1. COURSE
CS2B1. Platform Based Development (Mandatory)

2. GENERAL INFORMATION
2.1 Credits	:	3
2.2 Theory Hours	:	1 (Weekly)
2.3 Practice Hours	:	2 (Weekly)
2.4 Duration of the period	:	16 weeks
2.5 Type of course	:	Mandatory
2.6 Modality	:	Face to face
2.7 Prerequisites	:	CS112. Computer Science I. (2nd Sem)

3. PROFESSORS

Meetings after coordination with the professor

4. INTRODUCTION TO THE COURSE
The world has changed due to the use of fabric and related technologies, rapid, timely and personalized access to the information, through web technology, ubiquitous and pervasive; they have changed the way we do things, how do we think? and how does the industry develop? Web technologies, ubiquitous and pervasive are based on the development of web services, web applications and mobile applications, which are necessary to understand the architecture, design, and implementation of web services, web applications and mobile applications.

5. GOALS

- That the student is able to design and implement services, web applications using tools and languages such as HTML, CSS, JavaScript (including AJAX), back-end scripting and a database, at an intermediate level.

- That the student is able to develop mobile applications, administration of web servers in a Unix system and an introduction to web security, at an intermediate level.

6. COMPETENCES

c) An ability to design a system, component, or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability. (**Usage**)

d) An ability to function on multidisciplinary teams. (**Usage**)

g) The broad education necessary to understand the impact of computing solutions in a global, economic, environmental, and societal context. (**Usage**)

i) An ability to use the techniques, skills, and modern computing tools necessary for computing practice. (**Usage**)

7. SPECIFIC COMPETENCES

c3) Use different tools and programming languages in the software components (*Full stack*).

c4) Design and implement scalable software architectures in different platforms.

c5) Describe how platform-based development differs from the general purpose of programming.

c6) Apply the advantages and disadvantages of cross-platform constraints.

c7) Apply or implement Web platform constraints in software development

c8) Apply web standards.

c9) Apply development standards for mobile devices

c10) Implement software as a service.

d1) Collaborative software development using code repositories and version management (e.g., Git, Bitbucket, SVN)

d2) Developing group presentations and reports on specific topics.

g1) Develop solutions that solve an existing problem in our society.

i2) Use programming languages and environments that allow the implementation and debugging of solutions.

8. TOPICS

Unit 1: Introduction (5)	
Competences Expected: g	
Topics	**Learning Outcomes**
- Overview of platforms (e.g., Web, Mobile, Game, Industrial) - Programming via platform-specific APIs - Overview of Platform Languages (e.g., Objective C, HTML5) - Programming under platform constraints	- Describe how platform-based development differs from general purpose programming [Familiarity] - List characteristics of platform languages [Familiarity] - Write and execute a simple platform-based program [Familiarity] - List the advantages and disadvantages of programming with platform constraints [Familiarity]
Readings : [Fie00], [Gro09], [ADC13], [TC15]	

Unit 2: Web Platforms (5)	
Competences Expected: c,g,i	
Topics	**Learning Outcomes**
- Web programming languages (e.g., HTML5, JavaScript, PHP, CSS) - Web Platform constraints: Client-Server, Stateless-Stateful, Cache, Uniform Interface, Layered System, Code on Demand, ReST. - Web platform constraints - Software as a Service (SaaS) - Web standards	- Design and Implement a simple web application [Familiarity] - Describe the constraints that the web puts on developers [Familiarity] - Compare and contrast web programming with general purpose programming [Familiarity] - Describe the differences between Software-as-a-Service and traditional software products [Familiarity] - Discuss how web standards impact software development [Familiarity] - Review an existing web application against a current web standard [Familiarity]
Readings : [Fie00]	

