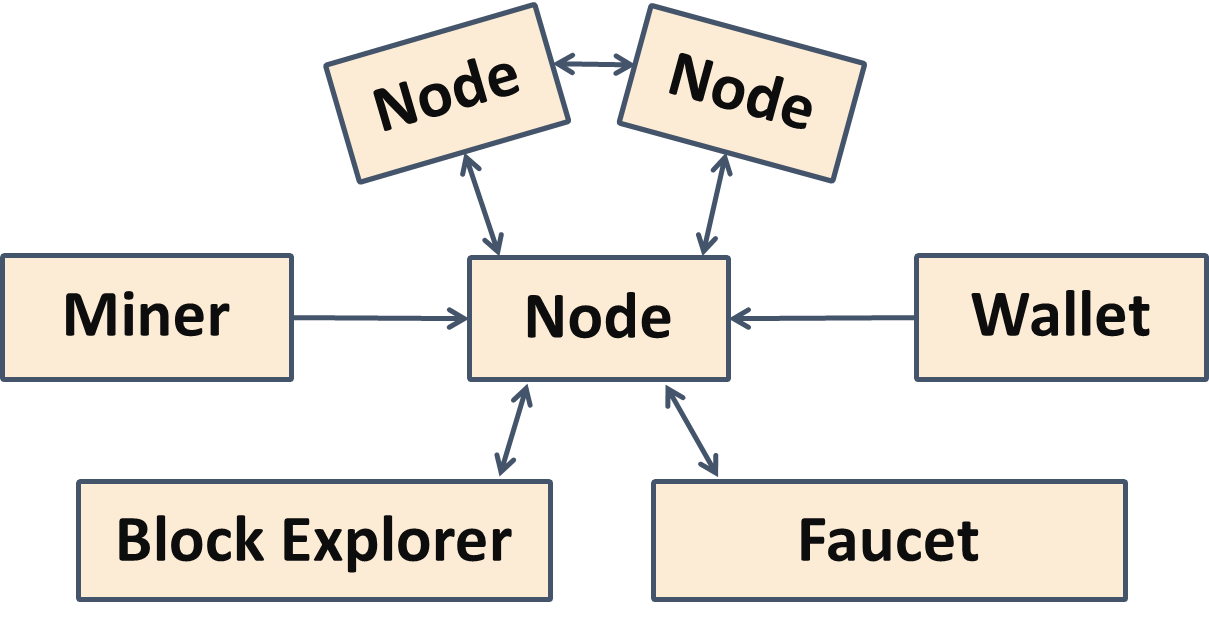
# Teamwork Project: Build Your Own Blockchain Network

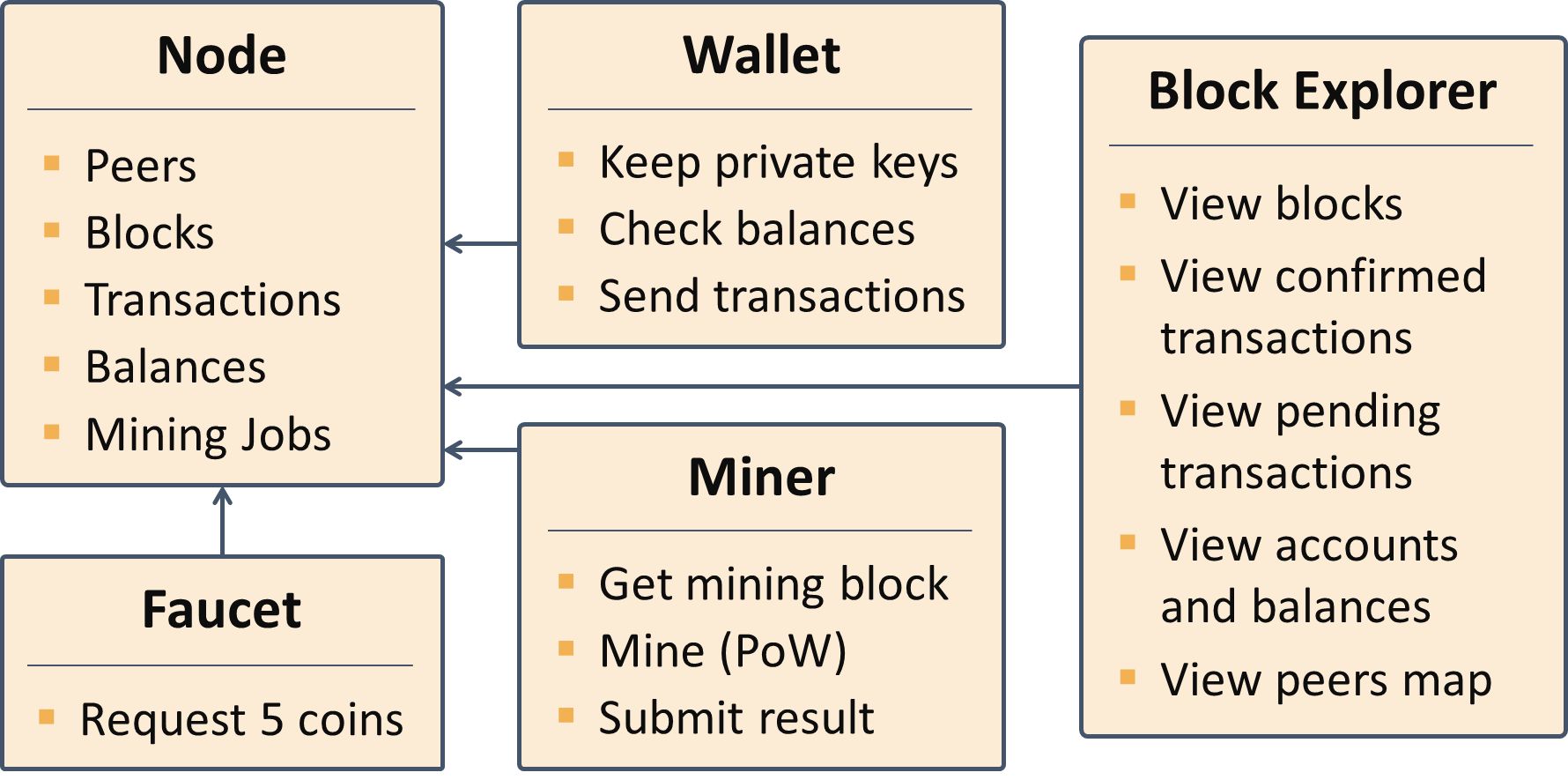
Implement a **fully-functional blockchain network** consisting of **Nodes** (peer-to-peer communication, blocks, transactions, consensus, REST API), **Mining** software (proof-of-work), **Wallet** software (handle private keys + sign and send transactions), **Faucet** app (to request free coins) and **Block Explorer** app (to browse the blocks and transactions).

