|  |  |
| --- | --- |
|  | **软件工程学期实训III** |
| **文档编号：SEST2023-[TEAMID]-[TEAMNAME]** |

[项目LOGO]

**[项目名称]**

**[Project Name]**

**项目开发文档**

**Version: [Version Number]**

[Team LOGO]

**[Team Name]**

**[YYYY-MM-DD]**

**All Rights Reserved**

目录

[1 项目概述 1](#_Toc56720659)

[1.1 项目背景 1](#_Toc56720660)

[1.2 项目定位 1](#_Toc56720661)

[1.2.1 应用场景 1](#_Toc56720662)

[1.2.2 目标人群 1](#_Toc56720663)

[1.3 项目方案 1](#_Toc56720664)

[1.4 项目目标 1](#_Toc56720665)

[1.5 项目价值 1](#_Toc56720666)

[2 开发计划 2](#_Toc56720667)

[2.1 最终呈现形式 2](#_Toc56720668)

[2.2 主要功能描述 2](#_Toc56720669)

[2.3 运行环境 2](#_Toc56720670)

[2.4 验收标准 2](#_Toc56720671)

[2.5 关键问题 2](#_Toc56720672)

[2.6 进度安排 2](#_Toc56720673)

[2.7 开发预算 2](#_Toc56720674)

[3 可行性分析 3](#_Toc56720675)

[3.1 技术可行性分析 3](#_Toc56720676)

[3.2 资源可行性分析 3](#_Toc56720677)

[3.3 市场可行性分析 3](#_Toc56720678)

[4 需求分析 4](#_Toc56720679)

[4.1 数据需求 4](#_Toc56720680)

[4.1.1 静态数据 4](#_Toc56720681)

[4.1.2 动态数据 4](#_Toc56720682)

[4.1.3 数据词典 4](#_Toc56720683)

[4.1.4 数据采集 4](#_Toc56720684)

[4.2 功能需求 4](#_Toc56720685)

[4.2.1 \*\*功能模块 4](#_Toc56720686)

[4.3 性能需求 5](#_Toc56720687)

[4.3.1 时间特性 5](#_Toc56720688)

[4.3.2 适应性 5](#_Toc56720689)

[4.4 界面需求 5](#_Toc56720690)

[4.5 接口需求 5](#_Toc56720691)

[4.5.1 硬件接口 5](#_Toc56720692)

[4.5.2 软件接口 5](#_Toc56720693)

[4.6 其他需求 5](#_Toc56720694)

[5 概要设计 6](#_Toc56720695)

[5.1 处理流程 6](#_Toc56720696)

[5.2 总体结构设计 6](#_Toc56720697)

[5.3 功能设计 6](#_Toc56720698)

[5.4 用户界面设计 6](#_Toc56720699)

[5.5 数据结构设计 6](#_Toc56720700)

[5.6 接口设计 6](#_Toc56720701)

[5.6.1 外部接口 6](#_Toc56720702)

[5.6.2 内部接口 6](#_Toc56720703)

[5.7 错误/异常处理设计 6](#_Toc56720704)

[5.7.1 错误/异常输出信息 6](#_Toc56720705)

[5.7.2 错误/异常处理对策 6](#_Toc56720706)

[5.8 系统配置策略 6](#_Toc56720707)

[5.9 系统部署方案 6](#_Toc56720708)

[5.10 其他相关技术与方案 6](#_Toc56720709)

[6 数据库设计 7](#_Toc56720710)

[7 手机端侧部署设计 8](#_Toc56720711)

[7.1 手机环境需求 8](#_Toc56720712)

[7.2 调用AI Unit能力接口设计（可选） 8](#_Toc56720713)

[7.3 AI Boost端侧模型加速设计（可选） 8](#_Toc56720714)

[8 详细设计 9](#_Toc56720715)

[8.1 \*\*功能模块 9](#_Toc56720716)

[8.1.1 功能描述 9](#_Toc56720717)

[8.1.2 性能描述 9](#_Toc56720718)

[8.1.3 输入 9](#_Toc56720719)

