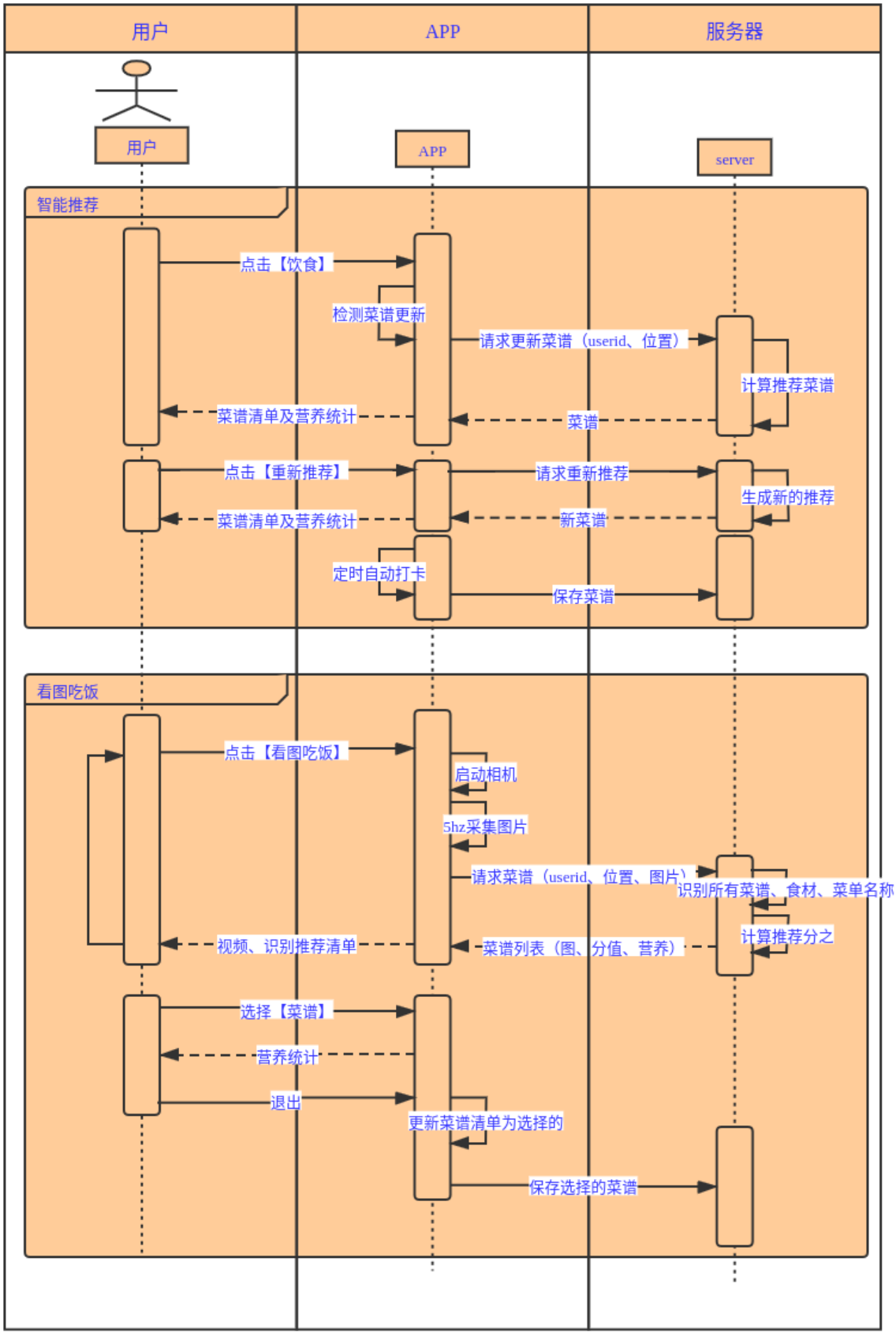
智能饮食服务器开发文档

交互流程图

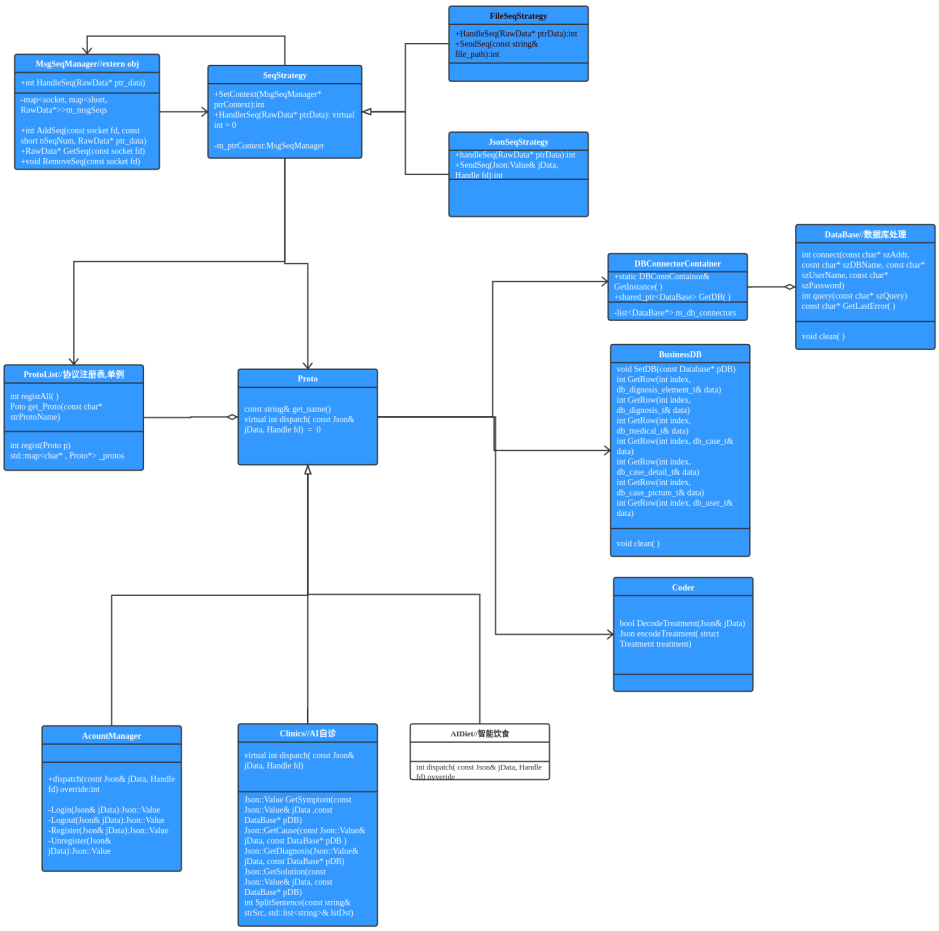


智能饮食时序图

说明：

- 当用户进入APP饮食界面时，触发菜谱更新请求；APP获取用户位置和userID向服务器请求菜谱；
- 服务器会根据《智能健康饮食》中提到的算法计算菜谱清单，并返回结果；
- 用户可能不满意推荐结果，允许重置推荐结果；
- 看图吃饭是基于场景的智能推荐，用户打开相机采集成品菜、菜单或者食材，APP将图文数据上传到服务器进行识别，服务器根据识别结果及用户健康数据返回识别及推荐结果；

类与对象图



说明：

1. 蓝色是已经实现的类，白色是待实现类，扩展业务模块仅实现Proto接口类即可；
2. 上图仅仅包括业务模块的类与对象结构关系，整体采用template设计模式，为支持文件、json传输使用strategy设计模式；
3. MsgSeqManager类实现消息队列管理，Strategy类实现文件、json等字节流解析，Proto类实现所有具体业务处理，ProtoList是具体Proto类的管理容器，DataBase对象封装MySQL接口，DBConnectorContainer管理数据库连接数量；

实现方法

- 代码路径

```
$https://github.com/XiangLiHealthy/XiangLiHealthyLife/tree/master/serv_linux

#编译方法
$https://github.com/XiangLiHealthy/XiangLiHealthyLife/blob/master/ReadMe.md
```

- 继承Proto类，实现int Proto::dispatch(const Json::Value& jData, Handle fd)即可

```
/*************************************************************************
 *说明: Proto作为所有业务协议的抽象接口,每个业务接口只要实现各自的接口即可
 *      线程模块和网络模块不需要做任何修改!不过需要在ProtoManager构造函数
 *      里面注册自己;
 * ************************************************************************/
class Proto
{
    public:
                                Proto(){};
        virtual                 ~Proto(){};

        /*************************************************************
         *功能: 获取协议对象的名称;每个类具体实现的时候都需要唯一指定自
         *      名称,然后prottoManger会根据名称map一个协议对象,接收到请求
         *      时直接根据消息里面的协议名称找到具体的协议对象;
