

Vilniaus kolegija/University of Applied Sciences
Faculty of Electronics and Informatics
Software Development Department

METHODICAL INSTRUCTIONS

FINAL PROJECT

SOFTWARE ENGINEERING
6531BX028

Prepared and Approved by
Software Development Department
2020-10-28

Table of Contents

1. Introduction.....	3
2. Final Project.....	3
3. Final Project Software.....	3
4. Final Project Description	4
4.1. Text Formatting Requirements	4
4.2. Final Project Description Structure.....	5
4.2.1. Introduction.....	5
4.2.2. Professional Competences	6
4.2.3. Task Formulation.....	6
4.2.4. Task Analysis.....	7
4.2.5. Software Implementation.....	7
4.2.6. User Manual.....	7
4.2.7. Conclusions and Recommendations	8
4.2.8. List of Information Sources	8
4.2.9. Appendixes.....	8
5. Final Project Preparation Progress and Monitoring.....	9
5.1. Introductory preview.....	9
5.2. First preview	9
5.3. Second preview.....	9
5.4. Third preview.....	10
5.5. Defence at department meeting.....	10
5.6. Defence at final project defence committee meeting.....	11
APPENDIX No 1. Thesis title page.....	12
APPENDIX No 2. Annotation.....	13
APPENDIX No 3. Competencies	14
APPENDIX No 4. Final Project Assignment	15

1. Introduction

Methodological instructions are assigned to Software Engineering study programme (state code 6531BX028) undergraduates of Vilniaus kolegija / University of Applied Sciences Faculty of Electronics and informatics (hereinafter VK EIF). The document describes the subject of the final project, and the procedure and stages of the preparation of the project are regulated by Vilniaus kolegija / University of Applied Sciences Academic Council Resolution No. ATN-5 “Description of Vilniaus kolegija / University of Applied Sciences final work (project) preparation and defence order” of June 16, 2020.

2. Final Project

VK EIF Software Engineering study programme is completed by the final project. In regard to VK study regulations, only undergraduates without academic debts are allowed to prepare the final project.

Preparing the final project undergraduates consolidate theoretical and practical knowledge gained during studies through implementation of the task of the final project. A software product, considering the task corresponding software and hardware, should be created for the final project. The description of the final project, corresponding the task, should also be prepared.

The final project is a qualification work, which shows the undergraduate’s professional preparation and reveals his professional competences.

The final project can be done by a group of 2-3 undergraduates. In this case, each undergraduate prepares a separate description of the final project corresponding his task.

The final project consists of the following main parts:

- Description of the final project;
- Software part.

The final project (description and software part) is the property of VK EIF. The final project description and software part implementation source code should be recorded on a storage media (CD / DVD or flash drive) which is presented with the final project. The undergraduate, wishing to use the name of an institution (except VK EIF) in the description and (or) software implementation, for which software implementation is assigned, should give the consent and the letter signed by the head of the institution or his authorized person regarding the planned or actual use of the results of the final project.

3. Final Project Software

The final project software part can be:

- a program which addresses applied tasks, intended for one or a group of users;

- smart device application;
- dynamic WEB application;
- software for embedded devices and data stream control.

The final project software part should not be:

- already successfully defended parts of the final project;
- constructed from the actual software implementation) without the use of any inserts or fragments of the undergraduate source code;
- designed using software that is not used at VK EIF and the undergraduate cannot freely dispose the license of the chosen software (except when it is possible to demonstrate designed software on a virtual or remote machine).

4. Final Project Description

This section describes text formatting requirements and the components of the final project description in detail.

4.1. Text Formatting Requirements

The final project paper should be written in the language in which the studies are conducted in accordance with the applicable regulations and standards of the organization and also in accordance with the instructions given in this document. The author of the work is responsible for the correctness of the text: grammar and language mistakes.

Entire text is compose on standard A4 (21 x 29.7 cm) paper. Page orientation is a portrait. Margins of the page: 3 cm from the left, 2 cm from the top and the bottom, 1 cm from the right. The page header - 1.27 cm. The pages of the final project description, starting with the content, are numbered in the lower margin, centered, the text is printed on one side of the page.

The final project description is in black Times New Roman font. The text is in 12pt font size. The indent of the first line of the paragraph is 1.27 cm, line spacing is 1.5, space after the paragraph before the section or sub-section is set to auto. The entire text (except in the tables) is justified.

The final project description is divided into sections. Section headings are in 14pt bold, numbered (except the list of information sources and appendixes), 1, 2, and so on, and centered. The headings of subsections are in 12pt bold, numbered - 1.1, 1.2. and so on. The headings of subsubsections are in 12pt bold, numbered - 1.1.1, 1.1.2. and so on.

