

# Pet's Caregiver 宠物信息管理系统

## 详细设计规约文档

编写日期	SE PG	版本	说明	作者
2022.12.13		1.0	完成基础设计类规约。	王方远，黄绍华
2022.12.21		1.1	根据老师的意见，详细改善了类接口调用的规范和类状态图的构建。	王方远，黄绍华

评审时间	评审参与人员	评审后修改批准日期	确认签字人员
2022.12.28	王方远，黄绍华	2022.12.31	王方远，黄绍华

## 目录

1. 引言.....	2
1.1 编写目的.....	2
1.2 背景及依据.....	2
1.3 参考资料.....	2
2. 系统软件结构.....	2
3. 微服务设计.....	3
4. 类设计.....	4
4.1 登录注册微服务.....	4
4.2 系统设置微服务.....	9
4.3 预约管理微服务.....	15
4.4 宠物信息管理微服务.....	21
4.5 宠物日志管理微服务.....	24
4.6 宠物健康史管理微服务.....	27
4.7 健康信息微服务.....	30

## 1. 引言

### 1.1 编写目的

本文档是用于细化 Pet's Caregiver 平台系统的详细设计的，我们将通过微服务设计、接口设计、类详细设计三个角度规范平台开发中的编码方向、不同微服务之间的接口调用以及方法设计等。

### 1.2 背景及依据

本项目名为 Pet's Caregiver，旨在为 PC 端的用户以及宠物医疗机构内的宠物医生和管理员提供一个体验良好，灵活性强，交互性好的轻量级宠物健康管理、医疗预约、信息管理的应用，用户可以通过该网站完成与宠物健康、医疗相关的活动，并通过其他相关的功能获得更好的体验。

本项目任务的提出者为软件工程课程设计需要；开发者为软件工程专业两名同学（王方远、黄绍华）；项目预期用户是宠物饲主、宠物医院的医生和管理员。

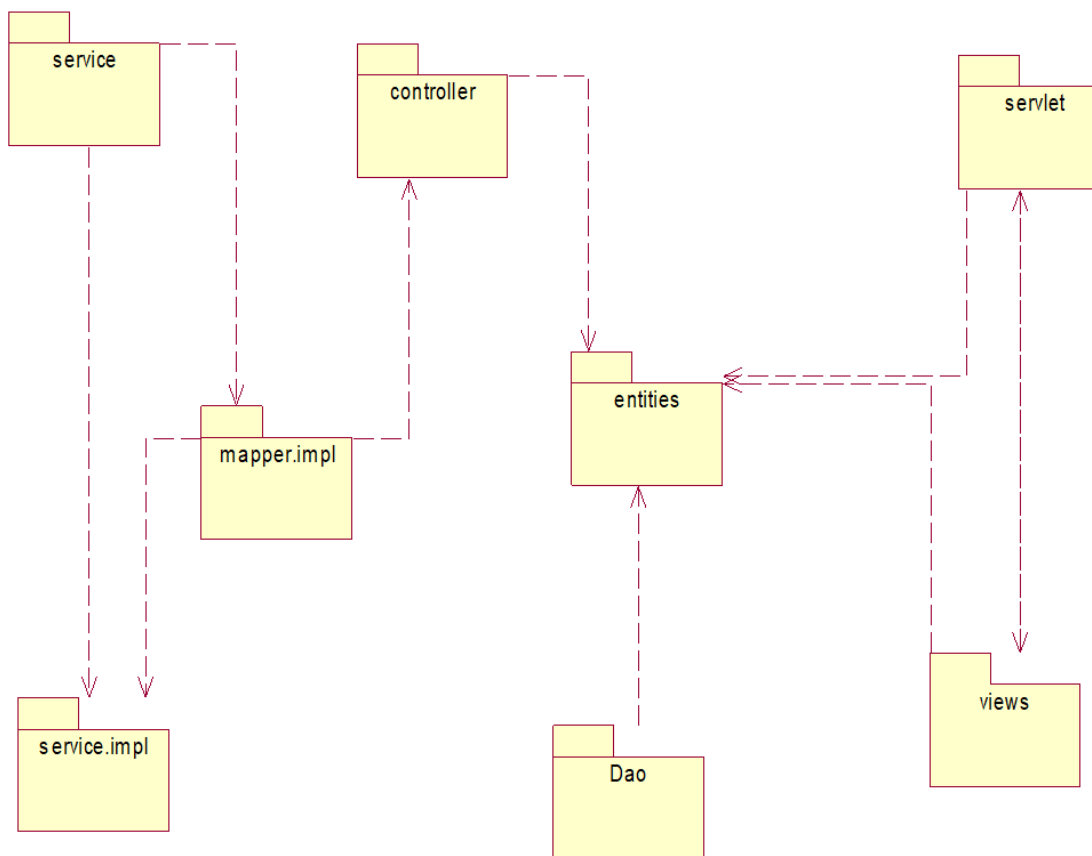
为了更好的实现所需要的业务逻辑，本系统调用了其他系统的接口以更好的完成本系统的功能，包括 e-chart 的图表接口、JQuery 接口以及 Shiro 接口等

### 1.3 参考资料

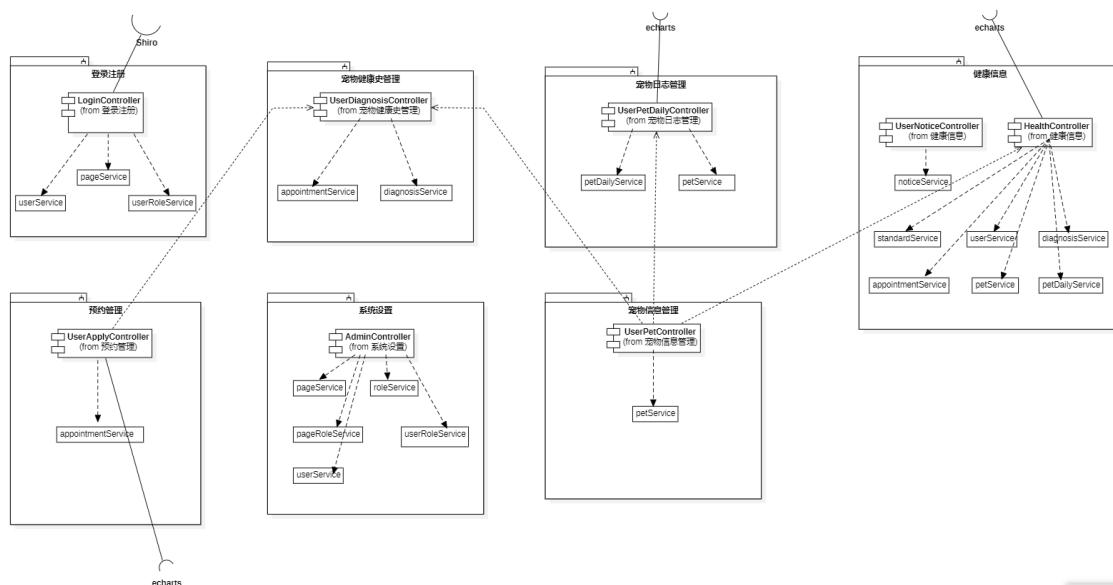
1. 《SpringBoot 更好的开发》
2. 《HTML5+CSS3 从入门到精通》
3. 《更好的软件架构，更好的设计》
4. 《Design Patterns》
5. 《Clean Architecture》
6. 《Clean Code》

