图片包含 游戏机, 物体, 钟表, 画

描述已自动生成

![卡通人物

描述已自动生成]()

软件工程综合实验

配置管理计划

WeChatterBot

|  |  |
| --- | --- |
| 团队编号 | E 组 |
| 团队成员 | 郭维泽 李书缘 吕江枫 |
|  | 王元玮 王云杰 张延钊 |

**Version 1.0**

版本变更历史

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 变更时间 | 主要编制人 | 备注 | 审核人 |
| 1.0 | 2020/03/23 | 吕江枫 | 初稿 | 郭维泽，李书缘，王元玮，王云杰，张延钊 |

配置管理说明

# 1. Github配置管理

## 1.1 目录结构

小组成员向GitHub的成果提交须严格遵守以后目录结构和要求。不得未经许可在根目录下创建新目录或将文件归类于错误的目录下。本组GitHub目录结构如下：

* 0-参考资料：用于存放必要的参考资料，如原项目说明文档、参考教材等
* 1-会议记录：存放每一周会议的记录文件
* 2-课堂报告：存放每周用于课堂报告的PPT文件或其他展示所用的资料
* 3-项目源码：存放WeChatterBot源代码
* 4-项目计划：存放项目计划书
* 实验1：软件需求分析：存放软件需求分析相关资料
* 实验2：软件需求评审：存放软件需求评审相关资料
* 实验3：软件产品改进与展示：存放软件产品改进与展示相关资料
* 实验4：软件测试：存放软件测试相关资料
* 实验5：软件测试评审：存放软件测试评审相关资料
* 实验6：软件进度计划与控制：存放软件进度计划与控制相关资料
* 实验7：配置管理：存放配置管理相关资料
* 实验8：工作量估计与统计分析：存放工作量估计与统计分析相关资料

## 1.2分支使用

小组成员均能够熟练的使用Git进行配置管理，在实验过程中，实验文档及其补充资料的提交由每周选定组员统一提交于master分支。

在代码开发中，每个功能一个分支，多人完成一个功能，从master分支上的基础上进行开发。在分支开发完成之后，组员提交贡献请求，由除开发者之外至少一名组员进行代码评审，合并到主分支。

## 1.3文件命名规范

1. 实验报告文件

命名规范：E\_document\_Vx.y.z.ext

例子：E\_软件需求分析说明书\_V1.0.0.docx

修订号更新规则：只做了向下兼容的修正时才递增，即针对不正确的内容进行内部修改。

次版本号更新规则：在有新功能加入或者功能被改进时递增。递增时必须整合之前所有的修订号改动，并将修订号归零。

主版本号更新规则：整合所有当前的次版本改动，并经过确认后，更新一个主版本，并将次版本号和修订号归零。

1. 补充文件

命名规范： filename.ext

例子：UML模型.mdj；第一周展示.ppt

1. 代码文件

## 1.4提交信息规范

命名规范：month.day\_document\_description

例子：3.25\_软件需求分析说明书\_完成概述部分

## 1.5文件提交

* 实验文档提交：只集中提交每周所要展示的最终版文档，中间结果一律在OneDrive处理。
* 补充文件提交：提交本周文档时一同提交与该文档相关的全部补充资料，包括课堂展示用ppt和会议记录等。
* 代码提交：个人代码提交全部提交到对应的功能分支，在功能完成开发内容和测试后通过贡献请求合并到主分支。

# 2. OneDrive配置管理

## 2.1 目录结构

小组成员向OneDrive的成果提交须严格遵守以后目录结构和要求。本组OneDrive目录结构与上文GitHub目录结构相同，用于存放对应的文档文件和补充文件。