UNIVERSITÄT SIEGEN

Co-funded by the Erasmus+ Programme of the European Union

Erasmus+ SkoPS

<Course Plan Template (General structure of courses)>

Project Title	Empowering the European Workforce Development through Online/Virtual Skills Training for Digital Transformation towards Mitigating the Impact of Pandemic Situations (SkoPS)		
Project Acronym	SkoPS	**Project Number**	2020-1-DE01-KA226-HE-005772
Date	2021-11-03	**Deliverable No.**	D0.18
Contact Person	Ali Behravan	**Organisation**	USI
Phone	-	**E-Mail**	Ali.Behravan@uni-siegen.de
Version	1.0	**Confidentiality level**	Public

UNIVERSITÄT SIEGEN Petanux Universitat d'Alacant European Institute for Local Development

Version History

Version No.	Date	Change	Editor(s)
0.1	03.11.2021	Initial draft	

Contributors

Name	Organization
Bahareh Kiamanesh	USI
Ali Behravan	USI
Setareh Majidi	USI

Disclaimer

This project has been funded with support from the European Commission. This publication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Table of Contents

Version History 2
Contributors 2
1. Introduction 4
1.1 Abstract 4
1.2 Purpose of the document 4
1.3 Relations to other deliverables 4
2. Template 5

1 Introduction

Course Plan Template can help the partners to prepare the course materials that helps us to organize the materials and contents of courses in the SkoPS project in common form. There are also some other Forms and templates for Student's assessment Plan and Evaluation Questionaries Plan for each Course can be also add to this Template.

1.1 Abstract

The methodology in creating the SkoPS courses will be as follows: Each partner is mainly responsible for preparing their own materials. All other partners and reviewers based on the table of reviewers in the project management handbook will review the prepared materials and will provide feedback in order to improve the quality of the materials. A recurring review process with all partners involved will further enhance the quality of the courses. Right at the beginning of the project (directly after the kick-off meeting), a requirement analysis will be carried out by all project partners in order to precise the content of the courses which are outlined in this proposal. A second meeting will be conducted six months after the kick-off meeting as an internal course development Hackathon. Each partner will invite one or two external experts for this Hackathon to get their feedback for improving the quality of the course. To ensure the quality and to narrow down the courses to the requirements of the industry, SkoPS will create an advisory board with experts from industry and research. Each partner will invite at least one expert to the scientific advisory board. Furthermore, the scientific advisory board will provide guidelines for the course materials.

1.2 Purpose of the document

The purpose of this document is having the same template for all courses. In this Template for each course some features are considered.

1.3 Relation to other deliverables

This Report is part of the D0.18 which is General structure of each course.

2 Template

Course Plan Template			
Course ID and Title:	Dependability and Safety		
Course Duration:	2 - 3 Months	**Course ECTS:**	2
Leading Organization:	USI		
Course Media:	Text File, Video		
Laboratory (Yes/No)	Yes		

Course Description:

Safety-critical applications require high reliability in the computing and electronic systems, which is achieved by designing fault-tolerant systems. This course presents the fundamentals of fault-tolerant systems and basic fault-tolerance tools at both hardware and software levels, e.g., redundancy and re-execution. It will give a good understanding of how to detect faults, design computing components to tolerate faults, and measure the reliability of systems. This course will help students in how to design and analyze reliable computing systems.

Course Materials and Equipment (Prerequisite)

- ✓ This course is suitable for first-year graduate and senior-level undergraduate students.
- ✓ Students should have a basic knowledge in ICT systems design and anlysis and probability theory.
- ✓ Laptop or PC is required for doing programming excercies.
- ✓ Journal papers and books in this field can be used as supplementary materials.