## 2. 系统软件结构

体系结构包图：



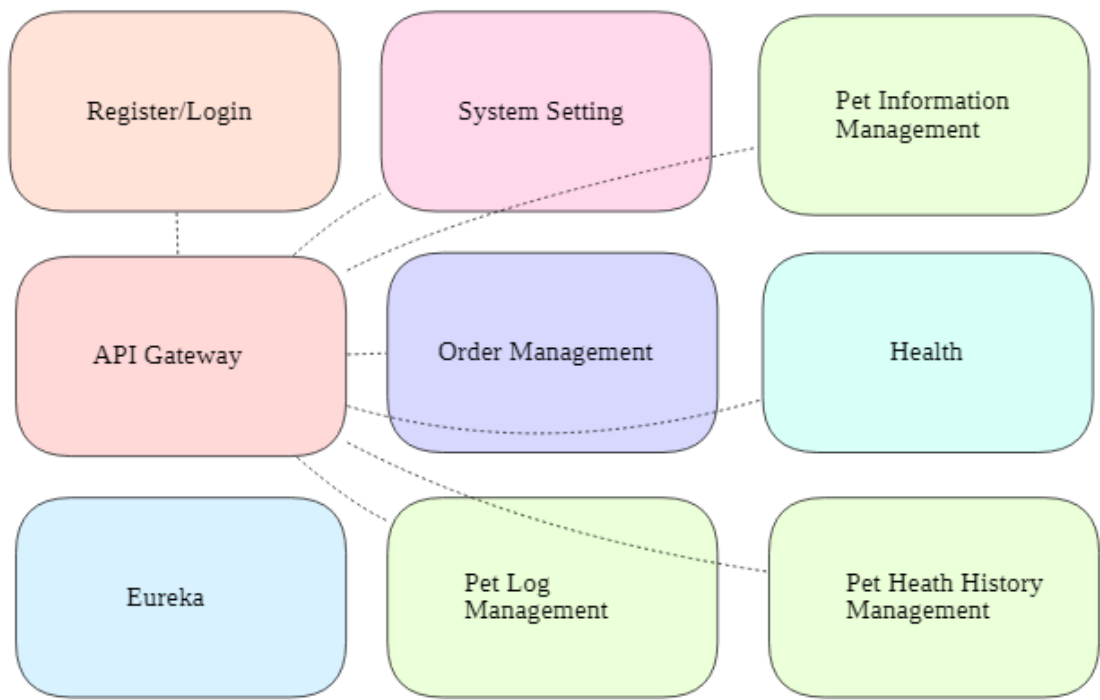
软件架构设计：



### 3. 微服务设计

平台在微服务划分上一共分为 7 个微服务和一个 API Gateway、一个 Eureka 服务发现与注册中心。API Gateway 的功能是提供统一的后端访问入口，Eureka 的作用是提供统一的服务 IP 地址管理。其余微服务的功能将在接下来详细类设计中说明。

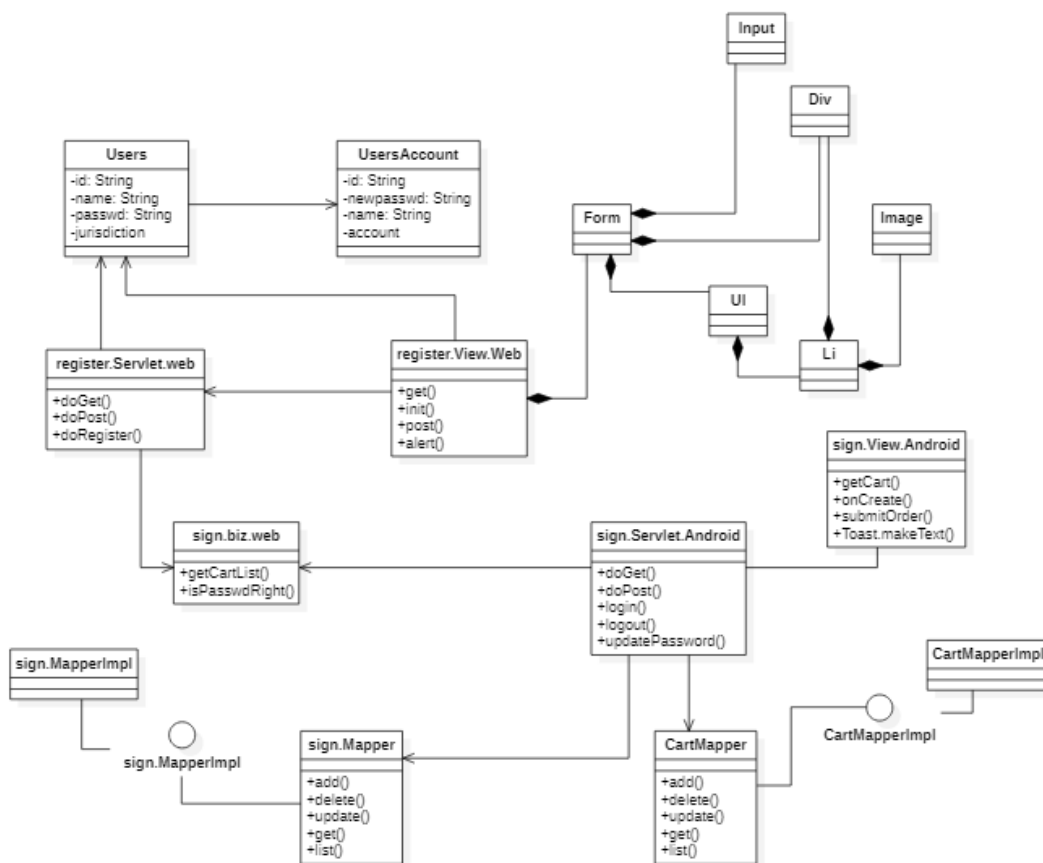
以下是基于微服务的系统架构划分图：



## 4. 类设计

### 4.1 登录注册微服务

在登录注册微服务中，一共有两个实体类，Users 与 UsersAccount，用来管理用户实体记录以及账户状态的管理。同时通过控制类 Servlet 来完成用户登陆、注册以及登陆标识管理的工作，用 View 完成页面初始化。控制类连接边界接口类 Mapper 来完成增删改查操作。如下图所示：



## 4.1.1 Users 类

### 4.1.1.1 功能

登录注册模块中的 **User** 类主要负责记录用户的登录账号（在宠物管理系统中是用户名或者手机号码）和密码。并且通过内部的方法实现检查用户登录账号和密码是否匹配，并且通过修改密码方法管理自身密码。

