|  |  |  |  |
| --- | --- | --- | --- |
|  | **文 档 编 号** | **产品版本** | **密级** |
|  |  |  |
| **产品名称：** | | **共 页** |

Travo服务端详细设计说明书

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 拟制： | 申明远 |  | 日期： | 2014-02-28 |
| 审核： |  |  | 日期： | yyyy-mm-dd |
| 核准： |  |  | 日期： | yyyy-mm-dd |
| 签发： |  |  | 日期： | yyyy-mm-dd |
| **文档版本：** | | | | |

**文件修改记录**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 修改日期 | 版本 | 修改页码、章节、条款 | 修改描述 | 作者 |
| 2014-02-28 | 1.0.0 | 全部 | 初拟文档 | 申明远 |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

目录

# 引言

## 编写目的

对系统的各功能模块进行过程描述，设计功能模块的内部细节，包括算法和详细数据结构，为编写源代码提供必要的说明。

## 名词释义

* travo：项目名称
* Handler：请求的处理类
* Travel：游记
* Note：游记中的一项，包含文字或图片

## 缩略语说明

|  |  |  |
| --- | --- | --- |
| 缩略语 | 英文全名 | 中文解释 |
| sync | synchronize | 同步 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## 参考资料

*列出本文档的所有参考文献（包括：正式/非正式出版物、公司/部门文档等），格式如下：*

*作者＋书名（或杂志、文献、文档）＋出版社（或期号、卷号、公司文档编号）＋出版日期*

***模板使用说明：***

***[1]注明可选的部分，可以根据实际情况选择是否填写；如果不必说明，请务必保留相关的章节标题，同时在该可选章节的内容中填入“无”；未注名可选的，则必须描述；如果有些设计此模版中没有合适的地方填写，则补充在最后的其他栏目中***

***[2]模版中斜体字相当于撰写指南，最后文稿请将本模板中所有的斜体字部分全部删除。***

***[3]模板里并不说明设计技术和方法，而只是说明应包含哪些内容，以及如何描述、组织这些内容。***

***[4]对于移植的代码，描述可以相对简单一些，但以下这些章节不能省略：代码框架描述、第3节可以简写，第4节可以根据项目实际情况简写***

# 代码框架描述

*本部分描述系统的源代码分布框架，说明源代码存放的目录结构、各源文件的功能。总体设计中有一个开发视图，这一部分就是对总体设计中开发视图的进一步细化。*

## 目录结构

|  |  |  |
| --- | --- | --- |
| 目录名称 | 目录说明 | 包含源文件列表 |
| travo/ | 根目录，包括服务器的配置与启动，各模块请求参数和数据的基本处理 | travo.py |
| default.py |
| config.py |
| location.py |
| note.py |
| sync.py |
| travel.py |
| user.py |
| utils.py |
| travo/service | 包括各模块的服务文件，处理主要的业务逻辑和数据的存取 | \_\_init\_\_.py |
| LocationService.py |
| NoteService.py |
| SyncService.py |
| TravelService.py |
| UserService.py |

## 源文件说明

|  |  |
| --- | --- |
| 源文件名称 | 文件描述 |
| **travo/** | |
| travo.py | 定义请求与Handler之间的映射、启动服务器，是系统的主要入口 |
| default.py | 定义所有Handler的父类、常用函数及异常 |
| config.py | 系统路径、全局常量、响应码 |
| location.py | 所有Location模块的Handler |
| note.py | 所有Note模块的Handler |
| sync.py | 所有Sync模块的Handler |
| travel.py | 所有Travel模块的Handler |
| user.py | 所有User模块的Handler |
| utils.py | 常用工具类、方法 |
| **travo/service/** | |
| \_\_init\_\_.py | 数据库连接、连接池；包公共函数 |
| LocationService.py | 所有Location模块的业务逻辑函数 |
| NoteService.py | 所有Note模块的业务逻辑函数 |
| SyncService.py | 所有Sync模块的业务逻辑函数 |
| TravelService.py | 所有Travel模块的业务逻辑函数 |
| UserService.py | 所有User模块的业务逻辑函数 |

# 子模块（实体）1详细设计说明

*主要从数据结构和函数（算法）两个方面进行详细设计说明。对于详细设计，如果你觉得还有一些要素的描述可以帮助理解本设计文档，则可以增加一个或多个小节进行描述，这个由设计人员自行把握，不做强制约定。*

***注：因为详细设计的目的是为了编码，所以其设计输出应当尽可能地使后面的编码工作变得方便快捷，同时达到设计目标要求。可以说，前面的总体设计工作主要在于阐述设计思想，目的是为了使项目的审核者、维护者、编码者更好地理解设计原理，也为了他们在必要的时候更好地理解代码。而详细设计则详细具体地描述了设计目标的实现方法和步骤。所以，本节是编码的直接依据（编码的输入），是对前面总体设计工作的具体实现（设计的输出）。***

## 数据结构

*具体定义模块1相关的数据结构，并加以解释说明。*

## 处理流程详细说明

*各模块的函数之间是存在关联关系的，为了增加可读性，子模块的每个功能实现流程在这里说明。*

*(1) 功能1*

*这里描述实现这个子模块功能有几个步骤*

*(2) 功能2*

*…*

*简单的步骤可以用文字描述，复杂的必须使用流程图表示（可以使用函数间的调用关系或流程来表示）。这个要和前面的总体设计模板中的5.3.1节关联起来，是对该节的进一步细化设计和说明。*

## 4.3 编码设计

*本节须说明模块1涉及的源文件，主要函数（或称子模块、子程序）名称和说明，并用程序描述语言（如伪代码）实现主要函数*

主要函数列表格式如下：

| 主要函数名称 | 函数简要说明 |
| --- | --- |
| 文件一 | |
| 函数1 | 简要描述函数的功能 |
| 。。。，。。。 |  |
| 函数n |  |
| 文件二 | |
| 函数1 |  |
| 。。。，。。。 |  |
| 函数n |  |

4.3.1 函数1

【功 能】

*简要描述函数的功能*

【参数】

*列出此函数的各输入项，说明其类型、名称、含义、取值范围等*

【返回值】

*列出此函数的各输出项，说明其类型、名称、含义、取值范围等*

【算 法】

*用程序描述语言描述此函数的处理过程、算法逻辑。对于逻辑比较简单的函数，可以概括地说一下算法，或者略去此项。*

***对于数据结构访问需要上锁、关中断等操作必须在这里说明。***

***需要提供调试内容的，需要在这里说明。***

【使用说明】

*说明该函数的外部定义包含在哪个头文件里。*

*说明调用此函数时的注意事项，以及设计者认为应当交待的其它事项*。

4.3.2 函数2

。。。。

4.3.n 函数n

。。。。

5 子模块（实体）2详细设计说明

*用类似方式，说明第2个模块乃至第N个模块的设计考虑。*

6 附录 子系统定义表

*如果本详细设计所对应的最后实现代码并未组织成一个子系统的话，就不需要填下面的表格。*

填表说明：

|  |  |
| --- | --- |
| 表格项 | 填 写 说 明 |
| Name | 该子系统的名字 |
| Class | 该子系统的类别 |
| Version | 该子系统自身的版本信息； |
| Init routine | 该子系统初始化函数 |
| Dependance | 该子系统的依赖信息 |
| Sequence | 该子系统的启动顺序要求 |

模块子系统信息表

|  |  |  |  |
| --- | --- | --- | --- |
| Name |  | Class |  |
| Version |  | Init Routine |  |
| Dependance |  | | |
| Sequence |  | | |