SYLLABUS

1. Information regarding the programme

1.1 Higher education institution	**Babes-Bolyai University**
1.2 Faculty	**Faculty of Mathematics and Computer Science**
1.3 Department	**Department of Computer Science**
1.4 Field of study	**Computer Science**
1.5 Study cycle	**Bachelor**
1.6 Study programme / Qualification	**Computer Science**

2. Information regarding the discipline

2.1 Name of the discipline				**Blockchain: Smart Contracts**			
2.2 Course coordinator				**Assoc. Prof. Ing. Florin Craciun**			
2.3 Seminar coordinator				**Assoc. Prof. Ing. Florin Craciun**			
2.4. Year of study	**3**	2.5 Semester	**6**	2.6. Type of evaluation	**C**	2.7 Type of discipline	**Optional**

3. Total estimated time (hours/semester of didactic activities)

3.1 Hours per week	3	Of which: 3.2 course	2	3.3 seminar/laboratory	1
3.4 Total hours in the curriculum	36	Of which: 3.5 course	24	3.6 seminar/laboratory	12
Time allotment:					hours
Learning using manual, course support, bibliography, course notes					8
Additional documentation (in libraries, on electronic platforms, field documentation)					7
Preparation for seminars/labs, homework, papers, portfolios and essays					8
Tutorship					2
Evaluations					8
Other activities:					-
3.7 Total individual study hours	89				
3.8 Total hours per semester	125				
3.9 Number of ECTS credits	5				

4. Prerequisites (if necessary)

4.1. curriculum	Fundamentals of Programming, Algorithms and Data Structures, Object-Oriented Programming, Advanced Programming Methods, Logic and Functional Programming
4.2. competencies	Basic knowledge in Python, Java, C#, C++

5. Conditions (if necessary)

5.1. for the course	Projector for lecture presentations
5.2. for the seminar /lab activities	Computers for practical assignments

6. Specific competencies acquired

Professional competencies	• Good programming skills in high-level languages • Better understanding of the program execution • Better knowledge about program semantics • Better knowledge about automated program verification • Better knowledge about writing correct code • Better knowledge about code optimization
Transversal competencies	• Ability to design and build dependable software systems • Ability to design and build critical systems

7. Objectives of the discipline (outcome of the acquired competencies)

7.1 General objective of the discipline	• Understanding of the main concepts and techniques of blockchain technology, with main focus on smart contracts
7.2 Specific objective of the discipline	• To understand the execution model of Ethereum platform • To understand bitcoin concepts • To understand the execution of smart contracts • To learn how to wite smart contracts • To become familiar with the tools which automatically analise, optimize and verify smart contract

8. Content

8.1 Course	Teaching methods	Remarks
1. Introduction into Blockchain foundations and applications	• Interactive exposure • Explanation • Conversation • Didactical demonstration	
2. Basics of Ethereum	• Interactive exposure • Explanation • Conversation • Didactical demonstration	
3. Introduction in Smart contracts	• Interactive	

	exposure • Explanation • Conversation • Didactical demonstration	
4. Smart contracts. Design patterns in Solidity	• Interactive exposure • Explanation • Conversation • Didactical demonstration	
5. Advanced topics on Solidity	• Interactive exposure • Explanation • Conversation • Didactical demonstration	
6. Decentralized Applications	• Interactive exposure • Explanation • Conversation • Didactical demonstration	
7. Bitcoins. Foundations	• Interactive exposure • Explanation • Conversation • Didactical demonstration	
8. Bitcoins. Advanced topics	• Interactive exposure • Explanation • Conversation • Didactical demonstration	
9. Consensus protocols. Foundations	• Interactive exposure • Explanation • Conversation • Didactical demonstration	
10. Consensus protocols. Advanced topics	• Interactive exposure • Explanation • Conversation • Didactical demonstration	
11. Security in Ethereum	• Interactive exposure • Explanation • Conversation • Didactical demonstration	

12. Mining strategies, Mining attacks	• Interactive exposure • Explanation • Conversation • Didactical demonstration	
13. Advanced topics on Blockchain verification	• Interactive exposure • Conversation	
14. The future of Blockchain	• Interactive exposure • Conversation	

Bibliography

1. (Main textbook-free available)Narayanan, Bonneau, Felten, Miller and Goldfeder, *Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction*

2. Bonneau, Miller, Clark, Narayanan, Kroll and Felten , Research Perspectives and Challenges for Bitcoin and Cryptocurrencies

3. Jeremy Clark , an extensive online bibliography of Bitcoin research papers

4. Bitcoin Developer Reference

5. Satoshi Nakamoto , Bitcoin: A Peer-to-Peer Electronic Cash System

6. Ethereum extensive wiki

7. Bitcoin Wiki

8. A.M. Antonopoulos, G. Wood , Mastering Ethereum: Building Smart Contracts and DApps O'Reilly Media, 2018

9. A.M. Antonopoulos , Mastering Bitcoin , O'Reilly Media, 2017

10. A. Bahga, V. Madisetti , Blockchain Applications: A Hands-On Approach, VPT Publishing House, 2017

11. Solidity: https://solidity.readthedocs.io/en/v0.5.10/

8.2 Seminar / laboratory	Teaching methods	Remarks
1. Configuration of Ethereum client	Conversation, debate, case studies, examples	The laboratory is structured as 2 hours classes every second week
2. Tools:Ganache, Remix, Mycrypto	•	
3. Solidity	•	
4. Project assignment		
5. Metatask and Design Patterns	•	
6. Decentralized Applications	•	
7. Project evaluation	•	
	•	
	•	

Bibliography

1. Jeremy Clark , an extensive online bibliography of Bitcoin research papers
2. Bitcoin Developer Reference
3. Satoshi Nakamoto , Bitcoin: A Peer-to-Peer Electronic Cash System
4. Ethereum extensive wiki
5. Bitcoin Wiki
6. A.M. Antonopoulos, G. Wood , Mastering Ethereum: Building Smart Contracts and DApps O'Reilly Media, 2018
7. A.M. Antonopoulos , Mastering Bitcoin , O'Reilly Media, 2017
8. A. Bahga, V. Madisetti , Blockchain Applications: A Hands-On Approach, VPT Publishing House, 2017
9. Solidity: https://solidity.readthedocs.io/en/v0.5.10/

9. Corroborating the content of the discipline with the expectations of the epistemic community, professional associations and representative employers within the field of the program

- The course respects the IEEE and ACM Curriculla Recommendations for Computer Science studies
- The content of the course is considered by the software companies as important for average software development skills

10. Evaluation

Course	- know the basic principle of the domain; - apply the course concepts in problem solving	Written Final Exam	50.00%
Seminar/lab activities	- be able to use course concepts in solving the real problems	Laboratory Work	50.00%

- At least grade 5 (from a scale of 1 to 10) at written final exam and at each laboratory assignment.

Date	Signature of course coordinator	Signature of seminar coordinator
	Assoc. Prof. Florin Craciun	Assoc. Prof. Florin Craciun

Date of approval

Signature of the head of department