School: .... Campus: ....

Academic Year: .... Subject Name: .... Subject Code: ....

Semester: .... Program: .... Branch: .... Specialization: ....

Date: ....

Applied and Action Learning

(Learning by Doing and Discovery)

Name of the Experiement : **Chains Beyond Ethereum _ Platform Comparisons**

* Coding Phase: Pseudo Code / Flow Chart / Algorithm

Introduction:

Blockchain technology has expanded far beyond Ethereum, giving rise to multiple Layer-1 and Layer-2 platforms with unique consensus mechanisms, scalability solutions, and interoperability features.

Platforms such as **Binance Smart Chain (BSC)**, **Polygon**, **Solana, Avalanche** aim to enhance transaction, throughput, reduce fees, and improve developer experience.

This experiment compares these platforms based on architecture, consensus algorithms, transaction speed, and ecosystem maturity to understand their advantages and trade-offs in decentralized application (dApp) development.

Algorithm / Procedure:

1. **Step 1:** Select popular blockchain platforms beyond Ethereum — e.g., **BSC, Polygon, Solana, Avalanche, Cardano**.
2. **Step 2:** Study each chain's **consensus mechanism** (PoS, DPoS, PoH, etc.) and scalability features.
3. **Step 3:** Compare **transaction speed, gas fees, and developer tools** available on each platform.
4. **Step 4:** Identify **use cases** and **ecosystem strengths** (e.g., DeFi, NFTs, interoperability).
5. **Step 5:** Summarize findings in a **comparison table** or chart to highlight key differences.
6. **Step 6:** Conclude by analyzing which chain offers the most balanced trade-off for scalability, security, and decentralization.

* Softwares used

- **Remix IDE** – for writing, compiling, and deploying smart contracts.
- **MetaMask Wallet / Chain Wallets** – for interacting with dApps across Ethereum, BSC, Polygon, and Avalanche.
- **Alchemy / Infura** – for connecting to blockchain nodes and managing RPC endpoints.
- **Block Explorers (Etherscan, BscScan, Polygonscan, Snowtrace)** – for tracking transactions and verifying contracts.
- **Command-Line Interfaces (Hardhat, Solana CLI, Cardano CLI)** – for testing, debugging, and deploying on respective networks.

*As applicable according to the experiment.
Two sheets per experiment (10-20) to be used.

Implementation Phase: Final Output (no error)

Different blockchain platforms offer unique architectures, consensus mechanisms, and development tools beyond Ethereum.

- Chains like **BSC** and **Polygon** provide Ethereum compatibility with faster transactions and lower gas fees.
- **Avalanche** ensures high throughput, **Solana** uses Proof-of-History for speed, and **Cardano** applies academic PoS research for security.
- Each chain balances scalability, decentralization, and performance in its own way.
- Comparing these platforms helps developers choose the most suitable one for building efficient and secure dApps.

Platform	**Consensus Mechanism**	**Average Transactions Per Second (TPS)**	**Transaction Cost**	**Key Differentiator**
Ethereum (ETH)	Proof-of-Stake (PoS)	~15–30	High (variable)	Large, established ecosystem with high security and institutional trust
Solana (SOL)	Proof-of-Stake (PoS) + Proof-of-History (PoH)	Up to ~65,000 (theoretical)	Very Low (~$0.02)	Extremely fast and scalable — ideal for gaming and high-frequency dApps
Cardano (ADA)	Ouroboros (PoS)	~250 (theoretical)	Low (~$0.10)	Research-based, formally verified, focused on sustainability and security
Polkadot (DOT)	Nominated Proof-of-Stake (NPoS)	~1,000	Moderate	Enables cross-chain interoperability via parachains
BNB Smart Chain (BNB)	Proof-of-Stake (PoS)	~100–300	Low (a few cents)	EVM-compatible with low fees — widely used for DeFi and dApps
				Provides near-instant

*As applicable according to the experiment.
Two sheets per experiment (10-20) to be used.

* Implementation Phase: Final Output (no error)

• **Smart Contract Validation:**
Performed thorough testing of smart contracts to identify logical errors, security flaws, and vulnerabilities such as reentrancy or overflow through both manual review and automated tools.

• **Security & Functional Assurance:**
Verified transaction integrity, data flow, and permission handling to ensure secure, reliable, and error-free contract execution before deployment.

• **Standardized QA Practices:**
Applied Ethereum-compatible QA tools like **Remix, Mythril, and Slither** to maintain uniform testing protocols and transparent validation results.

• **Optimized Testing Framework:**
Created a scalable testing setup that supports iterative testing, modular contract checks, and continuous refinement for improved system reliability.

* Observations

- Understood how to identify and fix common smart contract vulnerabilities using tools like Remix and Slither.
- Gained hands-on experience in testing logic flow, security, and transaction validation before deployment.
- Learned the importance of a standardized QA framework to ensure reliable and transparent smart contract performance.

ASSESMENT

Rubrics	Full Mark	Marks Obtained	Remarks
Concept	10		
Planning and Execution/ Practical Simulation/ Programming	10		
Result and Interpretation	10		
Record of Applied and Action Learning	10		
Viva	10		
Total	**50**		

Signature of the Student:

Name :

Regn. No. :

Signature of the Faculty:

****As applicable according to the experiment.***
Two sheets per experiment (10-20) to be used.