



**浙江大學**  
ZHEJIANG UNIVERSITY

# **C++ 项目管理及工程实践**

## **项目说明**

### **C++ Project Management and Engineering Practices**

组长 \_\_\_\_\_

组员一 \_\_\_\_\_

组员二 \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

# 目 录

<b>第 1 章 项目描述</b>	<b>1</b>
1.1 概述	1
1.2 开发平台介绍	1
1.2.1 JetBrains CLion	1
1.2.2 Qt	2
1.3 游戏介绍	3
<b>第 2 章 项目需求</b>	<b>5</b>
2.1 交互需求	5
2.2 呈现功能	5
2.3 迭代需求	5
<b>第 3 章 开发技术规划</b>	<b>7</b>
3.1 技术架构选择	7
3.2 开发阶段计划	7
<b>第 4 章 开发工具</b>	<b>9</b>
4.1 版本控制	9
4.2 持续集成	9
4.3 开发环境	9
4.4 编程语言	9
4.5 框架结构	9
4.6 文档撰写	9
4.7 交流讨论	9
<b>第 5 章 小组分工与计划</b>	<b>10</b>
5.1 项目成员及分工	10

# 第 1 章 项目描述

## 1.1 概述

《Dad n' Me》由 The Behemoth 游戏公司在 2011 年 5 月 19 日出品，是一款经典的 Flash 小游戏，因其优质动画风格和休闲特性，而受到众多玩家的好评。玩家将控制主角：紫色的暴力儿子，躲避孩子王的攻击、顺利通过地形，以完成各种场景的挑战，曾经在千禧一代之间非常火热。本 Project 便是一次仿照 The Behemoth 游戏公司经典游戏的复刻学习。

## 1.2 开发平台介绍

### 1.2.1 JetBrains CLion

*JetBrainsCLion* 是由 *JetBrains* 公司开发的一款专为 C 和 C++ 语言设计的跨平台集成开发环境（IDE）。CLion 提供了一系列强大的功能，旨在提高开发者的生产力和代码质量。

1. **智能编码辅助:**CLion 提供了智能的代码补全、代码分析和快速修复建议，帮助开发者更快地编写代码。
2. **深入的代码理解:**CLion 能够深入理解 C 和 C++ 代码结构，提供上下文相关的代码导航和查找功能。
3. **高级重构工具:**支持安全重构，包括但不限于重命名、移动、更改签名等，确保代码更改的一致性和正确性。
4. **代码生成:**CLion 能够根据代码上下文生成构造函数、析构函数、getter/setter 方法等，简化编码工作。
5. **集成调试器:**内置的调试工具支持设置断点、单步执行、观察变量和调用堆栈，方便开发者调试程序。
6. **版本控制集成:**支持 Git、SVN 等主流版本控制系统，方便代码的版本管理和团队协作。
7. **测试支持:**CLion 集成了对 Google Test、Boost.Test 和 Catch 等测试框架的支持，简化了测试流程。
8. **跨平台支持:**支持 Windows、macOS 和 Linux 操作系统，开发者可以在不同的平台上使用相同的开发环境。

9. **自定义和扩展性:** 提供丰富的自定义选项, 允许开发者根据自己的喜好调整界面和 workflow。同时, 支持插件扩展, 以适应特定的开发需求。
10. **性能优化:** JetBrains 一直在优化 CLion 的性能, 确保即使在处理大型项目时也能保持流畅的用户体验。
11. **持续更新:** 定期发布 CLion 的更新, 引入新功能、改进现有功能并修复已知问题。
12. **教育支持:** JetBrains 为学生和教师提供了免费的 JetBrains 工具包, 包括 CLion 在内, 以支持教育和学习。

CLion 的这些优势使其成为许多 C 和 C++ 开发者的首选 IDE, 尤其是在需要高效、智能编码辅助和强大项目管理功能的场景中。

### 1.2.2 Qt

QtCompany 是一家提供跨平台软件开发工具的公司, 其旗舰产品 Qt 是一个广泛使用的应用程序框架, 用于开发图形用户界面 (GUI) 应用程序、命令行工具和嵌入式系统。以下是按照您要求的格式整理的 Qt 软件及其优势列表:

1. **跨平台支持:** Qt 支持多种操作系统, 包括 Windows、macOS、Linux、Android、iOS 和嵌入式系统, 允许开发者编写一次代码, 然后在多个平台上运行。
2. **丰富的 API:** 提供一套完整的 API, 包括用于创建 GUI、网络编程、数据库访问、XML 处理等的模块。
3. **模块化架构:** 模块化设计允许开发者根据项目需求选择性地包含所需的模块, 从而减小应用程序的体积和复杂性。
4. **高性能:** Qt 应用程序通常具有高性能, 特别是在图形渲染和用户界面响应方面。
5. **易于学习和使用:** 提供清晰的文档和示例, 使得学习和使用 Qt 成为一个相对简单的过程。
6. **强大的开发工具:** Qt Creator 是 Qt 的官方 IDE, 提供了代码编辑、项目管理、调试和测试等工具。
7. **社区和支持:** Qt 拥有一个活跃的开发者社区和商业支持, 为开发者提供帮助和资源。
8. **可扩展性:** 允许开发者通过插件和自定义组件来扩展其功能。
9. **硬件加速:** 应用程序可以利用 GPU 硬件加速, 提高图形渲染的性能。

10. **国际化和本地化:** Qt 提供了强大的国际化和本地化支持, 使得应用程序可以轻松地适应不同的语言和文化。
11. **安全性:** 框架注重安全性, 提供了多种安全特性来保护应用程序和用户数据。
12. **商业和开源:** 提供商业版和开源版 (Qt Open Source), 满足不同开发者和企业的需求。
13. **企业级应用开发:** Qt 被许多大型企业和组织用于开发复杂的企业级应用程序。
14. **物联网和嵌入式系统:** 也适用于物联网 (IoT) 和嵌入式系统开发, 支持多种硬件平台。
15. **持续更新和创新:** Qt Company 持续更新 Qt 框架, 引入新特性和改进, 以适应不断变化的技术需求。

Qt 的这些优势使其成为开发跨平台应用程序的强大工具, 尤其适合需要高性能、可扩展性和易用性的项目。

### 1.3 游戏介绍

”Dad n’ Me” 是一款基于 Qt 设计框架的自由主题游戏, 灵感来自游戏公司 The Behemoth 的经典作品。玩家将控制紫色的暴力儿子, 躲避孩子王的攻击、顺利通过地形, 以完成各种场景的挑战。每个关卡都有独特的地形和怪物设计, 玩家需要巧妙运用连招技能克服各种障碍。

#### 1. 游戏界面进入

游戏载入完毕后, 点击 “PLAY”, 可选择单人、双人模式。选择关卡后即可开始游戏。

游戏开始界面



图 1-1 Start scene

#### 2. 操作方法

移动: ← (左), → (右), ↑ (上), ↓(下)

攻击: A (手击), S (腿击)

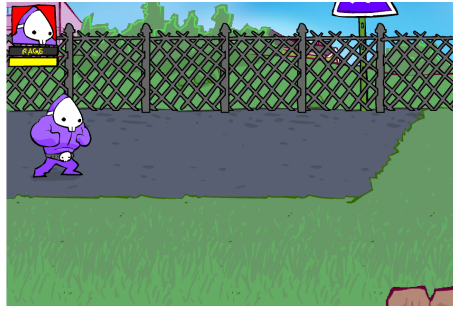


图 1-2 Game scene1

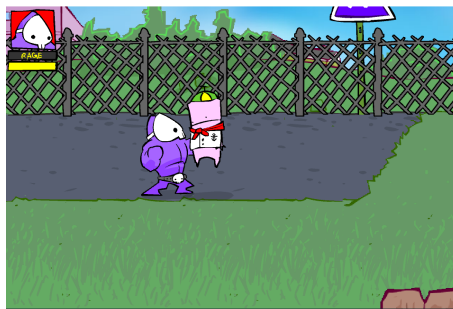


图 1-3 Game scene2

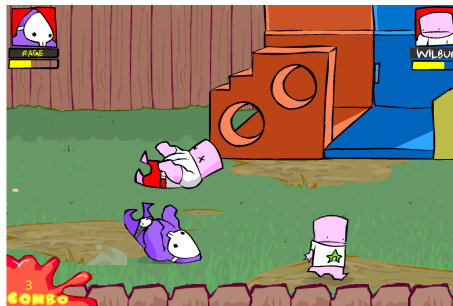


图 1-4 Game scene3



图 1-5 Gamewin scene



图 1-6 Gameover scene

## 第 2 章 项目需求

### 2.1 交互需求

1. 用户可以控制 Hero 击打小怪，并实时显示得分。这使得用户能够立即看到他们的游戏进展和成就，增加游戏的互动性和趣味性。
2. 用户使用技能时会播放对应的音效，为用户提供更好的沉浸式体验。这种音效反馈不仅可以增强游戏的娱乐性，还可以帮助用户更好地掌握技能使用的时机和效果。
3. 用户可以随时重置或者退出游戏。这一功能保证了游戏的灵活性和用户体验，允许用户在遇到困难或需要暂停时能够方便地重新开始或退出游戏。

