总体结构如图所示：

发送过程：

1. 在android应用程序中生成需要发送的json数据，然后通过mimo-node提供的sdk发送到mimo-node服务器特定的存储空间（Theme）。
2. Java服务器向mimo-node服务器订阅（subscribe）特定的存储空间（Theme）。
3. mimo-node服务器在接受到新的数据之后通过newMessages事件向java服务器推送数据，如果java服务器不在线则对数据进行缓存，当java服务器登录之后再推送过去。

具体执行步骤：

1. mimo-node服务器的启动

在linux系统中可以通过 rdesktop -f 10.107.29.248命令打开服务器的远程控制桌面。

之后依次执行 cd mongodb/bin

./mongd

cd redis/src

./redis-server

cd air-quality-monitor2

node –harmony app.js来启动服务器

1. 启动java服务器。
2. 启动android程序。

原理分析：

mimo-node是基于socketio开发的，程序主要是使用newMessages、subscribeOnTheme、publishOnTheme等关键字来进行通信，其中publishOnTheme是向mimo-node服务器发送数据，subscribeOnTheme是订阅服务器上存储数据的空间，以便服务器在收到新的数据之后向其推送数据，newMessages用来接收服务器的推送数据。

使用mimo-node之前要先注册用户名、密码、Space、Theme。

java服务器的test.java文件中有对publish、subscribe、login等api的封装，以及main函数。

android应用程序中通过jsonsender对api进行封装，在MultiThreadJsonReceiveThread类中对封装函数进行调用。

注意：android和java的使用socketio的jar包不同，用错则显示不能连接；并要注意版本问题，版本不对则只能连接不能发送数据。