

Project Report

Aliaxis VIP Contact

Implemented by: Luong Dinh Hieu

Email: hieu.luong@devoteam.com

August 12, 2016

TABLE CONTENT

Project summary	3
OBJECTIVE	3
REQUIREMENTS:	3
SOLUTION	4
PROJECT OUTLINE	4
Setup environment	6
REPOSITORY	6
INITIATE MOBILE SERVICE	7
INSTALL NPM	13
INSTALL FRAMEWORKS	14
INSTALL PLUGINS	15
ADD PLATFORMS	16
INSTALL IOS	17
INSTALL ANDROID OS	18
INSTALL WINDOWS OS	20
CONFIGURATION	21
BUILD & RUN	22
MORE INFO:	23
Screenshot	24

PROJECT SUMMARY

Objective

To develop an application that allows the users to import all “VIP” contact of Aliaxis company from Azure Active Directory server on mobile device.

Requirements:

1. Develop the Back-End
 - We will use Azure Active Directory as an authentication provider
 - We will create a mobile service on Windows Azure Directory, accessing the API to load all user contact information
 2. Develop the Client-Side (Application)
 - We need a framework that can run cross-platform (iOS, Android OS, Windows Phone OS, etc.)
 - The features we are going to develop are:
 - A. Login (use Azure Active Directory Authentication)
 - B. Import directly Aliaxis contact list
 - C. Logout
 3. Devices supported:
 - Android OS
 - iPhone OS
 - Windows Phone OS (from version 10.0 or higher)
 - Blackberry (from version 10.2.1 or higher, using Android apk file to install)
-

Solution

- Framework: [ionic2](#) to build our application, running on different mobile OS
- Web Technology: HTML5, Angular2, TypeScript
- Plugins:
 - A. cordova-plugin-ms-adal: to connect with the Azure Active Directory
 - B. cordova-plugin-x-contact: to work with the mobile user contact

Project Outline

TASK	DUE DATE	DONE	INITIALS
Set up Azure AD Server <ul style="list-style-type: none">- create a service to access user contact- create a contact list for test)		done	
Add plugin to connect with Azure AD		done	
Add plugin to work with the mobile user contact		done	
Create menu		done	
Create mobile user contact page		done	
Create aliaxis user contact page		done	
Implement login feature		done	
Implement Logout feature		done	
Implement feature “import” the aliaxis contact into the mobile user contact		done	
Implement feature “call”		done	
Implement feature “text”		done	

TASK	DUE DATE	DONE	INITIALS
Implement feature “share”		wait for the design	
Create About page		wait for the design	
Create Account page		wait for the design	
Implement Loading dialog		done	

SETUP ENVIRONMENT

Repository

- the source code can be found at:

<https://github.com/hieuld/aliascontact>

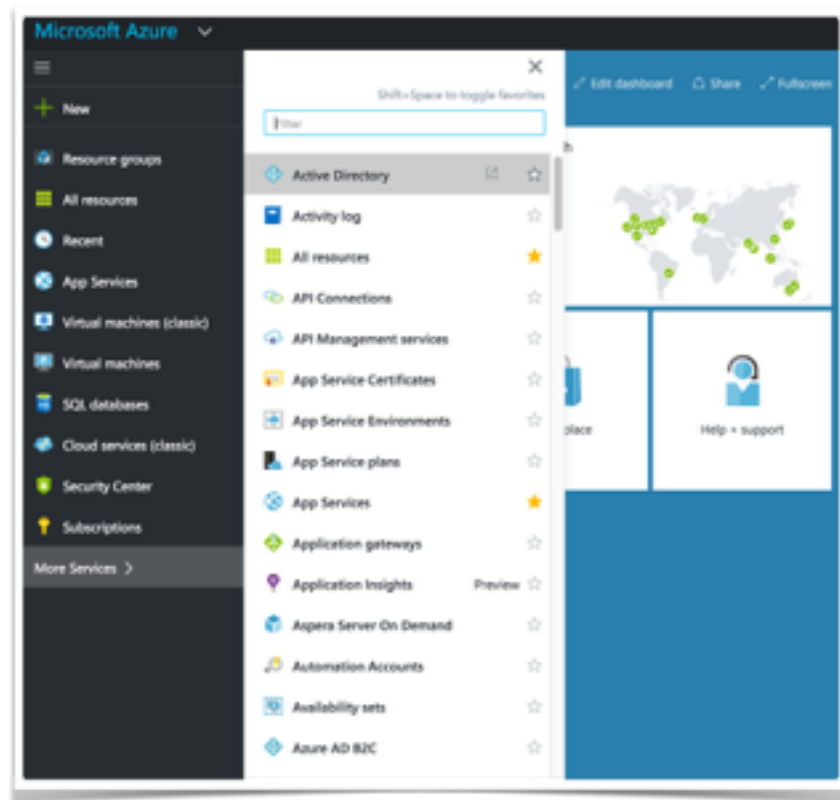
- download or clone the source code using command line:

git clone <https://github.com/hieuld/aliascontact> **aliascontact**

Initiate Mobile Service

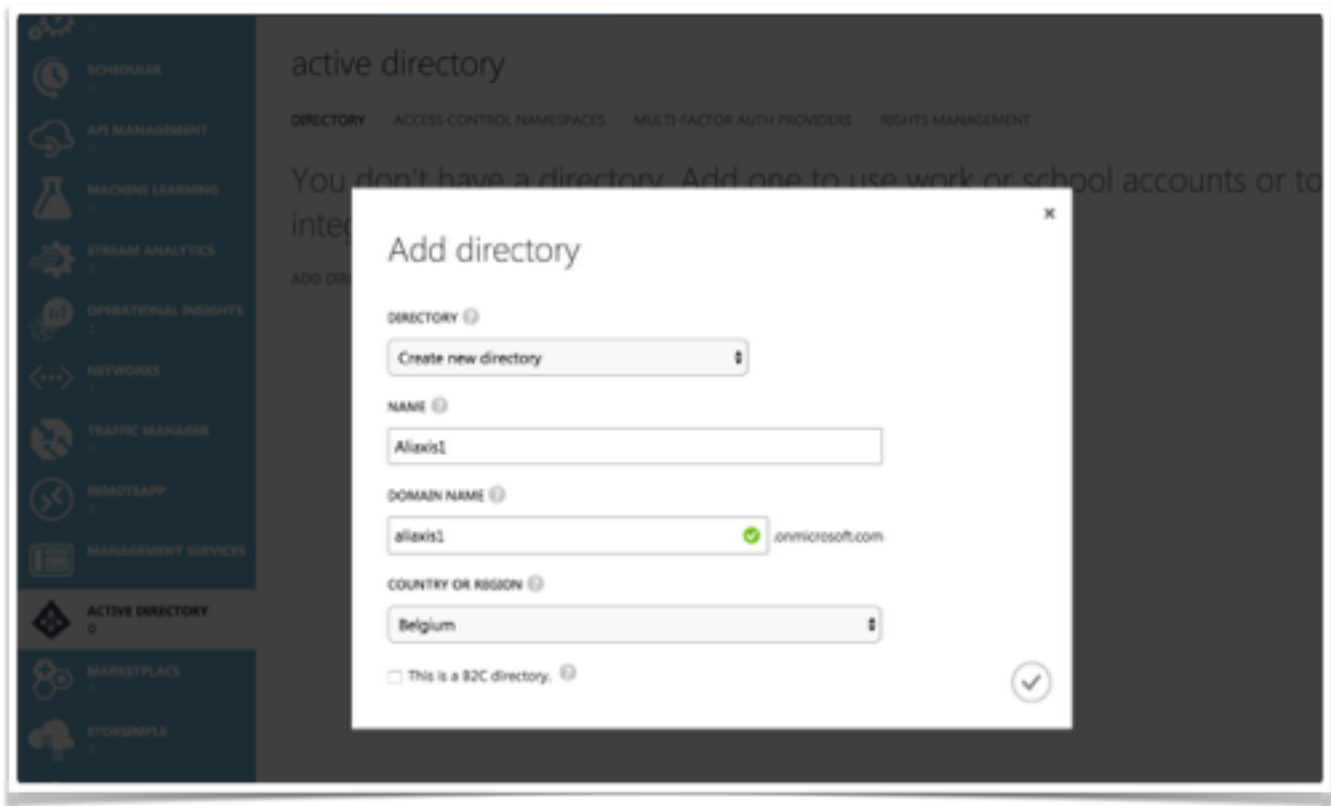
1. Login into Azure AD

- open portal.azure.com
 - login with the admin account
 - open Active Directory
- open More Service - Active Directory - login again (if needed)



2. Create a directory

open Active Directory - add directory (or use existing directory)



Name: Aliaxis1

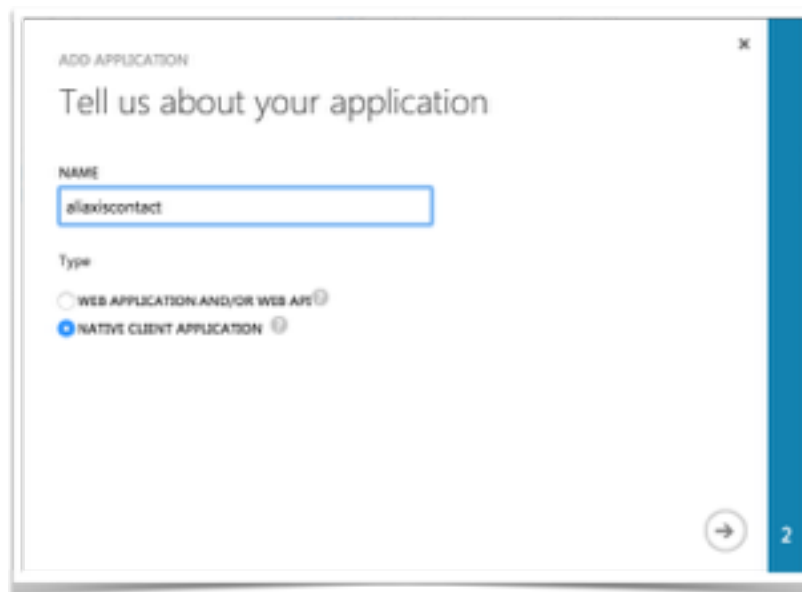
Domain Name: aliaxis1.onmicrosoft.com

Country or region: Belgium

All information above is up to you. You can choose whatever name you want if the name doesn't exist.

3. Register an application with Azure AD

- double click on the directory that we have created (Aliaxis1)
- select the tenant (aliaxis1)
- click the **Application** tab, and click **Add** in the bottom drawer
- add an application my organization is developing
- follow the prompts and create a new **Native Client Application**.

