

TUGAS BLOCKCHAIN



Dosen Pengampu:

Ade Kurniawan, S.Pd., M.Pd.T

Disusun Oleh:

Galant Anefsyah Pratama (22346031)

**PRODI INFORMATIKA
FAKULTAS ELEKTRONIKA
UNIVERSITAS NEGERI PADANG
TAHUN AJARAN 2025/2025**

Laporan - Membuat Smart Contract Sederhana

A. Apa itu Smart Contract?

Smart contract adalah program yang berjalan di atas **blockchain** dan dieksekusi secara otomatis ketika kondisi yang telah ditentukan terpenuhi. Smart contract digunakan untuk mengotomatisasi perjanjian atau transaksi tanpa memerlukan perantara pihak ketiga.

Karakteristik Smart Contract

1. Otomatis & Terdesentralisasi

- a. Dieksekusi secara otomatis tanpa perlu perantara seperti bank atau notaris.
- b. Berjalan di jaringan blockchain yang terdesentralisasi.

2. Transparan & Tidak Bisa Diubah

- a. Semua pihak dapat melihat kode dan transaksi smart contract.
- b. Setelah dideploy ke blockchain, kode tidak bisa diubah (immutable).

3. Keamanan Tinggi

- a. Menggunakan **kriptografi** untuk melindungi dari manipulasi.
- b. Tidak bisa dimanipulasi oleh satu pihak setelah disebarkan ke jaringan.

B. Bahasa Pemrograman Smart Contract: Solidity

Solidity adalah bahasa pemrograman tingkat tinggi yang digunakan untuk menulis **smart contract** di jaringan **Ethereum** dan blockchain lainnya yang kompatibel dengan **Ethereum Virtual Machine (EVM)**. Solidity dirancang untuk mengelola transaksi otomatis tanpa perantara.

Karakteristik Solidity

1. Tipe Statis

Variabel harus dideklarasikan dengan tipe data yang spesifik seperti uint, string, address, dll.

2. Kontrak Berorientasi Objek

Menggunakan konsep seperti **class** dalam pemrograman berbasis objek (OOP).

3. Berjalan di Ethereum Virtual Machine (EVM)

Smart contract yang ditulis dengan Solidity dieksekusi di dalam EVM.

4. Mendukung Inheritance (Pewarisan)

Kontrak bisa mewarisi properti dan fungsi dari kontrak lain.

C. Membuat Smart Contract Sederhana

1. Alat Yang Digunakan

- a. Remix IDE (<https://remix.ethereum.org>)
- b. Metamask Wallet (Testnet Sepolia)
- c. Faucet untuk Testnet ETH (<https://sepoliafaucet.com>)

2. Langkah – Langkah

- a. Buka Remix IDE.
- b. Buat file baru: SimpleContract.sol.
- c. Salin kode berikut:

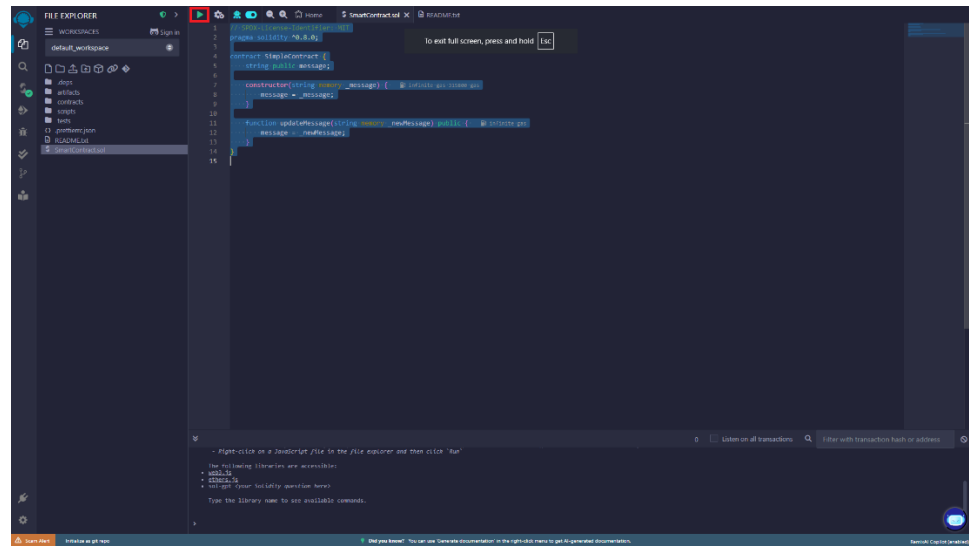
```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;

contract SimpleContract {
    string public message;

    constructor(string memory _message) {
        message = _message;
    }

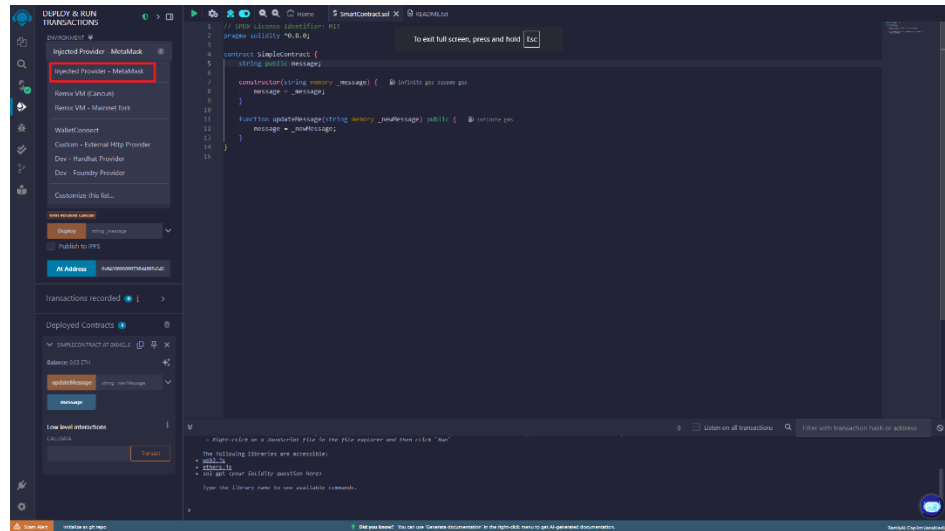
    function updateMessage(string memory _newMessage) public {
        message = _newMessage;
    }
}
```

d. Compile file



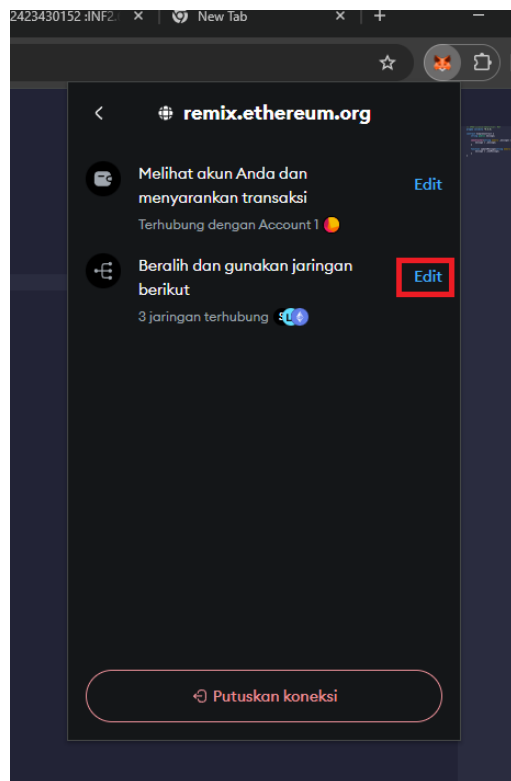
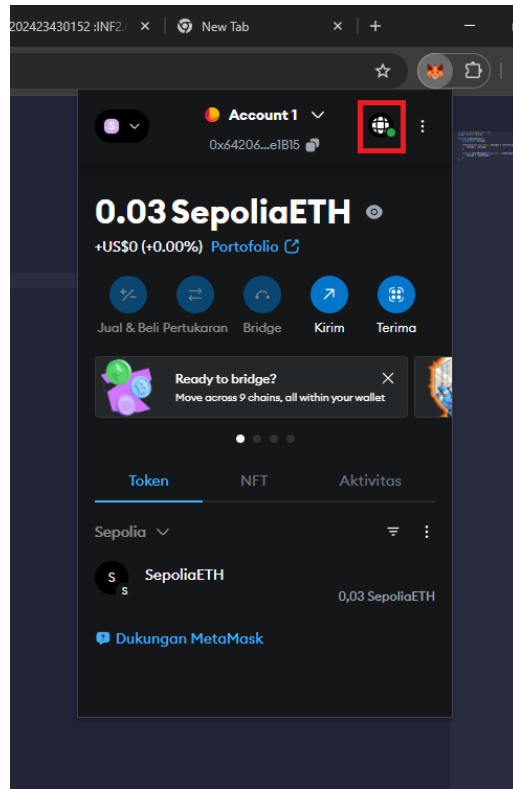
e. Pilih Environment: Injected Provider (MetaMask)

