|  |
| --- |
| 河南三次元智能科技有限公司 |
| 软件著作申请 |
| 三次元智能科技有限公司软著申请文档 |

|  |
| --- |
| 作者：三次元智能科技有限公司  2019-8-12 |

目录

**1,简介1**

1.1智能家居的变革2

1.2智能办公功能详述3

1.3产品范围4

**2，产品概述5**

2.1产品技术6

2.1.1射频技术7

2.1.2传感技术8

2.1.3网络通信技术9

2.1.4嵌入书系统技术10

2.1.5云计算平台11

**2.2产品种类12**

2.2.1智能门锁13

2.2.2智能网关14

2.2.3智能情景面板15

2.2.4智能语音背景音乐16

2.2.5智能电动卷帘17

2.2.6智能室内云摄像头18

2.2.7能开关控制19

2.2.8智能调色灯带20

2.2.9智能移动插座/计量插座21

2.2.10智能平移推窗器22

2.2.11智能红外控制/红外人体控制23

2.2.12智能水浸探测器24

2.2.13智能燃气探测器25

2.2.14智能晾衣系统26

2.2.15完整的智能生态体系设计逻辑27

2.3 **场景应用案例**28

**3，移动端功能模块**29

3.1登录功能30

3.2注册功能31

3.3效果展示32

**简介**

近几年，随着科技的发展，智能家居行业兴起，可是谁都不知道往后几年智能家居这个行业是否还会像现在这般火热，智能家居市场是否会爆发，对于想加入智能家居行业在准备或考察的朋友来说，都想知道这门生意好不好做。

**一、智能家居的利润空间**

**市场需求大，自然有利润**

    在目前的经济形势以及技术发展形势中，智能家居代表的已经是新一轮信息技术革命中价值最大的产业环节之一，市场发展空间大，赚钱自然不在话下。

    由于智能家居尚属发展阶段，在利润率上一般是70%-100%，如5000元的普通家庭智能建设工程，利润大约在3000元左右，每月接到3-5个这样的订单，月收入就能轻松上万，利润空间不是其他任何创业项目能与之媲美的。

这只是智能家居其中的一个环节，如果从用户购买到售后服务一系列下来，利润空间更大。

**二、哪些智能家居单品生意好做**

**1、智能门锁**

    智能门锁可以远程控制开门、指纹开锁、密码开锁、刷卡开锁、防撬报警等，这些功都受到广大消费者的喜爱，很多用户都以智能门锁作为自己智能家居生活开始的第一步。

**2、安防类智能家居产品**

    像门磁传感器、可燃气体传感器、烟雾传感器、漏水传感器、智能摄像头等，安装十分简单，不仅能够起到预防意外事故的发生还能起到远程监控的作用，自带的语音对讲功能还能够让相隔两地的人进行语音对讲，有效起到保护家庭安全的作用，同时还可以远程看护孩子、老人。

**3、智能插座**

家里有智能插座，家电可以定时开关，远程开关，咖啡机、面包机、煮蛋器等设定开机时间，这样早晨一起床就有美味的早餐等着我们了；不仅省电省钱，还可以为用户节省大把的时间。

**三、智能家居投入与产出**

**投入**：在一线城市开个150平米左右的智能家居店需要投资大约50万元;在二线城市开个120平米左右的智能家居店需要投资大约36万元;在三线城市开个100平米左右的智能家居店需要投资大约27万元。

**产出**：在一线城市开个150平米左右的智能家居店一年的利润大约为66万元;在二线城市开个120平米左右的智能家居店一年的利润大约为51万元;在三线城市开个100平米左右的智能家居店一年的利润大约为42万元;

     通过这些数据可以看到，做智能家居生意前期的投入只需一年就能够全部回本并赚钱。

**四、智能家居店的经营策略**

**1、多种渠道进货**

    智能家居的技术和产品的种类范围非常广泛，如果进货渠道多的话，则产品的功能会更加的完善、全面，用户的选择也会更多，也能满足所有家庭的需要。

**2、与商家合作**

    还应该与房地产开发商、装饰工程公司、灯饰及建材专业批发市场合作，扩展业务。

**3、根据不同户型进行推广**

    不同的户型所需要的智能家居产品会有所不同，有些需要布线有些客户不需要布线，还有装修前和装修后的智能家居产品搭配使用的考虑，可以多制定几套方案，可以是经济实惠，也可以是奢华高调的。解决方案可以适应不同的需求。

**4、建智能家居网店**

    开通网络销售渠道，加大产品推广，如果方法得当，网上也会卖得很火。

看了本篇文章的朋友们，都应该知道做智能家居这门生意其实并不难，而包括智能家居在内的任何生意只要坚持、努力一般都可以成功，当然也并不排除失败的可能性，但是任何事情都有两面性，智能家居这门生意也不例外，所以在做智能家居生意的时候也需谨慎。

     在国内的智能家居行业在市场与技术不断成熟之下，未来智能家居将会迎来爆发期，2012年到2018年中国智能家居市场规模高达639亿美元，智能家居必然会高速发展。

**1.1，5G+AI 智能家居迎来深度变革**

迈进2019年，5G离我们更近了，5G不只是网速的提升，更让万物互联更进一步，加速物联网在全行业的普及，未来将结合云端和人工智能等技术，推动社会变革，进入一个万物具有感知的智能社会。5G时代，一部容量为2GB的电影，仅需几秒钟就能下载完成。

