chayunyolong

Celestial Quest: The Journey of the Immortal Emperor

大帝封神

一款基于文本的角色扮演游戏，融合了冒险、探索和谜题解决的元素。



设计文档版本： 1.0

倪家诚 f23011126

2024/1/4

**1. 项目简介**

**2. 游戏概念**

**3. 游戏机制**

**4. 技术规格**

**5. 用户界面设计**

**6. 剧情设置**

**7. 测试和质量保证**

**8. 发布计划**

**9. 附录**

#### 1. 项目简介

* **项目名称**：《大帝封神》
* **项目描述**：《大帝封神》是一个基于文本的角色扮演游戏，让玩家沉浸在一个充满魔法和神秘的仙道世界。游戏中，玩家将探索这个奇幻世界，寻找传说中的无始大帝，解开隐藏的秘密，并面对重重挑战。
* **目标平台**：Windows
* **游戏类型**：文本冒险、角色扮演

#### 2. 游戏概念

* **游戏背景**：在《大帝封神》中，玩家将进入一个古老的仙道世界，这里充满了未解之谜和传说故事。玩家扮演一名冒险者，踏上了寻找无始大帝的征程，探索神秘的山谷、海域和神圣之地。
* **核心玩法**：玩家需要通过一系列的选择来推动故事发展。这些选择包括探索不同的神秘地点、与游戏世界中的角色互动，以及解决谜题。每个决策都会影响游戏的进程和最终结果。
* **目标受众**：游戏面向喜爱奇幻、冒险和角色扮演的玩家，特别是那些享受沉浸式故事和复杂决策制定的玩家。

#### 3. 游戏机制

* **探索**：玩家将探索各种神秘的环境，如神源之地、碧海须臾界等，每个地点都藏有不同的秘密和挑战。
* **决策与后果**：游戏强调玩家决策的重要性，每个选择都有可能带来不同的后果，包括推动故事发展或导致游戏结束。
* **分数系统**：玩家的选择不仅影响故事走向，还会影响他们的分数。分数是衡量玩家在游戏中表现的关键指标，不同的决策会带来不同数量的分数增加或减少。
* **登录和安全系统**：游戏包含一个账户系统，玩家可以创建和保存自己的游戏进度。系统采用文件存储来保存用户凭证，保证了基本的安全性。在连续登录失败后，系统提供重置密码的选项。

#### 4. 技术规格

* **编程语言**：C。这种高效且功能强大的语言非常适合开发复杂的游戏逻辑和处理游戏中的各种数据交互。
* **开发工具和库**：
  + **Visual Studio**：作为主要的集成开发环境（IDE），用于编写、编译和调试游戏代码。
  + **Windows API**：用于处理Windows平台上的用户输入、文件操作和其他系统级交互。
* **数据存储**：
  + **存档文件处理**：游戏进度通过二进制文件形式保存，每个存档包含了玩家的分数、游戏进度和其他关键游戏状态。使用标准的文件输入/输出操作进行读写。
  + **安全性**：虽然游戏的存档文件是以明文形式存储，但考虑到游戏的性质，对安全性的要求不高。未来可以考虑引入加密措施提高数据安全性。

#### 5. 用户界面设计

* **界面布局**：
  + **主界面**：清晰的文本显示区域，用于展示游戏故事和描述。界面顶部或底部设有简洁的菜单栏，包括存档、加载和退出等选项。
  + **游戏界面**：主要以文本为中心，辅以简单的文本框架和色彩，增强可读性和视觉吸引力。对话和描述文本清晰分隔，易于玩家跟踪。
* **用户交互**：
  + **输入机制**：玩家主要通过键盘输入进行交互，如输入数字选择不同的选项或进行游戏决策。特定场景下，玩家可能需要输入文本（如保存文件名）。
  + **反馈机制**：游戏以文本和简单的视觉提示（如高亮选项）来反馈玩家的操作结果。例如，选择一个选项后，游戏会显示新的故事线或游戏结果。

**6. 剧情设置**

**详见附录 剧本.docx**

**详见附录 结局.png**

**7. 测试和质量保证**

**详见附录 测试报告.docx**

**8. 发布计划**

#### 8.1 发布目标

* **最终发布日期:** 确定游戏《大帝封神》的最终发布日期是关键。建议设定一个具体的日期，考虑到测试反馈和最后的调整时间。例如，假定最终发布日期为2024年1月4日。
* **发布平台:** 游戏将通过GitHub进行发布。GitHub不仅是一个代码托管平台，还允许开发者发布和分享他们的作品。这将使游戏容易被开发社区发现，并促进开放源代码的协作和迭代。

#### 8.2 版本控制

* **版本命名规则:** 对游戏的每次更新和迭代使用语义化版本命名规则（如v1.0.0, v1.0.1等）。第一个数字代表主要更新，第二个数字代表次要更新，第三个数字代表补丁或者小修正。
* **更新日志:** 每次发布新版本时，都应提供详细的更新日志。这应该包括新功能的描述、改进点、修复的错误等。
* **发布流程:**
  1. **准备阶段:** 确保所有的新功能都已经完全集成并通过了测试。
  2. **打包阶段:** 将游戏源代码打包，包括所有必要的文件和资源。
  3. **发布:** 在GitHub上创建一个新的发布版本，并上传打包好的游戏文件。
  4. **通知:** 通过社交媒体、电子邮件订阅等渠道，通知社区和潜在的玩家新版本的发布。
* **更新策略:** 根据用户反馈和游戏表现，定期发布更新。对于紧急的错误修复，应及时发布补丁。

#### 8.3 维护和支持

* **问题跟踪:** 使用GitHub的问题跟踪系统来管理用户报告的问题和请求新功能。
* **社区参与:** 鼓励玩家在GitHub上提交问题报告和功能请求，促进社区的参与和反馈。

#### 8.4 后续发展

* **长期支持:** 计划对《大帝封神》进行长期的支持和更新，以维持游戏的活力和吸引力。
* **迭代开发:** 基于用户反馈和市场趋势，持续迭代开发，引入新内容和功能。

**9. 附录**

[**剧本.docx**](剧本.docx)

[**结局.png**](结局.png)

[**测试报告.docx**](测试报告.docx)