COURSE DESCRIPTION

1. Program Information

1.1 University	**"Alexandru Ioan Cuza" University of Iasi**
1.2 Faculty	**Computer Science**
1.3 Department	**Computer Science**
1.4 Study Domain	**Computer Science**
1.5 Study Cycle	**Masters**
1.6 Study Program / Qualification	**Masters (Distributed systems/Software engineering/Computational Optimization/Advanced Studies in Informatics) / Masters in Informatics**

2. Course Information

2.1 Course Name			**Event Based Systems**				
2.2 Course Teacher			**Associate Professor Emanuel Onica, Ph.D.**				
2.3 Seminary Teacher			**Associate Professor Emanuel Onica, Ph.D.**				
2.4 Study Year	**I/II**	**2.5** Semester	**2**	**2.6** Evaluation	**E**	**2.7** Course Status*	**OP**

** OB – Compulsory / OP – Optional*

3. Total estimated hours (hours per semester and didactic activities)

3.1 Hours per week	**4**	in which: **3.2** course	**2**	**3.3** seminary/laboratory	**2**
3.4 Hours in curriculum	**56**	in which: **3.5** course	**28**	**3.6** seminary/laboratory	**28**

Time Distribution	hours
Manual study, Course support, Bibliography, and others	**25**
Supplementary Documentation in library, in electronic forums, and on the field	**25**
Seminaries/laboratories preparation, homeworks, reports, portfolios and essays	**50**
Tutoring	**-**
Evaluation	**4**
Other activities	**-**

3.7 Total hours individual study	**100**
3.8 Total hours per semester	**160**
3.9 Credits	**7.5**

4. Preconditions (if necessary)

4.1 Of Curriculum	**Software Engineering, Computer Networks, Information Security, Advanced Programming (Java), Web Technologies**
4.2 Of Skills	**- Ability of programming using an object oriented language (Java)**

5. Conditions (if necessary)

5.1 For Course Operation	**-**
5.2 For Seminary/Laboratory Operation	**-**

6. Specific Skills Acquired

Professional Skills	**C1.** Knowledge of architectural organization and engineering details of distributed event based systems. **C2.** Capacity to identify and offer solutions for scalability issues, fault tolerance problems, and security requirements found in event based systems. **C3.** Knowledge of technologies used in the software industry for the development of distributed event based systems.
Transversal Skills	**CT1.** Knowledge of fundamental notions useful in the development of distributes systems : - information dissemination paradigms ; - algorithms for establishing consensus and synchronization ; - distributed hash tables ; - distributed databases ; - etc.

7. Course Objectives (from the grid of specific skills acquired)

7.1 General Objectives	Knowledge of primary notions related to distributed event based systems, of specific problems appearing in such systems, and of techniques used for solving such problems.
7.2 Specific Objectives	Using a distributed event based platform (e.g., a publish/subscribe system) for disseminating information, with the following sub-objectives: a) Handling scalability issues generated by large number of clients; b) Ensuring fault tolerance; c) Handling specific security issues.

8. General Description

8.1	**Course**	**Teaching Methods**	**Observations** (hours and bibliographic references)
1.	Introduction. The event-based interaction model.	Slides presentation	2
2.	Generic architectures for stream processing using operators.	Slides presentation	2
3.	Publish/Subscribe architectures. The topic based model.	Slides presentation	2
4.	The content based publish/subscribe model. Subscription storage.	Slides presentation	2
5.	Distributed message dissemination. Routing algorithms.	Slides presentation	2

6.	Routing – advanced aspects. Topology shifts.	Slides presentation	2
7.	Optimizing and testing for stream processing. Parallel processing. Case study.	Slides presentation	2
8.	Partial evaluation.	Examination	2
9.	Advanced aspects I. Security. Techniques for maintaining subscription confidentiality.	Slides presentation	2
10.	Advanced aspects II. Lamport timestamps. Message ordering.	Slides presentation	2
11.	Advances aspects III. Fault tolerance. Replication techniques.	Slides presentation	2
12.	Advanced aspects IV. Gossip protocols for dissemination. Consensus algorithms.	Slides presentation	2
13.	Partial evaluation.	Examination	2
14.	Complex event processing. Case study.	Slides presentation	2

Bibliography

Main references:

G. Mühl, L. Fiege, P. Pietzuch. *Distributed Event-Based Systems*. Springer, 2006.
O. Etzion, P. Niblett. *Event Processing in Action*. Manning, 2011.

