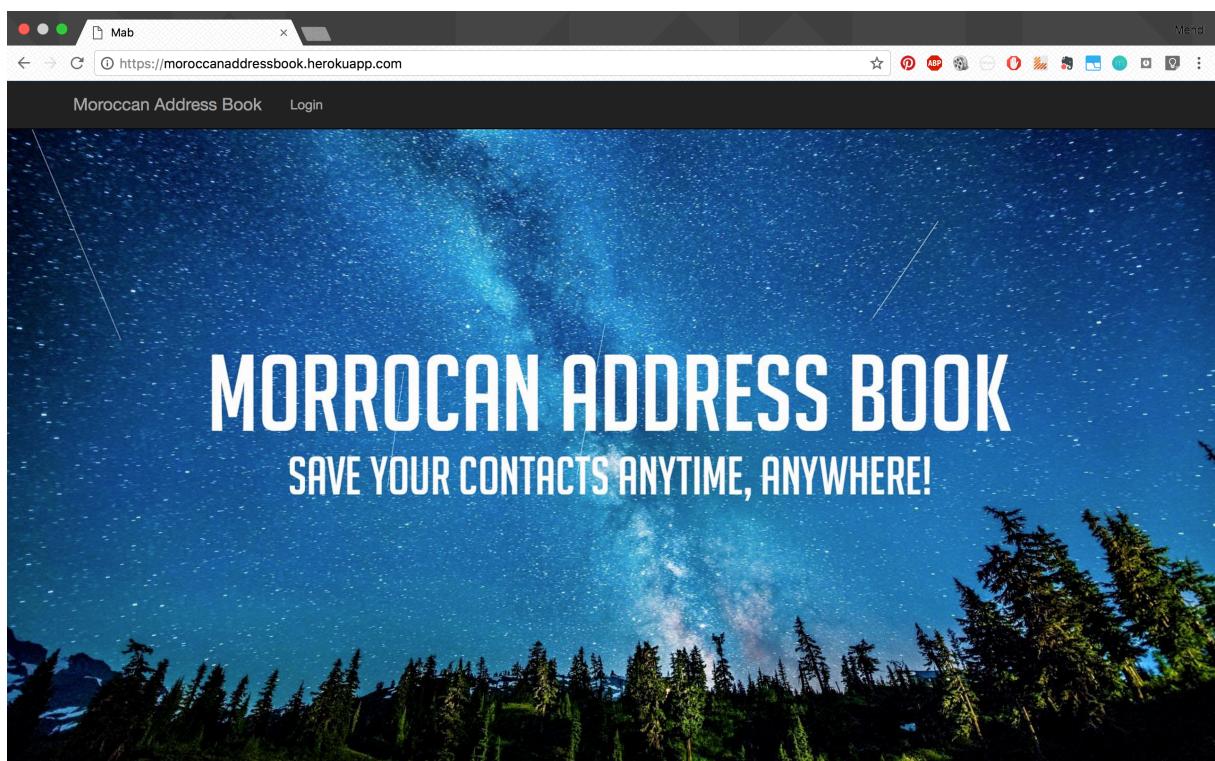
```

1 class AddressBook < ApplicationRecord
2   has_many :contacts
3   belongs_to :users
4 end
5

```

```
1 class Ability
2   include CanCan::Ability
3   def initialize(user)
4     user ||= User.new # guest user (not logged in)
5     if user.isAdmin?
6       can :manage, :all
7     else if user.isReg?
8       can :access, :rails_admin
9       can :dashboard
10      can :manage, AddressBook, user_id: user.id
11      can :manage, AddressBook, isPrivate: false
12      can :manage, Contact, user_id: user.id
13    else
14      can :read, [AddressBook, Contact]
15    end
16  end
17 end
18 end
```

1.2.3. GUI



The screenshot shows the Mab Admin dashboard with a teal header bar. The header includes the title "Site Administration | Mab Admin", a "maillist.pdf" tab, a "Secure" connection indicator, the URL "https://moroccanaddressbook.herokuapp.com/admin/", and user information "mehdi@laziri.com" and "Log out". A navigation sidebar on the left lists "Address books", "Contacts", and "User". The main content area has a green header "SITE ADMINISTRATION" and a blue success message "Signed in successfully.". Below is a "Dashboard" section with a "Model name", "Last created", and "Records" table:

Model name	Last created	Records	
Address books	about 12 hours ago	1	[grid, +, -]
Contacts	about 12 hours ago	2	[grid, +, -]
User	10 days ago	2	[grid, +, -]

The screenshot shows the "Address books" list page from the Mab Admin interface. The header is identical to the dashboard. The navigation sidebar shows "Address books" is selected. The main content area has a green header "LIST OF ADDRESS BOOKS" and a "List" view. It includes a "Filter" input, a "Refresh" button, and a "Selected items" dropdown. The table below lists one address book:

<input type="checkbox"/>	Id	Name	Isprivate	Created at	Updated at	User	...
<input type="checkbox"/>	72	AUI_Friends	<input checked="" type="checkbox"/>	May 02, 2017 06:48	May 02, 2017 06:48	12	[info, edit, delete]

1 address book

Contact | Mab Admin

Secure https://moroccanaddressbook.herokuapp.com/admin/contact

Mab Admin

Navigation Address books Contacts User

Dashboard Home mehdi@laziri.com Log out

Dashboard / Contacts

List Add new Export Add filter Selected items

Filter Refresh Export found Contacts

<input type="checkbox"/>	Id	Name	Postalcode	Address	City	Phone	...
<input type="checkbox"/>	112	Maha Laziri	142536	28, Gold Street	New York	+1124124891	
<input type="checkbox"/>	102	Othmane Mahfou	162738	Bld21,103	Ifrane	069647284728	
<input type="checkbox"/>	92	Maha Hamdi	27382	Hay Nahda	Rabat	06273843	
<input type="checkbox"/>	82	Achraf Mamdouh	12345	Avenue Med VI	Beni Me	0637482746	
<input type="checkbox"/>	72	Dr. Cavalli Sforza	123456	12 Hay Riad	Ifrane	0678637482	
<input type="checkbox"/>	62	Dr. Falah Bouchaib	12345	Hay Marjane	Casablanca	0612345678	

6 contacts

User | Mab Admin

Secure https://moroccanaddressbook.herokuapp.com/admin/user

Mab Admin

Navigation Address books Contacts User

Dashboard Home mehdi@laziri.com Log out

LIST OF USER

Dashboard / User

List Add new Export Add filter Selected items

Filter Refresh Export found User

<input type="checkbox"/>	Id	Email	Reset password sent at	Remember created at	Sign in count	Current sign in at	...
<input type="checkbox"/>	12	manal@hamdi.com	-	-	3	May 02, 2017 06:45	
<input type="checkbox"/>	2	mehdi@laziri.com	-	-	12	May 02, 2017 06:46	

2 user

NEW USER

Dashboard / User / New

Email Optional. Length up to 255.

Password Optional. Length of 6-128.

Password confirmation Optional. Length up to 255.

Reset password sent at Optional.

Remember created at Optional.

Sign in count 0 Optional.

Current sign in at Optional.

Last sign in at Optional.

Current sign in ip Optional. Length up to 255

Last sign in ip

Select fields to export

Select All Fields

Fields from user

<input checked="" type="checkbox"/> Id	<input checked="" type="checkbox"/> Email	<input checked="" type="checkbox"/> Reset password sent at	<input checked="" type="checkbox"/> Remember created at
<input checked="" type="checkbox"/> Sign in count	<input checked="" type="checkbox"/> Current sign in at	<input checked="" type="checkbox"/> Last sign in at	<input checked="" type="checkbox"/> Current sign in ip
<input checked="" type="checkbox"/> Last sign in ip	<input checked="" type="checkbox"/> Name	<input checked="" type="checkbox"/> Isadmin	<input checked="" type="checkbox"/> Created at
<input checked="" type="checkbox"/> Updated at	<input checked="" type="checkbox"/> Isreg		

