

# CheeseBox 测试文档

---

课程: 软件工程

软件名称: CheeseBox 知识管理软件

组名: 盒子猫

## 0. 测试环境

1. Linux(Manjaro 20.2, Kernel 5.8.18), Python3.9 & Python 3.7
2. Windows 10, Python 3.9 & Python 3.7

测试框架选用PyTest, (如果有时间的话)自动测试UI的框架参考[QTest](#)

## 1. 测试范围

考虑到开发进度, 采用单元测试、集成测试以及系统测试对于软件的功能进行验证和测试, 此外不进行特别的程序性能以及其他非功能的测试, 比如程序的静态测试。

## 2. 测试策略

首先根据类图的设计, 对于功能类进行单元测试, 测试覆盖软件util和pipe包下所有的类。单元测试完成后对于各个模块进行组合, 按照UI、逻辑两个思路进行集成测试。单元测试和集成测试都使用白盒方法。最后对于整体软件系统, 按照软件需求中的涉及到的用例, 使用黑盒测试覆盖所有用例。

## 3. 单元测试

对于软件中的最小可测试单元进行验证和检查, 包括但不限于数据结构、网络通信等非UI的部分, 此外, 将软件的独立单元与软件的其余部分独立开来进行测试。

对于程序的白盒测试, 在编写程序同时进行。使用PyTest进行自动化测试, 重点测试非UI功能的实现(比如数据结构)。

## 4. 集成测试

在单元测试的基础上按照设计架构封装为的子系统集成测试, 重点测试每个子系统是否功能工作正常, 此外系统之间通信是否正常(UI相关, 测试信号与槽)。

## 5. 系统测试

考虑到需求文档里面涉及到的用例, 使用黑盒测试的方式对于整个软件系统进行测试

TODO: 按照用例设计测试报告

## 6. 测试结果

目前由于编写进度的原因, 主要测试了逻辑相关的代码, 修改了一些bug。UI相关的代码正在开发中, 需要之后进行测试。