         *
         *返回: 协议名称
         * ************************************************************/
        const string&   getName() {return m_name;}

        /*************************************************************
         *功能: 执行具体的业务;
         *
         *参数: Json:Vaule & jData:包含协议数据的json对象
         *
         *返回: 需要返回给客户端的数据,用json格式包装
         * ************************************************************/
        virtual int     dispatch(const Json::Value& jData, Handle fd) = 0;
    protected:
        string                  m_name;
};
```

- 在serv_linux/business目录创建AIDiet.h

```
#ifndef AIDIET_
#define AIDIET_

#include"proto.h"
#include"business_data.h"

class AIDiet : public Proto
{
    public:
                            Clinics();
                            ~Clinics();

    virtual int     dispatch(const Json::Value& jData, Handle fd);
```

```
private:

public:
};

#endif
```

- serv_linux/business目录下创建AIDiet.cpp

```
AIDiet::AIDiet()
{
    m_name = "AIDiet";//必须在这里指定模块名称
}

int AIDiet::dispatch(const Json::Value& jData, Handle fd)
{
    //1.解析json协议，获取userid、用户位置

    //2.从数据库获取用户基础信息：年龄、性别、体重

    //3.结合营养推荐量表计算当日营养

    //.....


    //发送菜谱列表
    Json::Value jSend;
    jSend["method"] = m_name;
    //jSend["data"] = jData;//这里填推荐菜谱结果

    if (JsonSeqStrategy::SendSeq( fd, jSend) < 0)
    {
        LOG_ERROR("send json failed");
        return -1;
    }

    return 0;
}
```