### 4.1.1.2 性能

无约束

### 4.1.1.3 输入项

该类通过用户的手机号、用户名以及密码进行初始化。修改密码时，需要提供新、旧两个密码进行验证。

### 4.1.1.4 输出项

检查密码时，只会返回“正确”或“错误”进行反馈（布尔值）。其余情况下没有输出项。

### 4.1.1.5 属性描述

属性名	类型	说明
id	String	用户的电话号码。接受该参数初始化，之后不会再改变。初始化时会检查电话号码是否已经被注册。
passwd	String	用户的密码
name	String	用户的用户名。不允许超过 20 个字符。
jurisdiction	String	权限，分为普通用户、医生、超级管理员

#### 4.1.1.6 限制条件及出错处理

- 用户电话号码需要符合中国真实电话号码的格式，已注册的号码不能重复注册。

### 4.1.2 register.Servlet.Web 类

#### 4.1.2.1 功能

管理用户的注册行为，提供用户通过手机号码、用户昵称和密码注册的功能。同时还会通过帮助类获取全局唯一的用户标识符（ID）赋予注册成功的用户。

#### 4.1.2.2 性能

允许并发注册

#### 4.1.2.3 输入项

1. 该类维护了一个用户列表，通过接受用户的手机号码、用户名、地址和密码来进行注册；
2. 同时该类还可以单独通过手机号码来判断用户是否已经注册；
3. 用户输入密码可以在该类中检查是否符合要求。

#### 4.1.2.4 输出项

1. 用户注册是否成功，布尔值；
2. 该手机号是否已经注册系统中的用户，布尔值；

#### 4.1.2.5 方法详细描述

##### 4.1.2.5.1 doRegister

方法签名：public ResultMap doRegist(User user)

功能说明：通过参数用户手机号码、用户昵称、用户地址和密码来进行注册。

接口调用：

- 调用自身 `getUserByPhoneAndName` 方法，检查想要注册账号的用户手机号是否是新注册的，若不是则注册失败。

详细说明：

返回类型	ResultMap(写好的一个 Map 类型的数据格式)	
返回值	用户是否注册成功, "该手机号已注册!"表示手机已被注册, "注册失败"表示注册因不明原因失败, 需要查看日志, "注册成功"表示注册成功	
方法限定符	public	
参数列表 (参数名 / 类型 / 说明)		
name	String	用户的用户名
passwd	String	用户的密码
phone	String	用户的手机号码
addres	String	用户地址

### 4.1.3 sign.Servlet.Android 类

#### 4.1.3.1 功能

用于用户登录时检查用户账号和密码是否正确的类。还能通过用户的 ID 查看用户的登录情况。

#### 4.1.3.2 性能

允许并发登录

#### 4.1.3.3 输入项

1. 接受用户的用户名和密码，登录用户；

2. 接受用户的 ID，查询用户当前的登录情况。

#### 4.1.3.4 输出项

1. 输出用户是否登录成功；
2. 输出用户的登录状态。

#### 4.1.3.5 方法详细描述

##### 4.1.3.5.1 Login

方法签名：public ResultMap login(String username, String password)

功能说明：通过参数用户昵称和密码来进行登录。

接口调用：

- 调用自身 `getUserByName` 方法，检查密码与用户名是否符合。

详细说明：

返回类型	ResultMap(写好的一个 Map 类型的数据格式)	
返回值	用户是否登录成功，若登陆成功调用 index 显示主页，若登陆失败返回"用户名或密码错误"	
方法限定符	public	
参数列表（参数名 / 类型 / 说明）		
username	String	用户的用户名
password	String	用户的密码

##### 4.1.3.5.2 Logout

方法签名：public String logout()

功能说明：退出登录，返回登陆页面

接口调用：

- 调用自身 `SecurityUtils.getSubject()` 方法，获取当前帐号，并返回登陆界面

详细说明：



返回类型	String	
返回值	“退出登录”： 在日志中输出	
方法限定符	public	
参数列表（参数名 / 类型 / 说明）		
无		

#### 4.1.3.5.3 updatePassword

方法签名：public String updatePassword(String oldpassword, String newpassword)

功能说明：通过参数用户旧密码与新密码来进行密码修改。

接口调用：

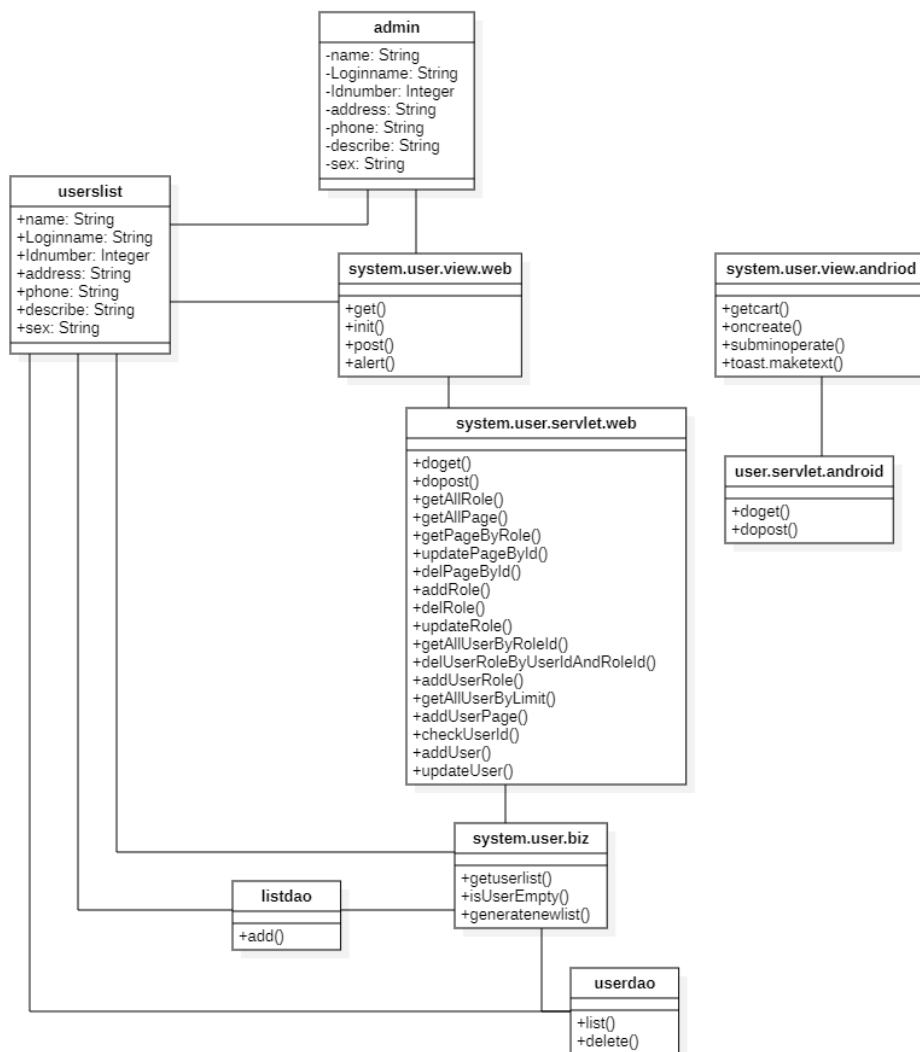
- 调用自身 getOldPasswordByName 方法，检查旧密码是否符合。