[8.1.4 输出 9](#_Toc56720720)

[8.1.5 程序逻辑 9](#_Toc56720721)

[8.1.6 限制条件 9](#_Toc56720722)

[8.2 \*\*功能模块 9](#_Toc56720723)

文档修订历史

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序号** | **修订原因** | **版本号** | **作者** | **修订日期** | **备注** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

# 项目概述

## **项目背景**

随着内部电子设备的老化，笔记本电脑经常遭受各种不同类型的损坏，其中键盘故障是常见的问题之一。当键盘出现故障时，我们通常会优先考虑修复键盘，但对于键盘与笔记本电脑一体化的设备来说，键盘维修的费用通常相当昂贵。此外，尽管可以通过Windows键盘和鼠标点击来替代键盘操作，但这种方式会大大降低输入效率。因此，我们的注意力转向了外接键盘。尽管外接键盘解决了花费和效率方面的问题，但在空间方面又会引发了尴尬的位置布局难题。

所以我们现在急需一种虚拟键盘app。它可以将笔记本与手机连接，在手机屏幕上显示一个可操作的键盘界面，以模拟实际键盘的输入。这样，您就可以使用鼠标或触控板点击虚拟键盘上的按键来进行输入。相比修复或更换键盘，使用虚拟键盘app是一种更加经济；虽然相比实际键盘来说，使用虚拟键盘可能会稍有降低输入效率，但它仍然比使用鼠标点击要高效得多。使用虚拟键盘应用程序可以解决外接键盘的位置布局难题，因为手机可以随身携带，并且占用空间不多。

目前，市场上的类似软件主要采用蓝牙配对的方式来连接手机和电脑。尽管蓝牙连接非常简便，但与Wi-Fi相比，蓝牙存在无法同时连接多个设备、连接范围有限以及速度一般等问题。因此，现在迫切需要一款基于Wi-Fi的应用程序控制器，以实现多设备连接、稳定连接和更广范围的功能。

另外，手机连接到电脑并模拟键盘和鼠标的方式，还可以为用户带来更多的舒适度和多人游戏的连接方式。这种方法允许用户使用手机作为虚拟键盘和鼠标来操作电脑，无需直接触摸电脑设备。相比传统的键盘和鼠标操作，这种方式更加便捷和灵活，为用户提供了更多的舒适度。此外，通过手机连接电脑，多人游戏也可以得到更好的连接方式。多个手机可以同时连接到同一台电脑，每个手机充当一个独立的游戏控制器。这样，多个用户可以同时参与游戏，享受到更加互动和协作的游戏体验。

## **项目定位**

### **应用场景**

1. 笔记本电脑键盘损坏无法正常使用时。
2. 鼠标控制键盘，但效率过低无法达到使用需求时。
3. 需要在有限空间内使用键盘的情况下。

### **目标人群**

1. 笔记本电脑用户。
2. 常需要移动工作环境的专业人士。
3. 需要在有限空间内使用键盘的用户。
4. 需要同时连接多个设备的用户。

## **项目方案**

对于笔记本键盘损坏无法正常使用的问题，本项目通过开发一款基于Wi-Fi的多功能模拟器来解决。该应用程序将笔记本与手机连接，通过在手机屏幕上显示一个可操作的键盘界面，模拟实际键盘的输入。用户可以使用鼠标或触控板点击虚拟键盘上的按键进行输入操作。相比修复或更换键盘，使用多功能模拟器是一种更经济实惠的选择。虽然相比实际键盘来说，使用多功能模拟器可能会稍有降低输入效率，但它仍然比使用鼠标点击要高效得多。此外，使用多功能模拟器可以解决外接键盘的位置布局难题，因为手机可以随身携带，并且占用空间较少。通过基于Wi-Fi的连接方式，实现了多设备连接、稳定连接和更广范围的功能，弥补了蓝牙连接的局限性。

