Blockchain for Developers DeCal (Fall 2025)

Lectures: Wednesday 5-6:30PM, Location: SOCS 20

Office Hours: By appointment

Course Email

dev-decal@blockchain.berkeley.edu

Course GitHub

https://github.com/BerkeleyBlockchain/fa25-dev-decal

Instructor

Subham Mitra (subham.mitra@berkeley.edu)

Introduction

Welcome to Blockchain @ Berkeley's Blockchain for Developers DeCal for the Fall 2025 semester! This decal is designed to provide students a comprehensive overview of relevant topics in blockchain development, as well as hands-on experience in developing and deploying their own smart contracts & sovereign Cosmos blockchains.

Blockchain is currently one of the fastest-growing industries in the technology sector. In response to this shortage of blockchain developers, this course aims to teach students the technical fundamentals of blockchains, Solidity programming language, Cosmos ecosystem, as well as industry-relevant tools such as Metamask, Wagmi, Foundry, Rust, and Cosmos SDK such that students will be equipped with industry-relevant experience in an accessible, collaborative environment. We hope that through this course, students will become more confident in their

ability to develop and deploy blockchain-based solutions on important industry issues.

Class Format

The course will be split into two parts. The first half of the course will provide a quick overview of the blockchain and a deep dive into Ethereum and its smart contracts. In this process, you will learn how to write your own smart contract and dApp frontends using industry-standard developer toolings like Foundry and Wagmi. In the second half of the course, students will be getting an introduction to the Rust programming language.

Each class will be divided into two parts as well. The first hour of each class will be dedicated to a lecture covering the topics planned for that week. We will then switch gears in the second hour of class to focus on more hands-on work with that week's topics.

Starting from 09/10/2025, each weekly lecture will cover:

Week #	Date	Lecture Title / Content
Week 1	9/10	❑ Introduction to Blockchain Development: A High-Level Overview
Week 2	9/17	❑ Solidity and Developer Tools
Week 3	9/24	❑ Advanced Solidity
Week 4	10/1	❑ Ethers.js, Wagmi, and Connecting to the Web
Week 5	10/8	❑ Writing Secure and Efficient Solidity
Week 6	10/15	❑ dApp Integrations: Compiling Contracts, Upgradability, Oracles
Week 7	10/22	❑ Memory Management in Blockchain Development
Week 8	10/29	❑ Rust Basics
Week 9	11/5	❑ Rust 2
Week 10	11/12	❑ Rust 3
Week 11	11/19	❑ Rust 4
Week 12	12/3	❑ Finale / Guest Lecture

Prerequisites

On a high level, this course will require some level of understanding of the fundamental Computer Science concepts. Completion of CS 61A (Structure and Interpretation of Computer Programs), CS 61B (Data Structures), and ideally CS 61C (Machine Structures) or relevant programming experience is highly recommended.

The bulk of the course will be focused on teaching Solidity (similar to JavaScript), Rust, and using touches of JavaScript (React) and Python. We do not require any prior experience with any of these languages, but do plan to allocate some extra time to pick it up if you haven't used a certain programming language before.

As previous experience with programming is a prerequisite for this course, we do expect students to be familiar with developing software already. However, it is likely that many of the tools and concepts introduced in this course are unfamiliar to those new to the blockchain space. If you face any challenges with your dev setup, please do not hesitate to reach out to any of the DeCal staff for help! We are here because we want every student in this course to succeed and/or achieve their personal academic goals for this course. We will do our best on our end to create an environment where students feel comfortable asking for help.

Enrollment

This course is open-enrollment. **We request that you honestly assess your level of preparedness for this class. We will not be offering debugging support for basic Python bugs - this is not a class that teaches you how to code!**

Note that this course requires a background in programming (whether it be through the Berkeley classes such as CS 61A or equivalent experience - you should know object oriented programming in at least one language). We recommend having taken CS61A, CS61B, and highly recommend having taken CS61C. You should also be broadly familiar with blockchain at the level of the Blockchain Fundamentals course - we will be doing a quick recap at the start of the semester but we will not be spending much time on this. If you do not meet these criteria, we may request you to take the class at a later semester.