详细说明：

返回类型	String	
返回值	用户是否更改密码成功，若成功返回“密码更改成功”，若失败返回“密码更改失败”或“旧密码错误”	
方法限定符	public	
参数列表（参数名 / 类型 / 说明）		
oldpassword	String	用户的旧密码
newpassword	String	用户的新密码

## 4.2 系统设置微服务

在系统设置微服务中，一共有两个实体类，admin 与 userslist，用来管理管理员实体记录以及用户列表。同时通过控制类 Servlet 来完成用户管理、页面管理、角色管理的工作，用 View 完成页面初始化以及发布获取功能。dao 作为数据层直接对数据库操作。如下图所示：



## 4.2.1 Admin 类

### 4.2.1.1 功能

系统设置微服务中的 Admin 类主要负责记录管理员的信息。

### 4.2.1.2 性能

无约束

### 4.2.1.3 属性描述

属性名	类型	说明
name	String	管理员名称

Loginname	String	管理员登录用户名
Idnumber	Integer	管理员 ID
address	String	地址
phone	String	电话号码
describe	String	描述
sex	String	性别

## 4.2.2 Userslist 类

### 4.2.2.1 功能

系统设置微服务中的 userslist 类主要负责记录用户列表的信息。

### 4.2.2.2 性能

无约束

### 4.2.2.3 属性描述

属性名	类型	说明
name	String	用户名称
Loginname	String	用户登录用户名
Idnumber	Integer	用户 ID
address	String	用户地址
phone	String	用户电话号码
describe	String	用户描述
sex	String	用户性别

4.2.2.4 限制条件及出错处理

需要遵循数据类型规定

4.2.3 system.user.servlet.web 类

4.2.3.1 功能

在系统设置微服务中执行用户管理、页面管理、角色管理相关功能，可以获取所有用户，获取所有页面，获取所有角色，并对上述的数据进行增删改查以及更新列表操作。

4.2.3.2 性能

无约束

4.2.3.3 输入项

- 1. 接受管理员的菜单选择操作，查询对应的列表信息
- 2. 接受管理员的增删改查操作，更新列表信息

4.2.3.4 输出项

- 1. 查询的列表信息展示
- 2. 更新的列表信息展示

4.2.3.5 方法详细描述

4.2.3.5.1 getAllRole

方法签名：public List<Role> getAllRole()

功能说明：获取所有权限信息

接口调用：

- 调用自身 roleService.getAllRole()方法，获取所有权限信息

详细说明：

返回类型	List<Role>
返回值	返回所有权限的一个 list
方法限定符	public
参数列表（参数名 / 类型 / 说明）	

无		
---	--	--

#### 4.2.3.5.2 getAllPage

方法签名：public List<Page> getAllPage()

功能说明：获取所有页面

接口调用：

- 调用自身 pageService.getAllPage()方法，获取所有页面信息

详细说明：

返回类型	List<Page>	
返回值	返回所有页面的一个 list	
方法限定符	public	
参数列表（参数名 / 类型 / 说明）		
无		

#### 4.2.3.5.3 addRole

方法签名：public String addRole(String name)

功能说明：增加角色

接口调用：

- 调用自身 roleService.addRole(name)方法，增加角色

详细说明：

返回类型	String	
返回值	增加成功，返回“增加成功”；失败则返回“增加失败”	
方法限定符	public	
参数列表（参数名 / 类型 / 说明）		
name	String	角色名

#### 4.2.3.5.4 addPageRoleByRoleId

方法签名: public String addPageRoleByRoleId(Integer roleId, Integer[] pageIds)

功能说明: 增加不同角色的页面

接口调用:

- 调用自身 pageRoleService.addPageRoles(roleId, pageIds)方法, 增加该角色页面

详细说明:

返回类型	String	
返回值	增加成功，返回“Success”；失败则返回“Error”	
方法限定符	public	
参数列表（参数名 / 类型 / 说明）		
roleId	Integer	角色 id
pageIds	Integer[]	要增加的页面 id 数组

#### 4.2.3.5.5 getAllUserByRoleId

方法签名: public Object getAllUserByRoleId(Integer roleId, Integer page, Integer limit)

功能说明: 得到某角色下的所有用户

接口调用:

- 调用自身 userService.getAllUserByRoleId(roleId, page, limit)方法, 返回用户列表

详细说明:

返回类型	Object
返回值	返回用户列表
方法限定符	public
参数列表（参数名 / 类型 / 说明）	

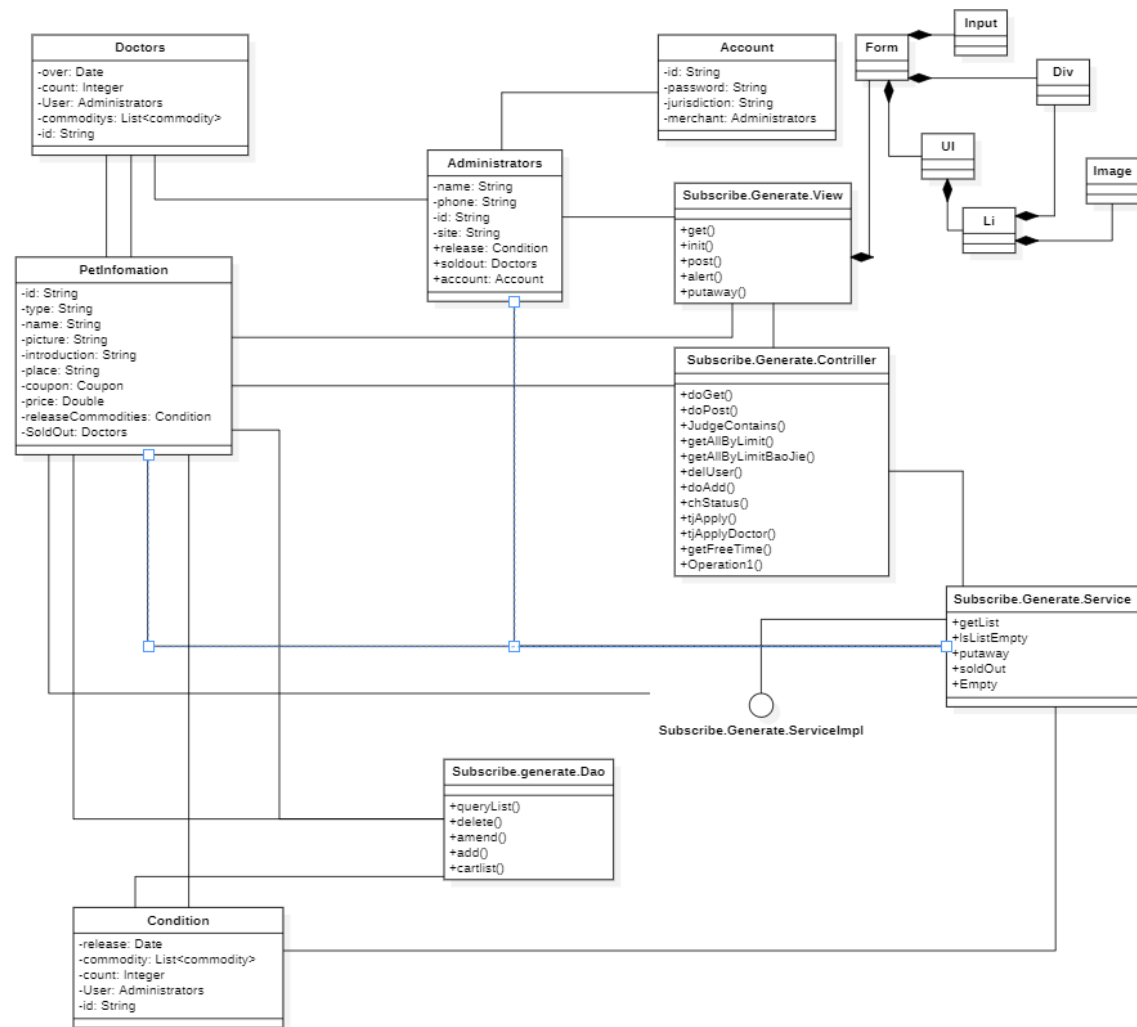
roleId	Integer	角色 id
page	Integer	页数
limit	Integer	每页的限制数量