Fields from associated address books

<input checked="" type="checkbox"/> Id	<input checked="" type="checkbox"/> Name	<input checked="" type="checkbox"/> Isprivate	<input checked="" type="checkbox"/> Created at
<input checked="" type="checkbox"/> Updated at	<input checked="" type="checkbox"/> User		

Fields from associated contacts

<input checked="" type="checkbox"/> Id	<input checked="" type="checkbox"/> Name	<input checked="" type="checkbox"/> Postalcode	<input checked="" type="checkbox"/> Address
<input checked="" type="checkbox"/> City	<input checked="" type="checkbox"/> Phone	<input checked="" type="checkbox"/> Created at	<input checked="" type="checkbox"/> Updated at

Options for csv

Encode to Search Choose output encoding. Leave empty to let current input encoding untouched: (UTF-8)

No header Do not output a header (no fields description)

Column separator Search Leave blank for default (",")

Export to csv Export to json Export to xml Cancel

User | Mab Admin

maillist.pdf

Secure https://moroccanaddressbook.herokuapp.com/admin/user/2/edit

Mab Admin

Navigation

Address books

Contacts

User

Name Optional Length up to 255

Isadmin

Isreg

Address books

Search + Add a new Address book

AUI_Friends Choose all Clear all

Contacts

Search + Add a new Contact

Maha Laziri
Othmane Mahfou
Maha Hamdi
Achraf Mamdouh
Dr.Cavalli Sforza
Dr.Falah Bouchaib Choose all Clear all

Save Save and add another Save and edit Cancel

User | Mab Admin

maillist.pdf

Secure https://incompetech.com/beta/maillabel/maillist.pdf

Achraf Mamdouh Avenue Med VI Beni Mellal, Azilal 12345	Dr.Cavalli Sforza 12 Hay Riad Ifrane, Meknes-Fes 12345	Dr.Falah Bouchaib 13.Hay Marjane Casablanca, Grand Casablanca 12345
First Last 111 1st Street Anytown, NY 10101	Maha Laziri 28 Gold Street Newyork, NY 20390	Maha Hamdi Hay Nanda Rabat, Rabat-Zemour 12345

Font: Arial

Data:

Directions: The data you paste into this box needs to be **tab-delimited**, in the following format...
FName LName Address City State Zip

First Last 111 1st Street Anytown NY 10101
Maha Laziri 28 Gold Street Newyork NY 20390
Maha Hamdi Hay Nahda Rabat Rabat-Zemour 12345
Achraf Mamdouh Avenue Med VI Beni Mellal Azilal 12345
Dr.Cavalli Sforza 12 Hay Riad Ifrane Meknes-Fes 12345
Dr.Falah Bouchaib 13,Hay Marjane Casablanca Grand Casablanca 12345

Your data may overflow the bounds of this box, and may look odd - but it will still work.
If your spreadsheet application uses " for the data, that's fine... they will be stripped out.

