

**《软件工程综合项目实战》**

**课程设计报告**

**设计题目: 基于Git\Github的代码托管及版本控制**

**学 院： 明德学院**

**专 业：计算机科学与技术**

**班 级： 计科1811**

**姓 名： 李义芳**

**学 号： 2030030553**

2021年 6月30日

**贵州大学明德学院本科课程设计**

**诚信责任书**

本人郑重声明：本人所呈交的课程设计，是在导师的指导下独立进行研究所完成。课程设计中凡引用他人已经发表或未发表的成果、数据、观点等，均已明确注明出处。

特此声明。

课程设计 作者签名：

日 期： 年 月

基于Git\Github的代码托管及版本控制

# 摘要

Git是一个开源的分布式版本控制系统，可以有效，高速的处理从很小到非常大的项目版本管理，他也是目前世界上最先进的分布式版本控制系统。它具有备份、记录操作时间线、多端共享代码、自动闭合，缺少了版本控制系统，就会给我们的工作带来很多不便，git的每个机器上都是一个完整的库，平时开发代码的中央服务器其实和我们自己的机器上的库内容基本完全一样

论文主要使用设计使用GitHub仓库、Pycharm工具设计一个基于Python的飞机大战游戏，游戏设计是基于Pygame框架，使用Pycharm工具进行设计，用Github进行版本控制。游戏是用户进入游戏开始游戏击落敌机产生特殊效果，碰撞敌机也会有特殊效果，生命会自动减一，减到三时候游戏结束。

关键字：飞机大战，github，Python，pycharm

The System Based on HTML5 Web Manuscript of Submission And Review

# Abstract

Git is an open source distributed version control system, which can effectively and high-speed deal with the version management of projects from small to very large. It is also the most advanced distributed version control system in the world. It has backup, record operation timeline, multi-terminal shared code, and automatic closing. Without version control system, it will bring a lot of inconvenience to our work. Every machine of git is a complete library, and the central server for developing code is basically the same as the library content on our own machine. This paper mainly uses GitHub warehouse and Pycharm tool to design a Python-based aircraft war game. The game design is based on Pygame framework, using Pycharm tool to design, and using Github for version control. The game is when the user enters the game and starts the game to shoot down enemy planes to produce special effects. Collision with enemy planes will also have special effects, and life will be automatically reduced by one to three, when the game is over.

Key Words: Aircraft Wars, github, Python，pycharm

**目 录**

[摘要 III](#_Toc73608573)

[Abstract IV](#_Toc73608574)

[第一章 课程设计背景 1](#_Toc73608575)

[1.1项目开发背景 1](#_Toc73608576)

[1.2 项目开发目的和意义 1](#_Toc73608577)

[1.3 国内外研究现状 2](#_Toc73608578)

[1.4 发展趋势 3](#_Toc73608579)

[1.5 本文主要工作 3](#_Toc73608580)

[第二章 课程设计主要技术及资源 5](#_Toc73608581)

[2.1 Python 5](#_Toc73608582)

[2.2 Pygame 6](#_Toc73608583)

[2.2 Pycharm 6](#_Toc73608584)

[2.3 Git 6](#_Toc73608585)

[2.4 Github 6](#_Toc73608586)

[2.5 Gittee 6](#_Toc73608587)

[第三章 项目需求分析 7](#_Toc73608588)

[3.1 功能需求 7](#_Toc73608589)

[3.2 系统功能设计 10](#_Toc73608590)

[3.2 非功能需求 12](#_Toc73608591)

[**第四章 项目详细设计** 14](#_Toc73608592)

[4.1 数据库需求设计 14](#_Toc73608593)

[4.2 概要设计 14](#_Toc73608594)

[4.2.1 系统E-R图 14](#_Toc73608595)

[4.2.2 实体图 15](#_Toc73608596)

[4.3 逻辑设计 16](#_Toc73608597)

[**第五章 结论** 19](#_Toc73608598)

