SYLLABUS

1. Course title:

Programming Tools in Power Engineering

2. Code:

EEMS101

3. Cycle of study:

1

4. ECTS credits:

6

5. Type of course:

● Mandatory ○ Elective

6. Prerequisites:

[EEMS001] Introduction to Energy Systems

7. Class restrictions:

8. Duration / semester:

1 4

9. Weekly contact hours:

9.1. Lectures:	3
9.2. Seminars:	1
9.3. Laboratory/Practice classes:	1

10. Faculty:

Faculty of Electrical Engineering

11. Department/study program:

Electrical Engineering and Computer Science

12. Lecturer:

Ph.D. Amir Nuhanović, full prof.

13. Lecturer's e-mail:

amir.nuhanovic@untz.ba

14. Web site:

15. Course aims:

The goal of this course is to introduce students with the properties and structure of the most commonly used software tools to solve problems in power engineering. Through examples of the application to learn about the practical use of available tools.

16. Learning outcomes:

After accepting of subject matter students are qualified for: use a common software tools (Matlab/Octave), use a free software tools for electrical power systems calculation (PSAT) and tools for statistical calculations, use VBA to solve the basic problems formulated by spreadsheets and some widely used Software tool for power systems calculation (PSS®E).

17. Course content:

Software tools, software development environments and tools in electrical engineering. Types and features, classification according to different criteria. Commercial and noncommercial tools. Open source tools. General-purpose tools, specialized tools. The usual structure of software tools in power engineering. Various examples of structures. Tools for design and analysis of power networks. Basic features of available tools. The elements of a development environment. Opportunities and types of data visualization and performance of individual tools. Technical and economic problems that are commonly solved using the tools in power engineering. Relevant standards and their implementation. Training simulators. A brief introduction to the mathematical background of the individual modules. Preparation of data. Data libraries. Files and commonly used file formats. Data conversion. Comparison and evaluation of software tools, decision-making on procurement and implementation. Practical examples of the use and application.

18. Learning methods:

Lectures, oral and laboratory exercises: The lectures cover theoretical basics with simpler examples when needed, on oral part students work on numerical exercises using appropriate software tool, and in laboratory exercises, students solve problems using the appropriate software tool.

19. Assessment methods:

At the half of the semester written examination of the knowledge that covers up the half of subject matter is organized, enabling the student to achieve a maximum of 30 points. The attendance at the course is evaluated from 0 to 5 points, with 5 points being awarded if the student is present at all lectures and exercises, and each absence is penalized with one point. Laboratory exercises makes maximum of 30 points, and The final exam makes maximum of 35 points and consists of a written and/or oral exam of the second part of the subject matter.

20. Assessment components:

The examination mark is based on the total number of points the student has obtained by fulfilling the preconditions and passing the final exam. Students can achieve a maximum of 100 points (5 + 30 + 30 + 35).

21. Required reading list:

F.Milano, “Power System Modelling and Scripting”, London, 2010.
E.P. Leite, “Matlab - Modelling, Programming and Simulations”, Sciyo, 2010.
Wessex Institute of Technology, “Software for Electrical Engineering Analysis and Design”, WIT Press, 2001

22. Web sources:

23. Applicable starting from the academic year:

2016/2017

24. Adopted in the Faculty/Academy session:

04.04.2016