**MARKII实验项目需求与设计文档**

1. **项目环境**

1.1 **项目总体目标**

本软件项目为虚拟仿真实验MARKII算法实验，旨在通过虚拟仿真实验过程帮助学生完成实验：当前学校开展软件工程工作量估计实验时，实验数据仍需要从虚拟项目图中获取，撰写实验报告的方式仍采用手写，不仅教学效果偏低，实验报告完成效率也偏低，我们试图提供一套完整的MARKII方法软件功能点估算仿真实验过程，能够覆盖从输入项目参数到生成实验报告整个流程，帮助完成软件规模度量。

1.2 **项目原理**

MK II的实验原理如下：

**第一步：确定计数的视角和目的**

确定功能点计数的视角方便了后续计数边界的确定。下面是三种常用的视角————  
 1.项目视角：为了确定以软件开发项目形式所要交付功能的规模，计数的目的是要利用功能 规模测量结果来评估项目开发生产率， 或者估算开发项目工作量。在此情况下，边界通过项目用户的功能需求范围来确定；  
 2.业务视角： 为了确定支持特定商业功能领域的应用程序的全部软件功能规模， 这些商业功能由一个业务主管所领导的一一个组织(业务团队)来管理，计数的目的是想要跟踪每年变化的业务领域功能需求， 或者是评估业务团队对整个领域的支撑工作生产率。在此情况下，边界通过业务团队支撑的功能范围来确定；  
 3.企业视角：为了对企业进行资产评估而确定企业应用程序功能组合的规模。在此情况下， 边界 通过企业的整个应用程序功能组合来确定。

**第二步：确定计数的边界和接口**

绘制FP计数申请的边界取决于计数的意图观点，目的和用途。绘制边界将确定将在计数中包含哪些功能， 以及排除哪些功能。当绘制应用程序边界时，它定义了软件和“用户”之间的概念边界。  
 来自用户的“输入数据”越过边界进入应用程序。 “输出数据”离开了应用程序，并越过边界 到达了用户那里。 在边界内是数据实体类型（为简单起见，以下简称“实体”） ，它们由应用程序的逻辑事务类型（以下简称“逻辑事务”或简称“ 事务”） 处理。  
 当为特定的FP计数而绘制应用程序边界时，它可能会导致识别到其他 应用程序的 “接口”。在一般的理解中，“接口”这个词用于两个物 理上不同的进程。两种类型的接口：  
 1.“数据”文件（如主数据记录、“事务”、一串消息等）可以作为接口文件从一个 应用程序与另一个应用程序共享。  
 2.“数据”通过一个接口，一次一个或一批，从一个应用程序传输到另一个应用程序

**第三步：识别逻辑事务，对数据实体类型进行识别和分类，计算输入数据元素类型、所引用的数据实体类型和输出数据元素类型。**

逻辑事务是软件应用程序支持的最低级别的业务流程。每个逻辑事务包括三个元素：跨应用程序边界的输入、 一些相关的处理和跨应用程序边界的输出  
 在识别逻辑事务集时，考虑以下问题：  
 “谁在使用它呢？”  
 “为什么还需要它呢？”  
 “它是做什么的？”  
 “它是什么时候启动的？”  
 “它与其他逻辑事务有何不同？”  
 实体类型是业务用户在现实世界中想要保存信息的某种东西（严格地说， 是某种类型的东西）。数据元素类型保存有关实体类型的信息。例如：·“员工”将是一个人事系统中的一个实体类型。识别并 对数据实体类型进行分类有助于我们对每个逻辑事务的分析以及其三元素的识别分析。  
 每个逻辑事务都由输入、流程和输出三个元素组成。 MkII FPA对这三个元素 的功能大小做出了以下基本假设： 输入元素的大小与构成事务输入侧的唯一处理数据元素类型（“DET’s”）的数量成正比； 处理元素的大小与逻辑事务过程中引用的数据实体类型（或“实体”）的数量成正比； 输出元素的大小与组成事务输出端的唯一处理DET的数量成正比。 MkII FPA的一个约定是，每个逻辑事务必须至少有一个输入数据元素类型，必须对一个实体类型进行一个引用， 并且必须有一个输出数据元素类型。 提供这三种不同类型的信息处理所涉及的功能都是不同的。因此，在MkII FPA中，使用了三个加权因子， 使这三种功能能够组合为一个功能大小的单个值。（权重已根据分析、设计、编程、测试、实施这些组件的行业平均相对工作量， 以便MkII FP功能规模量表为这些活动提供工作产出的行业平均相对度量。） 对于每个逻辑事务，计算输入数据元素类型的数量（Ni）、引用的数据实体类型的数量（Ne）以及输出数据元素类型的数量（No）。