[5.1 总结 19](#_Toc73608599)

[5.2 展望 20](#_Toc73608600)

[参考文献 21](#_Toc73608601)

第一章 课程设计背景

1.1背景

随着社会经济实力的迅速发展,人们的就业压力越来越大。长期的暴力和忙碌的工作压力使人们感到紧张。为了缓解紧张工作带来的压力，我们应该劳逸结合，采用适当的休闲娱乐方式放松自己。有很多方法可以减轻压力。玩游戏为自己选择正确的方式也是缓解压力的许多方法之一。看电影、电视和音乐也可以缓解压力,但人们可以被动地听,而不参与互动。而游戏有很好的互动性,玩家既要综合看角色如何移动,又要在角色中工作。在游戏中,玩家的行为也会影响游戏。除了有趣的人,游戏还可以用来缓解压力。当你休息的时候,你不必考虑任何事情,压力也会降低。  
 随着个人电脑的普及，游戏并没有被广泛接受。主要原因是不能充分发挥计算机的性能。游戏程序不能最大限度地利用PC机的硬件资源，而机器与人之间的交互作用还不够。因此，由于图形化、刚性操作和低性能，PC游戏不被接受。然而在计算机硬件的迅猛发展的前提下，同时伴有C、C++、Java、Python等高级语言不断涌现，这一问题已经被解决，游戏质量得到大幅度提升。比如从暴雪1998年发布的星际争霸，世界魔兽争霸，再到今天的英雄联盟等，带给所有玩家一个伟大的乐趣和一个良好的游戏体验。当然，除了游戏本身的卓越品质外，它还可以支持广泛的硬件设备和操作系统功能，不容忽视，高、中、低端的电脑都有很好的流畅性。

1.2 课程设计目的和意义

自网络游戏发展以来，市场竞争十分激烈，为了迎合市场的需求，这就促使开发者快速有效地响应玩家的需求，做到更好，以争取网络游戏市场的一席之地。在传统的游戏开发中，C/C++技术得到了广泛的应用。虽然C/C++具有明显的优势性和来发优势，但很多时候我们并不关心程序的效率，而关注程序开发效率、维护效率、跨平台性等。  
Python坚持清晰划一的编码风格，可读性较好，相较于其他语言更易维护。此外，Python还是免费开源的，具有极其丰富与强大的类库，支持代码热更新。综合时间及需求考虑，构建基于飞机战争的系统使用Python来开发是非常合适且具有意义的。

1.3 国内外研究现状和发展趋势

Python的多样性,意味着可以横跨多个领域,绝不仅限于Web开发、桌面程序、移动应用,甚至包含硬件开发等。所以并没有被束缚在单一的平台之上,Python具有良好的可移植性，在游戏开发上使用Python可以大大方便项目的开发和维护。  
由上面论述可知，通过Pygame工具的应用，可以简化项目的开发，Pygame使开发者不必过多的在意一些琐碎的问题，因为这些问题Pygame工具会帮我们解决，这让开发者拥有更多的时间和精力放在游戏的关键部分，给开发者带来了相当大的便利。最关键的是，Pygame拥有跨平台的特性，这使得基于它开发的项目可以自由的在各个操作系统上运行，这省去了很多繁琐的修改。随着Pygame开发工具的流行，该工具以其独特的优越性，越来越多的被应用于各种游戏的开发