#### 4.2.3.6 限制条件及出错处理

所有数据需要符合类型与格式规定。数据项不能为空，否则抛出格式异常。

### 4.3 预约管理微服务

在预约管理微服务中，一共有五个实体类，Doctors、Account、PetInfomation、Administrator、Condition，分别用来管理医生实体记录以及登录状态、账户状态、宠物信息、管理员实体记录、状态列表。同时通过控制类 Subscribe.Generate.Controller 来完成具体预约管理的工作。控制类以及服务 Service 连接一个边界接口类 Impl 来实现服务。如下图所示：



## 4.3.1 Doctors 类

### 4.3.1.1 功能

预约管理微服务中的 Doctors 类主要负责记录医生的登录信息。

### 4.3.1.2 性能

无约束

### 4.3.1.3 属性描述

属性名	类型	说明
id	String	宠物 ID
type	String	宠物种类



name	Integer	宠物名
picture	String	图片
introduction	String	介绍
address	String	地址

## 4.3.2 PetInformation 类

### 4.3.2.1 功能

预约管理微服务中的 PetInformation 类主要负责宠物列表信息。

### 4.3.2.2 性能

无约束

### 4.3.2.3 属性描述

属性名	类型	说明
over	String	更新日期
Doctorname	String	医生登录用户名
count	Integer	医生编号
id	String	医生 ID

## 4.3.3 Administrators 类

### 4.3.3.1 功能

预约管理微服务中的 Administrators 类主要负责记录管理员的信息。

### 4.3.3.2 性能

无约束

### 4.3.3.3 属性描述

属性名	类型	说明
name	String	管理员名称
soldout	Doctors	已预约医生列表
id	Integer	管理员 ID
address	String	地址
phone	String	电话号码
release	Condition	发布状态
account	Account	账户信息

#### 4.3.4 Subscribe.Generate.Controller 类

##### 4.3.4.1 功能

在预约管理微服务中执行预约查询，医生时间查询，预约增删改查相关功能。

##### 4.3.4.2 性能

响应时间<100ms

##### 4.3.4.3 输入项

1. 用户的操作
2. 超级管理员的操作

##### 4.3.4.4 输出项

1. 预约列表的更新
2. 医生时间表的更新

##### 4.3.4.5 方法详细描述

###### 4.3.4.5.1 getAllByLimit

方法签名：public Object getAllByLimit(Appointment appointment)

功能说明：得到该预约的所有信息

接口调用：

- 调用自身 `appointmentService.getAllByLimit(appointment)` 方法，返回预约信息

详细说明：

返回类型	Object	
返回值	返回预约信息	
方法限定符	public	
参数列表（参数名 / 类型 / 说明）		
appointment	Appointment	预约 id, 用户 id

#### 4.3.4.5.2 tjApply

方法签名：`public String tjApply(Model model)`

功能说明：普通用户预约统计

接口调用：

- 调用自身 `appointmentService.getAllByLimit(appointment)` 方法，返回预约信息

详细说明：

返回类型	String	
返回值	返回预约信息	
方法限定符	public	
参数列表（参数名 / 类型 / 说明）		
model	Model	当前用户的相关信息

#### 4.3.4.5.3 tjApplyDoctor

方法签名：`public String tjApplyDoctor(Model model)`

功能说明：普通用户预约统计

接口调用：

- 调用自身 `appointmentService.getAllByLimit(appointment)` 方法，返回预约信息

详细说明：

返回类型	String	
返回值	返回预约信息	
方法限定符	public	
参数列表（参数名 / 类型 / 说明）		
model	Model	当前医生用户的相关信息

#### 4.3.4.5.4 getFreeTime

方法签名：`public Object getFreeTime(Long id, String date)`

功能说明：用户查看医生空闲时间

接口调用：

- 调用自身 `appointmentService.getFreeTimeById(id, date+MyUtils.START_HOUR)` 方法，返回医生空闲时间

详细说明：

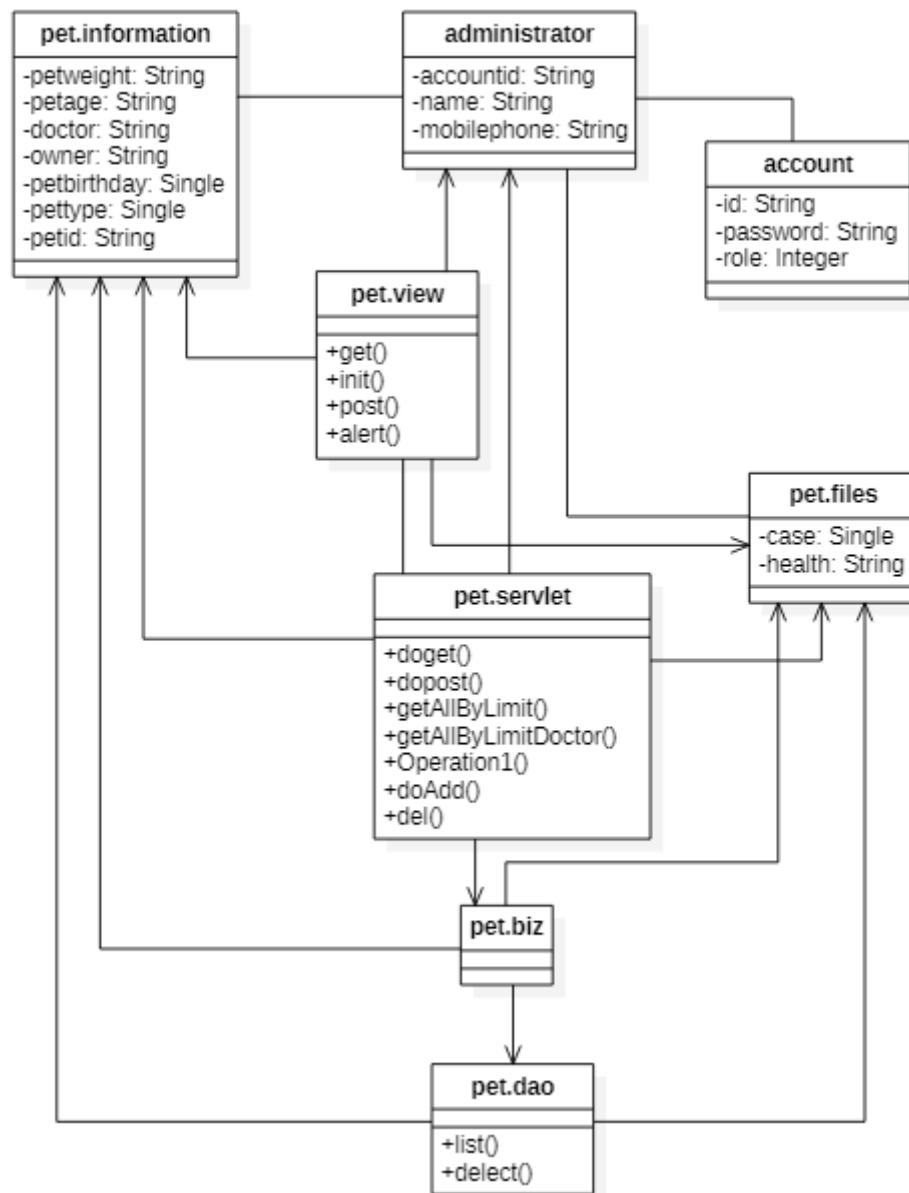
返回类型	Object	
返回值	返回空闲时间，包括日期、时分秒	
方法限定符	public	
参数列表（参数名 / 类型 / 说明）		
id	Long	医生 id
date	String	当前日期