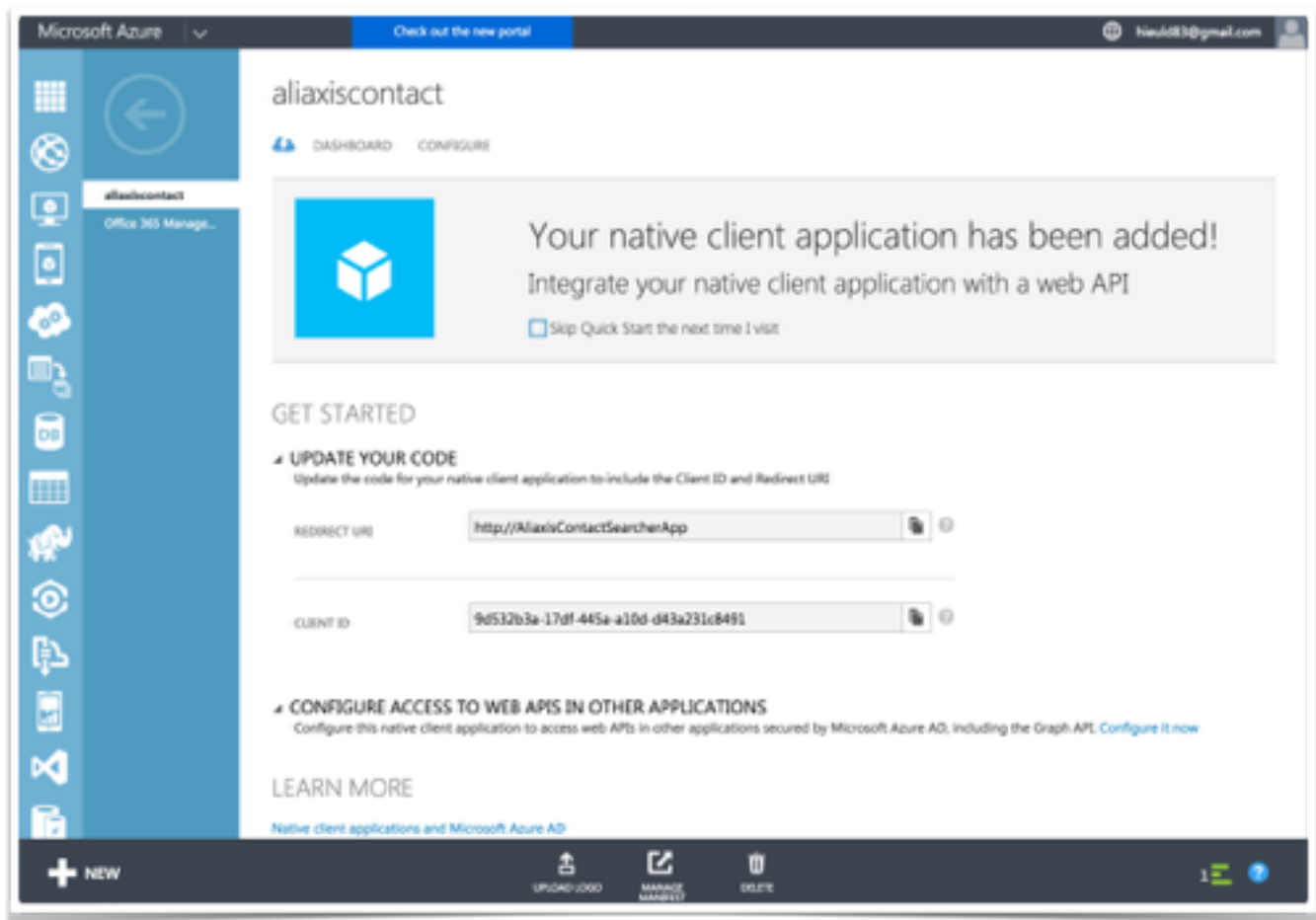


- the **Name** of the application will describe your application to end-users.
- the **Redirected URI** is the URI used to return tokens to your app. Enter:

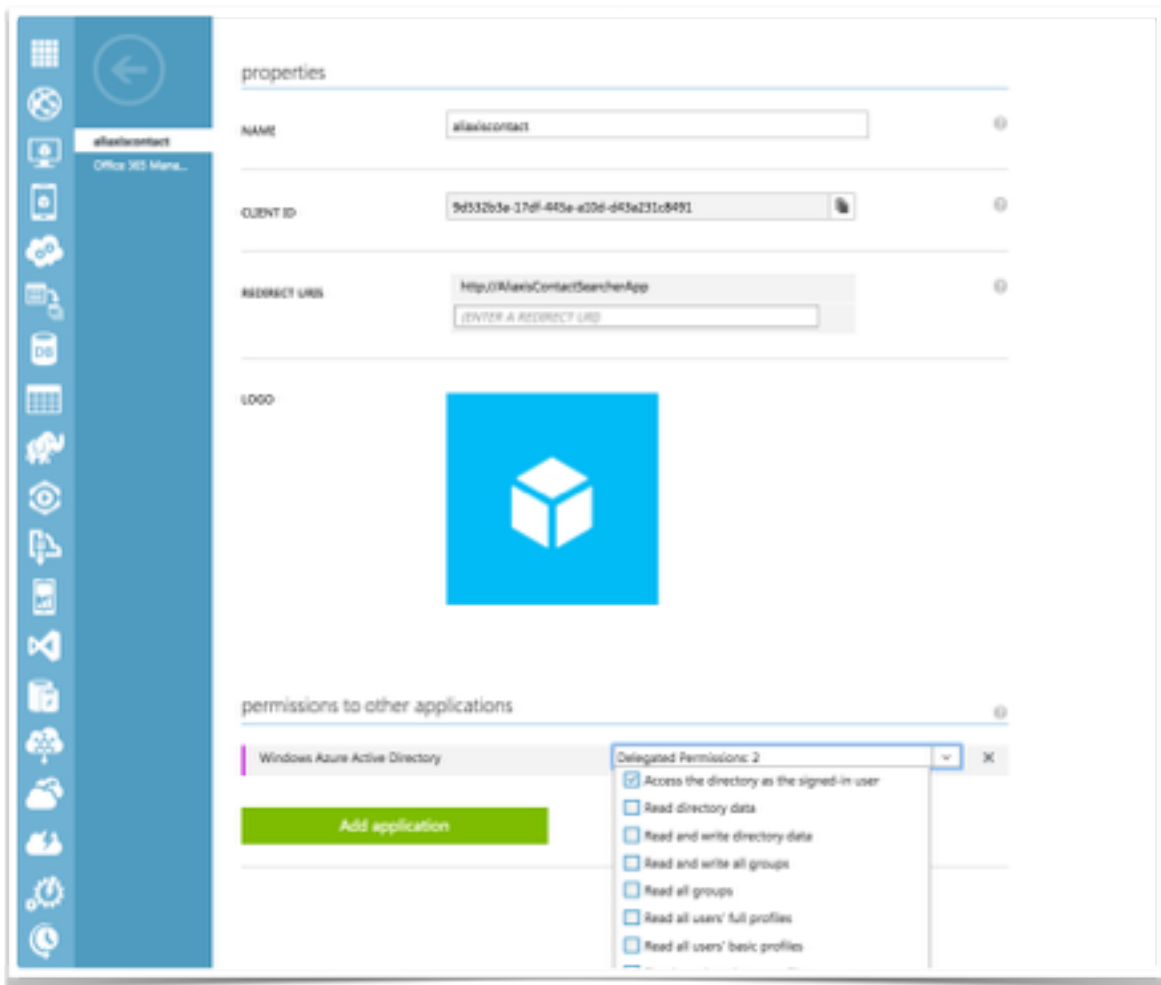
<http://AliaxisContactSearcherApp>

4. Configuration

- once you've completed registration, AAD will assign our app a unique client identifier (**Client ID**). We will need this value for the mobile application.