## 项目目标

开发一款基于Wi-Fi的多功能模拟器，用于解决笔记本键盘损坏导致无法正常使用的问题。

提供用户在手机屏幕上显示的可操作键盘界面，模拟实际键盘的输入。

实现通过鼠标或触控板点击虚拟键盘进行输入操作，以替代修复或更换键盘的成本高昂的解决方案。

解决外接键盘位置布局难题，使用户能够在有限空间内使用键盘，并随时携带手机进行操作。

## 项目价值

1. 经济实惠：相比修复或更换键盘，使用多功能模拟器是一种更经济实惠的选择，降低了维修费用或替换成本。
2. 提升效率：尽管相比实际键盘来说，使用虚拟键盘可能会稍微降低输入效率，但它仍然比使用鼠标点击要高效得多，提高了用户的输入效率和工作效率。
3. 解决空间难题：使用多功能模拟器解决了外接键盘的位置布局难题，用户可以随身携带手机，无需额外占用大量空间。
4. 多设备连接：通过基于Wi-Fi的连接方式，实现了多设备连接的功能，用户可以同时连接多个设备，提高了灵活性和便利性。
5. 连接稳定性和范围广：相比蓝牙连接的局限性，基于Wi-Fi的连接方式具有更稳定的连接和更广范围的覆盖，提供更好的使用体验。

# 开发计划

## 最终呈现形式

最终呈现形式将是一款基于Wi-Fi的多功能模拟器，可在用户的手机上展示一个可操作的键盘界面（同样可选择鼠标、手柄等），模拟实际键盘的输入。用户可以通过鼠标或触控板点击虚拟键盘上的按键进行输入操作，然后对应的连接相同Wi-Fi的电脑同步执行对应操作。

## 主要功能描述

连接笔记本电脑：通过Wi-Fi连接，将应用程序与笔记本电脑进行配对，以实现键盘输入的传输和控制。

键盘模拟：在手机屏幕上显示一个可操作的键盘界面，包含主要的字符键、功能键和控制键等，以模拟实际键盘的布局和功能。

鼠标模拟：在手机屏幕上显示一个可操作的鼠标界面，包含主要的触屏移动、鼠标左键和鼠标右键等，以模拟实际鼠标的布局和功能。

手柄模拟：在手机上显示一个可操作的手柄界面，模拟实际游戏手柄的布局和功能，包括方向键、动作按钮、摇杆等，以提供类似于使用实际手柄进行游戏的体验。

自定义布局：允许用户自由选择和调整界面元素的位置和样式，设置对应快捷方式，以满足其个性化需求和偏好。用户可以根据自己的喜好，自定义布局，包括调整组件的大小、位置和颜色等。

分享与下载：允许用户与他人分享或下载自定义布局。用户可以共享自定义布局给其他人，或者从他人那里下载布局到自己的设备。

## 运行环境

电脑端：操作系统：Windows10及以上版本。

手机端：操作系统：Android 9.0 及以上版本； 。

电脑端：支持无线网络连接（Wi-Fi），具备Wi-Fi适配器。

手机端：具备Wi-Fi功能，并安装了对应多功能模拟器。

网络连接：电脑和手机需要连接到同一个Wi-Fi网络。这样，它们可以通过该网络进行通信和数据传输。

## 验收标准

应用程序能够准确地模拟键盘和鼠标的输入，并将其传输到连接的电脑上。用户可以通过手机上的虚拟键盘和鼠标控制电脑的操作，包括输入文本、点击等。验收标准是应用程序在不同场景下的输入操作与电脑上实际输入一致，并能够满足用户的操作需求。

应用程序能够持续保持与电脑的连接，并实现实时的输入响应和传输，在连接过程中不出现频繁的断连或延迟现象。

应用程序能够同时处理多个手机的输入，并将它们准确传输到相应的电脑上，确保各个手机之间的输入不会相互干扰或冲突。

应用程序具有直观友好的用户界面，易于操作和配置,用户界面清晰明了，操作逻辑合理，用户可以轻松进行配置和调整。

应用程序具备良好的性能和响应速度，能够快速传输用户输入到电脑上，并实现实时的操作反馈，应用程序的响应速度快，输入延迟较低，用户可以顺畅地进行操作。

## 关键问题

设备兼容性：多功能模拟器需要依赖用户的手机和连接的电脑。不同型号和规格的手机可能会对应用程序的性能和稳定性产生影响。同样，电脑的操作系统和硬件配置也可能对应用程序的运行产生影响。

