

Developing cross-platform applications for mobile

Hands-on course of 4 days - 28h

Ref.: MOB - Price 2026: 2 470 (excl. taxes)

EDUCATIONAL OBJECTIVES

At the end of the training, the trainee will be able to:

Discover mobile development solutions.

Build a complete mobile application.

Understand the publication process of a mobile app.

Oversee a mobile project

Participants

A smartphone will be provided to attendees so that they can test the apps they will be developing throughout this course

TRAINING PROGRAM

THE PROGRAMME

last updated: 08/2024

1) » Reminders about basic technologies

- Overview of terminal families: Smartphones, tablets, their unique features.
- Reminder about HTML, CSS, and JavaScript technology.
- Reminders about modern development tools (NodeJS, NPM, Bower, Gulp, etc.).
- Overview of the Webkit, Gecko, etc. engines.
- Importance and integration of the useragent.
- Ajax communication: Importance and limit based on different devices.
- Development, testing, and simulation tools.
- Hands-on work ↗Setting up a work environment dedicated to mobile apps.

2) » On-board solutions

- Overview of existing technologies and their development environments: Limits, costs, portability.
- Objective-C and Swift (iPhone, iPad), Java (Android), C# .Net (Windows Phone), Tizen, Firefox OS.
- Method for publishing an app in the operators' official catalogs (App Store, GooglePlay, etc.).
- Administrative steps (iPhone Developer Program, Certifications, etc.).

3) » Differences between traditional and mobile Web applications

- Differences between browsers: JavaScript engines, CSS, plug-in limits (Flash, PDF, etc.), other unique features.
- Differences in size of screens and solutions.
- Hardware differences (CPU, memory, etc.).

4) » XHTML, HTML5 construction

- META and specific tags: Viewport and dynamic CSS configuration.
- Traditional page-building tags.
- Building tables.
- accesskey links and dedicated options.
- Images and the importance of images on terminals (format, image size).

TRAINER QUALIFICATIONS

The experts leading the training are specialists in the covered subjects. They have been approved by our instructional teams for both their professional knowledge and their teaching ability, for each course they teach. They have at least five to ten years of experience in their field and hold (or have held) decision-making positions in companies.

ASSESSMENT TERMS

The trainer evaluates each participant's academic progress throughout the training using multiple choice, scenarios, hands-on work and more. Participants also complete a placement test before and after the course to measure the skills they've developed.

TEACHING AIDS AND TECHNICAL RESOURCES

- The main teaching aids and instructional methods used in the training are audiovisual aids, documentation and course material, hands-on application exercises and corrected exercises for practical training courses, case studies and coverage of real cases for training seminars.
- At the end of each course or seminar, ORSYS provides participants with a course evaluation questionnaire that is analysed by our instructional teams.
- A check-in sheet for each half-day of attendance is provided at the end of the training, along with a course completion certificate if the trainee attended the entire session.

TERMS AND DEADLINES

Registration must be completed 24 hours before the start of the training.

ACCESSIBILITY FOR PEOPLE WITH DISABILITIES

Do you need special accessibility accommodations? Contact Mrs. Fosse, Disability Manager, at psh-accueil@ORSYS.fr to review your request and its feasibility.

- p and span texts. div blocks and canvas blocks in HTML 5.
- Forms. New data types (daterange, slider, etc.). Activating the number pad.
- CSS, importance in multichannel mode. Positioning of elements, navigation between elements (zindex, display, etc.).
- What CSS3 adds to HTML5. Simplifying the creation and maintenance of CSS with Bootstrap, Sass, Compass, and Less.
- Importance of DOM for multiterminal porting.
- Mobile environment (sensors, battery, network detection, etc.).
- Terminal-specific multitouch events (gesture, touch, drag-and-drop, etc.).
- Communication protocols (websocket, ajax), advantages of Node.js.
- Graphics library: Drawing API (rectangle, line, etc.), color palette, working with images.
- W3C compliance tests: Validation tools.
- Hands-on work ↗Implementation of input forms; buttons and components; rotation of the terminal by CSS; dynamic management

5) » Multiplatform framework

- Advantages of using a framework.
- Choosing a mobile framework.
- Cordova (PhoneGap) and its publication tools.
- AngularJS 2.0, a mobile-development-oriented framework.
- Mobile Angular UI: Building HTML5 Mobile Apps with Bootstrap and Angular JS.
- Ionic: Building native apps with Cordova and Angular JS.
- Frameworks for Material Design (Polymer, Materialize CSS, etc.).
- Hands-on work ↗Implementing and discovering different frameworks available on the market. Carrying out a Material Design

6) » Data storage in mobile devices

- Setting up an “off-line mode” strategy.
- Manifest and caching the mobile site's resources.
- Browser-integrated SQLite and indexedDB databases.
- SQL language and administration tools.
- Management and oversight in JavaScript (creating tables, queries, etc.).
- Using Google Gears for terminals not compatible with HTML 5.
- Managing the client-end cache for work in offline mode (localStorage, sessionStorage).
- Hands-on work ↗Creating a note manager with storage in the on-board database.

7) » Mapping and geolocation

- Managing Google Maps.
- Options for handling the device's geolocation.
- Hands-on work ↗Displaying the map with respect to the device's location and displaying markers on the map.

DATES

Contact us