

Create cross-platform mobile apps

**SII-301 4 Days (28 Hours)**

Description

This training provides an overview of techniques for creating cross-platform applications. Participants will learn the differences between traditional responsive web applications and mobile applications. They will also discover the different frameworks available and how to use mobile features such as data storage or geolocation.

Who is this training for ?

For whom

Mobility developers and project managers.

Prerequisites

- Good knowledge of the Web, HTML and CSS. Basic knowledge of JavaScript and XML.

Training objectives

- Identify the specificities of a mobile application
- Discover mobile development solutions
- Build a complete mobile application
- Understand the process of publishing a mobile application
- Manage a mobile project

Training program

Rappels sur les technologies de base

- Presentation of terminal families: Smartphones, tablets, their specificities.
- Reminder on HTML, CSS, JavaScript technologies.
- Reminder on modern development tools (NodeJS, NPM, Bower, Gulp.
-).
- Presentation of Webkit, Gecko engines.
- Importance and integration of the useragent.
- Ajax communication: importance and limits depending on mobile devices.
- Development, testing, simulation tools.
- Practical work Setting up a working environment dedicated to mobile.

Solutions embarquées

- Presentation of existing technologies and their development environments: limits, cost, portability.
- Objective-C and Swift (iPhone, iPad), Java (Android), C#.
- Net (Windows 10, Xamarin), Tizen, Firefox OS.
- Method of publishing an application in the official catalogs of operators (Apple Store, Google Play.
-).
- Administrative procedures (iPhone Developer Program, Certificates.
-).

Différence entre application Web traditionnelle et mobile

- Differences between browsers: JavaScript engines, CSS, plug-in limits (Flash, PDF.
-), other specificities.
- Difference in screen sizes and solutions.
- Hardware differences (CPU, memory).

Construction XHTML, HTML5

- META and specific tags: Viewport and dynamic CSS settings.
- Traditional page construction tags.
- Table constructions.
- Links and dedicated options accesskey.
- Images and the importance of images on terminals (format, weight of images).
- Texts p, span.
- Div blocks and canvas blocks in HTML 5.
- Forms.
- New data types (daterange, slider.
-).
- Activation of the numeric keypad.
- CSS, importance in multichannel mode.
- Positioning of elements, navigation between elements (zindex, display.
-).
- Bringing CSS3 to HTML5.
- Simplify the creation and maintenance of CSS with Bootstrap, Sass, Compass and Less.
- Importance of the DOM for multi-terminal porting.
- Mobile environment (sensors, battery, network detection.
-).
- Multitouch events specific to terminals (gesture, touch , drag and drop.
-).
- Communication protocols (websocket, ajax), advantages of Node.
- js.
- Graphics library: drawing API (rectangle, line.
-), color palette, image manipulation.
- W3C conformance tests: validation tools.
- Practical work Implementation of input forms; buttons and components; rotation of the terminal by CSS; dynamic management of "touch" events; of drag&drop; of mini drawing manager.
- Setting up a websocket server with Node.
- js.
- Detecting network outages.

Stockage des données dans le mobile

- Implementation of a "disconnected mode" strategy.
- Manifest and caching of mobile site resources.
- SQLite databases , indexedDB integrated into the browser.
- SQL language and administration tools.
- Management and control in JavaScript (table creation, queries.
-).
- Use of Google Gears for terminals not compatible with HTML 5.
- Client-side cache management for working in offline mode (localStorage, sessionStorage).
- Practical work Creation of a note manager with storage in the embedded database.

Cartographie et géolocalisation

- Managing the Google Maps map.
- Options for supporting mobile geolocation.
- Practical work Displaying the map in relation to the location of the mobile and display of markers on the map.