技术焦点：开发多功能模拟器需要涉及到多个技术领域，包括网络通信、Wi-Fi连接、输入模拟、多对一连接等。在不同技术领域的实现、整合和平衡上可能存在挑战。

## 进度安排



图2.1项目甘特图

## 开发预算

人力资源：开发团队的人员工资。

技术设备：开发过程中需要使用的计算机五台。

# 可行性分析

## 技术可行性分析

技术能力：开发团队多次使用Android Studio进行手机端app开发，并且曾使用蓝牙进行软件开发，开发团队具备开发所需的技术能力和经验，以确保项目能够成功实施。

技术可用性：Android Studio与Wi-Fi连接技术都经过多年的发展和改进，已经变得非常成熟。Android Studio提供了强大的功能和工具，以支持开发人员在使用Wi-Fi连接时进行应用程序的测试和调试。

技术风险：在Wi-Fi连接的过程中，可能会面临一些技术难题，例如连接不稳定、传输速度慢、数据丢失等。针对这些问题，可以采取对策：优化算法和协议，通过改进连接算法和协议，提高连接的稳定性和传输速度；采用实践技术更加熟练的蓝牙连接技术进行实现。

## 资源可行性分析

人力资源：评估团队成员五名均有多项项目参与经验、技能，确保项目能够按时完成。

物质资源：项目所需的设备、设施和物资均可获得，并且能够满足项目的需求。

财务资源：项目的预算充足资金来源稳定，确保项目能够按计划进行。

## 市场可行性分析

市场需求：现代用户对于控制方式的要求日益增长。他们需要一种灵活便捷的方式来控制电脑或其他设备，以提升工作效率或享受多人游戏的乐趣。他们希望能够通过自己的手机实现对电脑的控制，而不再局限于传统的键盘和鼠标。

竞争分析：目标市场上存在类似的产品例如“妙妙触控”、“蓝牙遥控”，存在功能不够丰富，设置麻烦，使用体验不佳以及连接不稳定的问题。

市场趋势：随着移动技术的发展，移动控制越来越受到人们的关注和青睐。随着多功能应用的普及和多人游戏社交的盛行，用户对于能够实现多人联机游戏和自定义布局的控制方式的需求也在增长。

# 需求分析

## 数据需求

### 静态数据

### 动态数据

### 数据词典

### 数据采集

## 功能需求

### \*\*功能模块

表1 核心功能模块描述

|  |  |  |  |
| --- | --- | --- | --- |
| **功能模块** | **功能** | **功能描述** | **优先级** |
|  |  |  |  |
|  |  |  |
|  |  |  |

表2 \*\*\*\*用例规约

|  |  |
| --- | --- |
| **用例名称** |  |
| **功能简述** |  |
| **用例编号** |  |
| **执行者** |  |
| **前置条件** |  |
| **后置条件** |  |
| **涉众利益** |  |
| **基本路径** |  |
| **扩展路径** |  |
| **字段列表** |  |
| **设计规则** |  |
| **未解决的问题** |  |
| **备注** |  |

## 性能需求

### 时间特性

### 适应性

## 界面需求

## 接口需求

### 硬件接口

### 软件接口

## 其他需求

# 概要设计

## 处理流程

## 总体结构设计

## 功能设计

## 用户界面设计

## 数据结构设计

## 接口设计

### 外部接口

### 内部接口

## 错误/异常处理设计

### 错误/异常输出信息

### 错误/异常处理对策

## 系统配置策略

## 系统部署方案

## 其他相关技术与方案

# 数据库设计

# 手机端侧部署设计

## 手机环境需求

## 调用AI Unit能力接口设计（可选）

## AI Boost端侧模型加速设计（可选）

# 详细设计

## \*\*功能模块

### 功能描述

### 性能描述

### 输入

### 输出

### 程序逻辑

### 限制条件

## \*\*功能模块