1.4课程设计主要工作

配置项目的开发环境

项目的运行和测试

配置Git

注册Github\Gittee 创建仓库

项目的版本控制和托管

2~3人的开发团队，创建master dev tom bob分支

Master分支负责发布成熟版本

ver 1.0（01 02 03 04）

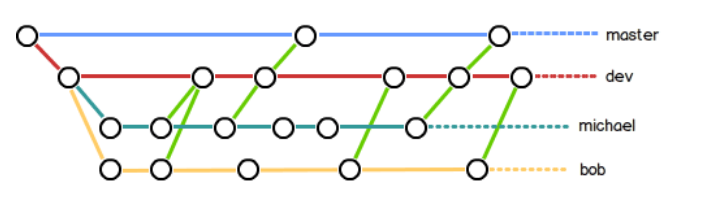
ver2.0（05 06 07 08）

ver3.0（09 10 11 12）

Dev分支负责开发

Dev分支创建至少2人的开发分支 tom bob

Tom bob的分支合并到Dev分支

****

**课程设计报告**

本系统将严格按照软件流程开发规范对社团CMS系统进行设计与开发。首先，分析系统可行性，从多层次分析技术、时间、法律、经济等可行性，推断该系统能否开发。可行性分析完成后，再对系统进行需求分析，得出社团CMS主体功能，例如：登录注册，管理员管理，资讯管理，留言管理等功能模块。之后，通过概要设计，对该系统进行详细的分析设计，将用户、轮播图[8]、资源下载、社团信息、在线留言[9]、管理员等实体类型所展现的具体内容设计出来。完成上述工作后，分析阶段任务完成，进入编码阶段，按照面向对象编程思想，对系统进行编程。在编码阶段，如果系统单个功能完成则可以开始测试相应的功能，当主体功能完成就可以进行单元模块测试，然后等系统所有功能完成，即可多次进行总体测试，如测试通过系统则开发实现，最后撰写相关开发材料。

# 第二章 课程设计主要技术及资源

2.1 Python

Python 是一种代表简单主义思想的语言。阅读一个良好的 Python 程序就感觉像是在读英语一样，尽管这个英语的要求非常严格！Python 的这种伪代码本质是它最大的优点之一。它使你能够专注于解决问题而不是去搞明白语言本身。

（1）易学

就如同你即将看到的一样，Python 极其容易上手。前面已经提到了，Python 有极其简单的语法。

（2）免费、开源

Python 是 FLOSS（自由/开放源码软件）之一。简单地说，你可以自由地发布这个软件的拷贝、阅读它的源代码、对它做改动、把它的一部分用于新的自由软件中。FLOSS 是基于一个团体分享知识的概念。这是为什么 Python 如此优秀的原因之一——它是由一群希望看到一个更加优秀的 Python 的人创造并经常改进着的。

（3）高层语言

当你用 Python 语言编写程序的时候，你无需考虑诸如如何管理你的程序使用的内存一类的底层细节。

（4）可移植性

由于它的开源本质，Python 已经被移植在许多平台上（经过改动使它能够工作在不同平台上）。如果你小心地避免使用依赖于系统的特性，那么你的所有 Python 程序无需修改就可以在下述任何平台上面运行。这些平台包括 Linux、Windows、FreeBSD、Macintosh、Solaris、OS/2、Amiga、AROS、AS/400、BeOS、OS/390、z/OS、Palm OS、QNX、VMS、Psion、Acom RISC OS、VxWorks、PlayStation、Sharp Zaurus、Windows CE 甚至还有 PocketPC、Symbian 以及 Google 基于 Linux 开发的 Android 平台！

（5）解释性

这一点需要一些解释。一个用编译性语言比如 C 或 C++ 写的程序可以从源文件（即 C 或 C++ 语言）转换到一个你的计算机使用的语言（二进制代码，即0和1）。这个过程通过编译器和不同的标记、选项完成。当你运行你的程序的时候，连接/转载器软件把你的程序从硬盘复制到内存中并且运行。而 Python 语言写的程序不需要编译成二进制代码。你可以直接从源代码运行程序。在计算机内部，Python 解释器把源代码转换成称为字节码的中间形式，然后再把它翻译成计算机使用的机器语言并运行。事实上，由于你不再需要担心如何编译程序，如何确保连接转载正确的库等等，所有这一切使得使用 Python 更加简单。由于你只需要把你的 Python 程序拷贝到另外一台计算机上，它就可以工作了，这也使得你的 Python 程序更加易于移植。