Tables and pictures are placed in text as objects. All objects must be numbered and have descriptive text. The numbering of the same object type must be continuous. Object headings are in bold 10pt font size. Table numbering and headings are placed above tables (Table 1), aligned with the right margin. The

pictures are centered in the text, their numbering (Fig. 1 ...) and headings are presented below the picture centered. If an object does not fit on one paper, or if a presented picture or table need to change orientation of the page - these objects are presented in appendixes, the text only contains a reference to the appendix.

Mathematical formulas are written using a special tool (Equations or any similar tool) starting a new line, centered and numbered. The formula number is in brackets and aligned with the right margin. Mathematical formulas do not change grammatical structure of the text, therefore punctuation marks in the text before and after a formula are placed according to the rules of punctuation. Interpretation of mathematical formula is presented in the next line separating notations by semicolons. The formation of the list of information sources is given in section 4.2.8.

Complete guidelines of text formatting can be found on the Faculty website under the following link <https://eif.viko.lt/studentams/metodiniai-nurodymai/>.

Other documents related to the final project description are: title page, task page and annotation page. The example of the title page of the final project description is given in Appendix 1. The example of the final project task page is given in Appendix 2 (task page should be printed on both sides of the page). The example of final project annotation is given in Annex 3.

4.2. Final Project Description Structure

The final project should consist of the following parts:

- Title page;
- Final project task page;
- Annotation page;
- Glossary;
- Table of contents;
- Introduction;
- List of professional competences;
- Task formulation;
- Task analysis;
- Software implementation description;
- User manual;
- Conclusions and recommendations;
- List of information sources;
- Appendixes (if any).

4.2.1. Introduction

The introductory part of the final project description, should include these aspects:

- Origin. Undergraduates should disclose origin of the choice of the final project topic, consider all factors that influenced it or present other sources that lead to the choice, review novelty from a practical point and solve problems in the subject area.
- Aims and objectives. In this aspect, it is necessary to disclose the aim of this final project. The aim should be realistic, measurable and achievable in time given for project preparation. The aim should express the desired result. Objectives related to the final project should also be named.
- Implementation tools. In this aspect the undergraduates provides a list of chosen software and hardware.

If the final project is prepared by a group of undergraduates, then the introductory part should include:

- A description of the contribution of each undergraduate and the tasks performed.
- A description of work plan. The undergraduates should submit a final Gantt diagram with a work plan that reveals task distribution between the undergraduates. Deadlines, responsibilities, the use of common resources, and so on.

The volume of the Introduction is 2 - 5 pages.

4.2.2. Professional Competences

This section should contain the description of professional competences that the undergraduate wants to demonstrate and prove preparing and defending the final project. The list of professional competences of the study program is given in Appendix 6 "General and professional competences" of these instructions.

Defending the final project the undergraduate should demonstrate that he has acquired general and professional (at least two) competencies. Evidence of the competences should be proved on how they were applied in the final project, indicating the sections or pages of the final project that reflect the realization of the listed competences. Mastery of the competences described in the final project and demonstrated is evaluated during defence.

The volume of Professional Competences section is 1 - 2 pages.

4.2.3. Task Formulation

The task formulation part describes functional and non-functional requirements for software implementation. Functional requirements describe what software implementation will be able to do. These requirements specify the main and auxiliary functions of the designed program. The main functions are intended to realize the designed program. The auxiliary functions are those that are influenced by

technological requirements. Typically, auxiliary functions are used to maintain or supervise software implementation (work reports, data archiving, statistics storage, etc.).

Formulating functional requirements for each function the initial data, actions performed by the function and the result are indicated. It also specifies the order of function execution and execution restrictions, if any.

Non-functional requirements define requirements that limit the range of possible design choices.

This section should also provide an analytical overview of similar systems.

The volume of the section is 3 - 5 pages.

4.2.4. Task Analysis

Exploratory task analysis should be performed in task analysis part:

- ER diagram and its description;
- Class diagram and its description;
- Use case diagram and its description;
- Activity diagram and its description;
- Other UML diagrams and their descriptions.

The volume of the section is at least 10 pages.

4.2.5. Software Implementation

Software implementation files, revealing their purpose, are described. The description of the classes and methods: actions, initial data, structure of the results is given. The physical model of the database (if any) and its description are presented. Other software constructions such as components, modules, and the relationships are described in detail.

The volume of the section is at least 25 pages.

4.2.6. User Manual

In this section, the undergraduate should provide an implementation guide:

- the software implementation dependence on other software products (to provide the description of system or other processes software implementation components without which implementation is not performed);
- identify computer hardware parameters on which software implementation was performed and tested;
- a detailed description of software implementation;
- a description of typical configuration (if any);

- a detailed description of software implementation (especially implementing mobile apps, websites or other services based on the Internet technologies);
- steps to eliminate software implementation.

