



Course guides

804247 - DMOB - Mobile Devices

Last modified: 20/09/2021

Unit in charge: Image Processing and Multimedia Technology Centre
Teaching unit: 804 - CITM - Image Processing and Multimedia Technology Centre.

Degree: BACHELOR'S DEGREE IN VIDEO GAME DESIGN AND DEVELOPMENT (Syllabus 2014). (Compulsory subject).

Academic year: 2021 **ECTS Credits:** 6.0 **Languages:** Catalan, Spanish, English

LECTURER

Coordinating lecturer: Fernández Duran, Pau

Others: Pau Fernández Durán

REQUIREMENTS

Knowing a high-level programming language (C, C++, Java, Python, etc.)

DEGREE COMPETENCES TO WHICH THE SUBJECT CONTRIBUTES

Generical:

CGFC4VJ. Apply basic algorithmic procedures of information technology to designing solutions for problems, analysing the suitability and complexity of the proposed algorithms.

CGFC5VJ. Efficiently design and use the most appropriate types and structures of data to solve a problem related to the development of video games.

Transversal:

01 EIN N3. ENTREPRENEURSHIP AND INNOVATION - Level 3. Using knowledge and strategic skills to set up and manage projects. Applying systemic solutions to complex problems. Devising and managing innovation in organizations.

TEACHING METHODOLOGY

Sessions with lectures intertwined with programming practice

LEARNING OBJECTIVES OF THE SUBJECT

- Show knowledge and be able to use libraries to create video games and applications on mobile devices and / or other devices.
- Show knowledge and mastery, and be able to explain, the technologies for the design and creation of video games and applications on mobile devices and / or other devices.
- Show ability to analyze the technical characteristics of the technologies for the creation of video games and applications on mobile devices and / or other devices.

STUDY LOAD

Type	Hours	Percentage
Self study	90,0	60.00
Hours medium group	30,0	20.00
Guided activities	12,0	8.00
Hours large group	18,0	12.00



Total learning time: 150 h

CONTENTS

1. The Dart programming language

Description:

Types and variables. Control flow. Functions. Advanced functions. Data Structures. Classes. Inheritance.

Full-or-part-time: 20h

Practical classes: 8h

Self study : 12h

2. Basic Flutter

Description:

Hello, world. Hot Reload. Basic Widgets. Layout Widgets. User defined Widgets.

Full-or-part-time: 20h

Practical classes: 8h

Self study : 12h

4. Advanced Flutter

Description:

StatefulWidgets. Buttons. Screens, parameter passing. Lists. Grids.

Full-or-part-time: 30h

Practical classes: 12h

Self study : 18h

4. Apps with Flutter

Description:

The model. State management. Provider. Local persistence.

Full-or-part-time: 30h

Practical classes: 12h

Self study : 18h

5. Firebase

Description:

Firebase Auth, Cloud Firestore and Firebase Storage. Futures. Streams. Cloud Firestore: collections and documents.

Full-or-part-time: 30h

Theory classes: 12h

Self study : 18h



6. Animations

Description:

Animation Widgets. Tweens. The Animation class. Sequences. Animations with Rave.

Full-or-part-time: 20h

Practical classes: 8h

Self study : 12h

ACTIVITIES

Deliverable 1: Layout for a screen

Description:

Given a professional design, develop a whole screen (only the graphic design).

Full-or-part-time: 5h

Guided activities: 5h

Deliverable 2: App without networking

Description:

Development of an app with a few screens with local persistence.

Full-or-part-time: 10h

Guided activities: 10h

Deliverable 3: App using Firebase

Description:

Development of an app which uses Firebase.

Full-or-part-time: 20h

Guided activities: 20h

GRADING SYSTEM

Mid-term exam: 25%

Deliverables: 40%

Final exam: 25%

Participation and disposition: 10%

Re-evaluation is available. Only the 50% corresponding to the two exams will be re-evaluated.

BIBLIOGRAPHY

Basic:

- Alberto Miola. Flutter Complete Reference: Create beautiful, fast and native apps for any device [on line]. Publicación Independiente, 2020 Available on: <https://fluttercompletereferece.com>.



RESOURCES

Hyperlink:

- Web de Flutter. <https://flutter.dev>
- Web de Dart. <https://dart.dev>