（6）面向对象

Python 既支持面向过程的编程也支持面向对象的编程。在“面向过程”的语言中，程序是由过程或仅仅是可重用代码的函数构建起来的。在“面向对象”的语言中，程序是由数据和功能组合而成的对象构建起来的。与其他主要的语言如 C++ 和 Java 相比，Python 以一种非常强大又简单的方式实现面向对象编程。

（7）可扩展性

如果你需要你的一段关键代码运行得更快或者希望某些算法不公开，你可以把你的部分程序用 C 或 C++ 编写，然后在你的 Python 程序中使用它们。

（8）丰富的库

Python 标准库确实很庞大。它可以帮助你处理各种工作，包括正则表达式、文档生成、单元测试、线程、数据库、网页浏览器、CGI、FTP、电子邮件、XML、XML-RPC、HTML、WAV 文件、密码系统、GUI（图形用户界面）、Tk 和其他与系统有关的操作。记住，只要安装了 Python，所有这些功能都是可用的。这被称作 Python 的“功能齐全”理念。除了标准库以外，还有许多其他高质量的库，如 wxPython、Twisted 和 Python 图像库等等。

（9）规范的代码

Python 采用强制缩进的方式使得代码具有极佳的可读性。

2.2 Pygame

Pygame是一个利用SDL库的写就的游戏库， 是一组用来开发游戏软件的 Python 程序模块。SDL，全名Simple DirectMedia Layer，SDL是用C写的，不过它也可以使用C++进行开发，当然还有很多其它的语言，Pygame就是Python中使用它的一个库。pygame允许你在 Python 程序中创建功能丰富的游戏和多媒体程序，是一个高可移植性的模块可以支持多个操作系统，用它来开发小游戏非常适合。

1. ORM

ORM 全拼Object-Relation Mapping，中文意为 对象-关系映射，在MVC/MVT设计模式中的Model模块中都包括ORM

1. DQL

数据查询语言DQL基本结构是由SELECT子句，FROM子句，WHERE，子句组成的查询块：

1. SELECT <字段名表>
2. FROM <表或视图名>

③WHERE <查询条件>

## 2.3 Pycharm

PyCharm是一种[Python](https://baike.baidu.com/item/Python/407313" \t "_blank) IDE（Integrated Development Environment，集成开发环境），带有一整套可以帮助用户在使用Python语言开发时提高其效率的工具，比如[调试](https://baike.baidu.com/item/%E8%B0%83%E8%AF%95/5852756" \t "_blank)、[语法高亮](https://baike.baidu.com/item/%E8%AF%AD%E6%B3%95%E9%AB%98%E4%BA%AE/9686751)、Project管理、代码跳转、智能提示、自动完成、单元测试、版本控制。此外，该IDE提供了一些高级功能，以用于支持Django框架下的专业Web开发。

## 2.4 Git

Git是一个开源的分布式版本控制软件,用以有效、高速的处理从很小到非常大的项目版本管理。 Git 最初是由Linus Torvalds设计开发的，用于管理Linux内核开发。Git 是根据GNU通用公共许可证版本2的条款分发的自由/免费软件

## 2.5 Github

（1）GitHub是一个基于Git的远程文件托管平台（同GitCafe、BitBucket和GitLab等）。

（2）Git本身完全可以做到版本控制，但其所有内容以及版本记录只能保存在本机，如果想要将文件内容以及版本记录同时保存在远程，则需要结合GitHub来使用。使用场景：

（3）无GitHub：在本地 .git 文件夹内维护历时文件

（4）有GitHub：在本地 .git 文件夹内维护历时文件，同时也将历时文件托管在远程仓库

## 2.6 Gittee

Gitee（码云）是开源中国社区推出的代码托管协作开发平台，支持Git和SVN，提供免费的私有仓库托管。Gitee专为开发者提供稳定、高效、安全的云端软件开发协作平台，无论是个人、团队、或是企业，都能够实现代码托管、项目管理、协作开发。

