AI 游戏开发定向简历

目标岗位：AI 游戏研发 / AI 应用开发实习生

计算机科学与技术本科 · 预计 2027 年毕业

邮箱：f1405670334@gmail.com | 可接受出差，工作地点可协商

核心方向： 游戏图形识别 · 本地 AI 引擎接入 · 游戏工具链 · AI 辅助开发

教育经历

计算机科学与技术 · 本科 预计 2027 年毕业

游戏与 AI 项目

个人项目 · 游戏棋盘视觉识别与本地 AI 分析工具 持续迭代

XiangqiOverlay ｜ Windows 游戏窗口视觉识别 + NNUE 引擎分析 GitHub

- **问题与目标：**针对 Windows 象棋游戏窗口，自动读取棋盘状态并展示本地 AI 分析结果，减少手工录入局面的成本；工具只做读取、分析和可视化，不执行自动落子。
- **视觉识别：**使用 OpenCvSharp 完成截图预处理、Canny 边缘、Hough 直线分组、HSV 墨迹校验和网格精修，自动定位 9×10 棋盘的 90 个交叉点；结合棋子图像特征、游戏高亮和象棋规则处理识别歧义。
- **AI 接入：**将识别到的棋盘状态转换为 FEN，通过 UCI 协议接入 Pikafish NNUE 本地引擎，解析搜索深度、评分、胜率和主变着法，并在透明悬浮层中展示结果。
- **工程实现：**按 Core / Vision / Engine / Platform / App 拆分 .NET 8 解决方案，完成窗口枚举、屏幕截图、坐标标定、状态更新、引擎异常重启和 FEN 合法性校验。
- **技术关键词：**C#、.NET 8、WPF、WinForms、OpenCvSharp、OpenCV、图形识别、Pikafish、NNUE、UCI、FEN。

个人项目 · AI 全栈工程（本地大模型应用） 持续迭代

AI Panel Studio ｜ 本地运行的 AI 圆桌讨论 Web 应用 GitHub 在线演示：https://ai.yuzhouwudibaolongzhanshi.com

- **问题与目标：**输入话题即生成主持人与专家阵容，确认后在演播厅实时观看 AI 观点交锋、共识与分歧更新；未配置模型 Key 时自动切换内置 Mock 演示，全程本地运行。
- **全栈实现：**npm workspaces 单仓（shared / server / web）；React 19 + Vite + Tailwind CSS v4 实现中文响应式演播厅；Node.js + Express 5 + node:sqlite 后端提供 REST 与 SSE，zod 契约前后端共享。
- **AI 工程化：**LLM 网关按任务路由 DeepSeek 质量档 / 速度档或 Mock，只提取 content、丢弃隐藏推理；出价调度按意图、紧急度、回应关系与公平性决策，不按座位轮转；API Key 只由后端读取。
- **实时与隔离：**SSE 事件带序号，断线凭 Last-Event-ID 补发；多讨论按频道隔离，Runner 状态全部为实例字段，持久事件先落库再广播。
- **质量保障：**按 SDD→DDD→TDD→E2E 规格先行开发，提交按「规格 → 组件 → 测试 → 实现」分层；299 个单元 / 集成测试 + 10 个 Playwright E2E 场景（隔离、重连、泄漏检测）全部通过。
- **技术关键词：**React 19、Vite、Tailwind CSS v4、Node.js、TypeScript、Express 5、node:sqlite、SSE、DeepSeek、Playwright。

实训项目 · LLM 智能体与机器人协同开发系统 持续迭代

LLM 智能体与机器人协同开发系统 ｜ RAG 问答助手 + 机器人智能体 + ESP32-S3 硬件 GitHub

- **问题与目标：**把大模型智能体落到真实硬件：一套 RAG 问答助手 Web 系统，加一条能驱动机器人动作与语音对话的智能体链路。
- **RAG 问答助手：**FastAPI + LangChain + SQLAlchemy + SQLite 后端，42 个 HTTP 接口 + 2 个 WebSocket；Vue3 + Element Plus 前端 12 个页面；分块 → 嵌入 → 检索 → 拼 Prompt，一次检索结果两用（喂模型 + 作出处），无 Key 时嵌入降级为本地 n-gram 稀疏向量。
- **机器人智能体：**OpenAI 兼容 /v1/chat/completions + SSE 流式；经 MCP（JSON-RPC over WebSocket）调用设备侧 12 个工具，控制 Otto 六舵机机器人 22 种预置动作；配套本地控制台与 MCP 客户端工具链，真机联调通过。
- **实时语音链路：**ESP32-S3 主控打通 唤醒 → ASR → LLM → SSE → TTS → 喇叭 全链路；6 轮语音实测全部出声，纯文本轮约 3.0 s，带动作轮端到端 4 ~ 11 s。
- **技术关键词：**Python、FastAPI、LangChain、SQLAlchemy、SQLite、Vue3、Element Plus、SSE、MCP、WebSocket、ESP32-S3。

AI 开发方式

- **AI 辅助工程：**使用国内外 AI 工具辅助检索 API、拆分模块、生成代码草稿和定位异常；对生成内容进行人工审阅，并用构建、测试、日志、运行截图和哈希清单复核结果。

技能与岗位匹配

游戏 AI：图形识别、棋盘状态建模、规则约束、本地 NNUE 引擎接入、UCI/FEN 协议。

工程工具：Git、.NET CLI、Android SDK、apktool、zipalign、apksigner、PowerShell。

桌面与图形：C#/.NET 8、WPF、WinForms、OpenCvSharp / OpenCV、截图与坐标标定。

代码与验证：模块化设计、异常处理、协议解析、构建脚本、契约测试、日志定位、SHA-256 产物校验。

求职补充

- 无正式实习经历，愿意接受游戏研发场景下的 AI 应用、图形识别、工具开发和客户端技术培训。
- 可准备项目源码、构建脚本、测试报告和运行截图。
- 喜欢游戏，持续关注玩法、交互和技术实现；学习能力较强，能够通过项目实践快速补齐新工具和新框架。
- 重视团队协作与沟通，习惯通过模块边界、日志、文档和可复现测试把问题讲清楚、跟进到底。