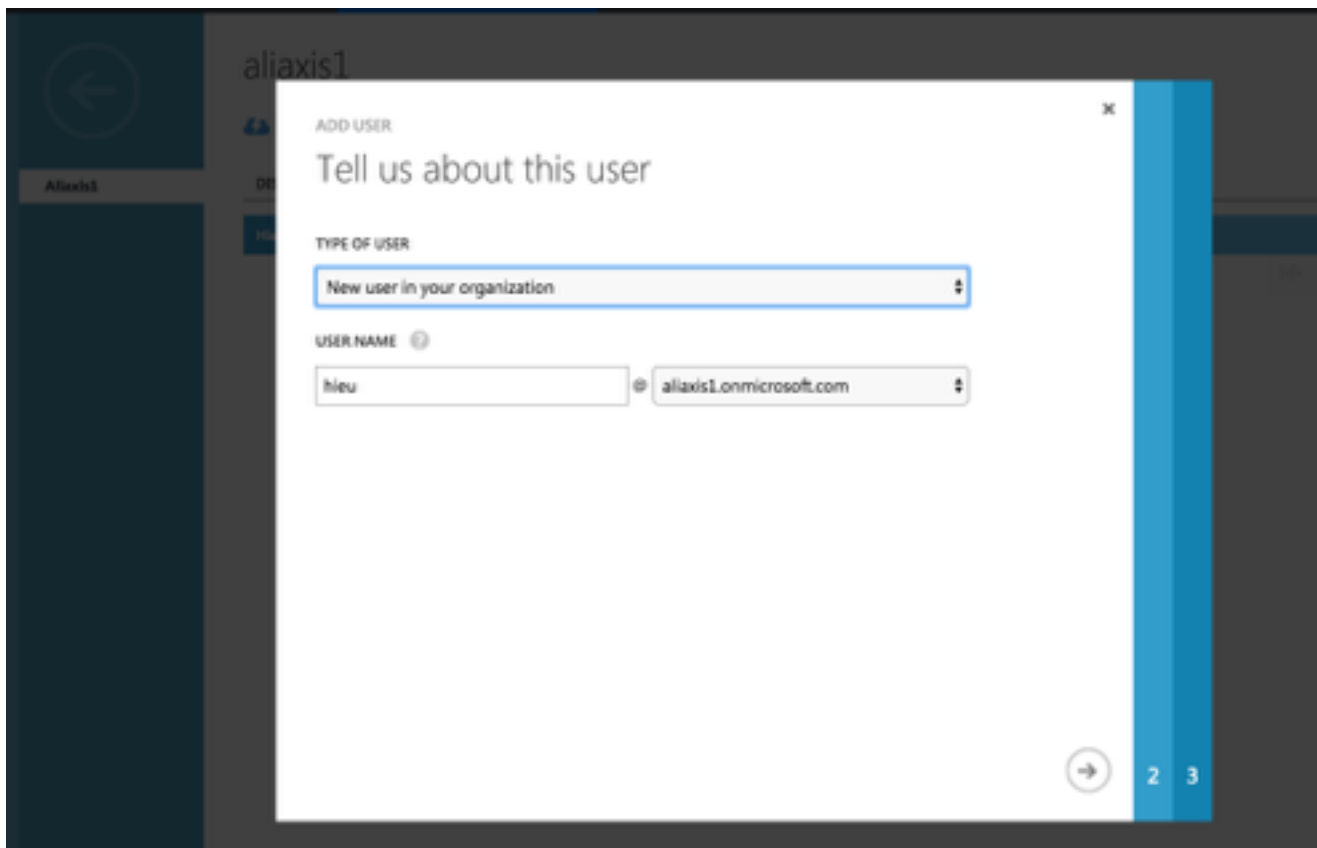


- grant the newly created app permission to query the Azure AD Graph API. In Configure tab, locate “Permissions to Other Applications”. For the “Azure Active Directory” application, add the **Access the directory as the signed-in user** permission under **Delegated Permissions**. This will enable your application to query the Graph API for users.
- **save** the setting.



5. Add Users

- click back to the Aliaxis1
- add Users (create a new one or choose the existing users)



The screenshot shows a web interface for 'Aliaxis1' with a dark sidebar. A modal dialog titled 'ADD USER' is open, with the subtitle 'Tell us about this user'. The dialog contains a dropdown menu for 'TYPE OF USER' with 'New user in your organization' selected. Below this is a 'USERNAME' section with a text input containing 'hieu' and a dropdown menu for the domain showing 'allaxis1.onmicrosoft.com'. At the bottom right of the dialog is a blue bar with a right arrow icon and the numbers '2' and '3'.

- if you create a new user, you will get the temporary password. Save it for the later use or reset password.
 - provide more information for the users (e.g., work place, mobile phone, display name, etc.)
-

Install npm

- download the npm from: <https://nodejs.org/en/download/>
- update the npm: `npm install npm -g`

Tips for windows users: I use **cygwin** - a similar Linux/Unix command line system in Windows. With **cygwin**, you can start using the command line. Download cygwin from here: <https://cygwin.com/install.html>

Install frameworks

- ionic2 is built on top of the framework cordova, so we need to install cordova first.

`npm install -g cordova`

`npm install -g ionic@beta`

- if the installation fails due to the lacking of any package, we can install these package.

`npm install -g package`

- install all packages needed, and then get back to install cordova and ionic2.

For more information: <http://ionicframework.com/docs/v2/getting-started/installation/>

Install plugins

- enter the directory of the project that we've downloaded.

`cd aliaxiscontact`

- install all packages

`npm install`

- install the azure plugins

`ionic plugin add cordova-plugin-ms-adal`

- install the contact plugin:

`ionic plugin add cordova-plugin-contacts`

- install the toast plugin:

`ionic plugin add cordova-plugin-x-toast`

Add Platforms

- add iOS platform:

`ionic platform add ios`

- add android platform:

`ionic platform add android`

- add windows platform:

`ionic platform add windows`

Another tips:

- how to list all platforms we've added: `ionic platform —list`
- how to remove a platform: `ionic platform remove ios [android | windows]`

Install iOS

Install Xcode from app store



Install the Simulator



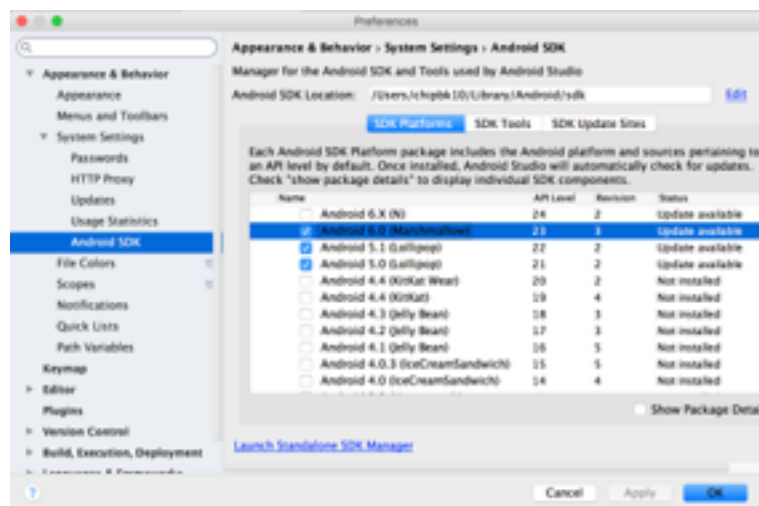
- open Xcode
 - from Menu - click on Xcode - Preference - Components Tab - Select iOS simulator
 - check and install now
-