Unit 3: Desarrollo de servicios y aplicaciones web (25)	
Competences Expected: c,d,g,i	
Topics	**Learning Outcomes**
- Describe, identify and debug issues related to web application development - Design and development of interactive web applications using HTML5 and Python - Use MySQL for data management and manipulate MySQL with Python - Design and development of asynchronous web applications using Ajax techniques - Using dynamic client side Javascript scripting language and server side python scripting language with Ajax - Apply XML / JSON technologies for data management with Ajax - Use framework, services and Ajax web APIs and apply design patterns to web application development	- Server-side python scripting language: variables, data types, operations, strings, functions, control statements, arrays, files and directory access, maintain state. [Usage] - Web programming approach using embedded python. [Usage] - Accessing and Manipulating MySQL. [Usage] - The Ajax web application development approach. [Usage] - DOM and CSS used in JavaScript. [Usage] - Asynchronous Content Update Technologies. [Usage] - XMLHttpRequest objects use to communicate between clients and servers. [Usage] - XML and JSON. [Usage] - XSLT and XPath as mechanisms for transforming XML documents. [Usage] - Web services and APIs (especially Google Maps). [Usage] - Macros Ajax for the development of contemporary web applications. [Usage] - Design patterns used in web applications. [Usage]
Readings : [FR11]	

Unit 4: Mobile Platforms (5)	
Competences Expected: c,d,g,i	
Topics	**Learning Outcomes**
- Mobile programming languages - Design Principles: Segregation of Interfaces, Single Responsability, Separation of concerns, Dependency Inversion. - Challenges with mobility and wireless communication - Location-aware applications - Performance / power tradeoffs - Mobile platform constraints - Emerging technologies	- Design and implement a mobile application for a given mobile platform [Familiarity] - Discuss the constraints that mobile platforms put on developers [Familiarity] - Discuss the performance vs power tradeoff [Familiarity] - Compare and Contrast mobile programming with general purpose programming [Familiarity]
Readings : [Mar17], [ADC13]	

Unit 5: Mobile Applications for Android Handheld Systems (25)	
Competences Expected: c,d,g,i	
Topics	**Learning Outcomes**
- The Android Platform - The Android Development Environment - Application Fundamentals - The Activity Class - The Intent Class - Permissions - The Fragment Class - User Interface Classes - User Notifications - The BroadcastReceiver Class - Threads, AsyncTask & Handlers - Alarms - Networking (http class) - Multi-touch & Gestures - Sensors - Location & Maps	- Students identify necessary software and install it on their personal computers. - Students perform various tasks to familiarize themselves with the Android platform and Environment for development. [Usage] - Students build applications that trace the lifecycle callback methods emitted by the Android platform and demonstrate the behavior of Android when device configuration changes (for example, when the device moves from vertical to horizontal and vice versa). [Usage] - Students build applications that require starting multiple activities through both standard and custom methods. [Usage] - Students build applications that require standard and custom permissions. [Usage] - Students build an application that uses a single code base, but creates different user interfaces depending on the screen size of a device. [Usage] - Students construct a to-do list manager using the user interface elements discussed in class. The application allows users to create new items and to display them in a ListView. [Usage] - Students build an application that uses location information to collect latitude, length of places they visit. [Usage]
Readings : [ADC13], [TC15]	

9. WORKPLAN

9.1 Methodology

Individual and team participation is encouraged to present their ideas, motivating them with additional points in the different stages of the course evaluation.

9.2 Theory Sessions

The theory sessions are held in master classes with activities including active learning and roleplay to allow students to internalize the concepts.

9.3 Practical Sessions

The practical sessions are held in class where a series of exercises and/or practical concepts are developed through problem solving, problem solving, specific exercises and/or in application contexts.

10. EVALUATION SYSTEM

********* EVALUATION MISSING ********

11. BASIC BIBLIOGRAPHY

[ADC13]   J. Annuzzi, L. Darcey, and S. Conder. *Introduction to Android Application Development: Android Essentials.* Developer's Library. Pearson Education, 2013. ISBN: 9780133477337.

[Fie00]   Roy Thomas Fielding. "Fielding dissertation: Chapter 5: Representational state transfer (rest)". In: *http://www. ics. uci. edu/~ fielding/pubs/dissertation/rest_arch_style. htm* (2000).

[FR11]   Eric Freeman and Elisabeth Robson. *Head first HTML5 programming: building web apps with JavaScript.* " O'Reilly Media, Inc.", 2011.

[Gro09]   R. Grove. *Web Based Application Development.* Jones & Bartlett Learning, 2009. ISBN: 9780763759407.

[Mar17]   Robert C Martin. *Clean architecture: a craftsman's guide to software structure and design.* Prentice Hall Press, 2017.

[TC15]   Trish and Richard Cornez. *Android Programming Concepts.* Jones and Bartlett Publishers, 2015. ISBN: 9781284070705.