# 系统概述

# 手机端

功能

上传emsi+经纬度

检测GPS和network，如果关闭，则打开；

接受服务端的配置信息，或者定期获取配置信息

如果没有network，则记录经纬度在本地，network好了之后，会自动上传

# 控制器设计（功能）

## 用户管理

### 批量导入用户数据

emsi

mdn

用户名称

用户其他信息

### 新建用户

新建一个用户

### 用户列表-文字

删除

修改

查看

轨迹

### 用户列表-地图

仅展示

## 系统管理

### 配置关键位置

基于地图添加关键位置

新建：右键，弹出info 窗口，录入 target 模型定义的信息，包括半径。

### 配置报警距离

普通表单即可。

事先在SysConf模型中创建几个参数，其中一个就是distance。

这个功能就是维护distance这条记录。

### 配置应急预案

普通表单，模型采用plan。

## 监控

### 实时告警

该页面，采用两部分。上面部分：地图展示告警关联人员的位置，下面部分：是告警关联人员列表。

### 实时动态

该页面，实现 每15秒的 人员动态位置图。

# 后台任务

计算人群聚集度，生成事件

计算靠近关键目标距离，生成告警事件

计算离线终端，生成告警事件

# 消息处理

位置信息上传

参照position模型中的字段；

服务端的处理方式：

1. 将信息保存在position中；
2. 将信息保存在past\_postion中

postion模型主要是为了记录position的log，核心线索是 时间。方便批量查找到某个时间点的位置信息；

past\_postion模型，则是以用户，日期为线索，记录某人某天的位置信息，方便画出轨迹。

离线时的信息处理

先记录在手机端，待上线，上传信息集合

# 模型设计

customer.rb

用户，或者客户模型；

主要字段：name，mdn，imsi，imei

incident.rb

事件模型；

字段：customer集合-customer\_ids, 发生的时间 occurence\_time

incident\_type.rb

事件类型：

past\_position.rb

以日期、人为核心的模型，记录某日某人的位置信息；

plan.rb

应急预案

尚在设计中

position.rb

位置信息：终端定期上传用户的信息

字段：

EMSI：卡的物理号，与customer对应；

地理位置信息 lat经度，lng纬度

手机电量：power；

上传时间：time

sys\_conf.rb

系统配置参数，name 和value组成；

target.rb

关键位置

字段为：name，名称；point 坐标点；radius 半径。

user.rb

# 服务设计

# 关键流程

告警显示的流程

1. 手机终端上传位置信息
2. 在保存位置信息的时候，position模型增加一个callback 方法，判断该位置信息是否与target中的距离在 sys\_conf 的参数distance 范围内，如果在范围内，则生成一条 incident记录
3. view层，采用定期从后台库中显示incident 信息，显示完毕后，将这个incident信息的计数 增加 1.表示已经显示过1次，如果。