1. INTRODUCTION AND OVERVIEW

1.1. **Presentation, objectives and learning objectives of the course.** Importance of the analysis and design activities: skills' acquisition. Concepts: software engineering and information systems. Concepts: organizational analysis, methodology, quality. Presentation of the topics and of the course program. Methodology of the course, teaching method. Examination arrangements.

2. SOFTWARE ENGINEERING

2.1. **Methodologies.** Software evolution. Direct and indirect costs. Maintenance. Logical design of information systems: Reality and Model

2.2. Models for software development. Waterfall Model. Applicability of the model. Prototyping cycle. Exploration model. Agile methodologies and eXtreme Programming. Incremental model. Iterative Model. Agile methodologies vs classical methods. Analysis process: adaptive vs. predictive. eXtreme Programming guidelines and phases.

2.3. **Requirements and Specifications**

2.4. ***Lab Exercise: Requirements***

2.5. **Design**

2.6. **Coding, Testing and Validation**

2.7. ***Lab Case: version control.*** *Management through the repository.*

2.8. **Software Project Management.** Company structure. Organization: teams and roles. Planning a project. Table of tasks

2.9. ***Lab exercise: production table of tasks, production graph of addictions.*** *Scheduling tasks. PERT. GANTT. Costs estimation*

2.10. **Gantt Planning**

2.11. ***Lab exercise: planning of a project.*** *Use of Microsoft Project.*

3. METHODS AND TOOLS

3.1. **Evaluation of the activity and cost estimates**

3.2. **Function point analysis.** Counting method. Identification of the elements. Calculation of UFP. Determination of the adjustment factor. Calculation of Adjusted Function Point (AFP). Example of UFP count.

3.3. ***Lab exercise Funcion Point Analysis.*** *Metodo di valutazione, con analisi e disegno della base dati.*

3.4. **Unified modeling language (UML).** Definition of a visual approach to design. The advantages of the diagrams in the design phase. Unified Software Development Process, characteristics and phases. UML structure. The views. The diagrams. Use cases. Actor. Relationship between actor and use case. Other types of relationship and association. How to make use cases. Meaning and description of diagrams. Diagrams: classes, objects, collaboration, sequence, activity, state, physical, components, deployment.

3.5. ***Lab Case: UML***

3.6. ***Lab Exercise: UML***

4. INFORMATION SYSTEMS

4.1. **Concepts, evolution.** Information system and system of informations. Components. Concept of information system. Quantity of information and uncertainty of the task. Information system and computer system. Characteristics, finality, modality and process information.

5. INTERNET INFORMATION SYSTEMS

5.1. **The phases of analysis, design, implementation and work tools.**

5.2. ***Lab Exercise: Website***

5.3. **Portals**

6. INFORMATION SYSTEMS FOR ANALYSIS AND DECISIONS

6.1. **Information systems for analysis and decisions.** Uses and users. Architecture. Multidimensional database. Data Cube. Operations: Drill-down, roll-up, Pivoting. Slice and dice. Ranking. Loading Access, Analysis and Reporting. DSS, EIS, Data Mining. Limitations. Time of realization. Data quality. Standards for metadata. Special database. Costs. Data marts.

6.2. **Presentation of data.** How to represent data graphically. Approach to the graphical representation of information. Some rules to build effective graphics. Multidimensional analysis with Excel. Analyze data. Produce Pivot tables and charts.

6.3. ***Lab Exercise: Information Systems for Analysis and Decisions, Lab Exercise: Data Graphic Representation***