#### 4.3.4.6 限制条件及出错处理

预约的数据格式要正确。

## 4.4 宠物信息管理微服务

在预约管理微服务中，一共有四个实体类，files、Account、PetInformation、Administrator，分别用来管理宠物档案、账户状态、宠物信息、管理员实体记录。同时通过控制类 pet.servlet 来完成具体宠物信息管理工作。dao 作为数据层直接对数据库操作。如下图所示：



### 4.4.1 petInformation 类

#### 4.4.1.1 功能

宠物信息管理微服务中的 PetInformation 类主要负责宠物列表信息。

4.4.1.2 性能

无约束

4.4.1.3 属性描述

属性名	类型	说明
petweight	String	更新日期
petage	String	年龄
petdoctor	String	宠物医生
owner	String	饲主
petbirthday	Single	宠物生日
pettype	Single	宠物种类
petid	String	宠物 ID
petheight	String	宠物身高

4.4.2 pet.servlet 类

4.4.2.1 功能

在宠物信息管理微服务中执行宠物信息管理，列表操作相关功能。

4.4.2.2 性能

无约束

4.4.2.3 输入项

1. 读取用户操作，包括新增，删除，编辑

4.4.2.4 输出项

1. 返回更新的列表

4.4.2.5 方法详细描述

4.4.2.5.1 getAllByLimit

方法签名：public Object getAllByLimit(Pet pojo)

功能说明：用户查询宠物信息

接口调用：

- 调用自身 petService.getAllByLimit 方法，返回查询的宠物信息

详细说明：

返回类型	Object	
返回值	返回宠物信息	
方法限定符	public	
参数列表（参数名 / 类型 / 说明）		
pojo	Pet	宠物部分信息 如名称等

4.4.2.5.2 doAdd

方法签名：public String doAdd(Pet pojo)

功能说明：将添加宠物加入数据库中

接口调用：

- 调用自身 petService.add(pojo)方法，插入宠物信息

详细说明：

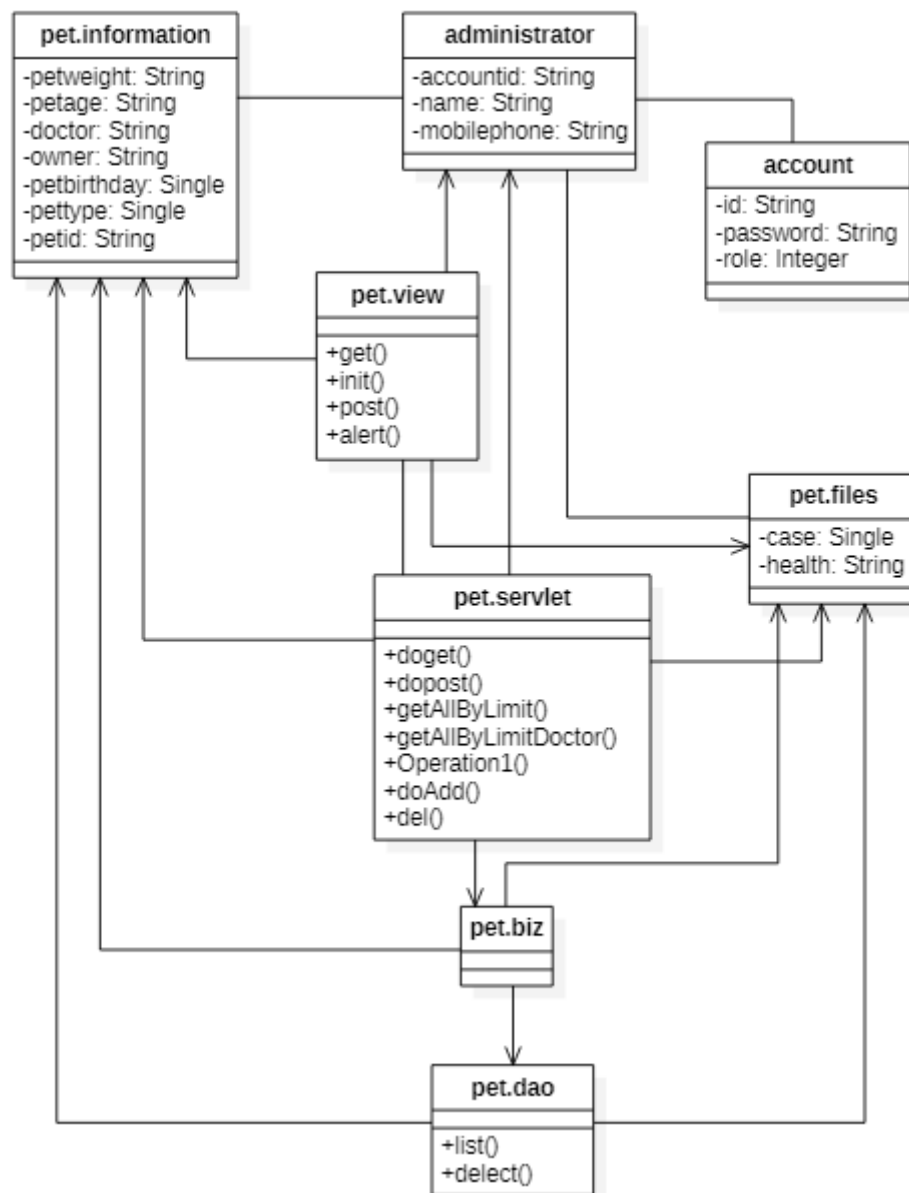
返回类型	String	
返回值	返回插入成功“Success”或失败“Failed”信息	
方法限定符	public	
参数列表（参数名 / 类型 / 说明）		
pojo	Pet	宠物所有信息