In addition, the undergraduate should provide user manual of software implementation - a description of the steps that address basic functional requirements.

The volume of the section is at least 15 pages.

4.2.7. Conclusions and Recommendations

In this section, the undergraduate should provide conclusions that are relevant to the final project. The conclusions should be substantiated, specific, related to the aim and objectives addressed in the final project. If the undergraduate failed to achieve the results he expected, or to solve all addressed objectives, it is necessary to state the reasons.

In this part, the undergraduate should also present development possibilities and ways for improving the program.

The volume of the section is 1 - 2 pages.

4.2.8. List of Information Sources

This section contains information sources quoted by the undergraduate in final project description. The section is not numbered. Information sources should be in alphabetical order. The quotation in the text is indicated by the source number in the list in square brackets (see Appendix 7, "Listing and quoting information sources"). Information sources may include:

- books;
- periodicals;
- electronic information sources, etc.

4.2.9. Appendixes

Appendixes may be any information not directly is related to the final project or information which representation needs to change formatting requirements of the text. Appendixes include:

- written consent of the institution to use the name of the institution in a final project description and (if necessary) software implementation;
- letter from the institution on the planned or actual use of the results of the final project;
- examples of software implementation results;
- examples of software implementation testing results;

- graphics objects, tables, pictures or other information that was mentioned in the final project description but not presented;
- copies of articles published by the undergraduate;
- conference certificates;
- other.

5. Final Project Preparation Progress and Monitoring

Final project preparation monitoring include previews and defence at the department meeting. The undergraduate who has not defended his project at Software Development department meeting loses his right to defend his final project at final project defence committee meeting. All information about final project preparation and monitoring is available and updated on the Faculty website <https://eif.viko.lt/studentams/metodiniai-nurodymai/>, also on Software Development Department billboard and on Moodle (course title *Final Project – Software Engineering*).

In regard to VK studies regulations, code of students' ethics and code of lecturers' ethics the undergraduate guarantees that his project is not plagiarized. The undergraduate who presents a plagiarized final project loses all rights to defend his project at final project defence committee meeting. Moreover, the undergraduate is administered according to the Republic of Lithuania Code of Administrative Offences (25 June, 2015, No. XII-1869) Article 123.

Final Project Preparation Stages

5.1. Introductory preview

At the introductory preview final project topics and supervisors are announced and further final project plan is created.

5.2. First preview

At this stage the undergraduate submits these parts of the final projects: title page, task page approved by the vice dean, introduction page, and task formulation and analysis pages. In regard to Remarks and comments on the submitted parts should be taken into account for the next preview.

5.3. Second preview

At this stage the undergraduate submits corrected parts of the first preview, professional competence description section and presents software implementation basic functionality. Remarks and comments on the submitted parts should be taken into account for the next preview.

5.4. Third preview

At this stage the undergraduate submits corrected parts of the second preview, completed final project description with user manual, conclusions and recommendations, list of information sources, appendixes, annotation, software implementation. All remarks and comments on the final project should be taken into account till defence in the department meeting

Final project annotation in English has to be presented to English advisor till the third preview.

5.5. Defence at department meeting

For defence at the department meeting the undergraduate should submit corrected third preview version of his final project, but not clipped, finished final project description, software implementation and final project presentation (aim and objectives, ideas, preparation progress, results, conclusions and recommendations). During defence at the department meeting the undergraduate should reveal gained competences presenting project description and software implementation to the members of the department. After that the members of the department decide if the undergraduate is allowed to defend his final project at final project defence committee meeting. After defence undergraduates are introduced with meeting protocol. The undergraduate, who has successfully defended his final project at the department meeting, should present it to the vice dean. According to Software Development department meeting protocol and by dean's order, the undergraduate is allowed to defend the final project at final project defence committee meeting and gets a reviewer.

Final project has to be clipped in this order:

1. Title page;
2. Task page;
3. Annotation page;
4. Glossary;
5. Table of contents;
6. Introduction;
7. Professional competences;
8. Task formulation;
9. Task analysis;
10. Software part;
11. User manual;
12. Conclusions and recommendations;
13. List of information sources;
14. Appendixes (if any).
15. Storage media with software

The undergraduate should bring a clipped final project to the department. The department hands it to a reviewer.

The undergraduate who has not defended his final project at the department meeting has the right to defend it next year in regard to studies regulations.

5.6. Defence at final project defence committee meeting

During defence at final project defence committee meeting the undergraduate should submit his final project, review and presentation. The undergraduate presents his presentation and software part of his final project.

