



மனோன்மணியம் சுந்தரனார் பல்கலைக்கழகம்
Manonmaniam Sundaranar University

**SYLLABUS FOR CERTIFICATE IN METAVERSE DESIGN PROGRAMME
OFFERED THROUGH DIRECTORATE OF VOCATIONAL EDUCATION
(COMMUNITY COLLEGES AND VOCATIONAL SKILL DEVELOPMENTCENTRES)
FROM 2023 - 2024**



CERTIFICATE IN METASVERSE DESIGN - 6 MONTHS

SCHEME OF EXAMINATION

Subject Code	Title of the Course	Credit	Hours	Passing Minimum
C23MD11 / E23MD01	Introduction to Web Technologies	6	90	40/100
C23MD12 / E23MD02	Unity Engine Essentials	6	90	40/100
C23MDP1 / E23MDP1	Practical I - Ethereum Block chain & Solidity	4	120	40/100
C23MDP2 / E23MDP2	Practical II - Web3 & Metaverse	4	120	40/100
C23MDPW / E23MDPW	Project	10	120	40/100

Eligibility for admission: Pass in 10th std examination conducted by the Govt. of Tamil Nadu board of secondary education, government of Tamil Nadu or any other equivalent examination.

Examination: Minimum Passing Score for each Course is 40%. Classification will be done based on percentage marks of the total marks obtained in all the courses and as given below:

40 % but less than 50 %	-	Third class
50 % but less than 60 %	-	Second class
60 % and above	-	First class

Course Duration: 6 Months

SYLLABUS

Course I	:	Introduction to Web Technologies
Course II	:	Unity Engine Essential
Course III	:	Practical I - Ethereum Block chain & Solidity
Course IV	:	Practical II - Web3 & Metaverse
Course V	:	Project

Program Objectives

- The Student should be able to build Metaverse application using Web Technologies and Unity Game Engine.
- To know how to create NFT.

COURSE I
INTRODUCTION TO WEB TECHNOLOGIES

Objectives:

- The Students will know how to design a webpage.
- Build an interactive landing page for the Metaverse application.
- To know how to write JavaScript events.

Unit I

15 Hrs

HTML FUNDAMENTALS

Introduction to HTML - History of HTML - Basic Structure of an HTML document - Tags and Elements - HTML Document Structure - Document type Declaration - HTML Document head and body elements - Meta tags and their importance - Text Formatting - Links (heading tags, paragraph, tags, line breaks) - Text Formatting tags(Bold, Italic, Underline) - Creating Hyperlinks - Buttons

Unit II

15Hrs

CSS STYLLING

Introduction to CSS - CSS syntax - Selectors - Inline Styles - Internal Style Sheets - External CSS files and Linking - Box model and Layout - Box model properties - Display and Position properties - Flexbox - Grid Layouts - Styling text and Backgrounds - Changing Font properties - Text color - Background color - Background Images Gradients

Unit III

20 Hrs

JAVASCRIPT BASICS

Introduction to JavaScript - Basic Syntax - Variables - Data Types - Operators - Control flow - Conditional Statements - Looping Structures - Functions - Parameter Passing

Unit IV

15Hrs

DOM MANIPULATION AND EVENT HANDLING

Introduction to the Document Object Model (DOM) - Accessing and Modifying HTML elements with JavaScript - Event Handling and Listeners - Dynamically Creating and Deleting Elements

Unit V

25Hrs

INTERMEDIATE JAVASCRIPT AND PROJECT DEVELOPMENT

Advanced JavaScript concepts (arrays, objects, closures, scope) - Error Handling (try -catch) - Introduction to ES6 features - Understanding Common Debugging Techniques, using browser developer tools to inspect and debug JavaScript code - Local Storage and Client - Side Data Management - Exploring the Local Storage and Session Storage APIs to store and Retrieve data locally on the client-side, Allowing persistent data management within your project

Web Reference:

1. HTML and CSS:

- **HTML and CSS: Design and Build Websites**" by Jon Duckett.
- **Learning Web Design: A Beginner's Guide to HTML, CSS, JavaScript, and Web Graphics**" by Jennifer Niederst Robbins.
- **CSS: The Missing Manual**" by David Sawyer McFarland.

2. JavaScript:

- **JavaScript: The Good Parts**" by Douglas Crockford.
- **Eloquent JavaScript: A Modern Introduction to Programming**" by Marijn Haverbeke.
- **"You Don't Know JS"** series by Kyle Simpson.

COURSE II

UNITY ENGINE ESSENTIALS

Objectives:

- The Students will know how to design using UI Toolkit.
- To know how to work in 3D Scenes.