Install Android OS

Install Android Studio

<https://developer.android.com/studio/index.html>

Install Android SDK



- open Android Studio
 - from menu - click on Android Studio - Preferences
 - on the left panel - click on Appearance & Behavior
 - System Settings - Android SDK
 - on the right panel - click on SDK platforms
 - select Android SDK - click OK
-

Install Android Emulator



- open Android Studio
 - from Menu - click on Tools - Android - AVD Manager
 - + Create Virtual Device ...
 - choose the Phone Device - Next - choose Android SDK && API level - Next
 - choose the name - Finish
-

Install Windows OS

Install Visual Studio

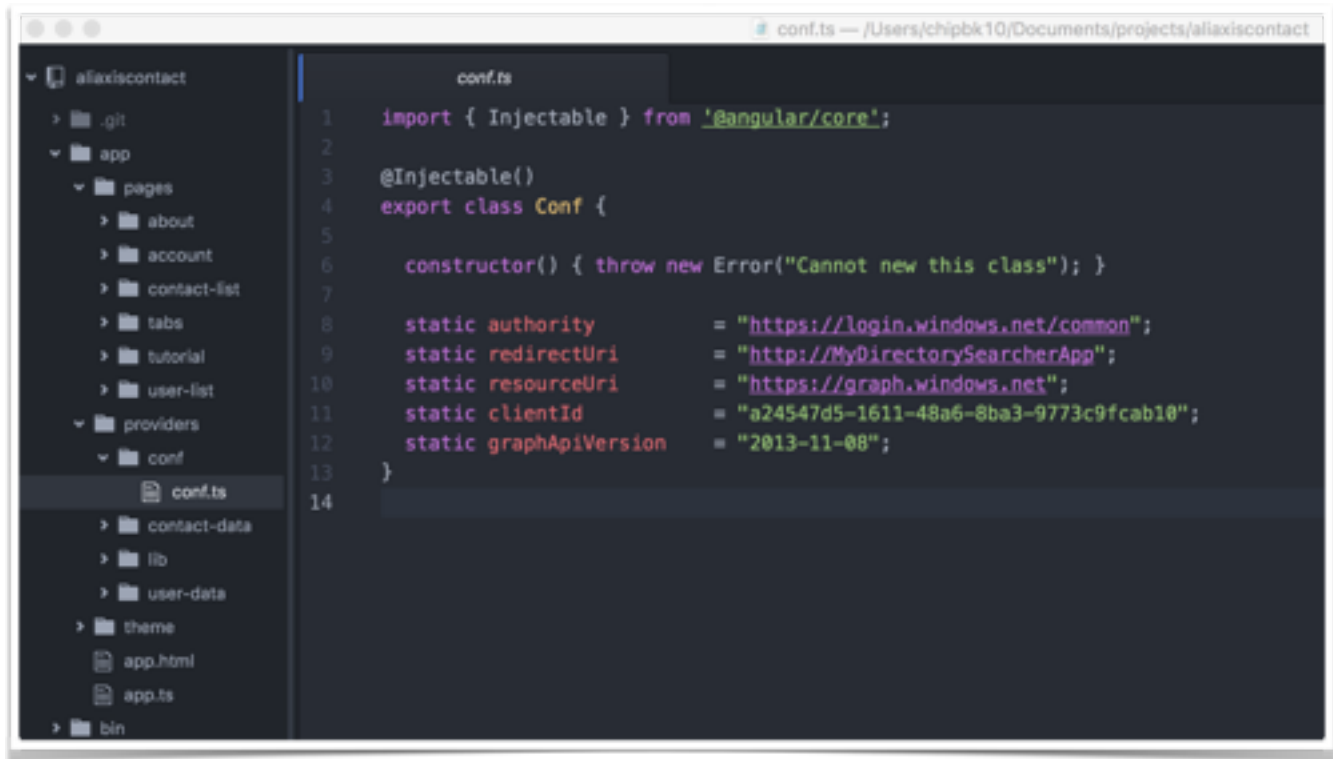
<https://beta.visualstudio.com/downloads/>

choose the Visual Studio Community for free download version.

Install Windows Phone SDK && Emulator

<https://developer.microsoft.com/en-us/windows/downloads/sdk-archive>

Configuration



- open the path: [aliaxiscontact/app/providers/conf/conf.ts](#)
 - set our redirect URI (see the **configuration** of the “**initiate mobile service**” section): [AliaxisContactSearcherApp](#)
 - set the client ID: ...
-

Build & Run

- build the project

`ionic build`

- plug the mobile device into your computer, and see the id of your device

`ionic run android —list`

`ionic run ios —list`

`ionic run windows —list`

- get the device's id above and deploy on the mobile device

`ionic run android —target="android-device-id"`

`ionic run ios —target="ios-device-id"`

`ionic run windows —target="windows-device-id"`

for example:

`ionic run android --target=3100cc6eafd7c2df`

(in which, 3100cc6eafd7c2df is the android device id)