The committee evaluate the final project by a mark and suggest to award a professional bachelor's degree in informatics and issue a professional bachelor's diploma.

Final project and defence are evaluated by marks. The lowest mark – 5, the highest mark – 10. Final mark is the average of final project defence committee members' marks and reviewer's mark. (Vilniaus kolegija / University of Applied Sciences Academic Council Resolution No. ATN-5 "Description of Vilniaus kolegija / University of Applied Sciences final work (project) preparation and defence order" of June 16, 2020.)

APPENDIX No 1. Thesis title page

VILNIAUS KOLEGIJA / UNIVERSITY OF APPLIED SCIENCES
FACULTY OF ELECTRONICS AND INFORMATICS



AUTHORIZED BY
Vice Dean of Electronics and Informatics
Faculty

_____ dr. Loreta Savulionienė

___ January, 20___

FINAL PROJECT TITLE

FINAL PROJECT
FP 6531BX028 PI _____

UNDERGRADUATE

FULL NAME

20_ - _ - _

SUPERVISOR

FULL NAME

20_ - _ - _

REVIEWER

20_ - _ - _

APPENDIX No 2. Annotation

Vilniaus kolegija/University of Applied Sciences
Faculty of Electronics and Informatics
Department of Software Development

State Code: 6531BX028
Date: __, __

Summary of the Final Project of Software Engineering Study Programme

Theme of the Final Project:

Undergraduate:

Supervisor:

Volume of the work – p. text without annexes, pictures, tables, ... bibliographical entries, annexes.

Annotation

The annotation of the final project must be prepared in English. The information provided would allow the reader to form an opinion about the content, main goal, tasks and results of the final project. Annotation must contain basic structure of the final project topic. The final work annotation must consist of 200-250 words but not less than 1200 characters

..

APPENDIX No 3. Competencies

General competences		Results of the study program (Student will be able to)	
1.	Ability to communicate orally and prepare reports.	1.1	Be able to communicate and share professional knowledge.
		1.2	Be able to argue their point of view in a professional discussion.
2.	Ability to work in a team, organize activities and take responsibility for teamwork results.	2.1	Be able to work in a team, organize activities.
		2.2	Be able to behave ethically and professionally.
3.	Be able to make decisions. Follow the principle of equal opportunities and tolerance. Evaluate and ensure the quality of work. Adapt to innovation in business and work.	3.1	Be able to evaluate changes in the market and make decisions.
		3.2	Be able to apply quality management techniques.
		3.3	Be able to study independently. Improve knowledge. Raise qualification.
		3.4	Be able to use a variety of software systems design and modeling methodologies.
		3.5	Be able to apply the acquired knowledge: in the software design, development, evaluation and implementation of application systems.
Subject competences		Results of the study program (Student will be able to)	
4.	Software development.	4.1	Be able to analyze and design data structures.
		4.2	Be able to design and implement algorithms with selected software tools.
		4.3	Be able to select and apply software solution testing methods.
		4.4	Be able to prepare user guide and technical documentation for software solutions.
		4.5	Be able to design, create and improve databases.
		4.6	Be able to design, create and improve user interface.
		4.7	Be able to design, create new and improve existing software solutions.
		4.8	Be able to analyze, select and apply data security solutions.
5.	Database development and management (specialization - Database systems).	5.1	Be able to design and develop data processing solutions.
		5.2	Be able to manage database management systems.
6.	Development and management of Internet services (specialization - Internet technologies).	6.1	Be able to create and develop online service solutions.
		6.2	Be able to manage online service systems.
7.	Smart Device Management (specialization - Smart Device Programming)	7.1	Be able to design and develop smart device management solutions.
		7.2	Be able to work with Internet of Things systems.

APPENDIX No 4. Final Project Assignment

**VILNIAUS KOLEGIJA / UNIVERSITY OF APPLIED SCIENCES
FACULTY OF ELECTRONICS AND INFORMATICS**

FINAL PROJECT ASSIGNMENT

Given to undergraduate **Name Surname** of group **PI**___ on October __, 2020.

Final Project Title: Write Actual Final Project Title

Final project description

This section is for describing the main project task and goal. Undergraduate must point out the key functional requirements and list the objectives that needs to be completed in order to reach the goal.

Final project will be defended in the meeting of Software Development department on January __, 20__.

Undergraduate.....
(signature) (name, surname)

Supervisor.....
(signature) (name, surname)

Verified by:

Head of Software Development Department
(signature) (name, surname)

Advisers for Technical Affairs:

.....
(signature) (name, surname)

.....
(signature) (name, surname)

Adviser for English Language:

.....
(signature) (name, surname)