Grading

Homework: In this course, you will be assigned homework that is based on the content of the previous week's lecture. Homeworks contains fundamental exercises to strengthen and assess your understanding of the lecture's content. This will mostly be small coding challenges and quiz questions, akin to CS61A's homework. All homework check-offs will be graded on completion due the night before the next lecture.

To pass the class, you must submit all the homework on time except two of your choice (two homework drops).

Homework will be posted weekly on the course GitHub repository:
https://github.com/BerkeleyBlockchain/fa25-dev-decal

Project: Our goal is to have you leave this class with real projects under your belt that set you up to keep growing after the end of the last lecture. To ensure your success, you will be implementing one project over the course of the entire Ethereum section, with one piece relevant to each lecture and another project over the course of the entire Rust section. During the Ethereum portion, you will build an efficient, secure, and upgradable NFT marketplace with a front-end that supports viewing and trading of NFTs. By the end, you'll even get to architect your marketplace to integrate with other protocols live on Ethereum to implement a custom feature of your choice. The Rust section will involve building an MEV bot which may be used to trade cryptocurrencies.

This is a perfect opportunity for you to put into practice the concepts you learned in lecture into a significant blockchain based application. Don't be intimidated! Roughly half of each class will be dedicated workshop time to work on that week's piece of the project.

If you have any questions about the project, please contact us, we'd be happy to talk about it. **Completion of these projects are mandatory to pass the class.**

Grade Breakdown:

- **Homework (weekly coding challenges & quizzes): 40%**
- **Ethereum Project (NFT marketplace): 30%**
- **Rust Project (MEV bot): 30%**

Percentage	Letter Grade
100% - 90%	A
89% - 80%	B
79% - 70%	C
69% - 60%	D
59% - 0%	F

Gradescope

Entry Code: TBA

Email Subham Mitra with any questions regarding joining the Gradescope course.

Policies

- Plagiarism and academic dishonesty are strictly prohibited and are treated with automatic failure of the course. Working collaboratively and discussing ideas are encouraged.
- Please do not post your solutions on a public repository on sites such as GitHub or GitLab. Please store them locally or use GitHub's private repository feature.
- If you have any questions, please feel free to contact any course staff.

Required Readings

- Introduction to Smart Contracts (https://ethereum.org/en/developers/docs/smart-contracts/)
- Ethereum Documentation (https://ethereum.org/en/developers/docs/intro-to-ethereum)
- Ethereum Whitepaper (https://ethereum.org/en/whitepaper)
- Bitcoin Whitepaper (https://bitcoin.org/bitcoin.pdf)
- UniswapV2 Whitepaper (https://app.uniswap.org/whitepaper.pdf)
- Rust Documentation (https://doc.rust-lang.org/stable/std/index.html)
- Solidity Documentation (https://docs.soliditylang.org/en/v0.8.30/)
- Flashbots MEV Documentation (https://docs.flashbots.net)
- ERC-721 Non-Fungible Token (NFT) Standard (https://eips.ethereum.org/EIPS/eip-721)

Supplemental Readings

- Working in Web3 Handbook (https://www.smsunarto.com/web3)
- useWeb3 (https://www.useweb3.xyz)
- EthSpring Resources (https://ethspring.com)
- Mastering Ethereum (https://github.com/ethereumbook/ethereumbook)
- In-depth EVM Guide (https://hackernoon.com/getting-deep-into-evm-how-ethereum-works-backstage-ac7efa1f 001)

- Cosmos Academy (https://tutorials.cosmos.network/academy/0-welcome/)
- A Cosmos Thesis (https://research.paradigm.xyz/cosmos-thesis

Past Class Recordings

- Blockchain Fundamentals Decal - Fall 2021 Video Recordings (https://www.youtube.com/watch?v=VFqtJosbBTk&list=PLSONl1AVlZNWoeYjazuvIVTeX7rx BtDNh)
- Blockchain for Developers Decal - Spring 2021 Video Recordings (https://www.youtube.com/watch?v=wKZaB4LNVg8&list=PLSONl1AVlZNWJVixT2vwY9-6O 7kgM4het)

RDI Certificate in Decentralization & Web3 Technology Innovations
For those interested in learning more about Web3 and Blockchain technologies, see the RDI certificate.