Supplementary references:

G. Coulouris, J. Dollimore, T. Kindberg, G. Blair. *Distributed Systems. Concepts and Design.* Addison Wesley, 2011.
C. Cachin, R. Guerraoui, L. Rodrigues. *Introduction to Reliable and Secure Distributed Programming*. Springer, 2011.

8.2	**Laboratory**	**Teaching methods**	**Observations** (hours and bibliographic references)
1.	Preparation of the work environment (virtual machine).	Recap of relevant aspects in the course. Presentation of specific technologies. Individual work for accomodation with these. Exercises and phase progressive work for developing a software project. Periodical presentations of relevant articles in the field.	2
2.	Apache Storm. Work on the message streaming platform.	Same as above	2
3.	Apache Storm. Parallel event stream processing. Utility for a simple publish/subscribe solution.	Same as above	2

4.	Apache Storm. Event stream grouping. Fault tolerance aspects.	Same as above	2
5.	Message encapsulation and serialization: Google Protocol Buffers. Practical application of various notions introduced by the course (e.g., message partitioning).	Same as above	2
6.	Semestrial project description. Homework evaluation.	Same as above	2
7.	Storm in cluster mode. Work on the project. Practical application of various notions introduced by the course. Presentation of relevant articles in the field.	Same as above	2
8.	Partial evaluation. Articles presentation.	Same as above	2
9.	Work on the project. Presentation of relevant articles in the field. Optional : using alternative platforms to Apache Storm (e.g., Apache Flink)	Same as above	2
10.	Work on the project. Presentation of relevant articles in the field.	Same as above	2
11.	Work on the project. Presentation of relevant articles in the field.	Same as above	2
12.	Work on the project. Presentation of relevant articles in the field.	Same as above	2
13.	Work on the project. Apache Storm. Advanced configuration options. Using Trident for persistent storage. Presentation of relevant articles in the field.	Same as above	2
14.	Final discussions on project status. Article presentations.	Same as above	2

Bibliography

S.T. Allen, M. Jankowski, P. Pathirana. *Storm Applied.* Manning, 2015.
Storm online tutorial - https://storm.apache.org/documentation/Tutorial.html

9. Course content synchronization with the expectations of the community representatives, professional associations and employers from the program domain

The course approaches a field of high interest in the enterprise software industry environment and covers architectural elements frequently used in practical distributed solutions. The content of the laboratory is focused on getting aquainted with platforms and tools used by top names in the cloud computing and distributed systems area, such as Google (Google Protocol Buffers), Yahoo! (Apache Zookeeper), Twitter (Apache Storm, Apache Thrift), Facebook (Apache Cassandra), LinkedIn (Apache Kafka).

10. Evaluation

Activity Type	10.1 Evaluation criteria	10.2 Evaluation methods	10.3 The weight of each evaluation form (%)
10.4 Course	- understanding of the distributed architecture, of the messaging paradigms and of the data storage in a distributed event based system - capacity to identify and solve various scalability, synchronization, consesus, fault tolerance, security issues in such systems - quality of stating the answers	Written tests – part of ongoing evaluation during semester (Note: in the context of the online evaluation, the tests will be replaced by a single equivalent evaluation covering all the subjects in the course curricullum)	30%
10.5 Laboratory	- the capacity to use diverse platforms and existing tools for the development of a publish/subscribe system - reading and summarizing the relevand information in various publications of the scientific field - the quality of the project development (scalability/fault tolerance/security)	Collaborative project (35%) Presentation of article (20%) – part of ongoing evaluation during semester Homeworks (15%) – part of ongoing evaluation during semester (Note: in the context of the online evaluation, the above presentations will be conducted via typical online means)	70%
10.6 Minimal performance standards			
For the theoretical side the fulfillment of the minimal conditions implies the understanding of the *basic* theoretical notions regarding the distributed event based systems architecture. For the practical side the fulfillment of the minimal conditions implies implementing a distributed application that offers the *minimal* functionality (without optimizations or additions) of a publish/subscribe service relying on content based filtering of the messages. Meeting the mentioned minimal conditions corresponds to accumulating a total minimum graduation score out of the maximum possible grade, following the evaluations during the semester time.			

Date:
23.09.2021

Course Teacher
Assoc. Prof. Emanuel Onica, Ph.D.

Laboratory Teacher
Assoc. Prof. Emanuel Onica, Ph.D.

Department Date of Approval

Director of the Department