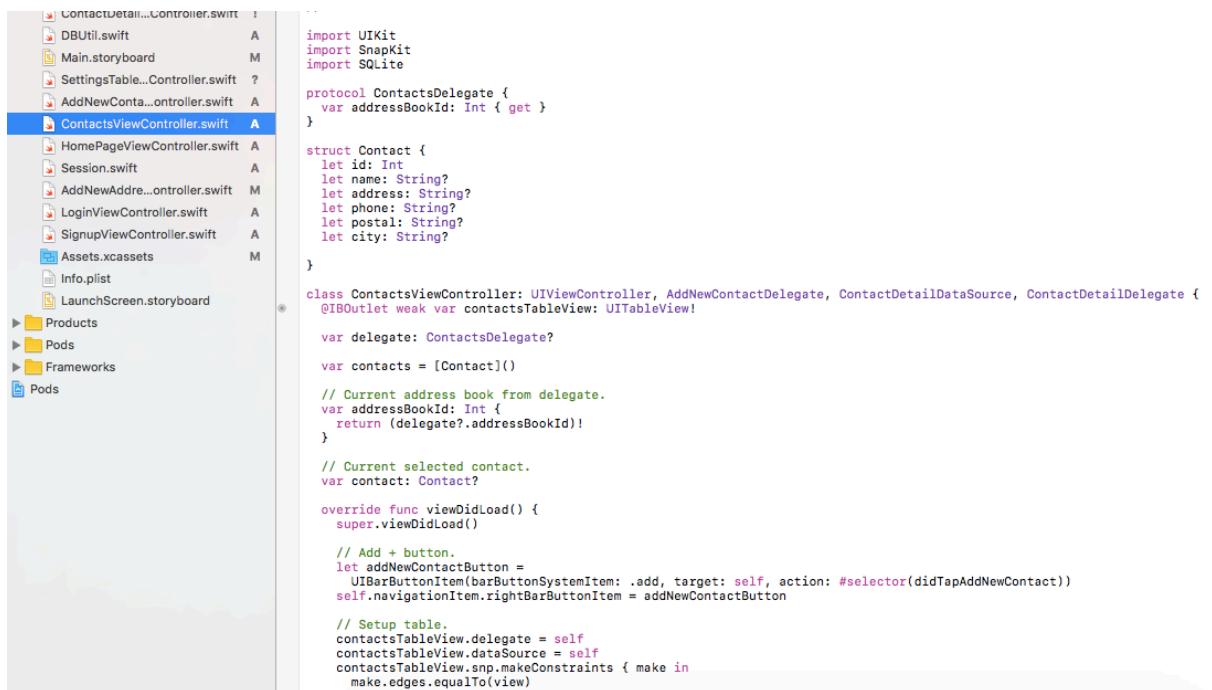
Get PDF

1.3. Mobile Application

1.3.1. Tools Used

- Swift language to code the mobile application.
- It is an IOS application.
- The IDE used is Xcode.
- SQLite is the database management system.

1.3.2. Code



The screenshot shows the Xcode interface with the 'ContactsViewController.swift' file selected in the left sidebar. The code editor displays the following Swift code:

```
import UIKit
import SnapKit
import SQLite

protocol ContactsDelegate {
    var addressBookId: Int { get }
}

struct Contact {
    let id: Int
    let name: String?
    let address: String?
    let phone: String?
    let postal: String?
    let city: String?
}

class ContactsViewController: UIViewController, AddNewContactDelegate, ContactDetailDataSource, ContactDetailDelegate {
    @IBOutlet weak var contactsTableView: UITableView!
    var delegate: ContactsDelegate?
    var contacts = [Contact]()
    // Current address book from delegate.
    var addressBookId: Int {
        return (delegate?.addressBookId)!
    }
    // Current selected contact.
    var contact: Contact?
    override func viewDidLoad() {
        super.viewDidLoad()
        // Add + button.
        let addNewContactButton =
            UIBarButtonItem(barButtonSystemItem: .add, target: self, action: #selector(didTapAddNewContact))
        self.navigationItem.rightBarButtonItem = addNewContactButton
        // Setup table.
        contactsTableView.delegate = self
        contactsTableView.dataSource = self
        contactsTableView.snp.makeConstraints { make in
            make.edges.equalTo(view)
        }
    }
}
```

```

DBUtil.swift A
Main.storyboard M
SettingsTable...Controller.swift ?
AddNewConta...ontroller.swift A
ContactsViewController.swift A
HomePageViewController.swift A
Session.swift A
AddNewAddress...ntroller.swift M
LoginViewController.swift A
SignupViewController.swift A
Assets.xcassets M
Info.plist
LaunchScreen.storyboard
Products
Pods
Frameworks
Pods

import UIKit
import SQLite

protocol AddNewContactDelegate {
    var addressBookId: Int { get }
    func didAddNewContact()
}

class AddNewContactViewController: UIViewController {
    @IBOutlet weak var firstNameTextField: UITextField!
    @IBOutlet weak var lastNameTextField: UITextField!
    @IBOutlet weak var addressTextField: UITextField!
    @IBOutlet weak var phoneTextField: UITextField!
    @IBOutlet weak var cityTextField: UITextField!
    @IBOutlet weak var postalCodeTextField: UITextField!
    @IBOutlet weak var errorLabel: UILabel!

    var delegate: AddNewContactDelegate?

    enum FormError {
        case error(String)
    }

    override func viewDidLoad() {
        super.viewDidLoad()

        // Do any additional setup after loading the view.
    }

    func handleError(with code: FormError) {
        switch code {
        case .error (let message):
            errorLabel.text = message
        }
        errorLabel.isHidden = false
    }

    @IBAction func didTapCreateAccount() {
        guard let delegate = delegate else {
            return
        }

        guard let firstname = firstNameTextField.text else {
            return
        }

        guard let lastname = lastNameTextField.text else {
            return
        }
    }
}