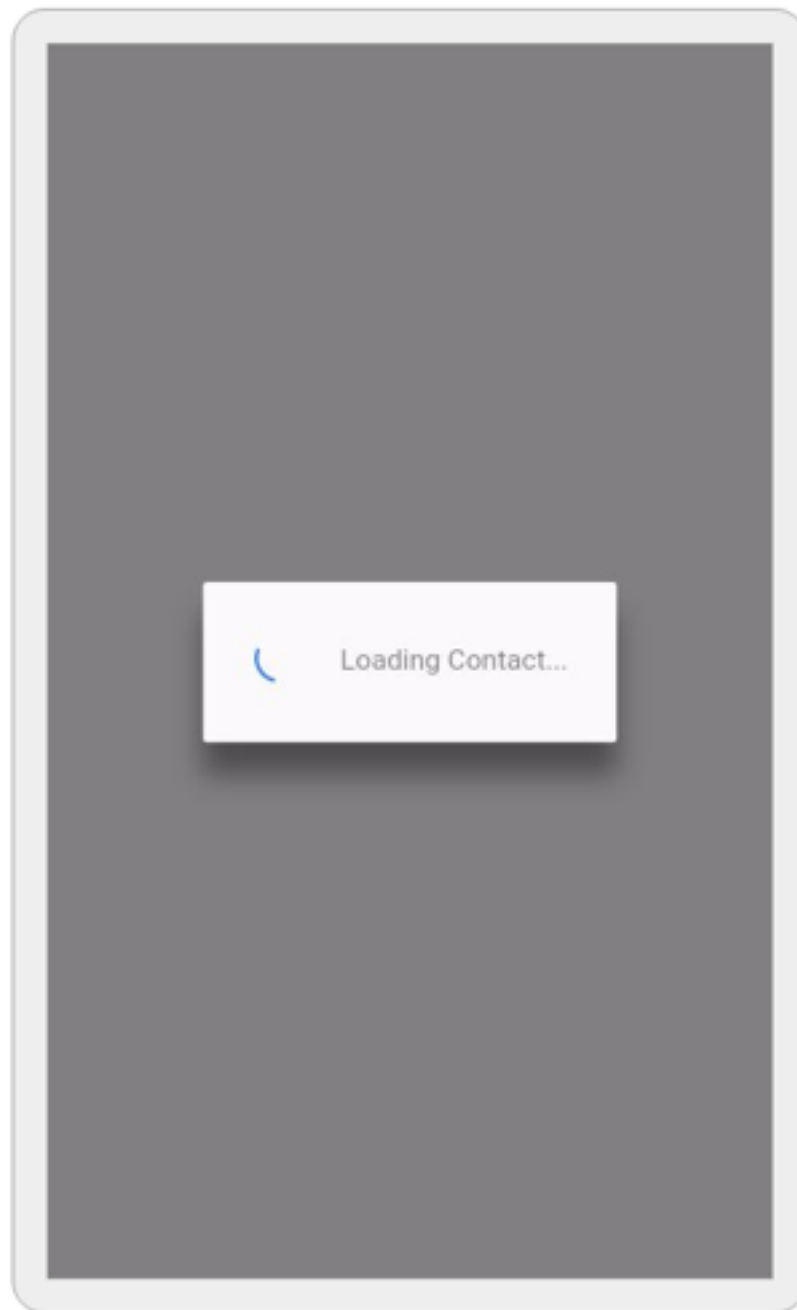
More Info:

For more information, please visit the web site :

<http://ionicframework.com/docs/v2/getting-started/installation/>

SCREENSHOT

1. LOADING CONTACT



2. AZURE AD LOGIN

MyDirectorySearcherApp

Work or school, or personal Microsoft account


Sign in

Back

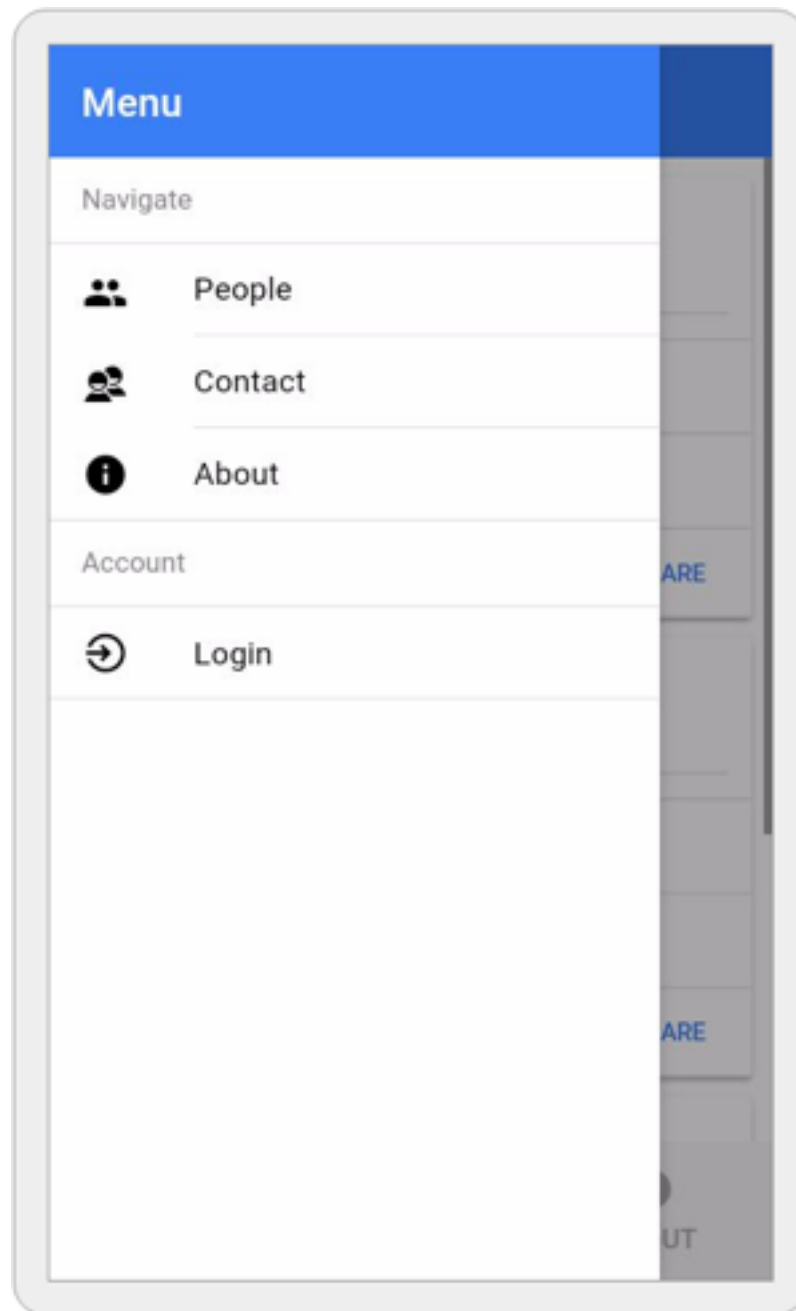
[Can't access your account?](#)