### 2.2 呈现功能

1. 为用户呈现出流畅的游玩画面。游戏界面需要设计简洁美观，动画效果流畅自然，确保用户在游玩过程中不会因为卡顿或画面质量问题而感到不适。

### 2.3 迭代需求

1. 在后续的迭代中实现更多可能功能。为了提高游戏的可玩性和用户黏性，未来的迭代中可以考虑以下扩展功能：
  - **多样化的敌人和关卡设计：**增加不同类型的敌人和具有挑战性的关卡，使游戏更加丰富多样。
  - **角色技能树和升级系统：**引入角色技能树和升级系统，让用户可以通过游戏中的表现和得分解锁新的技能和能力，增加游戏的深度和策略性。
  - **社交功能：**添加好友系统和排行榜，允许用户与好友进行比拼和互动，增强游戏的社交属性。
  - **剧情模式：**引入完整的游戏剧情和任务系统，使用户在游玩过程中有更强的代入感和目标感。
  - **自定义角色和装备：**允许用户自定义 Hero 的外观和装备，提供个性化的游戏体验。
  - **更多的音效和音乐：**增加游戏中的背景音乐和多样化的音效，提升游戏的音频体验。

- **优化性能和兼容性：**持续优化游戏性能，确保在不同设备和平台上都能流畅运行，并解决任何可能出现的兼容性问题。



## 第3章 开发技术规划

### 3.1 技术架构选择

- **C++ 和 Qt:** C++ 作为游戏逻辑和核心功能的实现语言, Qt 提供了丰富的图形界面和跨平台开发能力, 使得开发效率更高且易于维护。
- **MVVM 架构:** 采用 Model-View-ViewModel 架构, 将游戏逻辑 (Model)、用户界面 (View) 和逻辑与视图之间的交互 (ViewModel) 分离, 以提高代码的可维护性和可测试性。

### 3.2 开发阶段计划

#### 第一轮迭代: 基础框架与界面设计

##### 目标:

- 设计游戏程序的基本框架。
- 实现基本的角色控制和移动功能。

##### 任务:

- 创建项目结构和基础框架。
- 设计并实现基本游戏界面视图 (View)。
- 实现基本的角色和敌人 (Small Enemy) 的移动控制逻辑 (ViewModel) 和基本战斗交互。
- 进行视图和视图模型的数据绑定, 确保数据的正确显示和用户输入的响应。

#### 第二轮迭代: 游戏逻辑与交互设计

##### 目标:

- 实现主角与敌人 (Boss) 之间的基本交互和战斗逻辑。
- 增加游戏元素: 增加主角的攻击手段、显示血量。
- 添加基本的动画效果和音效支持。

##### 任务:

- 完善角色和敌人的战斗逻辑 (ViewModel)。

- 实现攻击动作和反馈效果的动画支持。
- 引入背景音乐和击打音效，增强游戏的沉浸感和交互性。

### 第三轮迭代：优化与扩展

#### 目标：

- 优化游戏性能和用户体验。
- 扩展游戏功能，增加 boss 战斗和更多的角色交互。
- 完善用户界面和视觉效果。

#### 任务：

- 进行代码优化和性能调优，确保游戏在不同平台上的流畅运行。
- 实现 boss 类型角色的战斗逻辑和动画效果。
- 增加游戏界面的交互元素和动态效果，提升用户体验。

## 第 4 章 开发工具

### 4.1 版本控制

本项目使用 Git 进行版本控制，并托管在 Github 上。

### 4.2 持续集成

持续集成工具选择了 Jenkins，以确保代码的质量和稳定性。

### 4.3 开发环境

开发环境使用了 QT 6.5.3，以提供良好的图形用户界面支持。

### 4.4 编程语言

项目的主要编程语言为 C++ 17，兼具性能和现代编程特性。

### 4.5 框架结构

采用了 MVVM 架构（模型-视图-视图模型），实现了前后端分离，高复用性和低耦合性。

### 4.6 文档撰写

文档撰写在 overleaf 在线平台协作进行，使用了 Markdown 和 LaTeX，确保文档的清晰性和可维护性。

### 4.7 交流讨论

团队内部的交流和讨论通过钉钉视频会议和线下进行，以保证沟通的效率和质量。

## 第 5 章 小组分工与计划

### 5.1 项目成员及分工

- 开发 App 层，组装其他两层的代码
- 参与度占比 1/3
- 开发 View 层和 Window 层
- 参与度占比 1/3
- 开发 ViewModel 和 Model 层
- 参与度占比 1/3