#### 4.4.2.6 限制条件及出错处理

列表数据格式需要符合类型

### 4.5 宠物日志管理微服务

在预约管理微服务中，一共有四个实体类，files、Account、PetInfomation、Administrator，分别用来管理宠物档案、账户状态、宠物信息、管理员实体记录。同时通过控制类 pet.servlet 来完成具体宠物日志管理的工作。dao 作为数据层直接对数据库操作。如下图所示：



#### 4.5.1 petInformation 类



### 4.5.1.1 功能

宠物信息管理微服务中的 PetInformation 类主要负责宠物列表信息。

### 4.5.1.2 性能

无约束

### 4.5.1.3 属性描述

属性名	类型	说明
petweight	String	更新日期
petage	String	年龄
petdoctor	String	宠物医生
owner	String	饲主
petbirthday	Single	宠物生日
pettype	Single	宠物种类
petid	String	宠物 ID
petheight	String	宠物身高

## 4.5.2 pet.servlet 类

### 4.5.2.1 功能

在宠物信息管理微服务中执行宠物信息管理，列表操作相关功能。

### 4.5.2.2 性能

无约束

### 4.5.2.3 输入项

1. 读取用户操作，包括新增，删除，编辑

### 4.5.2.4 输出项

1. 输出更新的列表

4.5.2.5 方法详细描述

4.5.2.5.1 getAllByLimit

方法签名：public Object getAllByLimit(PetDaily pojo)

功能说明：用户查询宠物日志信息

接口调用：

- 调用自身 petDailyService.getAllByLimit 方法，返回查询的日志信息

详细说明：

返回类型	Object	
返回值	返回宠物信息	
方法限定符	public	
参数列表（参数名 / 类型 / 说明）		
pojo	PetDaily	宠物日志部分信息如名称等

4.5.2.5.2 doAdd

方法签名：public String doAdd(PetDaily pojo)

功能说明：将添加宠物日志加入数据库中

接口调用：

- 调用自身 petDailyService.add(pojo)方法，插入宠物日志信息

详细说明：

返回类型	String
返回值	返回插入成功“Success”或失败“Failed”信息
方法限定符	public
参数列表（参数名 / 类型 / 说明）	

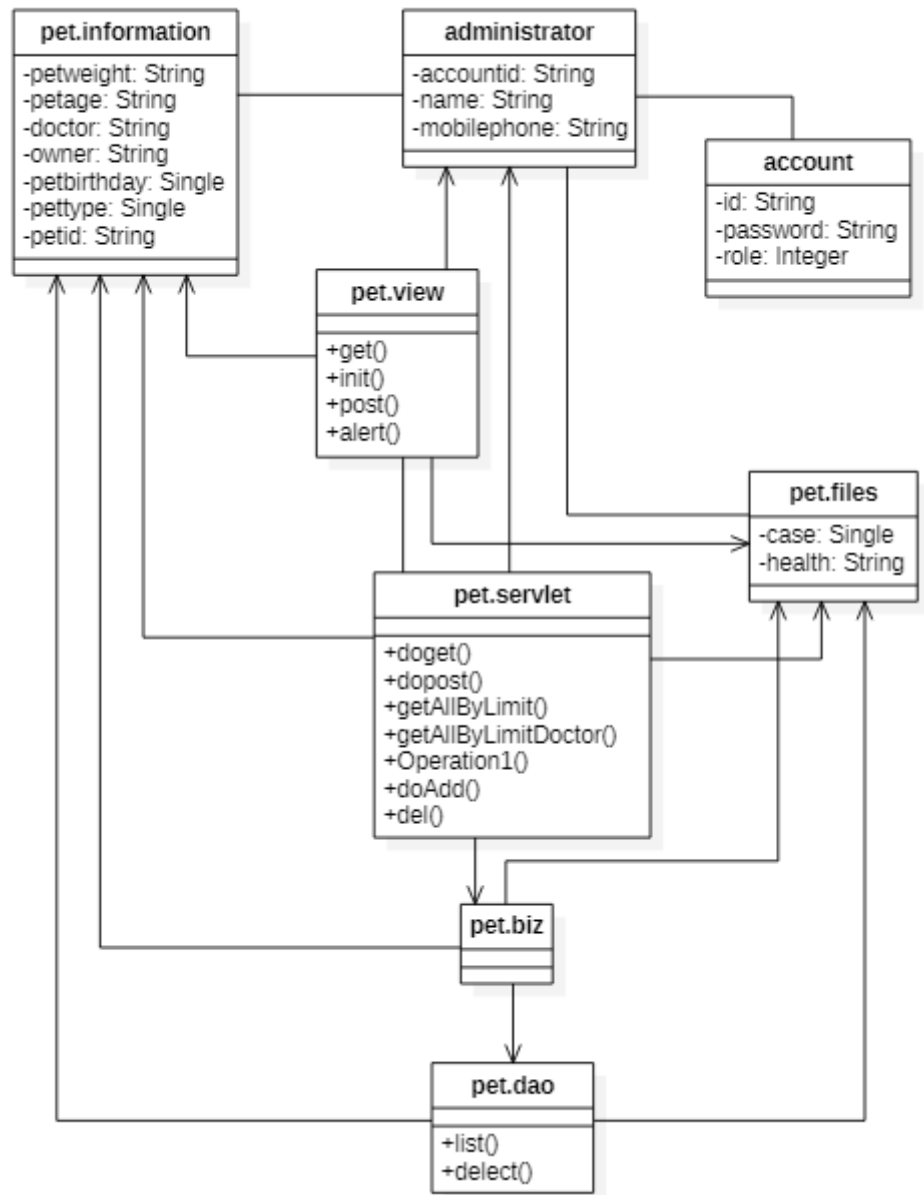
pojo	PetDaily	宠物日志所有信息
------	----------	----------

#### 4.5.2.6 限制条件及出错处理

列表数据格式需要符合类型

### 4.6 宠物健康史管理微服务

在预约管理微服务中，一共有四个实体类，files、Account、PetInforation、Administrator，分别用来管理宠物档案、账户状态、宠物信息、管理员实体记录。同时通过控制类 pet.servlet 来完成具体宠物健康史管理的工作。dao 作为数据层直接对数据库操作。如下图所示：



## 4.6.1 petInformation 类

### 4.6.1.1 功能

宠物信息管理微服务中的 PetInformation 类主要负责宠物列表信息。

### 4.6.1.2 性能

无约束

### 4.6.1.3 属性描述

属性名	类型	说明
petweight	String	更新日期
petage	String	年龄
petdoctor	String	宠物医生
owner	String	饲主
petbirthday	Single	宠物生日
pettype	Single	宠物种类
petid	String	宠物 ID
petheight	String	宠物身高

