**软件需求说明**

**版本 1.0**

**<<有注释版本 >>**

2024年6月5日

**Snake Game**

Lewei Jiang

1. **介绍**
   1. **目的**

本文档的目的在于展示关于贪吃蛇游戏项目的细致介绍，包含贪吃蛇游戏项目的目标，项目特征和约束操作，以及约束操作下应有的反应。

* 1. **文档约定**

无

* 1. **目标读者和阅读建议**

本文档的目标读者为此贪吃蛇游戏项目的所有利益相关者，主要包括开发人员，测试人员。

* 1. **项目范围**

本文档所展示的项目为贪吃蛇游戏项目，该项目为一个包含图形化界面的，用键盘进行输入的贪吃蛇游戏。更细致地说，该游戏允许使用者通过键盘和鼠标输入，与游戏进行交互，并通过其输入操作做出对应反应，并以图形化的界面显示在屏幕上。

* 1. **参考文献**

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.* IEEE Computer Society, 1998.

* 1. **概述**

在第二章（总体描述），本文档将概述产品的总体功能以及项目所需的各项需求。

第三章（具体需求）主要面对开发人员，本文档将详细地使用技术名称对本项目的功能进行阐明。

1. **总体描述**
   1. **产品视角**

本项目不依赖于任何其他项目，因此，没有相关项目。

* 1. **产品功能**

本项目将实现以下功能：

* 视觉和界面：为用户提供图形化的游戏界面。
* 进行游戏：用户可以开始一局贪吃蛇游戏。
* 游戏内交互：用户可以使用键盘操控蛇在游戏窗口内进行移动。
* 游戏逻辑：蛇碰撞不同物品有不同的效果，如加分和游戏结束。
* 实时计分：包含一个计分板，用于显示当前分数。
* 游戏结束：向用户显示游戏结束的信息。
* 重启游戏：在游戏结束后，用户可以重新开始游戏。
  1. **用户特征**

本项目的目标用户主要为任何对游戏有兴趣的人员。

* 1. **操作环境**

本项目需要使用Window系统，需要有Cmake 3.0或以上以及SFML (Simple and Fast Multimedia Library) 2.6版本或以上。

* 1. **设计和实现约束**
     1. **编程语言限制**

本项目需要使用C++进行开发。

* + 1. **硬件要求**

本项目需要兼容常见的配置，包括不同的显示分辨率和图形处理能力。

* + 1. **图形库限制**

游戏将使用特定的图形库，例如 SFML (Simple and Fast Multimedia Library) 来处理图形和多媒体需求。

* + 1. **内存使用**

项目需要高效地使用内存，并防止内存泄漏和不必要的资源消耗。所有动态分配的内存都必须在不再需要时释放。

* + 1. **版本控制**

开发过程中需要使用 Git 进行版本控制，以便于团队协作和代码管理。

* 1. **假设和依赖**

考虑到不同因素对项目的影响，本项目做如下假设：

* 用户有最低电脑配置，且使用Windows系统。
* 开发周期中，开发需求不会有巨大变动。
* SFML库提供稳定和充足的支持。

1. **具体需求**
   1. **用户界面**

用户界面能给用户提供直观的显示和直觉的交互，本项目提供图形化的界面给用户进行贪吃蛇游戏。

* 1. **功能**
     1. **视觉和界面**

用户可以观看图形化的游戏界面，而非仅观看Ascii字符。

* + 1. **进行游戏**

用户可以在程序运行后，点击开始游戏进行一局贪吃蛇游戏。游戏包含一条蛇，水果和四面墙壁，四面墙壁分别在游戏界面的四边。蛇的身体有一个初始长度，以一个速度从初始位置朝一个方向进行移动。水果在被蛇碰撞后消失，视为被食用，如果蛇食用了水果，则根据不同水果执行不同效果。如果蛇碰撞墙壁或自身身体的一部分，视为蛇死亡，蛇死亡则视为游戏结束。

* + 1. **游戏内交互**

在游戏中，用户可以使用键盘输入，操控游戏中的蛇，在朝一个方向移动的过程中进行转向操作，转向另外一个方向移动。

* + 1. **游戏逻辑**

水果分为三类：苹果，橘子和地雷：食用苹果后玩家本局分数+1且蛇的身体增长一节。食用橘子后玩家本局分数+5，蛇的速度变快且蛇的身体增长一节。食用地雷后蛇视为死亡。在非地雷水果被食用后，应重新在窗口内随机位置生成一种非地雷的水果。

* + 1. **实时计分**

在食用不同水果后，用户本局分数会根据水果种类实时变化。分数直观显示在游戏界面上，便于用户得知自己本局游戏当前的分数，

* + 1. **游戏结束**

在游戏结束时，蛇应该停止移动，用户不能再对蛇进行操控。此时显示游戏结束以及本局游戏当前分数。

* + 1. **重启游戏**

在游戏结束后，提供给用户选择：选择1退出游戏：如果用户选择此项则程序结束。2重启游戏：如果用户选择此项则重新开始一局游戏，此时本局当前分数清零，蛇的位置和初始长度重置。用户可以用鼠标选不同的选项，并做不同的处理。

* + 1. **错误处理**

如果游戏出现错误，对错误进行处理并向用户显示对应错误信息。

* 1. **性能需求**

**3.3.1 响应时间**

应保证游戏的响应时间，保证游戏的刷新率，一秒钟显示30帧画面。

* 1. **非功能需求**
     1. **安全需求**

在游戏运行时不应对用户设备产生危害。

* + 1. **安全性需求**

本游戏应不收集和保留任何用户的个人信息，在运行的过程中不调用摄像头，麦克风以及不访问未分配的内存和其他文件。本游戏应不使用网络。

* + 1. **软件性需求**
       1. **可用性**
* 在用户启动时，游戏应正常启动并稳定运行。
* 在游戏运行过程中，不应有卡死或者程序崩溃的情况。
* 如果出现错误，应提供错误信息并有对错误有对应处理。
  + - 1. **正确性**

本项目应在用户进行输入操作的情况下，正确输出符合游戏逻辑的结果，如按用户输入的方向正确对蛇进行转向。

本项目应正确生成游戏内容，如水果的重新生成。

* + - 1. **可维护性**

本项目应易于维护，包括良好的注释，命名规范和文档。

* + - 1. **可扩展性**

本项目应有良好架构和顶层设计，便于之后开发对功能的扩展。