



Blockchain Developer

Become a blockchain developer

TERM 1: FUNDAMENTALS

Blockchain Fundamentals

Project 1: Create Your Identity on Bitcoin Core

Blockchains are a public record of completed value transactions that are immutable and decentralized. These transactions often contain sensitive data such as personally identifiable information—so how does a blockchain secure against this information leaking into the wrong hands? The answer to this question lies in how “identity” is created, established, and maintained on a blockchain.

Your blockchain “identity” is an encrypted private-public key pair that allows you to anonymously interact with information on the blockchain, while still allowing others to verify your data. In this project, you will generate a private-public key pair on Bitcoin Core and learn how to use it to securely authenticate your identity for the transactions you post.

Supporting Lesson Content: Blockchain Fundamentals

Lesson Title	Learning Outcomes
BLOCKCHAIN BASICS	<ul style="list-style-type: none">→ Identify the purposes and characteristics of blockchain platforms→ Explain core components that make a blockchain secure and powerful→ Define blockchain protocols and their key differences
MANAGING BLOCKCHAIN TRANSACTIONS	<ul style="list-style-type: none">→ Describe how transactions occur between 2 entities (e.g. Bob and Alice) using a private key, public key, wallet on a public blockchain→ Create and manage identity on the Bitcoin Blockchain and establish proof-of-ownership with blockchain transactions, without the need to provide sensitive information

Project 2: Create Your Own Private Blockchain

A blockchain is a shared database that features added immutability as a safe and accurate alternative to existing data storage methods. You'll learn the basics of how the blockchain data model works by creating your own private blockchain using Node.js and Leveldb.

In this project, you'll learn the fundamentals of architecting a collection of data into a blockchain data model. You'll configure how each block stores data, validate blocks, add new blocks to the chain, and create methods to validate the chain integrity.

Supporting Lesson Content: Blockchain Data

Lesson Title	Learning Outcomes
BLOCKCHAIN: DATA OVERVIEW	→ Describe the general concepts of a blockchain data model
BLOCKCHAIN INTERACTION: BITCOIN CORE INTRO	→ Identify the tooling available to explore the Bitcoin Core public blockchain
BITCOIN CORE TESTNET	→ Explain the benefits of utilizing the Bitcoin Core testnet → Describe the difference between the public testnet and regression testing
BLOCKCHAIN DATA	→ Learn the relationship between different stages of transaction lifecycle using Bitcoin Core
PRIVATE BLOCKCHAINS	→ Explain the value of a private blockchain, and prepare for the course project

Project 3: Connect Private Blockchain to Front-End Client via APIs

To be able to build real-world applications on private blockchain, it needs to integrate with a web service so the data is consumable by client applications such as desktop, mobile and IoT.

For this project, you'll create a back-end API web service, and then migrate your private blockchain—built in Project 2—to the web service. In the process, you'll learn how to post new blocks to the blockchain via a RESTful web client, and learn how to validate blockchain health. By the end of the project, you'll have transformed your private blockchain into a valuable web service.

Supporting Lesson Content: Blockchain Web Services

Lesson Title	Learning Outcomes
BLOCKCHAIN WEB SERVICES	→ Use and build blockchain web services with Bitcoin Core Remote Procedure Call utilizing Node.js

Project 4: Secure Digital Assets on a Private Blockchain

With a blockchain that ensures data security, one of the most important features is the ability for a transaction on the blockchain to be signed and validated. These signed transactions provide a great way to prove ownership of any data on the blockchain at any given point in time.

Data stored on a blockchain can vary from digital assets (e.g. documents, media) to copyrights and patent ownership.

In this project you'll build on the skills learned in Project 3. You'll build an API service to claim ownership of a digital asset—in this case, your favorite star. The project will focus on building out the back-end infrastructure with a pre-developed front end. You'll learn how to encrypt and decrypt the unique digital collectibles on a private blockchain, and how to publicly prove who rightfully owns it. You'll also configure your web service using a RESTful web client to post new blocks to the blockchain. Along the way, you'll also explore how to manage the limitations of block size when handling digital data.

Supporting Lesson Content: Digital Assets on Blockchain

Lesson Title	Learning Outcomes
DIGITAL ASSETS ON BLOCKCHAIN	→ Encode and decode digital assets on a private blockchain, and publicly prove ownership of the assets using digital identity

Project 5: Build CryptoStar Dapp on Ethereum

With Project 5, your focus moves from Bitcoin to Ethereum blockchain. You'll begin by building a decentralized app (Dapp) that allows you to create, sell, and transfer ownership of unique star token (CryptoStar) on the Ethereum blockchain using smart contracts and the non-fungible (ERC721) token standard. This service is designed to demonstrate how to claim and transfer ownership of unique digital asset (e.g. document, deed, agreement, media, etc.) on Ethereum blockchain.