# 第三章 项目需求分析

3.1 功能需求

3.1.1 用例分析

玩家进入游戏如果击中敌机就会获得积分，当玩家碰撞敌机就会减去一条当前的生命中总共三条生命值，当三条生命值都没有时候游戏结束，显示最终得分，如图3.1所示。

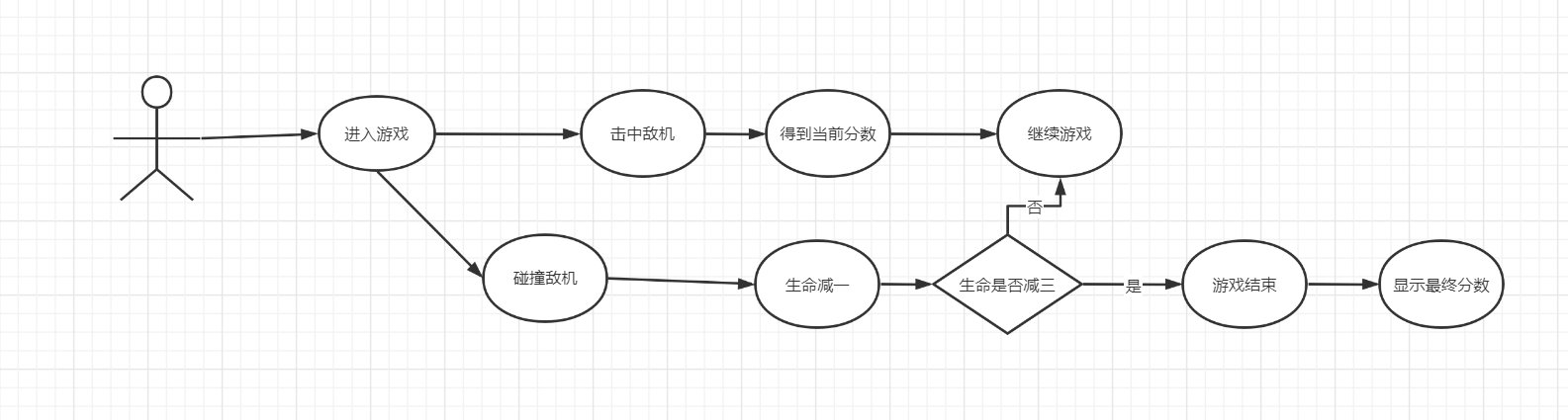


图3.1 系统用例图

2. 活动图

除了使用用例图描述系统需求以外，还可以使用活动图来描述，活动图能更具体地描述该用例与角色的交互。以下是系统主要用例的活动图。

（1）玩家开始游戏

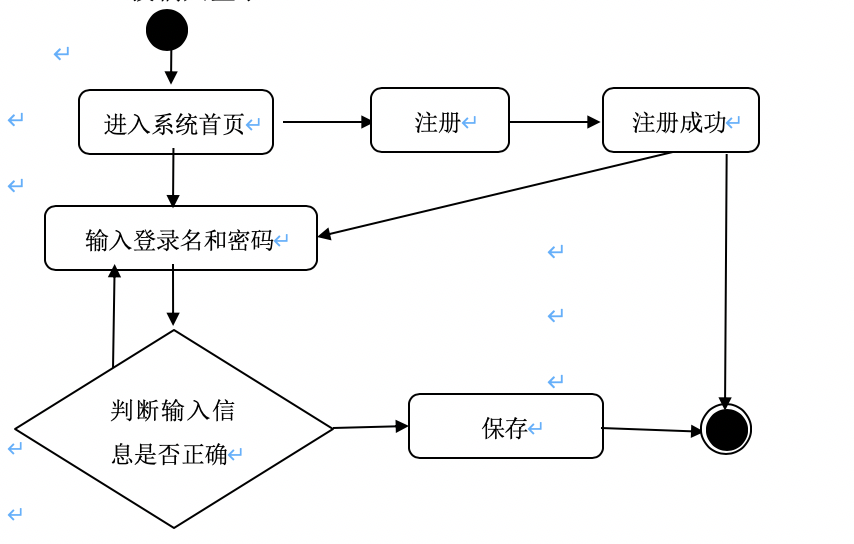
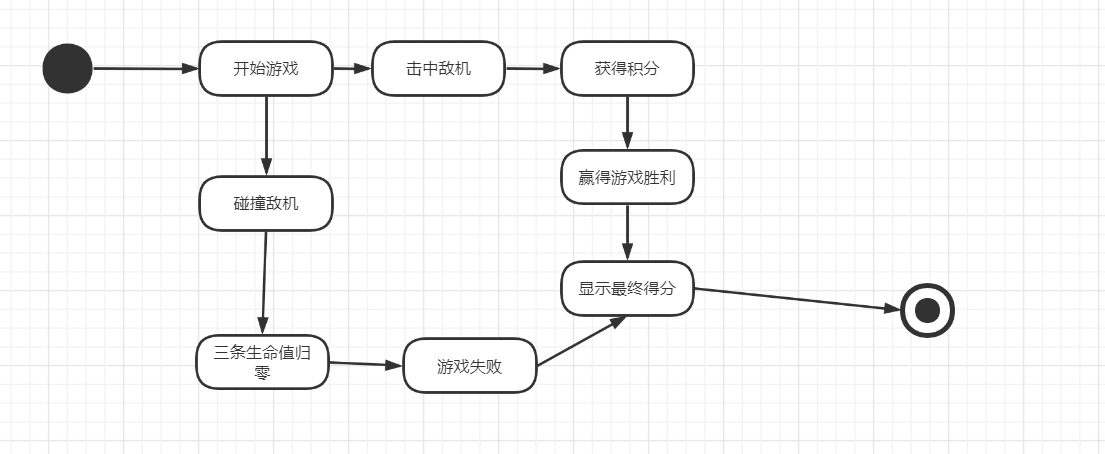


图3.1.2 玩家游戏进行活动图

## 3.2 系统功能设计

飞机大战分为四个模块，如图3.7所示。