## 4.6.2 pet.servlet 类

### 4.6.2.1 功能

在宠物信息管理微服务中执行宠物信息管理，列表操作相关功能。

### 4.6.2.2 性能

无约束

### 4.6.2.3 输入项

1. 读取用户操作，包括新增，删除，编辑

#### 4.6.2.4 输出项

1. 输出更新的列表

#### 4.6.2.5 方法详细描述

##### 4.6.2.5.1 getAllByLimit

方法签名：public Object getAllByLimit(Diagnosis diagnosis)

功能说明：用户查询宠物健康史信息

接口调用：

- 调用自身 diagnosisService.getAllByLimit 方法，返回查询的健康史信息

详细说明：

返回类型	Object	
返回值	返回宠物健康史信息	
方法限定符	public	
参数列表（参数名 / 类型 / 说明）		
diagnosis	Diagnosis	宠物健康史部分信息如名称等

##### 4.6.2.5.2 doAdd

方法签名：public String doAdd(Diagnosis diagnosis)

功能说明：将添加宠物健康史加入数据库中

接口调用：

- 调用自身 diagnosisService.add(diagnosis)方法，插入宠物健康史信息

详细说明：

返回类型	String	
返回值	返回插入成功“Success”或失败“Failed”信息	
方法限定符	public	

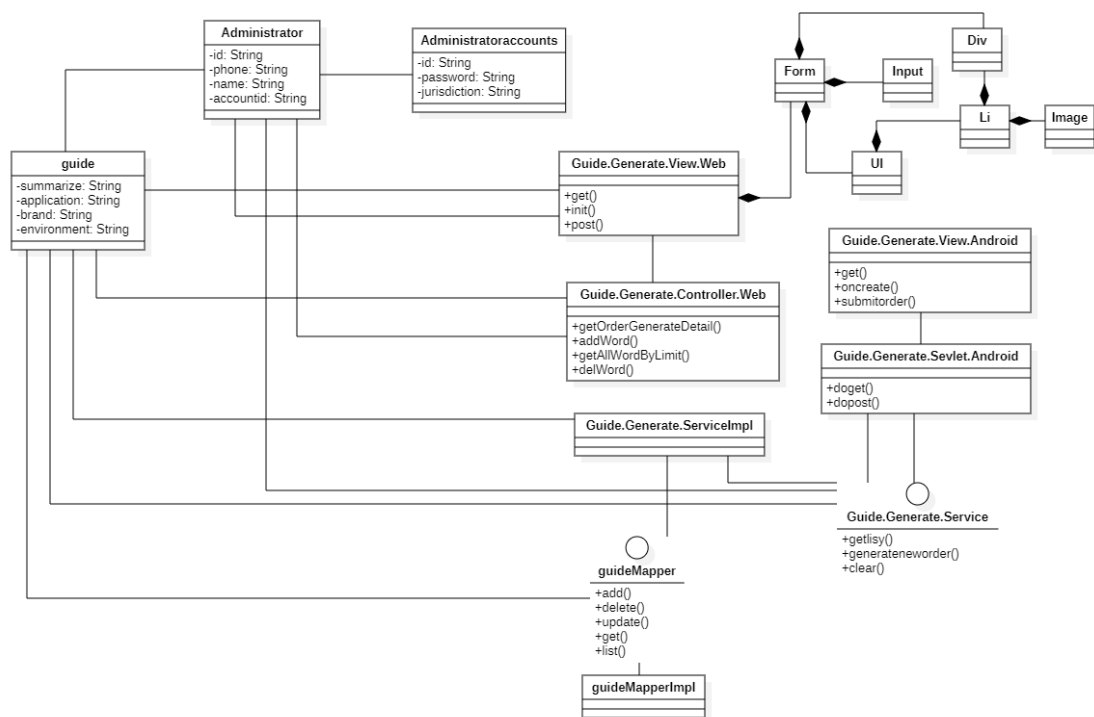
参数列表（参数名 / 类型 / 说明）		
diagnosis	Diagnosis	宠物健康史所有信息

### 4.6.2.6 限制条件及出错处理

列表数据格式需要符合类型

## 4.7 健康信息微服务

在健康信息微服务中，一共有三个实体类，guide、Administrator、Administratoraccounts，分别用来管理健康指南、管理员实体记录、管理员账户。同时通过控制类 Guide.Generate.Controller.Web 来完成具体健康信息的操作。Mapper 以及服务 Service 连接一个边界接口类 Impl 来实现服务。如下图所示：



### 4.7.1 Administratoraccounts 类

#### 4.7.1.1 功能

健康信息微服务中的 Administatoraccounts 类主要负责记录管理员的账户信息。

#### 4.7.1.2 性能

无约束

4.7.1.3 属性描述

属性名	类型	说明
id	Integer	管理员账户 ID
password	String	管理员账户密码
jurisdiction	String	权限

4.7.2 Administrator 类

4.7.2.1 功能

健康信息微服务中的 Administators 类主要负责记录管理员的信息。

4.7.2.2 性能

无约束

4.7.2.3 属性描述

属性名	类型	说明
name	String	管理员名称
id	Integer	管理员 ID
phone	String	电话号码
account	Account	账户信息

4.7.3 Guide.Generate.Controller.Web 类

4.7.3.1 功能

在健康信息微服务中执行健康标准管理、健康指南管理、健康监测等相关功能，可以获取宠物的健康监测分析，健康标准列表，健康指南列表，并对上述的数据进行增删改查以及更新列表操作。

#### 4.7.3.2 性能

无约束

#### 4.7.3.3 输入项

1. 接受管理员/医生的指南选择操作，查询对应的指南信息
2. 接受管理员/医生的增删改查操作，更新指南信息

#### 4.7.3.4 输出项

1. 查询的指南列表信息展示
2. 更新的指南列表信息展示

#### 4.7.3.5 方法详细描述

##### 4.7.3.5.1 addWord

方法签名：public String addWord(Notice notice)

功能说明：将健康指南加入数据库中

接口调用：

- 调用自身 noticeService.add(notice)方法，插入健康指南

详细说明：

返回类型	String	
返回值	返回插入成功“Success”或失败“Failed”信息	
方法限定符	public	
参数列表（参数名 / 类型 / 说明）		
notice	Notice	健康指南信息

##### 4.7.3.5.2 delWord

方法签名：public String delWord(String[] ids)

功能说明：将健康指南加入数据库中

接口调用：

- 调用自身 noticeService.deleteById 方法，删除健康指南



详细说明:

返回类型	String	
返回值	返回删除成功“Success”或失败“Failed”信息	
方法限定符	public	
参数列表（参数名 / 类型 / 说明）		
ids	String[]	健康指南 id 列表

4.7.3.5.3 **getGuideGenerateDetail**

方法签名：public String getGuideGenerateDetail(Long id, Model model)

功能说明：将健康指南加入数据库中

接口调用：

- 调用自身 noticeService.view 方法，显示健康指南详细信息

详细说明:

返回类型	String	
返回值	返回详细信息	
方法限定符	public	
参数列表（参数名 / 类型 / 说明）		
id	Long	指南 id
model	Model	相关指南的标题等

4.7.3.6 **限制条件及出错处理**

指南和标准的数据格式需要符合类型要求