

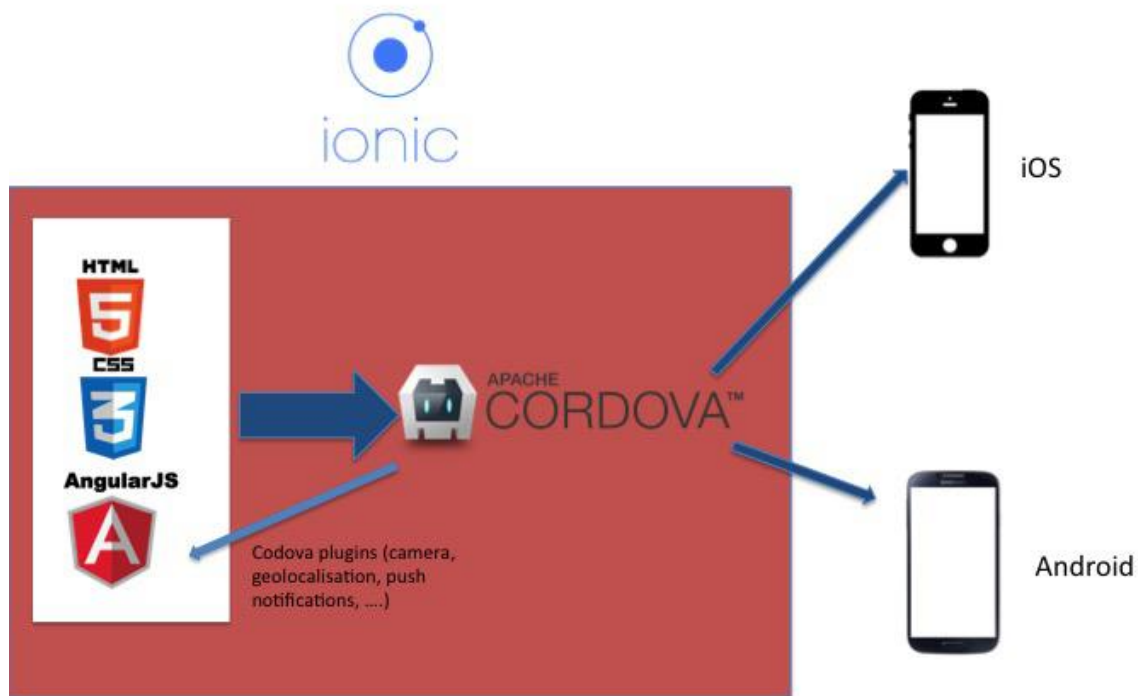
Software Architecture Documentation for the JRC – MYGEOSS app for Invasive Species project 2015.3724

Table of Contents

1	Architecture View	2
2	Application	3
2.1	Technologies Used	3
2.1.1	Apache Cordova v.5.4.1	3
2.1.2	Ionic v.1.1.0	3
2.1.3	AngularJS	4
2.1.4	ng-Cordova v0.1.20-alpha	4
2.1.5	leaflet.js v.0.7.7.....	4
2.2	Requirements	4
2.3	Target platforms and Devices	Error! Bookmark not defined.
2.4	Deployment	4
2.4.1	Folder Structure	4
2.4.2	Assets	6
2.5	Limitations	Error! Bookmark not defined.

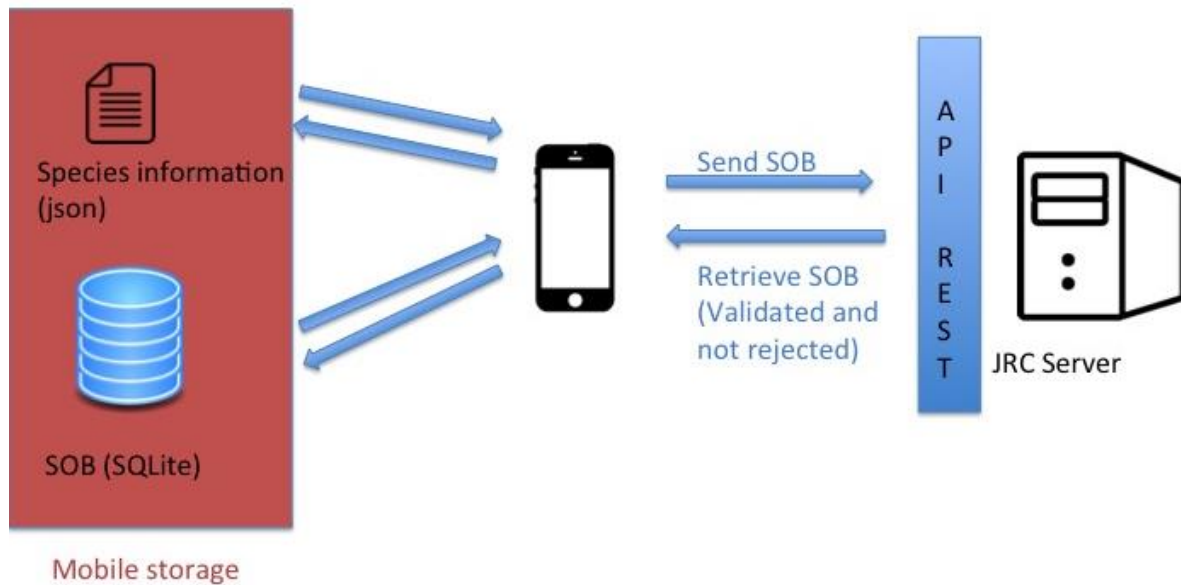
1 Architecture View

MYGEOSS is a hybrid mobile application available for Android (≥ 4.4) and iOS (≥ 8) that also runs on tablets. The application communicates with a webserver REST provided by the JRC to send and retrieve data.