```

```

ContactDetail...Controller.swift I
DBUtil.swift A
Main.storyboard M
SettingsTable...Controller.swift ?
AddNewConta...ontroller.swift A
ContactsViewController.swift A
HomePageViewController.swift A
Session.swift A
AddNewAddress...ntroller.swift M
LoginViewController.swift A
SignupViewController.swift A
Assets.xcassets M
Info.plist
LaunchScreen.storyboard
Products
Pods
Frameworks
Pods

import UIKit
import SnapKit
import SQLite

protocol ContactsDelegate {
    var addressBookId: Int { get }
}

struct Contact {
    let id: Int
    let name: String?
    let address: String?
    let phone: String?
    let postal: String?
    let city: String?
}

class ContactsViewController: UIViewController, AddNewContactDelegate, ContactDetailDataSource, ContactDetailDelegate {
    @IBOutlet weak var contactsTableView: UITableView!

    var delegate: ContactsDelegate?
    var contacts = [Contact]()

    // Current address book from delegate.
    var addressBookId: Int {
        return (delegate?.addressBookId)!
    }

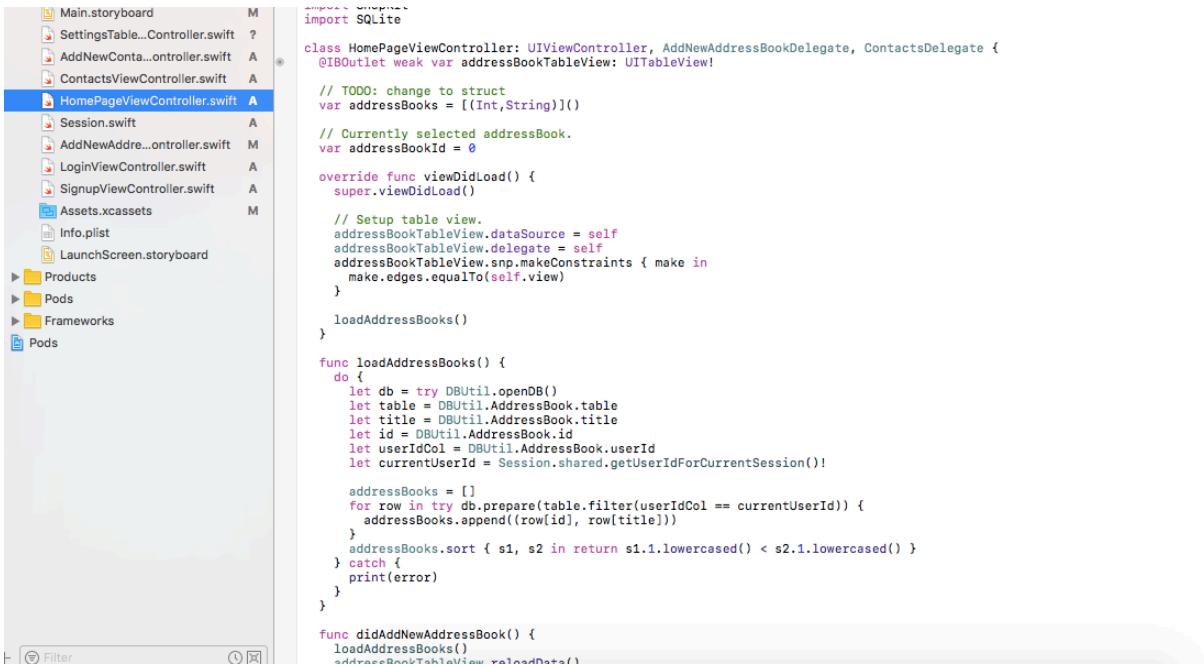
    // Current selected contact.
    var contact: Contact?

    override func viewDidLoad() {
        super.viewDidLoad()

        // Add + button.
        let addNewContactButton =
            UIBarButtonItem(barButtonSystemItem: .add, target: self, action: #selector(didTapAddNewContact))
        self.navigationItem.rightBarButtonItem = addNewContactButton

        // Setup table.
        contactsTableView.delegate = self
        contactsTableView.dataSource = self
        contactsTableView.snp.makeConstraints { make in
            make.edges.equalTo(view)
        }
    }
}

