一、导言

1.目的

该文档的目的是描述博物馆网站数据采集子系统经过测试的总结报告，其主要内容包括：

系统自身情况测试

系统使用情况测试

本文档的预期读者是：

项目管理人员

测试人员

2.范围

该文档定义了系统测试的结果，测试了子系统数据采集、存储等基本功能，并给出了测试的结论以及解决办法。

3.测试时间及人员

本次测试的时间、地点和人员总结如下：

测试时间：2021-5-10至2021-5-16

人员：小组全体成员以及应用本组子系统的其他小组部分成员

4.参考资料

①《计算机软件文档编制规范》

② 韩万江软件文档范例

③ 孙家广学生文档范例

二、系统自身情况测试

本系统可进行Museum、Collection、Exhibition信息的爬取，通过start.py入口启动爬虫，并根据参数进行个性化爬取。

1.启动入口

为便于维护和保证信息的完整性，我们为每个网站的不同字段分别进行了爬虫文件的编写，爬虫文件数量较多，因此本系统提供了专门的启动入口便于一次性启动多个爬虫文件。

爬虫的启动采用顺序和并发执行两种方案，顺序执行速度较慢，但爬取的信息完整度高，并发执行速度较快，但容易出现网络连接超时、内存溢出等问题，不能保证信息爬取的完整度。

```
顺序执行命令：python start.py 参数
并发执行命令：scrapy crawall（不推荐）
```

2.爬取方式

针对静态和动态的网页，我们的爬虫采用两种爬取方案，静态网页使用scrapy默认的爬取方式，动态网页使用selenium模拟chrome浏览器用户的操作触发动态网页的渲染后进行爬取。

3.爬取测试

①Museum基本信息

a.Museum基本信息爬取测试（顺序执行）：

```
运行命令：python start.py Museum
```

```
运行时间：155.21秒
```

b.Museum基本信息爬取测试（并发执行）：

```
运行命令：scrapy crawlall
```

```
运行时间：16.33秒
```

②Collection藏品信息

a.Collection藏品信息爬取测试（顺序执行）：

```
运行命令：python start.py Collection
```

```
运行时间：1228.38秒
```

b.Collection藏品信息爬取测试（并发执行）：

该字段数据量较大，并发执行对数据库压力较大，容易导致time out丢失数据，且scrapy官方文档推荐使用CrawlerRunner顺序执行的方式运行多个爬虫，故不采用并发执行方案

③Exhibition展览信息

a.Exhibition展览信息爬取测试（顺序执行）：

```
运行命令：python start.py Exhibition
```

```
运行时间：283.60秒
```

b.Exhibition展览信息爬取测试（并发执行）：

该字段数据量较大，并发执行对数据库压力较大，容易导致time out丢失数据，且scrapy官方文档推荐使用CrawlerRunner顺序执行的方式运行多个爬虫，故不采用并发执行方案

4.All启动所有爬虫

①All启动所有爬虫测试（顺序执行）：

```
运行命令：python start.py All
```

```
运行时间：2341.07秒
```

②All启动所有爬虫测试（并发执行）：

该字段数据量较大，并发执行对数据库压力较大，容易导致time out丢失数据，且scrapy官方文档推荐使用CrawlerRunner顺序执行的方式运行多个爬虫，故不采用并发执行方案

5.定时启动

①windows平台：

```
此电脑 → 管理 → 任务计划程序 → 创建基本任务
创建任务定时运行入口文件即可
```

②Linux平台：

```
通过crontab创建定时任务定时执行入口文件即可
```

crontab定时任务示例：

```
#安装crontab
yum install crontab
#编辑crontab服务文件
crontab -e
#实现每月1号6:30定时执行所有爬虫脚本
30 6 1 * * cd 工作目录/first_group && python start.py All
```

三、系统使用情况测试

我们小组从其他组收到的反馈及相应处理措施如下：

1.第三小组反馈博物馆名字符串含有不必要的'\n'和'\t'等字符，造成了页面布局不协调甚至出错。（如下图）

解决办法：

使用正则表达式过滤该类字符，核心代码如下：

```
import re

class StrFilter:
    # 过滤\n,\r,\t,[xxxx]
    r1 = re.compile(u"\\n|\\r|\\[.*?]|\\t")

    @staticmethod
    def filter(src):
        res1 = re.sub(StrFilter.r1, "", str(src))
        res2 = str(''.join(res1.split()))
        res3 = res2.replace(" ", "")
        return res3

    @staticmethod
    def filter_2(src):
        return StrFilter.filter(src).replace('[', '').replace(']', '')
```

2.第三小组反馈博物馆位置含有不必要的索引标记，如下图所示：

解决办法：

使用正则表达式过滤该类字符，核心代码同反馈1

3.第三小组反馈部分博物馆开放时间有冗余字，希望删去。

解决办法：

用正则表达式过滤中文字符，只保留数字和必需字符，核心代码如下：

```
class StrFilter:
    # 过滤中文
    r2 = re.compile(r"[\u4e00-\u9fa5]|\\(.*?)|（.*?）")

    # Museum表consultationTelephone
    @staticmethod
    def filter_Telephone(src):
        res = re.sub(StrFilter.r2, "", str(src)).replace(': ', '').replace(' ', '')
        if len(res[0]) >= 1 and res[0] == ':':
            return res[1:]
        return res
```

4.第三小组反馈字段为空时返回内容不一致，有些为NULL，有些为空字符串，有些为'None'，造成如下问题：

解决办法：

假设展览信息简介少于一定字符的均为无用信息，用”暂无介绍“代替，核心代码如下。

```
exhibitionIntroduction = StrFilter.filter_2(row[3])
        if len(exhibitionIntroduction) <= 4:
            exhibitionIntroduction = "暂无介绍"
```

5.第三小组反馈返回图片链接时，掺杂了一两个无用链接。

解决办法：

把含有某些特征的链接全部转换成默认图片链接，核心代码如下：

```
exhibitionImageLink = StrFilter.filter_2(row[6])
        if len(exhibitionImageLink) <= 6 or 'None' in exhibitionImageLink:
            exhibitionImageLink = "http://bucttalk.online/first_group/default.jpg"
```

四、测试评估

本次测试执行准备充足，完成了既定目标。但由于经验以及对工具使用不熟练，因此对系统性能测试还有待提高和加强。