**第四步：计算功能规模**

功能大小（FPI,Function Point Index）是所有逻辑事务、输入数据元素类型（Ni）、引用的数据实体类型（Ne） 和输出数据元素类型（No）的事务的加权和。因此，一个应用程序的FPI是：FPI = Wi \* ∑Ni + We \* ∑Ne + Wo \* ∑No。 其中“∑”表示所有逻辑事务的总和，每个输入数据元素类型、数据实体类型引用和输出数据元素类型的行业平均权重分别为： Wi = 0.58, We = 1.66, Wo = 0.26。

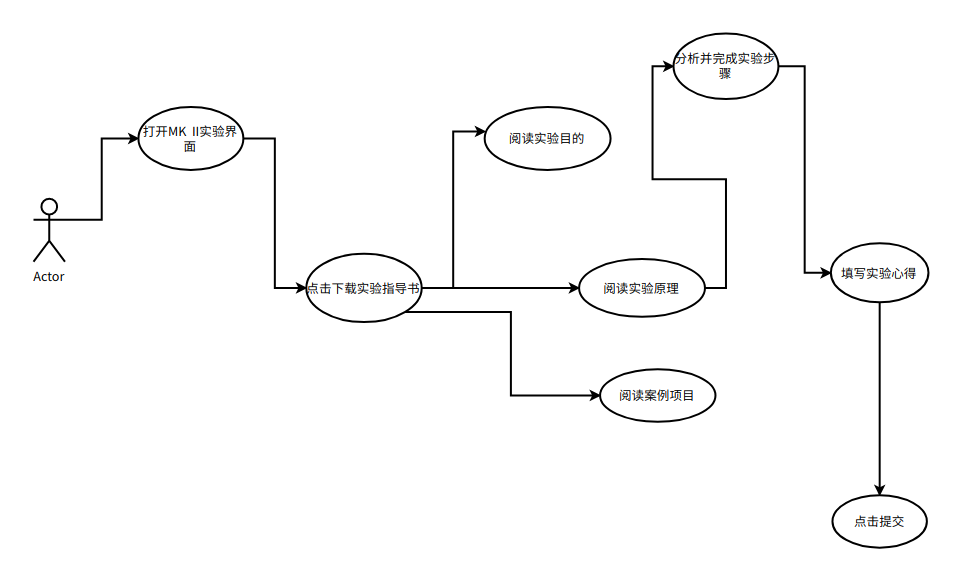
**第五步：评估每个技术复杂性调整特性的影响程度，并计算调整后的规模**

除了信息处理规模之外，FPA还提供了一种考虑到应用程序 的技术复杂性和某些质量要求的方法。 这些需求有时被称为“非功能性需求”。当FPI乘以技术复杂性调整时，结果称为调整函数点指数。 该方法试图在0到5的范围内测量19个（或更多）技术特征对应用规模的影响。 所有特征的和被用来计算要应用的因子。这个因素被称为技术复杂性调整（TCA）。 如果这19个（或更多）特征中的每个特征的得分的总数被称为总影响度（TDI） 即要求从数据流，分布式功能，性能等20个技术中挑选19个，并为其复杂度调整特性的影响程度打分， 0~5表示影响程度由小到大。 TCA =（TDI \* C）+0.65，其中当前行业平均值C为0.005。因此，TCA可以在0.65到1.15之间变化（ 如果所有的特征分别有最小或最大的影响）。调整后的功能点计算表示为： AFPI = FPI \* TCA， 其中AFPI =调整后的功能点指数，FPI =功能点指数，TCA =技术复杂性调整。