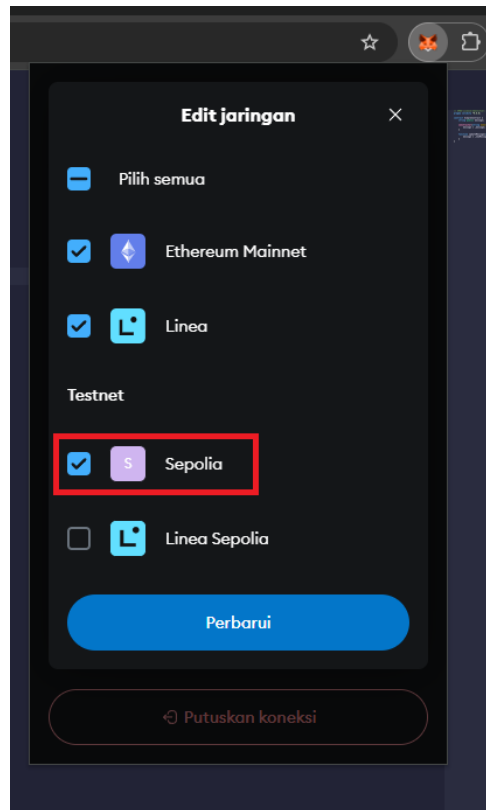
- Klik tab Deploy & Run Transactions
- Pada bagian Environment, pilih Injected Provider - MetaMask.
- Pastikan sudah menginstal dan masuk ke MetaMask di browser.



f. Hubungkan ke Jaringan Sepolia Testnet

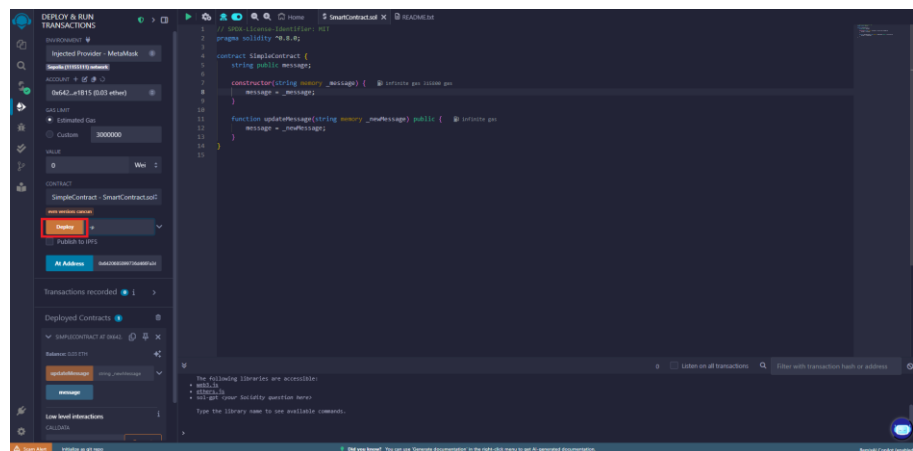
- Buka MetaMask, klik jaringan di bagian atas, lalu pilih Sepolia Testnet.
- Jika belum memiliki jaringan ini, tambahkan secara manual