Unit I

20 Hrs

FOUNDATION OF UNITY

Introduction to Blockchain Technology - What is Blockchain and its key features - The concept of decentralization and distributed ledger - Consensus mechanisms in blockchain - Use cases and benefits of blockchain technology

Unit II

10 Hrs

UNITY UI

Canvas - Rect Transform - Pivot - Anchors - Sprite Import - Image - TextMeshPro - Button - Input Field - Raw Image - Panel - Toggle - Slider - Scrollbar - Scroll View - Dropdown - Event System

Unit III

20 Hrs

UNITY ENGINE ESSENTIALS

Import 3D Asset - Design 3D Scene - Prefab - Asset Store - Package Manager - AV Components - Rigid Body & Colliders

Unit IV

30 Hrs

C# FOR UNITY

Variables - Functions - Class - Find Game Object - Get Component - Transformation - Text - Image - Events - Keyboard Input - Mouse Input - Touch Input

Unit V

10 Hrs

DESIGNING VIRTUAL MUSEUM

Scene creation - Import assets - Scene design - Light - FPS setup - Build settings - Export WebGL

Web Reference:

1. <https://docs.unity3d.com/Manual/UnityOverview.html>
2. <https://docs.unity3d.com/Manual/UIElements.html>
3. <https://docs.unity3d.com/Manual/AssetWorkflow.html>
4. <https://docs.unity3d.com/Manual/collision-section.html>
5. <https://docs.unity3d.com/Manual/AnimationSection.html>
6. <https://docs.unity3d.com/Manual/class-VideoPlayer.html>
7. <https://docs.unity3d.com/Manual/Audio.html>
8. <https://docs.unity3d.com/Manual/ScriptingSection.html>

COURSE III

PRACTICAL I - ETHEREUM BLOCK CHAIN & SOLIDITY

Objectives:

- The Students will know how to work in Ethereum Block Chain.
- To know how to write smart contract in solidity.
- To know how to create a Wallet.
- To know how to use OpenSea.
- To Know how to Build and deploy a virtual art museum application in Blockchain.

WRITING YOUR OWN SMART CONTRACT

48 Hrs

Overview of Ethereum - Syntax and data types in Solidity - Solidity language overview and Basic syntax - Data types - Variable declarations and Scoping rules - Type conversion and Casting - Functions, Modifiers, and Events - Defining and Calling functions in Solidity - Function modifiers and their uses - Events and event-driven programming in Solidity - Function visibility and access control - Error handling and Exception handling - Error handling mechanisms in Solidity - Exceptions and Revert statements - Error propagation and Handling unexpected scenarios - Handling gas-related issues in Solidity contracts - Ether (ETH) and Gas in the Ethereum network - Smart Contracts and their Applications - Understanding smart contracts and their significance - Solidity vs. other smart contract languages - Writing, deploying, and interacting with smart contracts

CONTRACT DEPLOYMENT AND INTERACTION

24 Hrs

Deploying smart contracts - Deploying contracts on test networks (Ropsten, Rinkeby, etc.) - Gas considerations and estimating deployment costs - Interacting with smart contracts from web applications - Web3.js and Ethereum JavaScript APIs - Reading and writing data to smart contracts - Calling functions and handling contract events - Front-end frameworks and Libraries for Ethereum integration - Testing and debugging smart contracts - Importance of testing smart contracts - Debugging techniques and Tools for smart contracts - Test-Driven Development (TDD) for smart contracts

CONFIGURING HARDHAT OR TRUFFLE

48 Hrs

Setting up a development environment - Installing and Configuring Hardhat or Truffle - Configuring development networks and accounts - Integrating with Ethereum test networks (Ganache, Geth, etc.) - Setting up project structure and dependencies - Compiling, Deploying, and Testing contracts with Hardhat - Compiling Solidity contracts using Hardhat - Deploying contracts to local or test networks - Running tests with Hardhat testing framework - Utilizing Hardhat plugins for additional functionality - Writing automated tests for smart contracts - Importance of automated testing for smart contracts - Testing patterns and best practices for smart contracts - Writing comprehensive test suites with Truffle or Hardhat - Coverage analysis and reporting tools

Web Reference:

1. Solidity (programming language for Ethereum smart contracts):

- **Mastering Ethereum: Building Smart Contracts and DApps**" by Andreas M. Antonopoulos and Gavin Wood.
- **Solidity Programming Essentials: A Beginner's Guide to Build Smart Contracts for Ethereum and Blockchain**" by Ritesh Modi.
- **Building Ethereum DApps: Decentralized Applications on the Ethereum Blockchain**" by Roberto Infante.

