

# 区块链基础应用技术方案

## 一、技术架构

1. 编程语言：采用 Rust 语言，利用其内存安全、高性能和并发性等特性，确保区块链系统的稳定性和效率。
2. 模块划分
  1. 核心模块 (**core**)：包含 `block.rs` 和 `blockchain.rs` 文件，负责定义区块和区块链的核心结构与逻辑，如区块的创建、哈希计算，区块链的添加区块等操作。
  2. 工具模块 (**utils**)： `coder.rs` 文件提供序列化、反序列化和哈希计算等工具函数，方便其他模块使用。
  3. 主程序模块 (**main**)：作为程序的入口，创建区块链实例并进行添加区块等操作。

## 二、关键技术实现

1. 区块结构
  1. **BlockHeader**: 包含时间戳(`time`)、交易哈希(`tx_hash`)和前一区块哈希 (`pre_hash`)，用于标识区块的基本信息和与其他区块的链接关系。

2. **Block:** 包含区块头 (`header`)、区块哈希 (`hash`) 和区块数据 (`data`)，通过 `set_hash` 方法计算区块哈希。

## 2. 区块链结构

1. **Blockchain:** 使用 `Vec<Block>` 存储一系列的区块，提供 `add_block` 方法添加新的区块到区块链中，`new_blockchain` 方法创建初始的区块链（包含创世区块）。

## 3. 数据处理

1. **序列化与反序列化:** 利用 `bincode` 库实现数据的序列化和反序列化，方便在不同模块间传输和存储数据。
2. **哈希计算:** 采用 `crypto` 库中的 `Sha3` 算法进行哈希计算，确保数据的完整性和唯一性。

# 三、开发环境

1. **操作系统:** 支持 Windows、Mac 和 Linux 等主流操作系统。
2. **开发工具:** 推荐使用 Visual Studio Code 等支持 Rust 语言开发的 IDE，配合 Cargo 包管理工具进行项目构建和依赖管理。