## Project Description

Implement a simple **fully-functional blockchain network** consisting of **nodes**, **miners**, **wallet** app, **faucet** app and **blockchain explorer** app. The architecture of the network is defined as follows:



The functionalities of each component and the interactions between the components are defined as follows:



**Detailed project functionality** with technical details are provided in **separate document** (the presentation slides).

## Technologies

You are free to use **technologies and platforms of your choice**, e.g.

* JavaScript + Node.js + Express + JS crypto-libraries
* C# + ASP.NET MVC + BouncyCastle.NET or other crypto libraries
* Java + Spring MVC + BouncyCastle or other crypto libraries
* Python + Flash or other REST framework + pycoin or other crypto libraries

You may mix the technologies, e.g. implement the Blockchain Node in JS, the Mining software in C# and the Blockchain Explorer in Java.

## Team by Choice

You are free to work in a **team by choice**.

* Your team should consist of **1** to **4** **people**.
* You are **free to work alone** if you don’t have a team.

Everyone in your team should use a **Git** and **GitHub**.

* Use **Git** as source control system for your project development.
* Use **GitHub** as your source-code repository and project management tool.
* Your **source code** should be published as **open-source project** in Internet.

## Project Scope Depends on the Team Size

The bigger your team is, the bigger should be the scope of the project. Generally, **1-2 components per team member** works well, but a team member can develop or contribute to several components as well. For example:

* A **single team member** may implement a **single component**, e.g. the Wallet of the Miner.
* It is essential to have a **Node** implemented in your team or taken from another team.

## Forbidden Techniques and Tools

* Your project should be created mainly by **you and your team**.
  + You are **not allowed to copy a project from Internet** and present it as your development.
  + If you **fork code** from some other team or from Internet, please **declare this explicitly**.
  + Your contribution should be **significant**, not just clone + adjust or copy / paste.
* You can use external libraries, frameworks, platforms and tools, but **not to clone an entire project** and present it as yours.

## Commit Logs

* Each team member should have **at least 5 commits** (changesets) in the project repository.
* Please **commit every day** during the project development to demonstrate your work progress.
* More commits (especially in more than the last 1-2 days) are valued higher during the project assessment.

## Deliverables

Submit the **URL of your project source code** as deliverable, e.g.

* <https://github.com/SoftUni/SoftUniChain>

Each team member submits the same **source code URL**. Put the following assets in the project repository:

* The complete **source code** of your project (JS / C# / Java / Python code, HTML, CSS, images, scripts, build files, config settings and other files).
* Any other project assets (optionally): documentation, design documents, tests, etc.

## Public Project Defense

Each team will have to deliver a **public defense** of its work in front of the course trainers and other teams.

The teams will have **15-20 minutes** for the following:

* **Demonstrate** the application’s functionality (very shortly).
* Show the **source code** and explain briefly how it works.
* Briefly explain the **contributions of each team member**.

At least **one team member** should come at the defense.

Hints for better presentation:

* Be **well prepared** for presenting maximum of your work for minimum time.
* **Open and start all project assets** beforehand to **save time**, e.g. run your Blockchain Node server, prepare the REST requests you want to demonstrate, start all your other project components, open your project GitHub page to show the commit logs, etc.
* Test how to connect your laptop with the **multimedia projector** before the defense to save time.

## Assessment Criteria

Each team member will get separate score during the evaluation, depending on the project quality, presentation and individual contribution. The **team score** (0…100) will be sum of the following components:

* **Node**: 0…**30** score
* **Wallet**: 0…**15** score
* **Faucet**: 0…**15** score
* **Miner**: 0…**15** score
* **Block Explorer**: 0…**20** score
* **Bonus**: 0…**5** score

Each team member will get **individual score**:

* **Excellent** work: **20-30** score
* **Good** work: **10-20** score
* **Some** contribution: **5-10** score