```



The screenshot shows the Xcode interface with the project navigation bar at the top. Below it is a list of files in the project structure:

- Main.storyboard
- SettingsTableController.swift
- AddNewContactController.swift
- ContactsViewController.swift
- HomePageViewController.swift (selected)
- Session.swift
- AddNewAddressController.swift
- LoginViewController.swift
- SignupViewController.swift
- Assets.xcassets
- Info.plist
- LaunchScreen.storyboard

On the right side, the content area displays the code for HomePageViewController.swift:

```
import SQLite

class HomePageViewController: UIViewController, AddNewAddressBookDelegate, ContactsDelegate {
    @IBOutlet weak var addressBookTableView: UITableView!
    // TODO: change to struct
    var addressBooks = [(Int, String)]()
    // Currently selected addressBook
    var addressBookId = 0

    override func viewDidLoad() {
        super.viewDidLoad()

        // Setup table view.
        addressBookTableView.dataSource = self
        addressBookTableView.delegate = self
        addressBookTableView.snp.makeConstraints { make in
            make.edges.equalTo(self.view)
        }

        loadAddressBooks()
    }

    func loadAddressBooks() {
        do {
            let db = try DBUtil.openDB()
            let table = DBUtil.AddressBook.table
            let title = DBUtil.AddressBook.title
            let id = DBUtil.AddressBook.id
            let userIdCol = DBUtil.AddressBook.userId
            let currentUserID = Session.shared.getUserIdForCurrentSession()!

            addressBooks = []
            for row in try db.prepare(table.filter(userIdCol == currentUserID)) {
                addressBooks.append((row[id], row[title]))
            }
            addressBooks.sort { $1, $2 in return $1.1.lowercased() < $2.1.lowercased() }
        } catch {
            print(error)
        }
    }

    func didAddNewAddressBook() {
        loadAddressBooks()
        addressBookTableView.reloadData()
    }
}
```

1.3.3. GUI



Noula AddressBook

username

password

Login

[I forgot my password](#) [Create new account](#)