Building hybrid mobile application with Ionic

We use the web technologies HTML 5, CSS3 and JavaScript to generate hybrid applications. In doing so we can manage a single code and asset repository for multiple platforms rather than having separate developments for each platform (In this case, Android and iOS).



How the application manages the data

The species information is embedded inside the app in a .json file, the Species Observation (SOB) is saved locally and uses SQLite to persistently store and retrieve information.

2 Application

2.1 Technologies Used

This application is built using the IonicFramework. This framework use Cordova to create hybrid mobile application and comes with a list of plugins to use the native functionality of the device. The applications are implemented as a browser-based WebView within the native mobile platform allowing it to use the common web technologies, more specifically HTML 5, CSS 3, and JavaScript.

Ionic also uses the Angular framework.

2.1.1 Apache Cordova v.5.4.1

Apache Cordova is a library that is used to create native mobile applications using Web technologies. The application is created using HTML, CSS and JavaScript and compiled for each specific platform using the native tools of the platform. Cordova provides a standard set of JavaScript APIs to access device features on all supported platforms.

<https://cordova.apache.org/>

2.1.2 Ionic v.1.1.0

Ionic is a complete open-source SDK for hybrid mobile app development. Built on top of AngularJS and Apache Cordova, Ionic provides tools and services for developing hybrid mobile apps using Web technologies such as CSS, HTML5, and Sass. Apps can be built with these Web technologies and then distributed through native app stores to be installed on devices through the use of Cordova.

<http://ionicframework.com/>

2.1.3 AngularJS

AngularJS is a structural framework for dynamic web apps. It lets you use HTML as your template language and lets you extend HTML syntax to express your application components clearly and succinctly. Angular's data binding and dependency injection eliminates much of the code you would otherwise have to write, all from within the browser, making it an ideal partner with any server technology.

This is the core of the application.

<https://angularjs.org/>

2.1.4 ng-Cordova v0.1.20-alpha

A library that optimises the Cordova plugins to work with AngularJS.

<http://ngcordova.com/>

2.1.5 leaflet.js v.0.7.7

Leaflet is an open-source JavaScript library for mobile-friendly interactive maps. At approximately 33 KB of JS, it has all the mapping features most developers need.

Leaflet is designed with simplicity, performance and usability in mind working efficiently across all major desktop and mobile platforms it can be extended with a large number of plugins, it has an easy to use and well-documented API.

It is used to display the map with the locations of the species.

<http://leafletjs.com/>

2.2 Platform Requirements

iOS 8+, Android 4.4+, smartphones and tablets.

Access to features of the device:

- geolocalization
- camera
- library
- network access

2.3 Deployment

The final deployment requires publishing the app on the Google app store and Apple app store, This is carried out by the Publications Office with information provided by the Author Service following the Publication Office document "*Guideline for mobile apps publishing on the European Union Accounts*".

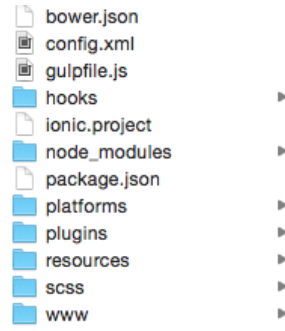
2.3.1 Folder Structure

We have three different structures. Cordova and Ionic provide one unique working directory (so both version iOS and Android use the same source code). The development code is inside the `www` folder. (More detail of this can be found in the *Programmers Manual*).

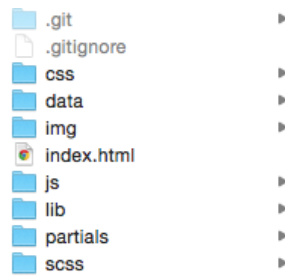
In the resources folder, we find the entire icon and splashscreen files for the entire available target.

We use Sass to generate the CSS files.

The `config.xml` file contains all of the important information like the bundle id or the version number of the application.

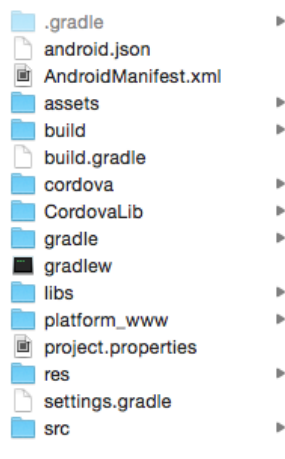


ionic/cordova working directory

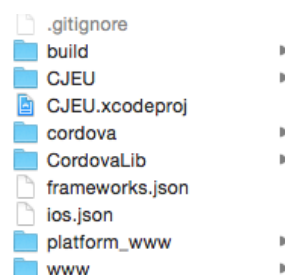


www directory

The code that is compiled to create native app is inside the platforms directory (Android and iOS). It is Cordova that generates these files. These files can be opened respectively with Android Studio and Xcode.



Android platform



iOS platform

2.3.2 Assets

The species information and species pictures (and thumbnails) are inside the `www/data` folder.

All of the icons and images for the app are inside the `img` folder.

END OF DOCUMENT

EPUB_Lot1_2015.3724_SAD_v020