© 2016 Microsoft

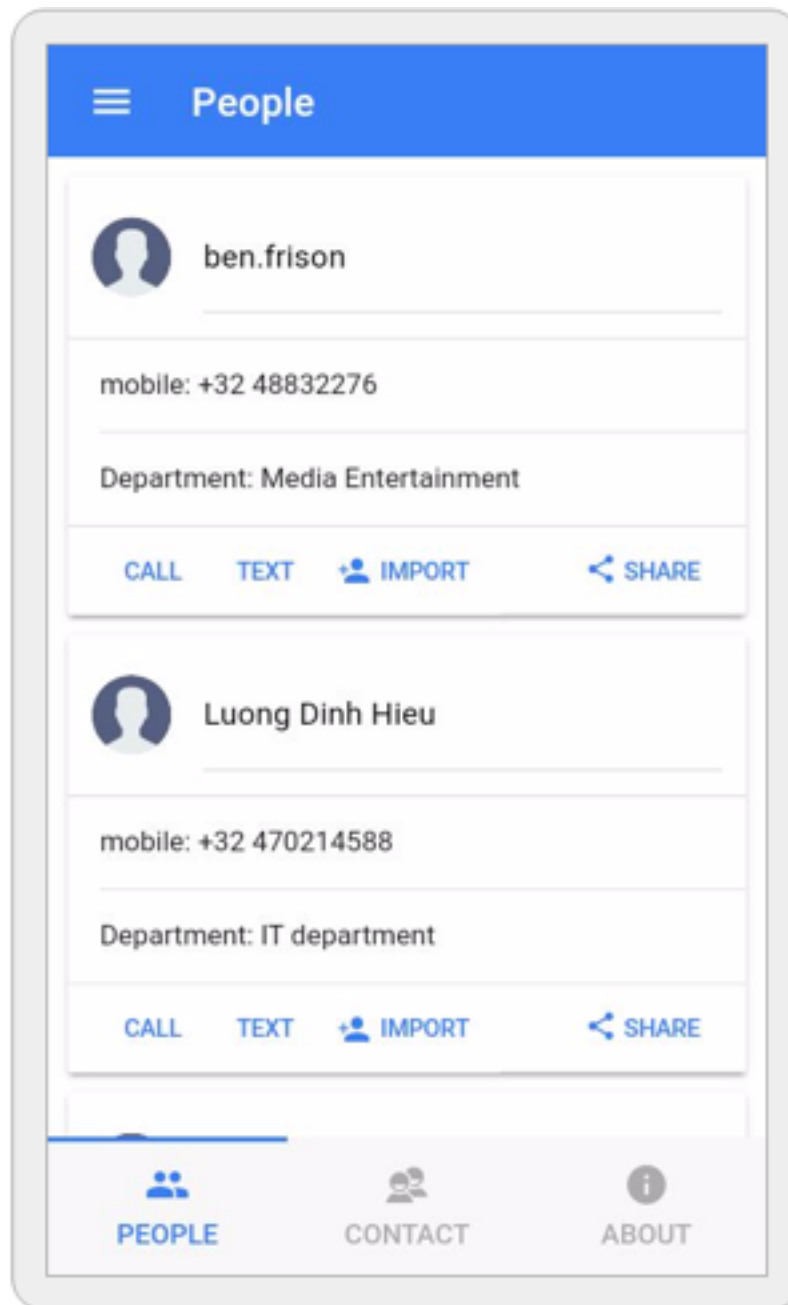
[Terms of use](#) [Privacy & Cookies](#)



