

# INSTRUCTOR-LED WEBINAR - SYLLABUS

## EOS DEVELOPER

<b>Duration:</b>	16 Hours
<b>Delivery:</b>	Live Instructor-Led Webinar - 16 Hours
<b>Instructor(s):</b>	TBD
<b>Office Hours:</b>	10:00 AM to 6:00 PM Eastern Standard Time
<b>Email:</b>	studentsupport@blockchainhub360.com
<b>Prerequisites:</b>	Minimum 1-year software development experience. C++ , Python, Rust, and Solidity - C++ is currently the best language for EOSIO. Rust, Python, and Solidity toolchains are available to learn and build. High-performance and secure smart contracts are best developed using C++

**Continuing Education Units:** 1.6

**Certification Exam:** Certified EOS Developer  
**Certification Body:** Blockchain Certification Association

### **Course Overview:**

The EOS Developer course is designed for web/app developers who wish to understand how to create and/or integrate EOSIO based apps. The course gives a top-down view of all the important aspects of creating such an app and a basic overview of how the EOSIO blockchain works. EOSIO is a blockchain platform designed for the real world. Built for both public and private use cases, EOSIO is customizable to suit a wide range of business needs across industries with rich role-based security permissions, industry-leading speeds, and secure application processing. Building on EOSIO follows familiar development patterns and programming languages used by existing non-blockchain applications so developers can create a seamless user experience using development tools they already well known.

### **Course Composition:**

Instructor Led Webinar: EOS Developer Modules 1 - 4

### **Learning Objectives:**

- Create a working environment for deploying and interacting with smart contracts
- Explain EOS Taken Structure
- Demonstrate ability to read, write, modify and deploy smart contracts
- Appraise the Interoperability of EOS chains

**Demonstration of Learning Outcomes:**

At the conclusion of the EOS Developer, course developers have the ability to deploy EOS based smart contracts, and working dApps (distributed application).

**Evaluation:**

Evaluation is based on participation and a final exam.

Weighted:

50% participation

50% on the final grade

80% overall grade is required in order to receive a Certificate of Completion.

**Grading Policy:**

Pass or Fail. No Credit (NC).

**Attendance Requirements:**

Students are expected to attend at least 70% of Instructor-Led Webinar Presentations. Should a student miss any portion of the live instruction instructor-led webinars is recorded and attached to the learning management. A Certificate of Completion will not be issued if attendance requirements are not met.

**Student conduct and etiquette:**

Students will be expected to be courteous in their conduct and communications to the instructor and classmates at all times whether such conduct or communication is in person, by telephone or electronic communications.

Behavior that persistently or grossly interferes with the instructor or other student activities is considered disruptive behavior and may be subject to disciplinary action. Such behavior inhibits other students' ability to learn and an instructor's ability to teach. The instructor may require a student responsible for disruptive behavior to leave the learning environment pending discussion and resolution of the problem and may report a disruptive student to the Student Affairs Office

Note: Disruptions or any other distractions in the learning environment may result in a failing grade.

**Course Evaluations**

Course evaluations and program surveys are important components of the educational process. Students are encouraged to complete the student course evaluation form issued at the conclusion of the course. The evaluation is anonymous.

**Computer/Information Literacy Expectations for Students enrolled in this class**

Students in this class are expected to:

1. Use a word processing program for writing assignments (e.g., Microsoft Word)
2. Be able to access assigned websites through the internet
3. Have access to PC or mobile device for participation in course content

## Course Module Overview:

### EOS DEVELOPER

#### Module 1 – Understanding Blockchains Public/Private

Origins of Blockchain

Blockchain today

- Cryptography Fundamentals
  - Blocks
  - Hash
- Understanding Consensus Models
  - Proof of Work (PoW)
  - Proof of Stake (PoS)
  - Delegated Proof of Stake (dPoS)
  - Proof of Elapsed Time (PoET)
  - Byzantine Fault Tolerant (BFT)
  - Other Consensus models
- Public, private, and hybrid networks
- Distributed Ledger Technology
- Provenance, Immutability, and Finality

Separation of Cryptocurrency and Blockchain

Common misconceptions of Blockchain

Drawbacks of Blockchain Technology

When to and when not to use Blockchain

The current state of Blockchain

Ethereum vs Hyperledger vs. Corda vs. EOS vs Tron

Understanding the development environment

Understanding Smart Contracts and Oracles

#### Module 2: Introducing EOS.IO

Introduction to EOS

EOS Design Requirements  
Delegated Proof of Stake (dPoS)  
Block Producers  
EOS usability, dApps, and Smart Contracts  
How to build/install EOSIO  
Setting up a Dev Environment  
EOS Blockchain Governance  
Overview of EOSIO libraries  
Overview of storage capabilities  
overview of inline actions  
EOS Testnet and Toolset  
EOSJS  
Using desktop/browser wallets  
Voting  
Memory, Net, and CPU  
EOS command line tools CLEOS and EOSC

### **Module 3 – Tools of the EOS platform**

Where are we in EOSIO dApp development capabilities  
Accounts and Actions  
Account Role-based Permissions  
Permissions Levels  
Delayed Actions  
Account Recovery  
Parallel Execution of Apps  
Structured of EOS blocks  
Action Handlers and Atomic Transactions  
Deferred Transactions  
Context-Free Actions

### **Module 4 – Smart Contract**

Introduction to Smart Contracts  
Creating a Smart Contract

EOSIO Smart Contract capabilities

Modify/Deploy your smart contract

Development walk-through of an EOSIO Smart Contract

EOS Token Structure

Measures, Costs & Workers

Scripting

Interoperability of EOS chains

### **Module 5 – Blockchain Security**

Security Program Basics

ICO Security

Fundamentals

Node and Wallet Security

Basic Blockchain Security

Tools

Practical Blockchain Usage

### **Final Exam**