You'll build the back-end infrastructure for the CryptoStar with a pre-developed front end. This will enable you to render the service on any modern web clients.

Supporting Lesson Content: Ethereum Smart Contracts, Tokens and Dapps

Lesson Title	Learning Outcomes
ETHEREUM FUNDAMENTALS AND DEVELOPMENT TOOLS	<ul style="list-style-type: none">→ Describe the fundamentals of Ethereum and how it is different from Bitcoin→ Build, compile, deploy, and test smart contracts using remix, ganache, truffle, and infura
SMART CONTRACTS WITH SOLIDITY	<ul style="list-style-type: none">→ Learn Solidity, a Turing complete smart contract language→ Learn about different token standards (ERC-721, ERC-20)→ Create a fungible (ERC-20) token on Ethereum using Solidity
ETHEREUM DAPP	<ul style="list-style-type: none">→ Create a fully-functioning Dapp that allows users to create, buy and sell their unique stars<ul style="list-style-type: none">❑ Write a smart contract with functions to support proof of existence❑ Compile and migrate the smart contract❑ Test smart contract code coverage❑ Deploy smart contract on a public test network❑ Create a user interface to interact with the smart contract❑ Interact with the dapp in a browser

TERM 2: ARCHITECTURE & APPLICATIONS

Project 6: Ethereum Dapp for Tracking Items through Supply Chain

By the completion of Term 1, you'll have learned the importance of Proof of Existence, which is used to verify whether a digital asset is authentic and can be trusted. In this project, you'll scale up to architect a solution that verifies authenticity for a product when multiple actors are involved.

You'll build a supply chain system on Ethereum blockchain that allows users to verify the authenticity of an item as it passes through different hands. You will architect a Dapp (Decentralized Application) authenticity management system backed by the Ethereum platform. To do so, you'll scope out the needs of the various actors in the supply chain and create smart contracts that help track product origination and verify product authenticity. You'll then tie this all together with a simple front-end that allows users to manage the product life-cycle as the product moves through the supply chain.

Supporting Lesson Content: Blockchain Architecture

Lesson Title	Learning Outcomes
PLANNING BLOCKCHAIN SOLUTIONS	<ul style="list-style-type: none">→ Learn the correct technology stack to layer services and provide software solutions→ Design supporting visuals with Unified Modeling Language (UML)
PRIVACY	<ul style="list-style-type: none">→ Implement several techniques to enhance privacy of blockchain such as merkle trees, zero-knowledge proofs
SECURITY & MAINTENANCE	<ul style="list-style-type: none">→ Identify architecture security and maintenance risks
DISTRIBUTED FILE SYSTEM	<ul style="list-style-type: none">→ Create your own website and Dapp on the new decentralized storage protocol

Project 7: Flight Delay Insurance Dapp

In the real-world, many smart contracts perform actions based on external triggers. These triggers can be caused by our actions, or by data received from outside sources using what are known as “oracles.” To make real-world decentralized applications, smart contracts need to respond autonomously to these triggers, thereby making these applications more interactive.

In this project, you'll build a decentralized application for a use case in which you have airlines that offer flight delay insurance, and passengers who carry this insurance, and who get paid in the event their flight is delayed. You'll build multiple smart contracts which are autonomously triggered by external sources, and which handle payments based on flight delay scenarios.

Supporting Lesson Content: Dapp with autonomous smart contracts and oracles

Lesson Title	Learning Outcomes
MULTIPARTY CONTROL AND PAYMENTS WITH SMART CONTRACTS	→ Build Dapp with secure, multi-sig smart contracts that autonomously receive, transfer, and pay funds.
ORACLES	→ Utilize third-party data sources to inform autonomous smart contracts
HANDLING SMART CONTRACT PAYMENTS	→ Create, and test, secure and cost-efficient smart contracts that handle, distribute, and test ETH payments to a smart contract

Project 8: Capstone Project

Here, you'll use all the new skills you've acquired to build your own supply chain blockchain project!

You'll work end to end—from building your architecture (centralized, decentralized), and choosing your technology stack (front end, back end), to surfacing the protocol and blockchain features to the Dapp layer, to experimenting, learning, and collaborating with other open-source projects to complete your own project.

Supporting Lesson Content: Capstone

Lesson Title	Learning Outcomes
SUPPLY CHAIN & DISTRIBUTION OF PRODUCTS & SERVICES	→ Identify the benefits of using a public blockchain to support distribution channels
CAPSTONE PROJECT PREP	→ Design supporting visuals with Unified Modeling Language (UML)