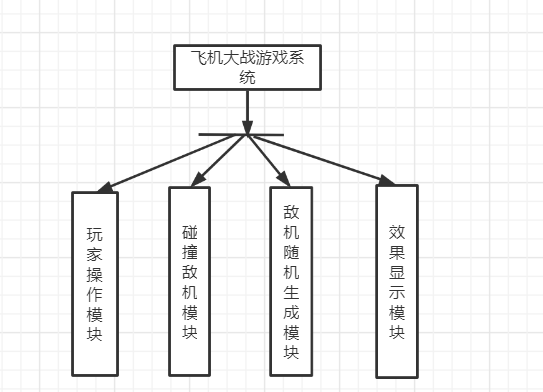


图3.2 系统功能框架图

1、玩家操作模块

玩家可以控制的自己的飞机进行移动，躲避与敌机的碰撞

2、敌机碰撞模块

当玩家的战机与敌机碰撞时候就会使我方战机生命值减一

3、敌机随机生成模块

我方战机会在敌机全部消失后自动随机生成

4、效果显示模块

当我方战机击中敌方战机或者碰撞敌方战机都会显示特殊的效果

3.2 非功能需求

根据系统模块特点和功能的不同，搭建系统前系统需要满足以下要求：

1. 用户界面：可大可小，根据自己的舒适度可以拉伸游戏界面大小。
2. 稳定性：系统的运行应当平稳，不会出现在无外部数据变化的情况下系统自行崩溃。
3. 安全性：游戏应当具有安全性校验，避免造成不必要的bug。
4. 软硬件环境：需要具有windows7或者windows8、windows10以及相关windows系统。

