NATIONAL UNIVERSITY OF ENGINEERING
COLLEGE OF INDUSTRIAL AND SYSTEMS ENGINEERING
SYSTEMS ENGINEERING PROGRAM

ST204 – SOFTWARE ENGINEERING WORKSHOP I

I. GENERAL INFORMATION

CODE	**:**	ST204 Software Engineering Workshop I
SEMESTER	**:**	7
CREDITS	**:**	3
HOURS PER WEEK	**:**	6 (Theory–Practice)
PREREQUISITES	**:**	ST213 Systems Analysis and Design
CONDITION	**:**	Compulsory
DEPARTMENT	**:**	Systems and Telematics

II. COURSE DESCRIPTION

In this course student analyze, design, construct and test a software-based system properly applying software engineering methods and techniques. Students apply different modeling process depending on the particular characteristics of the problem: evolving process modeling (spiral and incremental development), sequential linear modeling (cascade development). Students also apply agile methodologies for project management, as well as proper techniques for the analysis and design of software systems. Data bases area modeled and implemented using available information and communication technologies. CASE modeling tools are used, as well as visual programming and code generation tools. The project final report is submitted and defended at the end of the course.

III. COURSE OUTCOMES

At the end of the course, students:

1. Apply proper methods for the analysis and design of software-based systems.
2. Apply the different steps of software engineering projects: requirements definition, software design and construction, software testing, software configuration, software quality.
3. Properly select and justify the modeling method to be applied to the system to be developed.
4. Construct the prototype of the software project and test it subject to different operating conditions and quality requirements.

IV. LEARNING UNITS

1. SEMESTER ACTIVITIES AND STUDENT WORKING GROUPS

Software engineering steps / Formation of student working groups / Definition of project theme / Semester working schedule.

2. METHODOLOGIES FOR SYSTEMS DESIGN AND ANALYSIS

Elaboration of project profile: objectives and goals, scope, methods, resources, results / Revision of project profile / Deliverables.

3. DELIVERABLE 1

Structure analysis and main diagrams / Final diagrams / Analysis of technological architecture / Systems design – Part I.

4. **DELIVERABLE 2**
   Considerations for system design / Methodologies of systems modeling / Systems design – Part II / Submission and defense of Deliverable II: System Design.

5. **DELIVERABLE 3**
   Interfaces design and presentation / Coding – Part I / Submission and defense of Deliverable III: Main interfaces.

6. **DELIVERABLE 4**
   Final interfaces / Coding – Part II / Submission and defense of Deliverable IV: System coding / Connectivity to data bases / Development of prototype.
   Main interfaces.

7. **DELIVERABLE 5**
   Units testing / Software integral test / Software quality test / Submission and defense of Deliverable V / System configuration / Elaboration of manuals.

8. **FINAL REPORT**
   Submission and defense of final report.

V. LABORATORY AND PRACTICAL EXPERIENCES

Continuous use of computing laboratory throughout the academic semester.

VI. METHODOLOGY

The course is carried out in computer laboratory sessions. At the beginning of each class, the instructor presents the concepts, methods and techniques. Students apply the methods for developing the solution to the software engineering project, working in teams, and using software applications such as UML, SCRUM, and XP, as well as software and web tools such as PHP, Designer, BPWin, ERWIN, Dreamweaver, Data Bases: SQL Server, Oracle and MySQL. At the end of the course, each student team must submit and defend a design project. In all sessions, students' active participation is encouraged and graded.

VII. EVALUATION FORMULA

The Final Grade PF is calculated as follows (evaluation system: D):

$$\boldsymbol{PF = (4\ Best\ Partial\ Deliverables + MF)/5}$$

MF: Final Deliverable (Report)

VIII. BIBLIOGRAPHY

1. **PRESSMAN Roger**
   Software Engineering. A Practical Approach.
   McGraw Hill Editions, 2010.
   Prentice Hall Editions, 2012.

2. **WEITZENFIELD Alfredo**
   Object Orient Software Engineering using UML, Java and Internet.
   Thomson Editorial, 2005