2. **项目需求**

2.1 **需求描述**

总体的用例图如下：



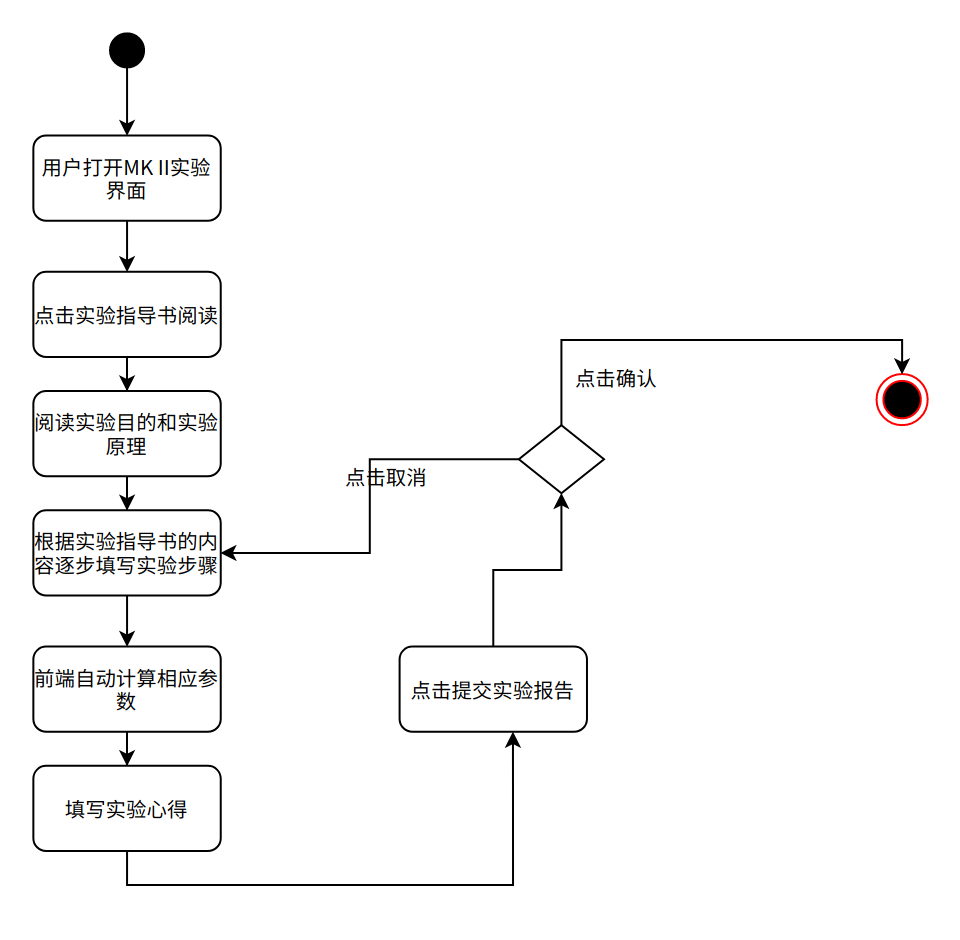
2.2 **软件需求**

2.2.1 **功能性需求**

功能型需求主要体现在用户交互操作方面，见下表：

|  |  |  |
| --- | --- | --- |
| 需求ID | 需求名称 | 详细描述 |
| 1 | 实验目的展示 | 在前端页面展现实验目的 |
| 2 | 实验原理展示 | 在前端页面展现实验原理 |
| 3 | 实验步骤一实现 | 用户计算输入数据元素类型Input Types个数，引用的数据实体类型Entity References个数，输出数据元素类型Output Types个数 |
| 4 | 实验步骤二实现 | 根据用户填写的步骤一的数据自动计算调整前的功能大小FPI |
| 5 | 实验步骤三实现 | 实现技术复杂性调整，在 0到5的范围内测量19个（或更多）技术特征对应用规模的影响。  即实现用户从数据流，分布式功能，性能等20个技术中挑选19个，并为其复杂度调整特性的影响程度打分， 0~5表示影响程度由小到大 |
| 6 | 实验步骤四实现 | 根据用户填写的步骤三的数据自动计算总影响度TDI |
| 7 | 实验步骤五实现 | 根据用户填写的步骤三的数据自动计算技术复杂度调整TCA |
| 8 | 实验步骤六实现 | 根据用写的步骤三的数据自动计算调整后的功能点指数（AFPI） |
| 9 | 实验步骤七实现 | 根据用户填写的步骤三的数据自动根据生产率计算项目工作量 |
| 10 | 实验心得填写 | 提供实验心得的填写功能 |

