

pems概要设计

1. 引言

1.1编写目的

本文档是PEMS——Python Ebooks Management System系统的概要设计。针对每个模块的功能需求，从技术的角度给出交互过程，描述功能实现。

文档的预期读者为：

- 系统开发人员
- 系统测试人员
- 拥有大量Ebook的用户

1.2背景

a) 项目名称：PEMS——Python Ebooks Management System

b) 本项目由西邮linux兴趣小组提出，开发者为西邮linux兴趣小组成员：冯立强，李磊，潘猛，米成刚，用户为在个人pc上拥有大量电子书并用我们编写的软件来管理它们的计算机用户。

1.3定义

PEMS: Python Ebooks Management System电子书管理系统

1.4参考资料

《python核心编程》第二版
《Dive Into Python》中文版——深入Python

2. 总体设计

2.1需求规定

2.1.1系统功能

PEMS——Python Ebooks Management System是一个用python语言编写的图书管理系统，主要的目标是对个人计算机上的电子书（如pdf，chm等格式的电子书）进行管理和阅读，系统的主要功能有

1. 搜索：

a) 搜索本地计算机的文件夹查找电子书（如pdf.chm此处格式可以自定义），后期会考虑对压缩文件内部进行搜索。提供快速搜索和详细搜索的功能，快速搜索只是按照文件后缀名进行匹配，查找所有符合条文件，详细搜索则会查看文件头，根据文件头来判断文件的格式。

b) 在已经存在的列表中查询用户需要的电子书，支持模式匹配。

2. 分类：电子书搜索之后默认按照文件格式（或者文件名）对电子书进行分类，搜索完成后，用户可以自定义分类，并按照自己的需要重新进行分类。对于分类好的电子书将会保存子列表里面，以方便用户下次访问。

3. 整理：可以将不在同一文件夹内的电子书整理到一个特定的文件夹当中。
4. 打开：使用系统默认的工具打开电子书，如pdf用Adobe reader打开。chm用Chmsee打开，可由用户选择。同时对电子书建立书签功能，方便下次访问。
5. 对列表中的文件进行管理，如复制，删除，移动等。
6. 查找重复文件，根据文件大小和文件名进行对比，选出重复的文件。在整理过程中，对比文件的大小和名字，并提示用户进行相应的操作。

2. 1. 2系统性能

2. 1. 2. 1精度

快速搜索不能确保信息的准确性，详细搜索可以确保信息的准确性。

2. 1. 2. 2时间特性要求

快速搜索要求在短时间内检索出定义的路径中的所有电子书，时间要求最多5分钟。

2. 1. 3输入输出要求

用户需要输入所要查询的文件的路径，输出搜索到的文件的信息，如：大小，创建修改时间，名称，路径，分类等。

2. 1. 4数据管理能力要求

使用配置数据文件进行数据的管理，将电子书列表等信息保存到文件中，暂时先使用文件，如果速度不能够满足需求，考虑使用ini.xml等方式，甚至数据库。

2. 1. 5其他专门要求

因为python本身是跨平台的环境，所以最终pems也会考虑跨平台的特性。前提是它可以顺利的在Linux上使用之后，才会考虑移植到Windows下。

2. 2运行环境

2. 2. 1设备

- a . 普通个人电脑，Windows或Linux操作系统。
- b. x86或x86_64及其兼容处理器，128M或以上内存
- c. 至少1M硬盘空间；
- d. 正常网络环境。
- e . 已经安装了不同格式电子书对应的阅读器

