

CSE4/526 Description

Bina Ramamurthy

2/1/2022

CSE4/526 Blockchain Application Development Spring 2022

Short Description

This course is intended for students interested in learning about blockchain technology and developing applications using blockchain concepts. It begins with the definition of the blockchain as a trust layer over the internet. Blockchain is meant for working with distributed resources and decentralized and disintermediated control. Topics include: Definition of a blockchain in terms of transactions, blocks, and chain of blocks, data structures enabling the blockchain protocol, and operational details involving algorithms and techniques such as peer-to-peer transactions, cryptography, digital signing and hashing, and consensus mechanisms. All of these concepts will be illustrated using Bitcoin and Ethereum blockchain. In the second part of the course, we introduce the idea of code execution on the blockchain and the program module called smart contract and a language, Solidity, for writing smart contracts, compiling, deploying, and testing the smart contracts on the Ethereum blockchain. In the last part of the course, we bring together the concepts into a decentralized application (Dapp) stack and explore problem-solving using blockchain. This process involves designing and developing a Dapp stack with the computational logic represented by the smart contract code, a user interface, and support for external data access through oracles and decentralized file systems. Students will work on hands-on end-to-end Dapp projects using the Ethereum blockchain and Truffle integrated development environment (IDE). The course will also discuss standards, best practices, current challenges, scalability and interoperability, and the respective solutions. On completion of the course, a student will analyze a problem and develop and implement an end-to-end blockchain solution on the Ethereum blockchain.

New this semester, a focus on crypto, dealing with it, and the new roles for investment and participation.

Course Outcomes

On completion of this course, a student will be able to design and implement a smart contract, and a blockchain-based decentralized web application (Dapp). Students will have good understanding and working knowledge of cryptocurrency eco systems.

Course Information

- Website for general information: <http://www.cse.buffalo.edu/~bina/cse426/fall2022>
- We will use **ublearns** for the ongoing course activities.
- Instructor: Bina Ramamurthy (bina@buffalo.edu)
- Lecture Time: TuTh 5:30PM - 6:45PM
- Lecture Location: Davis 101
- Office Hours: TTh: 12.00 Noon - 1.30 PM

Course Prerequisite

CSE250 Data Structures, equivalent, or permission of the instructor. You should have a good foundation in problem solving, design representation, and object-oriented design methodology and application and design and development in a highlevel language. You should also be familiar with object-oriented modeling, modern code design and debugging practices. Working knowledge of JavaScript language and web stack development are essential.

Text book

B. Ramamurthy. Blockchain in Action. Manning publishers. <https://www.manning.com/books/blockchain-in-action>, ISBN-13: 978-1617296338, 2020. The instructor will provide many reference material throughout the course to support the projects and blockchain application development.

Grading Policy

Grading Distribution Final letter grades will be based on the (combined) overall percentage of all the items listed below. A (95 –), A- (90 – 94), B+ (85 – 89), B (80 – 84), B- (75 – 79), C+ (70 – 74), C (65 – 69), C- (60 – 64), D+ (55 – 59), D (50 – 54), F (less than 50). This policy is subject to change. If needed, the individual components and the overall grades will be appropriately curved.

Grade components

Component (Quantity)	Points
Attendance including pop quiz	100
Project (1)	100
Exam 1 (1)	100
Exam 2 (1)	200

Project

There is one project with multiple phases. This strategy will allow for incremental development and manage your workload appropriately spread over the semester rather than rushing and panicking at the end of the semester. The due date for each project phase will be announced when it is assigned. All the source code, documentation, makefile, data files, and README files will be submitted online. The details of submission will be given along with your first project phase. You will have to follow the rules for the other phases too. I reserve the right to change the project specifications at any point before the due date to address problems that may arise during the project. If your design is modular, the changes will not be difficult to implement. A detailed grading guideline will be given to you along with the project specification. Use this as a guide for your design and implementation. It is necessary to keep up with the programming projects in the class. No late assignments will be accepted. Keep submitting partial project material as you develop them. Develop your code using the Incremental Development technique. Do not try to sit down and code the entire assignment in one sitting. Instead, take one section at a time, implement, test it, back up the code, and move on to the next section. You will turn in each phase before 11:59 PM on the due date via the departmental submit command. You must also include appropriate testing programs to show the validity of your solution. In addition, you must include external documentation discussing the “how’s and why’s” of your design and implementation. You will be required to demonstrate your lab to your TA. The TA will also run test examples against your code to check your solution’s overall correctness. The TA will provide a

demo schedule. It is your responsibility to demo your project, or you will receive a zero for that portion of the grade. When your grade is assigned for the project phases, the TA will indicate critical areas that must be fixed to solve the next phases.

