

# 范文辉 全栈工程师

✉ hu.work.pham@gmail.com  
📍 越南  
🔗 <https://huypham.vercel.app/>

## 轮廓

我是一名全栈开发工程师，致力于打造创新且高效的技术解决方案。凭借多年软件开发经验，我不断提升技术能力，致力于提供卓越的开发成果。我的技术栈涵盖多种编程语言、框架和工具，特别擅长前端、后端以及区块链开发。在设计可扩展架构、构建高效应用以及优化性能方面，我有着丰富的实践经验，能够有效解决复杂的技术挑战，交付高质量的软件产品。作为一名资深的全栈开发工程师，我具备较强的领导力，能够有效指导和培养初级开发者，推动团队合作和技术共享。我注重沟通，能够与利益相关者密切合作，确保项目目标的顺利达成，并在过程中不断推动技术创新和团队持续进步。

## 语言

中文	<div style="width: 100%;"></div>
英语	<div style="width: 100%;"></div>
德语	<div style="width: 100%;"></div>

## 教育

计算机科学学士学位  
*Lingnan University*  
09/2013 – 03/2017

## 兴趣

阅读、足球、绘画

## 专业经验

### 全栈工程师

*ecobased.ai*

03/2024 – 10/2025

- 使用 Next.js、Tailwind CSS 和 Daisy UI 设计和开发整个前端页面。
- 使用 Chainlink 的 POR 功能将气候数据和现实世界资产信息连接到区块链。
- 使用 web3.js 集成启动板智能合约和前端页面。
- 利用 corss-mint api 开发nft mint后端。

### 首席全栈开发人员

*muchbetter.ai*

01/2022 – 02/2024

- 开发身份验证、角色管理、销售情况跟踪、反馈及评估结果等功能的 API。
- 使用 ChatGPT API 根据所选环境、角色、销售情况和上下文生成客户端响应，提高系统的交互性与智能。
- 集成 Azure 存储 和 语音认知服务 SDK，实现音频合成，用于语音反馈 及其他相关功能。
- 利用 OpenAI 和 Azure 提供的 ChatGPT，根据上下文、环境和评估标准 进行角色扮演的评估。
- 使用 Docker 容器化应用，确保部署环境一致性。
- 通过 GitHub Actions 和 Azure CI/CD 流程，提升开发效率，实现更快速 的部署、更高效的管理。

### 区块链全栈工程师

*gempad.app*

03/2020 – 12/2022

- 使用 Next.js 提供快速的渲染和无缝用户体验，结合 Material UI 和 Tailwind CSS 让界面更加美观和响应式。
- Solidity 智能合约的应用使得销售过程自动化且透明，支持多种销售策略（如线性、按比例等）。
- 通过 Web3.js 实现前端与智能合约的直接交互，使用户能够方便地进行购买和查询。
- 在 Discord 上部署赠品机器人，促进社区互动，增加用户参与度。

### 区块链全栈工程师

*sewerratsocial.club*

05/2017 – 12/2019

- ERC20 和 ERC721 智能合约为生态系统提供了交易的基础，确保了代币 和 NFT 的安全流通。
- 质押池 功能鼓励用户持有并质押 Rats NFT，通过赚取 CHIZ 代币来获得 被动奖励。
- 使用 React.js 构建的仪表盘使用户可以方便地管理质押操作，查看实 时 奖励。
- Discord 机器人 提供实时买卖通知，增加了社区成员的互动和参与 感。

## 技能

---

### 前端开发

React, Next.js, Vue.js, TypeScript, Angular.js, JavaScript, Material UI

### 后端开发

Node.js, Python, Go, Rust

### 区块链

ERC-20/ERC-721 tokens, DeFi, Ethers.js, Web3.js, smart contract

### 数据库

MongoDB, PostgreSQL, Transactions, Indexing, Replication, Query Optimization

### 智能合约

Rust (Solana), Anchor, Solidity (EVM), Hardhat, Foundry

### 密码系统

Payment Gateways, Internal Transaction Systems, Wallets, Ledgers

### 系统设计

Architecture Design, Load Calculation, Scalability Planning

### 消息传递和流媒体

Kafka, RabbitMQ, ZeroMQ, Event-Driven Systems

### 微服务

Service Decomposition, Authentication, Inter-Service Communication

### 交易所和去中心化金融

DEX & CEX API Integration

### 绩效与规模

Horizontal Scaling, Concurrency Management, Fault Tolerance