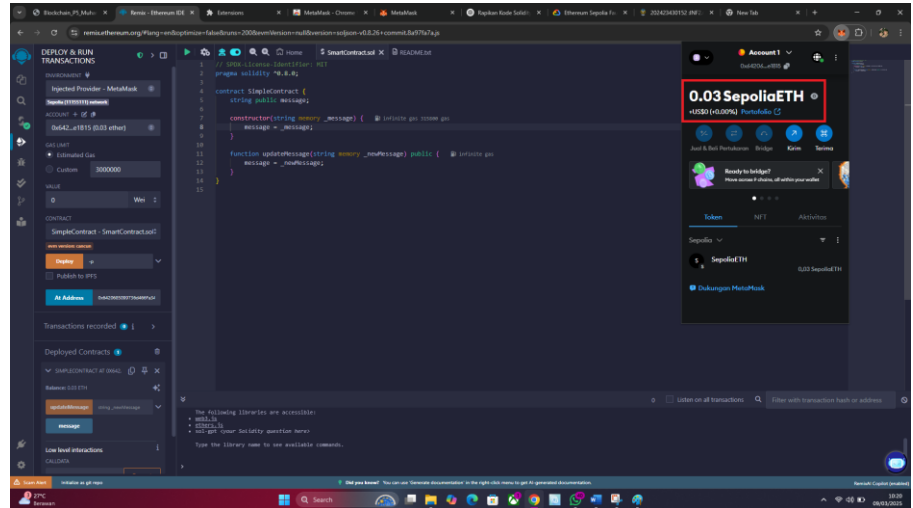
g. Deploy Smart Contract

- Pada Remix, pastikan Deploy menampilkan parameter string untuk constructor.
- Masukkan pesan awal, misalnya "Hello World!". • Klik Deploy dan konfirmasi transaksi di MetaMask.
- Tunggu hingga transaksi berhasil diproses.



h. Transaksi selesai

- Kontrak berhasil dideploy ke Sepolia Testnet.
- Konfirmasi transaksi muncul di MetaMask.
- Kontrak terlihat di Remix dan bisa digunakan.
- Ada perubahan saldo Sepolia ETH karena biaya gas.



D. Kesimpulan

Melalui praktikum ini, mahasiswa dapat memahami konsep dasar **smart contract** pada **blockchain Ethereum**, termasuk cara menulis, mengompilasi, dan menerapkan kontrak menggunakan **Remix IDE**. Selain itu, mahasiswa juga akan mempelajari bagaimana menghubungkan smart contract dengan jaringan **Sepolia Testnet** melalui **MetaMask**, yang merupakan langkah penting dalam pengujian dan simulasi transaksi tanpa menggunakan aset nyata.

Praktikum ini memberikan pengalaman langsung dalam **pengembangan smart contract**, yang merupakan elemen fundamental dalam teknologi blockchain. Mahasiswa akan memahami bagaimana kontrak pintar bekerja secara otomatis tanpa perantara, serta bagaimana konsep **desentralisasi, transparansi, dan keamanan** diterapkan dalam jaringan blockchain.

Selain itu, mahasiswa juga akan mendapatkan wawasan tentang **penggunaan gas fee, deployment smart contract, interaksi dengan fungsi kontrak, serta debugging dalam lingkungan pengembangan blockchain**. Dengan pemahaman ini, mahasiswa

akan memiliki dasar yang kuat untuk mengembangkan aplikasi berbasis blockchain, seperti **Decentralized Finance (DeFi)**, **Non-Fungible Token (NFT)**, serta berbagai solusi berbasis Web3.

Dengan demikian, praktikum ini tidak hanya memberikan pemahaman teoretis tetapi juga keterampilan teknis yang dapat diaplikasikan dalam industri blockchain yang terus berkembang.

E. Referensi

1. Ethereum.org - Smart Contract Documentation
2. Solidity Documentation - <https://docs.soliditylang.org>
3. Remix IDE - <https://remix.ethereum.org>
4. Metamask - <https://metamask.io>