Exams and quizzes

There will be a midterm that will be administered and graded before the resignation date. Exam1 will cover all lecture and reading assignments before the exam and concepts from the lab assignments. Exam2 is a comprehensive exam covering all lecture, lab, and homework areas. All exams are closed books, closed notes, and closed neighbors. We do not give makeup exams for any reason. If you miss an exam, you will receive a zero for that portion of the grade.

Quizzes will be pop-quizzes and may be unannounced. No makeup quizzes will be given.

Attendance Policy

You are responsible for the contents of all lectures and recitations (your assigned section). If you know that you are going to miss a lecture or a recitation, have a reliable friend take notes for you. Of course, there is no excuse for missing due dates or exam days. Attendance will be taken and will determine the attendance grade component to compute your final grade.

During lectures, we will be covering material from the textbook. We will also work out several of the problems from the text. Lectures will feature exploring several real-world problems not covered in the book. Lecture dynamics will depend very much on students' participation, and students are strongly encouraged to ask questions related to the material covered in the class. You will be given a reading assignment at the end of each lecture for the next class.

Office hours are designed to review difficult concepts in the class and to spend additional time discussing the lab work required for the course. This is your time to communicate with your TA about the course, project, exams, quizzes etc. Use the opportunity to the fullest.

Office Hour Policy

If you can't meet during the designated office hours, you will have to communicate via Email. Office hours are intended to resolve questions about the material that could not be answered in lecture or recitation. Come to office hours prepared! Office hours are NOT for the following: to repeat missed lecture material or have the instructor or TA solve an assigned problem for you. We will NOT write or debug your code for you during office hours! Instead, we will direct you to where to concentrate your debugging efforts.

Grading Policy

All assignments will be graded and returned promptly. When an assignment is returned, you will have a period of one week to contest any portion of the grade. The TA who graded your assignment will be the first person to resolve a grading conflict. The instructor will mediate the dispute if the conflict cannot be resolved. The judgment of the instructor will be final in all such cases. When contesting a grade, you must demonstrate how your particular solution is correct. Also, when contesting a grade, the instructor or TA reserves the right to reevaluate the entire lab or exam, not just the portion in dispute.

Incomplete Policy

We only grant incompletes in this course under the direst of circumstances. By definition, an incomplete is warranted if the student is capable of completing the course satisfactorily, but some traumatic event has

interfered with their capability to finish within the timeframe of the semester. Incompletes are not designed as stalling tactic to defer a poor performance in a class.

Academic Integrity Policy

UB's definition of Academic Integrity in part is, "Students are responsible for the honest completion and representation of their work". It is required as part of this course that you read and understand the departmental academic integrity policy located at the following URL: <https://engineering.buffalo.edu/computer-science-engineering/information-for-faculty-and-staff/academicintegrity.html> There is a very fine line separating conversation pertaining to concepts and academic dishonesty. You are allowed to converse about general concepts, but in no way are you allowed to share code or have one person do the work for others. You must abide by the UB and Departmental Academic Integrity policy at all times. NOTE: Remember that items taken from the Internet are also covered by the academic integrity policy! If you are unsure if a particular action violates the academic integrity policy, assume that it does until you receive clarification from the instructor.

Ublearns and Piazza

The CSE426/526 ublearns should be checked frequently for important news. Course assignments, slides, grade reporting, and general hints and tips will be posted on the ublearns blackboard LMS system. Also new this semester is the licensed version of piazza for your class discussions. Please enroll in it and make the best use of it. Piazza is only for any public discussion, be civil. Also do not post any code publicly on Piazza. Piazza cannot solve any individual grading question. In this case, you should contact your TA or teh instructor for grading questions.

Students with Accessibility Issues

If you have special needs due to a disability, you must be registered with the Office of Student Accessibility. If you are registered with them please let your instructors know about this so that they can make special arrangements for you.

Piracy and illegal use of course material

All materials prepared and/or assigned by me for this course are for the students' educational benefit. Other than for permitted collaborative work, students may not photograph, record, reproduce, transmit, distribute, upload, sell or exchange course materials, without my prior written permission. "Course materials" include, but are not limited to, all instructor-prepared and assigned materials, such as lectures; lecture notes; discussion prompts; study aids; tests and assignments; and presentation materials such as PowerPoint slides, Prezi slides, or transparencies; and course packets or handouts. Public distribution of such materials may also constitute copyright infringement in violation of federal or state law. Violation of this policy may additionally subject a student to a finding of "academic dishonesty" under the Academic Integrity Policy and/or disciplinary charges under the Student Code of Conduct.

Violations

Students who violate this policy will be required to complete an educational sanction about the value of intellectual property. More serious and/or repeat violations of this policy may be treated as acts of "academic dishonesty" under the Academic Integrity Policy 2 (<https://www.buffalo.edu/academic-integrity/policies.html>) or subject a student to disciplinary charges under the Student Code of Conduct.