# 第四章 版本控制及托管

**https://www.liaoxuefeng.com/wiki/896043488029600**

学习参考资料

**本章阐述相关概念、介绍命令、写出详细过程**

4.1 本地仓库

在Git中，我们将需要进行版本控制的文件目录叫做一个仓库（repository），每个仓库可以简单理解成一个目录，这个目录里面的所有文件都通过Git来实现版本管理，Git都能跟踪并记录在该目录中发生的所有更新。

现在我们已经知道什么是repository（缩写repo）了，假如我们现在建立一个仓库（repo），那么在建立仓库的这个目录中有一个“.git”的文件夹。这个文件夹非常重要，所有的版本信息，更新记录，以及Git进行仓库管理的相关信息全部保存在这个文件夹里面。所以，不要修改/删除其中的文件，以免造成数据的丢失。

### 4.11基本命令

1.git init初始化一个Git仓库

2. git statu可以查看仓库当前状态；

3. git add <file>把修改文件添加到暂存区；

4. git commit -m "message"提交更改，把暂存区的内容提交到当前分支；

5. git diff <file>查看该文件的修改；

6. git checkout <file>撤销对文件的修改；（撤销未提交的文件）

7. git rm <file>删除本地仓库文件；

8. git log查看提交日志；

4.2 远程仓库

多人协同作战的时候，通常找一台电脑充当服务器的角色，每天24小时开机，其他每个人都从这个“服务器”仓库克隆一份到自己的电脑上，并且各自把各自的提交推送到服务器仓库里，也从服务器仓库中拉取别人的提交文件。

4.2.1 基本命令

1、git clone + 路径从远程仓库拉代码，当然首先得在本地文件夹 git init 初始化仓库

2、git pull更新代码 （需要切换到项目分支再 pull 代码）

3、git branch -a 查看远程仓库的全部分支

4、git checkout + 分支名直接切换到该远程仓库分支，git checkout -b + 分支名 + 远程分支名在本地创建一个分支并且切到某一个远程分支上

5、git branch查看当前所在的分支及本地存在的分支

6、git diff <日志节点1> <日志节点2>比较两个节点的不同

7、git status 查看本地仓库的状态

**第五章 结论**

## 5.1 总结

基于Python的飞机大战游戏是采用python语言及其pygame模块开发完成的.本项目实现了游戏流程中的所有功能,界面布局更加人性化,并有着良好的可扩展性。

对于使用git本地仓和远程仓时候会出现很多的错误信息，比如出现配置命令重复，导致远程文件夹创建失败等问题，通过不停的测试以及抓住问题的要点来解决问题，是问题得到解决

## 5.2 展望

本次课题研究是基于兴趣爱好的一次实践，兴趣爱好给予了我很大的动力，这也使我在遇到各种问题时，总能通过多种渠道找到解决方案。虽然是第一次使用Python开发游戏，在一边进行学习，一边进行游戏开发和设计过程中,遇到了很多问题，也走了不少弯路，但锻炼了自我解决问题的能力

# 参考文献

[1][挪威] Magnus Lie Hetland著 凌杰 陆禹淳 顾俊 译 《Python算法教程》 人民邮电出版社  
[2] [美]Jonathan S.harbour 著 李强译 《Python游戏编程入门》 人民邮电出版社  
[3] [瑞典] Jan Erik Solem 朱文涛/袁勇 译《Python计算机视觉编程》 人民邮电出版社  
[4] [美] Al Sweigart著 李强译《Python游戏编程快速上手》 人民邮电出版社  
[5] [美]James Payne著 张春晖 译《Python游戏编程入门》 清华大学出版社  
[6] [美]Eric Matthes 著 袁国忠 译《Python编程：从入门到实践》 人民邮电出版社

# 附录A

### Github仓库地址

[*https://github.com/pengbin1971/plane.git*](https://github.com/pengbin1971/plane.git)

[*git@github.com:pengbin1971/plane.git*](mailto:git@github.com:pengbin1971/plane.git)

### Gittee仓库地址

[*https://gitee.com/zeng-deng/fang*](https://gitee.com/zeng-deng/fang)

git@gitee.com:zeng-deng/fang.git