**迭代计划（简化版）**

　　　　　　　　　　　　　制定日期：

|  |  |  |  |
| --- | --- | --- | --- |
| 组号 | 17 | 项目名称 | Robotrain：具身智能云端训练平台 |
| 迭代名称 | 技术原型迭代 | 计划起止日期 | 2025/7/1-7/10 |
| 任务、进度安排和人员分配：   |  |  |  |  | | --- | --- | --- | --- | | No | 任务 | 起止日期 | 人员 | | 1 | 设计架构，选定google python规范 实现测试任务调度算法 | 7/1-7/2 | 师文翔 | | 2 | 搭建基本训练平台框架 | 7/3 | 师文翔 | | 3 | 进行通用组件的单元测试与修复 | 7/4 | 师文翔 | | 4 | 使用真实训练脚本进行连接 | 7/5 | 师文翔 | | 5 | 尝试基于Ray搭建分布式训练框架并部署 | 7/6-7/7 | 师文翔 | | 6 | 使用RabbitMQ与后端通讯  使用MinIO util与对象存储连接 | 7/8 | 师文翔 | | 7 | 测试并准备技术评审 | 7/9-7/10 | 师文翔 | | 8 | 完善fake\_trainer和训练端的通信 | 7/1-7/2 | 潘奕宇 | | 9 | 测试Websocket | 7/3-7/4 | 潘奕宇 | | 10 | 具体对接并测试和前端通信部分 | 7/5-7/6 | 潘奕宇 | | 11 | 具体对接并测试和训练端通信部分 | 7/7-7/9 | 潘奕宇 | | 12 | 完成上一次迭代遗留的问题，如部分页面未完全开发，页面功能未细化等 | 7/1 | 徐丁杰 | | 13 | 完成前端与后端的API对接 | 7/2-7/3 | 徐丁杰 | | 14 | 测试前后端是否能够正常通讯，并根据测试结果对前端页面修改 | 7/4-7/5 | 徐丁杰 | | 15 | 添加模型测试相关前端页面 | 7/6-7/7 | 徐丁杰 | | 16 | 尝试使用ECharts，让训练结果能够以动态图表的形式在前端展示 | 7/8-7/9 | 徐丁杰 | | 17 | 使用官方数据集进行模型训练与评估，得到训练结果以及评估结果 | 7/1-7/2 | 石辰阳 | | 18 | 修改训练和评估脚本，完成和后端通信部分 | 7/3-7/4 | 石辰阳 | | 19 | 尝试为现有训练框架增加新的支持的模型 | 7/5-7/6 | 石辰阳 | | 20 | 进一步优化训练框架的项目结构，测试不同模型训练效果 | 7/7-7/8 | 石辰阳 | | 21 | 测试并完善最终展示的仿真环境 | 7/9-7/10 | 石辰阳 | | 22 | 完整项目测试，完成内部评审，撰写项目文档 | 7/10 | 师文翔、潘奕宇、徐丁杰、石辰阳 | | | | |
| 预期成果：  [列出本迭代计划交付的文档、模型、源代码、安装包等]  后端项目源码及docker部署  训练框架项目源码及服务器端部署  完善的、能与后端正常通信的前端页面  在仿真环境中展示训练好的模型运行结果的视频 | | | |
| 主要的风险和应对方案：  [分析当前项目风险，列出最大的3~5个风险，自高到低排列，并列出每个风险缓解和应急的措施。注：随着项目的进行，风险会不断变化，故每个迭代要重新对风险评估和控制。]  **风险**：Ray进行分布式训练可能存在技术困难；**应急措施**：暂时针对单机多卡场景支持分布式数据并行策略，而不考虑多机场景。  **风险**：服务器端部署仿真环境存在渲染相关问题；**应急措施**：在本机端侧进行训练好的模型在仿真环境中的部署和推理  **风险**：Websocket传递log的可行性；**应急措施**：改用http轮询  **风险**：训练调度的实现可能存在技术困难；**应急措施**：改用更加简单的调度逻辑 | | | |