Lecture 1:
Software economics. General information.

Studying process

- Lectures –22
- Laboratory Classes – 22
- Self-study – 64
- Total (hours/ECTS credits) – 108/3
- Homework (1) – 8th semester
- Graded Test – 8th semester

Studying process

Type of work	Module 1	Module 2	Final test
Practice work	24 (5 tasks)	24 (3 tasks)	
Homework	12		
Test	14	14	12
Total	**50**	**38**	**12**

Deadlines

Type of work	Module 1	Module 2
Homework	06/03/17	
Test	27/02/17	27/03/17

Deadlines

Practice work	Deadline
1.1. Description of software (requirement specification), what necessarily includes budget, labor costs, organizational structure and economic effect from implementation.	30/01/17
1.2. Make a plan of works, what necessarily includes identifying deliverables, identifying the activities needed to complete those deliverables, estimating the resource requirements for the activities, estimating time and cost for activities, developing the schedule, developing the budget and risk planning. Make a flow-chart.	30/01/17
1.3. Choose life cycle model for your project. Make a plan of works according phases of life cycle.	06/02/17
1.4. Make a plan of works and phase of life cycle according GOST 34.601-90 (par.2.1) / GOST 34.602-89.	13/02/17
1.5. Make a risk planning part of plan of works. Include different types of risks.	20/02/17
2.1. Count direct and indirect metrics in your software project.	07/03/17
2.2. Estimate your software project using algorithmic method and improve estimation by calibration the algorithmic model.	13/03/17
2.3. Prove economic & technological project efficiency.	20/03/17

General information

- *Economics* is a social science concerned chiefly with description and analysis of the production, distribution, and consumption of goods and services.

General information

- *Economics* is the study of how people make decisions in resource-limited situations.

Software price

- It depends on:
- 1. Size of the final product (for components written by hand). For example, number of lines of code.
- 2. Features of domain. For example, special documents.
- 3. Skills of developers.

Software price

- 4. Programming environment. Methods, tools, techniques, possibilities, else.
- 5. Quality of the final product, which includes functionality of software, reliability and adaptability.

Economics & Life cycle

- Feasibility Phase.
- Plans and Requirements Phase.
- Product Design Phase.
- Programming Phase.
- Integration and Test Phase.
- Maintenance Phase.
- Phaseout.

Economics & Life cycle

- *Feasibility Phase.*
- How much should we invest in information system analyses (user questionnaires and interviews, current-system analysis, workload characterizations, simulations, scenarios, prototypes) in order to converge on an appropriate definition and concept of operation for the system we plan to implement?

Economics & Life cycle

- *Plans and Requirements Phase.*
- How rigorously should we specify requirements? How much should we invest in requirements validation activities (automated completeness, consistency, and traceability checks, analytic models, simulations, prototypes) before proceeding to design and develop a software system?

Economics & Life cycle

- *Product Design Phase.*
- Should we organize the software to make it possible to use a complex piece of existing software that generally but not completely meets our requirements?

Economics & Life cycle

- *Programming Phase.*
- Given a choice between three data storage and retrieval schemes that are primarily execution-time efficient, storage efficient, and easy to modify, respectively, which of these should we choose to implement?

Economics & Life cycle

- *Integration and Test Phase.*
- How much testing and formal verification should we perform on a product before releasing it to users?

Economics & Life cycle

- *Maintenance Phase.*
- Given an extensive list of suggested product improvements, which ones should we implement first?

Economics & Life cycle

- *Phaseout.*
- Given an aging, hard-to-modify software product, should we replace it with a new product, restructure it, or leave it alone?

Mail for questions:

sidljarenkosidljarenko@sidljarenko@ukr.net

Andrii Sidliarenko

Practice task № 1.1

- Description of software (requirement specification), what necessarily includes budget, labor costs, organizational structure and economic effect from implementation.
- Real geographically distributed organization, what buys services / products and manufacturing services / products.

Requirement specification. Example of structure

- 1. Goal of development
- 2. **Application field**
- **3. Technical & economical block**
- **4. Main requirements**
- **5. Schedule**
- **6. Special rules from client**
- **7. Quality assessment**