**迭代计划**

　　　　　　　　　　　　　制定日期：2022年11月10日

|  |  |  |  |
| --- | --- | --- | --- |
| 组号 | 第11组 | 项目名称 | 数据流动态分析平台 |
| 迭代名称 | 迭代3构建阶段 | 计划起止日期 | 2022.11.12 - 2022.11.26 |
| 任务、进度安排和人员分配：   |  |  |  |  | | --- | --- | --- | --- | | No | 任务 | 起止日期 | 人员 | | 1 | 前端数据源注册界面（近期） | 11.22 - 11.24 | Naomi | | 2 | 前端数据源展示界面（近期） | 11.24 - 11.26 | Naomi | | 3 | 前端界面原型与路由 | 11.12 - 11.26 | Naomi | | 4 | 数据流算子“map”的抽象化（近期） | 11.22 - 11.24 | 李梦瑶、佘智龙 | | 5 | 后端DataDO六边形架构（近期） | 11.24 - 11.26 | 胡昊源、李梦瑶 | | 6 | 流数据模拟 | 11.12 - 11.18 | 胡昊源、李梦瑶 | | 7 | 抽象数据流 | 11.19 - 11.26 | 胡昊源、李梦瑶、佘智龙 | | 8 | 细化平台应用管理功能，细分成界面、功能点； | 11.12 - 11.15 | Naomi、胡昊源、李梦瑶、佘智龙 | | 9 | 依照设计进行编码，提供平台应用管理功能； | 11.15 - 11.22 | Naomi、胡昊源、李梦瑶、佘智龙 | | 10 | 测试平台应用管理能力，并进行系统测试； | 11.22 - 11.26 | Naomi、胡昊源、李梦瑶、佘智龙 | |  |  |  |  | |  |  |  |  | | | | |
| 预期成果：  数据流动态分析平台 界面原型  数据流动态分析平台 前端代码  数据流动态分析平台 后端微服务代码  数据流动态分析平台 系统测试用例  数据流动态分析平台 迭代3迭代计划  数据流动态分析平台 迭代3迭代评估报告 | | | |
| 主要的风险和应对方案：  **1、进度风险**  由于技术人员们缺少项目经验，对相关技术框架不熟悉，设计实现工作量较大导致开发进度缓慢。  **缓解和应急措施：**   1. 首先完成困难的核心需求，其余功能点可以放在后面的迭代中完成； 2. 首先考虑具体一个流程是否能够完全跑通，保证有基础的功能与效果； 3. 最后，如果进度风险仍然存在，适当降低数据流抽象程度，降低技术困难；   **2、技术风险**  对于数据流算子的抽象是当前最主要的技术难题。为了满足算子的泛用性，我们通过解析csv文件中，输入用户定义数据类型，动态生成类。目前尽管能够实现一定程度的抽象，但其程度仍有待加深。  **缓解和应急措施：**   1. 向老师和学长学姐等专家请教，是否有更加合适的抽象方法；   2）最后，如果技术风险仍然存在，适当降低数据流抽象程度，降低技术困难；   1. 架构风险   由于代码采用微服务架构，使用DDD六边形架构较为规范。而早先同学们使用了分层架构，虽然可以跑通，但技术风格不统一，加大了沟通难度，造成了一定程度上的反工。  **缓解和应急措施：**   1. 在保证迭代进度的前提下，进行架构的更改，使用DDD六边形架构； 2. 最后，如果架构风险已然存在，则需要大家完善开发文档，详细说明各个结构，减少交流成本； 3. 质量风险   实际开发代码质量较低，造成一部分反工  **缓解和应急措施：**  1）负责人做好 code review 工作，把控代码质量；   1. 需求风险   在开发过程中，实验室提出了新的需求，但整体仍在可控范围之内，故风险不是很大。  **缓解和应急措施：**   1. 做好项目模块化，敏捷应对需求变更； 2. 做好代码解耦，完善架构，减少返工； | | | |