2. Ethereum Blockchain:

- **Mastering Ethereum: Building Smart Contracts and DApps**" by Andreas M. Antonopoulos and Gavin Wood.
- **Ethereum: Blockchains, Digital Assets, Smart Contracts, Decentralized Autonomous Organizations**" by Henning Diedrich.
- **Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners**" by Chris Dannen.

COURSE IV
PRACTICAL II - WEB 3 & METAVERSE

Objectives:

- The Students should know how to use Web3 Stack.
- To know how to write a smart contract using solidity.
- To know how to deploy their Cryptocurrency to Test-net.
- To know how to Deploy DAPPS to Test-net.

SETTING UP NODE.JS FOR METAVERSE DEVELOPMENT 30 Hrs

Installing Node.js - NPM (Node Package Manager) - Role of Node.js in Metaverse development - Exploring the event - driven, Non-blocking architecture of Node.js-Creating and running a basic Node.js application for Metaverse projects CommonJS modules - Module - Exports to organize code in Metaverse - Projects Importing - External modules for Metaverse development - Working with the file system - Asset management in the Metaverse

CREATING REST API WITH EXPRESS.JS 15 Hrs

Introduction to Express.js framework - Setting up an Express.js project - Creating RESTful APIs with Express.js - Interaction within the Metaverse - Implementing middleware for authentication, authorization, and session management - Handling real-time communication using Web Sockets and Socket.io for Metaverse experiences

DATABASE DESIGN FOR METAVERSE 35 Hrs

Introduction to databases (SQL and NoSQL) - Storing Metaverse related data - Connecting Node.js to databases for persistent storage - Implementing asynchronous programming with Callbacks - Promises - Querying - manipulating data - Integrating Node.js with Metaverse platforms and technologies - Implementing Metaverse - specific features like multiplayer interactions and virtual economies - Security considerations - Data protection Performance - optimization - Deploying and scaling Node.js - Based Metaverse applications in production environments - Testing, debugging, and monitoring Metaverse applications Exploring emerging trends and future developments in Metaverse technology

CONFIGURE WALLET AND MORALIS INTEGRATION 40 Hrs

Introduction to Open Sea - Overview of NFTs and their significance - Exploring Open Sea marketplace features - Understanding the role of wallets and transactions - Creating and Minting NFTs - Overview of NFT standards (ERC721, ERC1155) - Creating metadata and assets for NFTs - Minting and listing NFTs on Open Sea - Buying and Selling on Open Sea - Browsing and searching for NFTs - Placing bids and making offers - Managing collections and wallets - Smart Contracts and Open Sea Integration - Integrating smart contracts with Open Sea's APIs - Adding royalties and secondary sales to NFTs - Implementing Open Sea storefronts and customizations - Introduction to Moralis Unity SDK - Incorporating block chain functionality in Unity projects - Creating decentralized applications (D Apps)

Web Reference:

1. Metaverse:

- **The Fourth Transformation:** How Augmented Reality & Artificial Intelligence Will Change Everything" by Robert Scoble and Shel Israel.
- **The Virtual Revolution:** How VR, AR, and AI are Changing Everything" by Luciano Floridi.
- **The Spatial Web:** How Web 3.0 Will Connect Humans, Machines, and AI to Transform the World" by Gabriel Rene.

2. Node.js:

- **Node.js in Action"** by Mike Cantelon, Marc Harter, TJ Holowaychuk, and Nathan Rajlich.
- **Learning Node.js:** A Hands-On Guide to Building Web Applications in JavaScript" by Marc Wandschneider.
- **Node.js Web Development:** Server-side Development with Node 10.x for Full Stack JavaScript Development" by David Herron.

3. Web Servers:

- **NGINX Cookbook:** Over 70 Recipes for High-Performance Load Balancing and Content Switching" by Dipankar Sarkar.
- **Apache Cookbook:** Solutions and Examples for Apache Administrators" by Rich Bowen.
- **Professional IIS 10 and ASP.NET Core 2.0"** by Christian Wenz.

4. Full-Stack Web Development (including Node.js and web servers):

- **Full Stack JavaScript:** Learn Backbone.js, Node.js and MongoDB" by Azat Mardan.
- **Web Development with Node and Express:** Leveraging the JavaScript Stack" by Ethan Brown.
- **Pro Express.js: Master Express.js:** The Node.js Framework for Your Web Development" by Azat Mardan.

5. **Moralis:** <https://docs.moralis.io/>

6: **W3School:** https://www.w3schools.com/js/js_htmlDOM_document.asp

7. **OpenSea:** <https://docs.opensea.io/>

COURSE V
PROJECT

List of Exercises:

120 Hrs

- Designing a Virtual University
- Developing a Metaverse Classroom
- Registering for a Webinar in Metaverse
- Digital Certificate