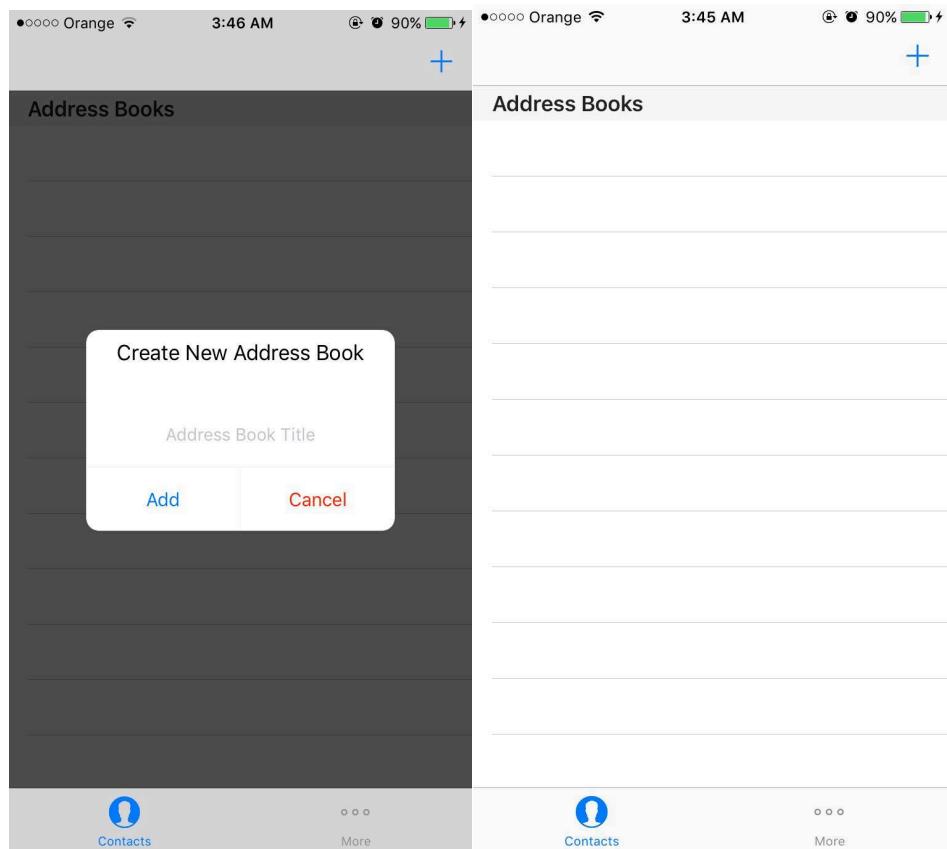
Sign Up

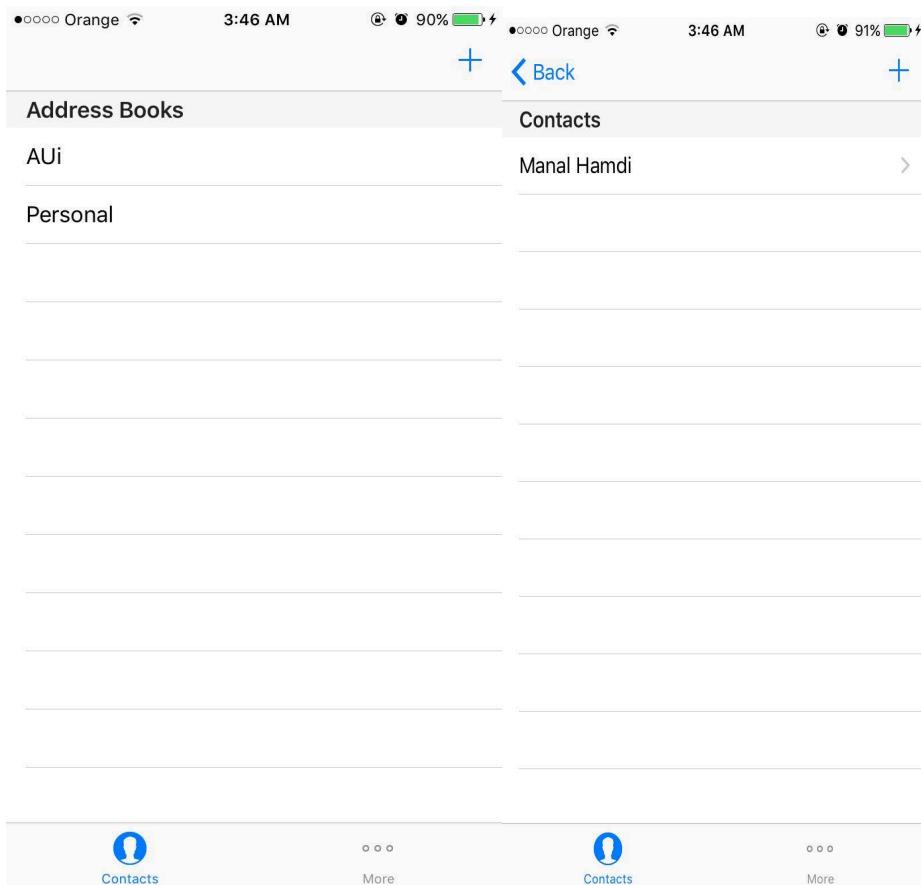
username

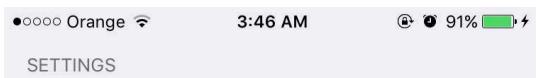
password

retype password

SignUp







SETTINGS

Logout

