INSTITUTION	NATIONAL AND KAPODISTRIAN UNIVERSITY OF ATHENS						
SCHOOL	SCHOOL OF SCIENCE						
DEPARTMENT	INFORMATICS AND TELECOMMUNICATIONS						
COURSE LEVEL	UNDERGRADUATE						
COURSE TITLE	**Software Development for Network and Telecommunication Systems**						
COURSE CODE	**K23β**		**Semester**	**7**	**ECTS**	**8**	
TEACHING HOURS per week	**THEORY**	**1**	**SEMINAR.**	**4**	**LABORATORY**	**1**	
COURSE TYPE	Project						

K	**E1**	**E2**	**E3**	**E4**	**E5**	**E6**
B						

URL	https://eclass.uoa.gr/courses/DI292/
EXPECTED PRIOR KNOWLEDGE/ PREREQUISITES AND PREPARATION:	K16 - COMMUNICATION NETWORKS I, Recommended K33
TEACHING AND EXAMINATIONS LANGUAGE:	GREEK
THE COURSE IS OFFERED TO ERASMUS STUDENTS	NO

COURSE CONTENT
Students have to develop a project which consists of an extensive implementation of network functions, protocols and applications. The approach aims at gradually familiarizing with programming at different network levels. More specifically, it includes software development for fixed and mobile networks and Future Internet applications, mobile and wireless systems. Applications and implementations also involve cognitive and remodeling mobile and wireless communications environments (using, for example, J2ME, Android). Indicative covered issues are as follows. • The JAVA programming language • The Android operating system and the application development environment on it • The git publishing control system • The Internet of Things • The MQ Telemetry Transport (MQTT) messaging protocol as well as software / development environments for MQTT

STUDENT LEARNING OBJECTIVES
Expected Learning Outcomes Upon successful completion of the course the student will be able to: • Design and implement software applications and systems using the JAVA programming language. • Design and implement mobile apps for the Android operating system. • Use software publishing control system. • Uses integrated software development environments (IDEs). • Design and implement applications based on client-server and publish-subscribe models. • Design and implement applications using the MQTT protocol. • Work in a team for software development. • Record and analyze the technical and operational requirements of network and telecommunication software.

TEACHING AND LEARNING METHODS - ASSESSMENT	
TEACHING METHOD	In Class (Face to Face)
USE OF INFORMATION AND COMMUNICATION TECHNOLOGIES	Learning process supported by the e-class platform (Teaching material, Videolectures, Discussions, Announcements, Task assignments) Email communication Live transmission of lectures Ability to watch recorded lectures Utilization of educational environments: Git based publishing control system https://anapgit.scanlab.gr/ Utilization of Special software: Android SDK, Java IDE
TEACHING ORGANIZATION *Describe in detail the way and methods of teaching:* *Enhanced Lectures,* *Online Lectures,* *Seminars,* *Tutorial,* *Laboratory,* *Laboratory Exercise,* *Study & analysis of literature,* *Practice (Positioning),* *Interactive teaching,* *Developing a project,* *Individual / group work* *Telework (reference to tools) etc.* *Details of the student's study hours for each learning activity and hours of non-guided study are shown to ensure that the total workload at the semester corresponds to the ECTS*	A series of lectures (Theory, Tutorial) introduces the concepts, technologies and software development environments to be used for the project. The project is about developing a distributed application system on mobile devices and servers. The application is different each academic year. Students work together in groups of 2 to 4 people, depending on the requirements of the project. Lectures are broadcasted live and recorded to be available to students at any time.

Activity	***Student Workload*** *(hours)*
Lectures	12
Tutorial	9
Laboratory	3
Code Development	200
Independent Study	26
Total Course	***250***

ASSESSMENT OF STUDENTS

Description of the assessment process

Assessment Methods, Formative or Concluding, Multiple Choice Test, Quick Response Questions, Test Development Questions, Problem Solving, Written Work, Report / Report, Oral Examination, Public Presentation, Laboratory Work, Other / Other

Fully defined evaluation criteria are mentioned and if and where they are accessible to students.

Students are assessed a) by delivering a project in 2 phases by oral examination with demonstration of the application in the laboratory, at the end of each phase and b) by a final written examination.
The rating of each phase of the project represents 50% of the degree of the project. The assignment is assessed with graduated criteria and communicated to the students.
The written examination covers both the theoretical part of the subject matter and the programming. Complaints and retrains are allowed.

The final grade is as follows:
If the difference between the project grade and the written examination is up to 2 points, then the final grade is the grade of the project.
Otherwise, the final grade is the average of the project grade and the grade of the written examination.

Assessment methods	***Number***	***Percentage***
Oral examination / demonstration / check of the first deliverable	1	50%
Oral examination / demonstration / check of the second deliverable	1	50%
Written examination	1	100%
Final grade		as described above

LITERATURE AND STUDY MATERIALS / READING LIST

The course does not have a specific workbook for students. The teacher / tutor proposes educational resources on the internet and manuals of the technologies, software and software development environments used for the project implementation.