Teaching and Learning Activities:

1. Providing lectures and contents,
2. Using graphs, images, and viedos to enhance the understanding of the subject
3. Solving different examples in modeling and evaluate to clarify the problem
4. laboratory projects assignments
5. Evaluation of excersices and laboratory projects and providing feedback

Course activities:

1. Content focus and presentation of information to students,
2. Look at pictures and watch videos
3. Presentation of real-world case studies
4. Discussion and ineractivity focus

Course Objectives:

The purpose of this course is to provide a solid knowledge of essential terminology and concepts used in specification, design, and evaluation, principles of fault-tolerant operations, and designing robust and resilient systems. It gives a good overview of dependability in the field of computing systems. This course increases the awareness of the potential impact of failures, dependability and safety as the important factors from the economical and social perspectives.

Course Summary:

This course includes 9 chapters. It starts with providing introductory information about the faults, errors, failures and etc. In Chapter 2, preliminary description regarding fault-tolerance techniques are given. Chapter 3 introduces the requirements of dependability. Chapter 4 presents hardware redundancy for fault-tolerance and probabilistic tools to measure reliability. Chapter 5 describes fault-tolerance techniques for software systems such as recovery blocks or N-version programming. Chapter 6 covers the fault tolerance of some network topologies. In Chapter 7, some of the real use cases of fault-tolerant systems are described. Chapter 8 deals with simulating a fault-tolerant system and measuring its reliability in case of injecting different faults. In the last chapter, we provide a set of examples for more practice.

Table of Contents:

1. Dependability Introduction
    1.1. Basic Concepts
    1.2. Faults, Errors, Failures
    1.3. Failure detection
    1.4. Error detection
2. Fault Tolerance
    2.1. Fault Hypothesis
    2.2. Fault Classification
    2.3. Fault-Tolerant Unit
    2.4. Measures of Fault Tolerance
    2.5. Types of Redundancy
3. Dependability Requirement

3.1. Reliability
3.2. Safety
3.3. Maintainability
3.4. Availability
4. Hardware Fault Tolerance
4.1. Hardware Failures
4.2. Failure rate and Reliability
4.3. Resilient Structures
4.4. Reliability Evaluation Techniques
5. Software Fault Tolerance
5.1. Basic Principles
5.2. Single-Version Fault Tolerance
5.3. N-Version Programming
5.3.1. Consistent Comparison Problem
5.3.2. Version Independence
5.4. Recovery Block Approach
5.4.1. Basic Principles
5.4.2. Success Probability Calculation
5.4.3. Distributed Recovery Blocks
5.5. Software Reliability Models
5.5.1. Jelinski–Moranda Model
5.5.2. Littlewood–Verrall Model
5.5.3. Musa–Okumoto Model
5.5.4. Model Selection and Parameter Estimation
6. Fault-Tolerant Networks
6.1. Common Network Topologies
6.2. Measures of Resilience
7. Case Studies
8. Simulation Techniques
9. Exercises

Laboratory Description and Equipment:

Laboratory projects are carried out in small groups implementing the simulation projects. Simulation program can be done by using high-level programming languages such as **C**, **C++** or **Java** or simulation language such as **GPSS** or **SIMAN**

Course References:

The 4 following books are considered as the major references for this course. Although other related books and journal papers can also be used as the supplementary materials.

1. Shooman, M.L., 2003. ***Reliability of computer systems and networks: fault tolerance, analysis, and design***. John Wiley & Sons.
2. Koren, I. and Krishna, C.M., 2020. ***Fault-tolerant systems***. Morgan Kaufmann.
3. Lee, E.A. and Seshia, S.A., 2017. **Introduction to embedded systems**: A cyber-physical systems approach. Mit Press.
4. Sorin, D.J., 2009. **Fault tolerant computer architecture**. Synthesis Lectures on Computer Architecture, 4(1), pp.1-104.

Evaluation and Assessment Methods:

1. Written examination
2. Simulation examples assignment
3. Evaluating excersices

The tasks lead to the production of the intellectual output and the applied methodology.

1. Designing lecture slides
2. Having a companion web site for additional resources
3. Providing simulation framework for the laboratory

3 Course Contents