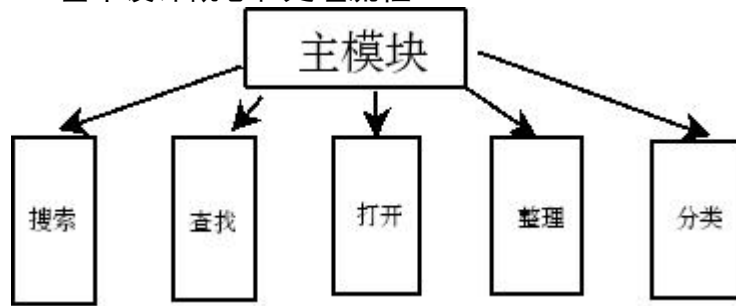
2. 2. 2支持软件

安装python环境，以及一些python需要的库。

2. 2. 3控制

本软件采用python的图形库编写的图形界面控制，尽量为用户提供最大的便利

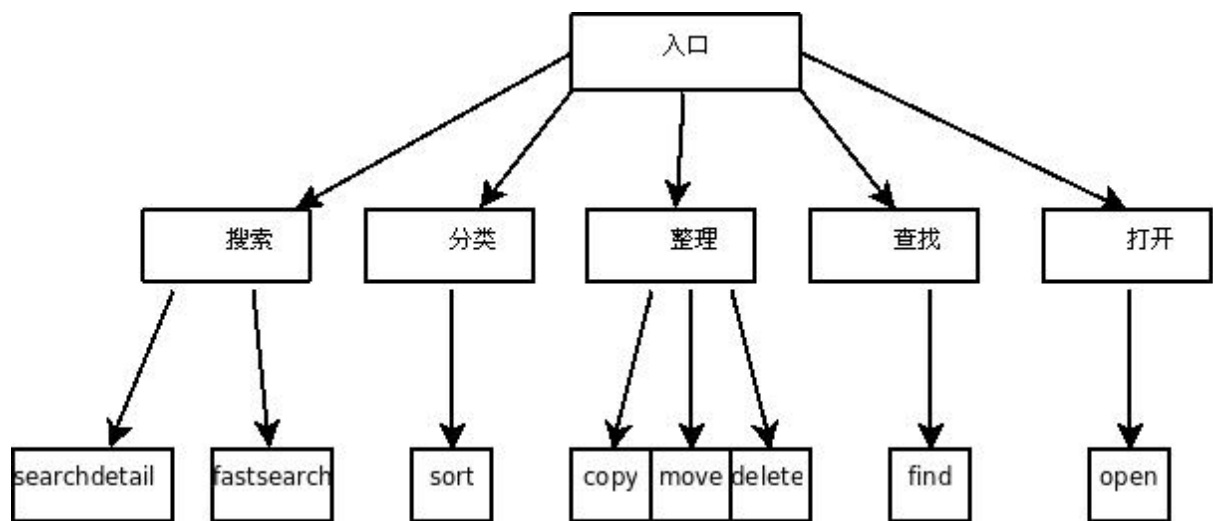
2.3基本设计概念和处理流程



处理流程：本系统初次运行时搜索配置文件目录，并根据文件来配置系统，若没有发现配置文件，会对用户发出提示，提示用户选择要搜索的文件夹，然后用户自己决定搜索方式（详细，快速）和文件类型，搜索过程中，系统会按照文件格式默认进行分类，并存储在文件中，然后用户可以自己添加分类方式进行自定义分类，并将用户选择的电子书添加到这个分类当中，用户也可以对其进行整理操作比如，复制删除移动等。打开电子书，根据用户的选择用系统默认的阅读器打开电子书。

2.4结构

2.5功能需求与系统模块的关系



	[search]	[sort]	[find]	[copy]	[move]	[delete]	[open]
[搜索]	√						
[查找]		√					
[整理]			√	√	√		
[分类]		√					
[打开]							√

2.6人工处理过程

对重复文件的选择删除或者重命名，用户必须对其进行选择性的操作。

2. 7尚未解决的问题

中文支持，界面设计，权限设置。

3. 系统数据结构设计

python字典，列表，元组

4. 系统出错处理设计

无权限，搜索或阅读该文件：

搜索不到文件：

系统中没有阅读器：提示用户安装