- 在proto_manager.cpp中注册实现的对象

```
/*****************************************************************************
 *下构造函数自动注册所有业务协议.不用再另外执行专门的注册动作,这样可以
 *简化实现;
 *
 *每个新定义的协议都要在这里添加注册代码
 *协议中不能保存业务缓存,因为每个协议可能同时被多个线程执行,对应着不同的
 *连接
 * ****************************************************************/
ProtoManager::ProtoManager()
{
    //每个协议都是一个单列.并实例化一个静态对象,这样协议名可以每个对象单独维护
    regist(new Clinics());
    regist(new Test());
    regist(new AccountManager());
    regist(new ClinicsRecord());
```

```
    //regist(new AIDiet());
}
```

通信协议

- 1 智能推荐

```
//app请求协议
{
    "protocol":"AIDiet",
    "data":{
        “method”:"get_menu_list",
        "user_id":123,
        "position":{
            "name":"浙江省杭州市西溪八方城顺旺基",
            "longitude":1234.234,
            "dimension"：234.234
        },
        "style":"single/picture/video"
        "picture":"WERWRESFDSFDSARE#@FDS",//经过base64编码的图片数据
    }
}

//服务器应答协议
{
    "method":"get_menu_list_response",
    "desc":"success/failed",
    "data":{
        "menu":[
            {
                "menu_id":123,
                "name":"麻婆豆腐",
                "picture":"BSDFAERFDAFDSAGDSAFD",
                "weight":500,
                "unit":"克、份、碗",
                "recommend_time":"2020-10-23 12:00:00",
                "type":"早餐",
                "ingredients":[
                    {
                        "food_id":123,
                        "name":"豆腐",
                        "percentage":0.8,
                        "energy":234,
                        "portein":23,
                        "fat":243,
                    },
                    {
                        "food_id":128,
                        "name":"低钠盐",
                        "percentage":0.03,
                        "Na":0.0007,
                        "K":0.0001,
                        "I":0.00000002,
                    }
                ]
```

```
            },
            {
                "menu_id":124,
                "name":"酸菜鱼",
                "picture":"BSDFAERFDAFDSAGDSAFD",
                "weight":600,
                "recommend_time":"2020-10-23 12:00:00",
                "type":"早餐、午餐、下午茶、晚餐",
                "ingredients":[
                    {
                        "food_id":123,
                        "name":"草鱼",
                        "percentage":0.8,
                        "energy":234,
                        "portein":23,
                        "fat":243,
                    },
                    {
                        "food_id":128,
                        "name":"酸菜",
                        "percentage":0.03,
                        "Vitamin_C":0.0007,
                        "K":0.0001,
                        "I":0.00000002,
                    }
                ]
            },
        ],
        "medication_task":[
            {
                "medical_id":13,
                "name":"碳酸镁铝片",
                "picture":"WREWQDASFDSAREWR",
                "weight":3,
                "unit":"颗、片、ml",
                "recommend_time":"2020-10-23 12:30:00"
            }
        ]

    }
}
```

- 2 重置推荐结果

```
//app请求协议
{
    "protocol":"AIDiet",
    "data":{
        "method":"reset_menu_list",
        "style":"one_menu/one_meal/one_day",
        "menu_id":[234,23,34444],
        "user_id":123,
        "position":{
            "name":"浙江省杭州市西溪八方城顺旺基",
            "longitude":1234.234,
```

```
            "dimension"：234.234
        },
        "style":"single/picture/video"
        "picture":"WERWRESFDSFDSARE#@FDS",//经过base64编码的图片数据
    }
}

//服务器应答协议与1相同
```

- 饮食打卡
- 看图吃饭请求
- 识别结果
- 选择推荐

数据表结构

- menu_recommend_cache（菜谱缓存表）

序号	字段名	字段类型	说明
1	user_id	int	用户id，很多用户都会缓存到这张表
2	munu_id	int	菜谱id，这个id可以查到包含的食材
3	weight	float	分量：g
4	recommend_time	datetime	推荐食用时间
5	flag	int	0 备用、1 推荐、2 不喜欢

说明：

1. 食用菜谱缓存表是为了应对重置推荐的情况，重置推荐仅需要从缓存表中选择食物，节约计算时间；

- meals_record(饮食记录表)

序号	字段名	字段类型	说明
1	user_id	int	
2	memu_id	int	
3	weight	float	
4	recommend_time	datatime	
5	meal_time	datatime	食用时间，可能与推荐时间不同

- medication_task

序号	字段名	字段类型	说明
1	user_id	int	用户id
2	medicine_id	int	药物id
3	weight	float	分量
4	unit	varchar(8)	分量单位，克、ml、包、颗粒、片
5	take_time	datetime	建议服用时间
6	time_type	int	0 任意时间 1饭前 2饭后
7	relative_time	datatime	相对于饭前或者饭后的时间值
8	state	int	0 待服用 1 已服用 2 过期