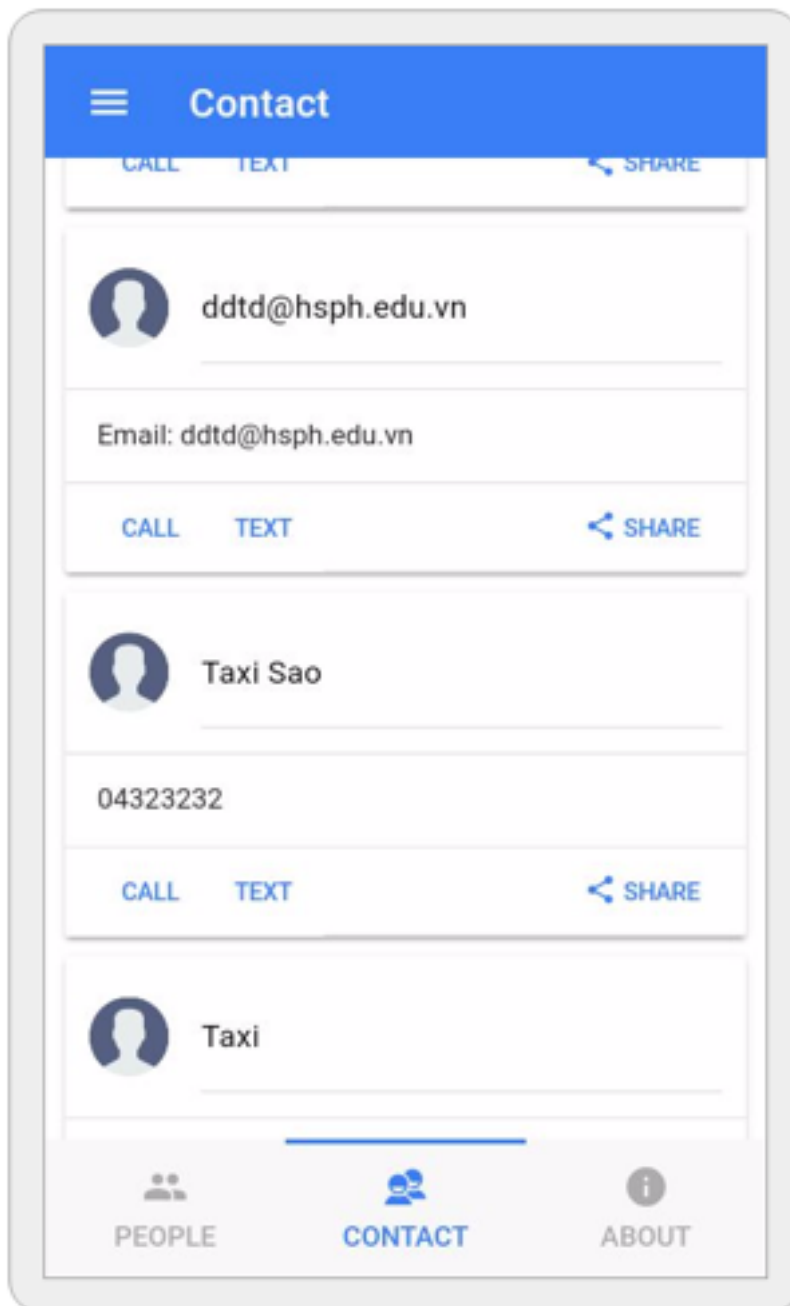
3. MENU



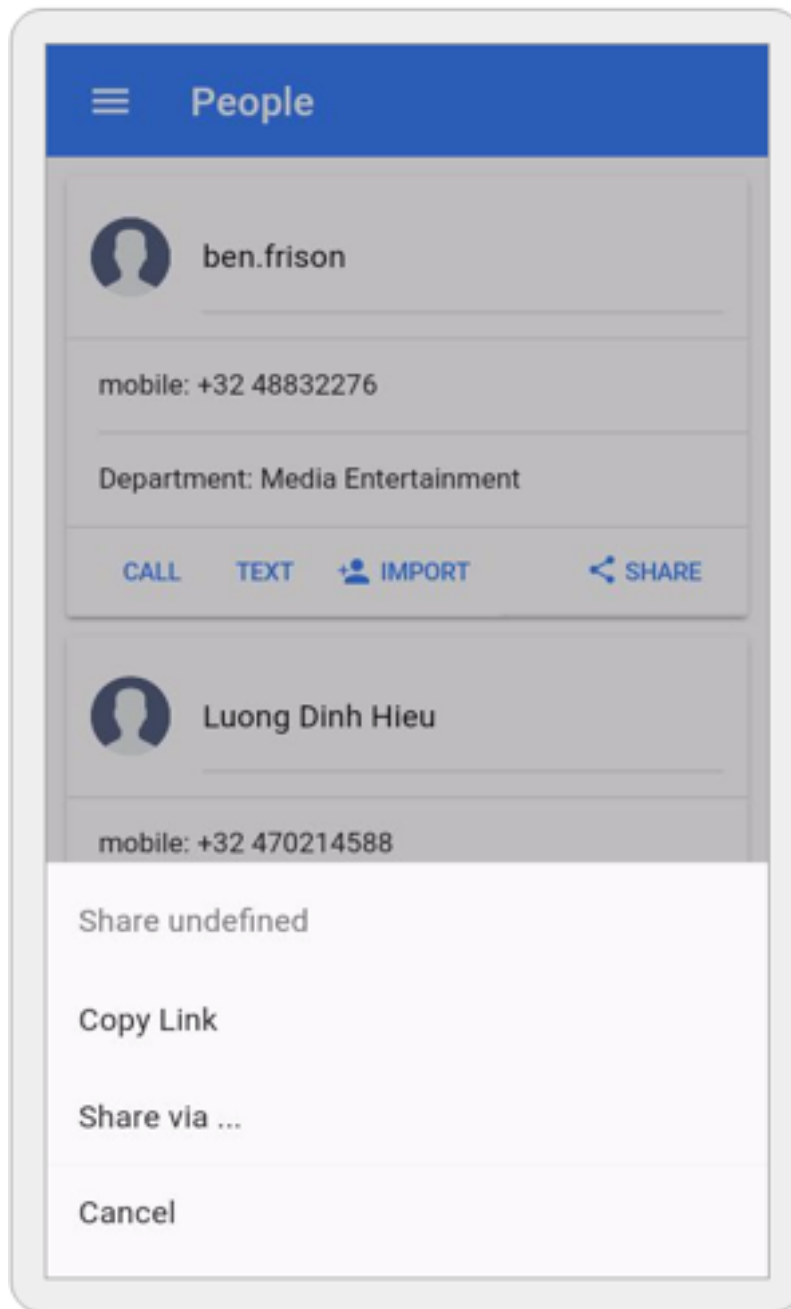
4. ALIAXIS USER CONTACT



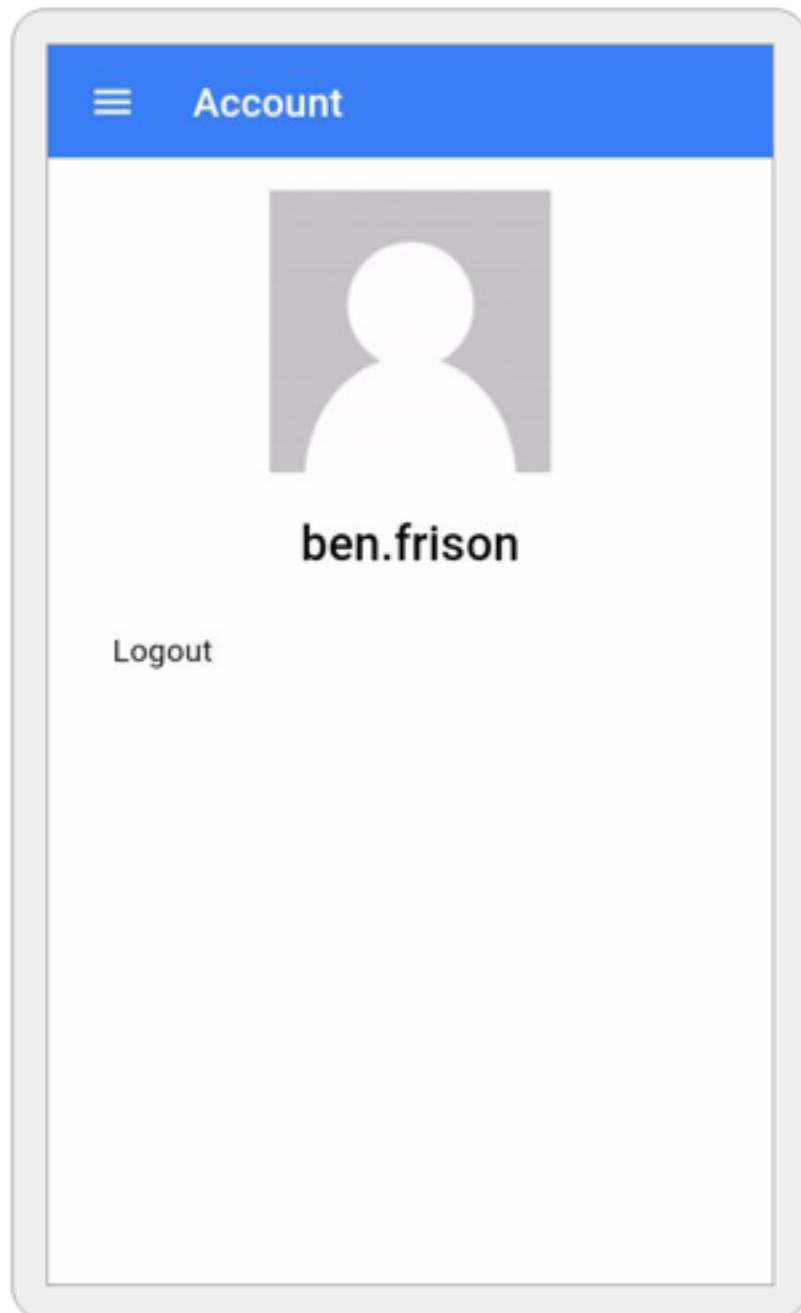
5. MOBILE USER CONTACT



6. SHARE FEATURE



7. ACCOUNT PAGE



8. ABOUT PAGE