使用活动图表示如下：



2.2.2 **非功能性需求**

1. 观感性

即页面的观看舒适度。界面舒适度很大的效果上是来自操作者的反馈，对于需求外观的期望、情绪和风格有一定要求。包括：

易接受性：色彩是否和当前系统类型一致，例如蓝色偏商务风等。

风格统一性：设计风格是否统一，考虑用户在多个系统之间进行系统切换的时候，打开多页面不迷失的问题。

1. 易用性

易理解：用户在使用该系统的时候，思维方式和常规人或软件的思维方式一样。

易学习：用户使用该系统所花费的成本是否过高。每个功能是否需要单独学习相关的操作和理解。

易操作：操作和控制该系统是否方便。例如页面排版上，数据是否方便填写和选择等，可以方便快速找到内容。

1. 可靠

在规定的时间和条件下，其稳定运行的能力。主要从以下几个方面考量

稳定性：无错运行率99.99%。

容错性：软件在错误的数据和环境下，对于错误的处理能力。

可恢复性：程序在宕机后，多长时间能恢复。

1. 兼容性

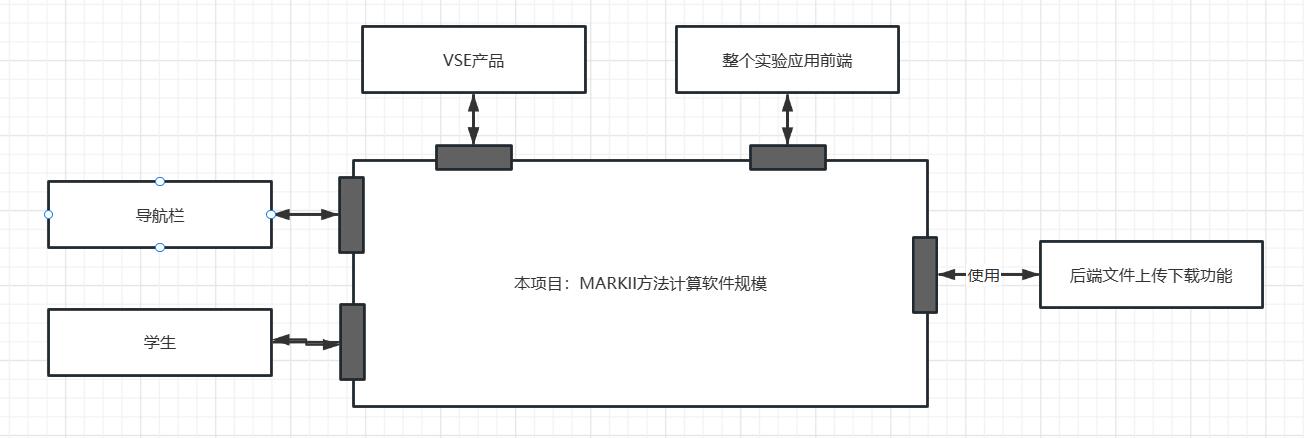
保证不同浏览器下前端界面的正常展示和使用。

3. **项目设计**

3.1 **概要设计**

3.1.1 **体系结构上下文图**

我们的项目是整个虚拟仿真实验平台中的一个模块，作为VSE体系结构的一部分，我们表示了该项目在整体虚拟仿真实验平台中的位置



3.1.2 **系统架构**

采用前端单页面应用（Single-Page Application）架构，基于Vue 3.0框架。

SPA 是一种现代化的 Web 应用程序设计方法，它通过动态地更新页面的部分内容，而不是每次用户操作都重新加载整个页面。这种设计使得应用具有更好的用户体验和响应速度。

3.1.3 **模块划分**

根据整个MARKII方法的软件规模不同过程，我们将整个实验划分成以下模块：

1. **实验目的和实验原理**

该模块承担实验介绍的功能，在这里，我们向学生介绍了什么是MARKII方法，以及MARKII的计算过程，属于学习性内容。

1. **实验步骤**

该模块是项目的主体部分，在这里，学生需要按照计算规模的一系列步骤，一步步地填写数据，并由我们的系统自动计算得出每一步结果。

|  |  |  |
| --- | --- | --- |
| 步骤 | 功能 | 交互方式 |
| 第一步 | 功能点分析 | 用户填写 |
| 第二步 | 计算调整前的功能大小 | 自动计算 |
| 第三步 | 技术复杂性调整 | 用户选择 |
| 第四步 | 技术复杂度调整 | 自动计算 |
| 第五步 | 计算总功能点数 | 自动计算 |
| 第六步 | 计算调整后功能点数 | 自动计算 |
| 第七步 | 根据生产率计算项目工作量 | 自动计算 |

1. **实验心得**

该模块主要组件为一张表单，学生需要在文本输入框中输入完成该实验的心得体会，作为最后生成的实验报告中的一部分。

3.1.4 **技术选型**

1. **Vue3**

Vue3 是一个流行的 JavaScript 框架，用于构建用户界面。它是 Vue.js 的最新版本，提供了许多改进和新功能。Vue3 采用了一种名为 Composition API 的新的组合式 API，使得在编写可复用的组件和管理组件状态时更加灵活和方便。它还引入了虚拟 DOM 的优化算法，提高了性能，并提供了更好的 TypeScript 支持。

1. **AntdV**

AntdV（Ant Design Vue）是一个基于 Vue 的 UI 组件库，提供了一套美观、易用且高效的组件，用于构建现代化的 Web 应用程序。AntdV 的设计灵感来自于 Ant Design，一个流行的 React UI 组件库。AntdV 提供了大量的组件，包括按钮、表单、弹出框、导航、图标等，以及一些复杂的组件如表格、树形控件和日期选择器等。它还提供了丰富的主题定制和国际化支持。

1. **Stepin Template**

Stepin Template 是一个前端项目模板，用于快速搭建基于 Vue.js 和 AntdV 的 Web 应用程序。它提供了一个预设的项目结构和配置，包括了常用的工具和依赖项，帮助开发者快速开始一个全功能的前端项目。Stepin Template 集成了 Vue CLI（Vue.js 的脚手架工具），并在此基础上做了一些定制和配置，以支持 AntdV 组件和特定的项目需求。开发者可以基于 Stepin Template 进行二次开发，快速构建符合自己需求的应用程序。

3.2 **前端页面设计**

经过初步分析，我们对前端页面做出了如下设计（第一稿）