5G与智能家居设备的结合被视为家居行业发展的必然趋势。

5G技术的商业化，肯定是先从手机开始。智能家居设备也会享受到这波红利：更高的带宽、更快的速度、更低的延迟。

首先近几年，越来越多的智能产品走进千家万户。像智能门锁、智能安防、智能空调、智能家电等设备已经有较高的普及度了，智能设备与消费者之间的距离不断缩短。那么5G的到来对智能家居会产生什么样的影响呢？

在4G网络下，信息传输慢，设备与设备交换信息存在延时情况，这种情况在家庭影音和视频通话之间显得尤为明显。5G速度每秒高达10Gb，能承载更多设备连接、传输更大流量。5G时代到来后，智能家居产品将更灵敏，其它传统家电设备响应速度也会大幅提升。

智能安防是智能家居市场上最成熟的领域之一，包括智能门锁、智能门磁、智能摄像头等各类产品。然而，因网络信号覆盖传输受空间布局等影响，让智能安防设备在使用过程中出现各类问题。

5G网络的出现，单位流量内资费费率肯定会逐步下降，困扰家庭安防、智能家居等一些信号、应用推广问题有可能都会迎刃而解，数以亿计家庭应用安防设备将走入普通家庭。

智能家居控制系统通过多个传感器把相互孤立的信息连接起来，但这些连接远远不够。每个厂家都专注于连接自己的产品，无法实现品牌互联。5G时代到来后，国际权威机构制订标准，有利于打破各大企业各自为战、割据一方，市场孤岛化的混乱局面。

所有智能家居设备都在低功率下串联，并且通过不同方式相互交换信息，增加了设备传输间延时和传输超窄带宽问题，影响了整个智能家居生活体验。

5G网络超高速率、极大容量、超低延时等优势可以支撑海量设备互联，大量数据包传输，人以及设备之间可以实现真正的连接，更多硬件都能在5G网络环境下连接传感器实现快速数据处理与智能操控。

5G最明显优势之一就是“传输速度快”，这种高速率有助于信息检测和管理，能使设备之间“感知”更加精确，特别是在家庭影音和视频通话之间尤为明显，有利于提高整个智能家居控制系统智慧化程度，将给用户带来更好体验。

AI技术提升了响应空间，通过AI能够将一个比较孤立的设备拉入场景化，于是智能家居就有了更加丰富的想象空间。

“AI+IoT”技术风口下，互联网企业和家电企业是推动智能家居产业发展的重要力量。包括房企、安企、设备商、第三方服务商等，都可以在这股浪潮中找到自已的定位。智能家居涉及消费者生活的方方面面，涵盖服务种类较多，大到房屋、家电、安控，小到门锁、插座、等，都将因新技术到来而受益智能音响是智能家居生态链中的一员，让用户具有了语音控制其他智能硬件的可能，在实际使用中，一个智能音箱就可以控制所有的智能家居，比如语音控制窗帘，空调，灯具等等，智能音箱是一个入口，一个总控平台，扩展了智能设备的响应数量和响应空间。

5G与AI的结合，将点燃智能家居市场的新一轮军备竞赛，直接影响智能家居产业的创新力，这些改变不仅让智能家居产品的体验进一步升级，同时也缓解了品牌之间不兼容的尴尬。

在未来，智能生态或智能平台，不能单纯的是智能设备生产商，或者是只具备销售能力的公司，需要有自己的一套生态体系。智能生态系统的建立，需要生态创建者有足够强的号召能力，最好能在研发、生产、销售、服务等层面有补齐生态短板的能力，能够起到产业赋能的职责，才能真正扛起建立智能生态系统的大旗和重任。这样看来，目前有条件构建智慧生态的公司并不多，各大公司能否迅速补齐短板，争夺一张5G时代的船票，将会成为未来的一大看点。

据业内人士预测，在2020年中国将会真正的进入5G时代，而全球智能硬件产品也将会达到250亿个，也就是说到了2020年，一切传统的生活方式将会被智能产品取代，真正的智能家居时代将会到来。

在5G商业化步步逼近的今天，三次元智能家居也将迎来更加快速的发展，凭借5G的超高速传输使系统与设备之间的“感知”更加精确，提高整个智能家居控制系统的智慧化程度。借助5G浪潮，加快产品的系统化集成和应用，从中国领先的智能家居厂商发展为世界物联网浪潮的领跑者。

**1.2，智能办公功能详述**

如今，智能化的概念开始逐渐渗透到各行各业以及我们生活中的方方面面，市场上相继出现了智能住宅小区、智能医院等，而现在智能化写字楼也渐渐出现了。

作为一家现代企业，特别是中型以上的企业，如何来提高企业的客户接待档次、产品展示的独特性、提升企业科技形象？如何来提高办公、会议的高效与现代化？如何来提高公司内部的办公效率与舒适性？除了人性化的机制与服务以外，企业办公智能化、会议办公智能化，产品展示智能化将成为大势所趋，这也是现代企业的高科技办公、会议、产品推介与展示全新科技手段。

舒适的办公环境能有效提升员工的工作效率、忠诚度，同时大大提升了企业的科技化形象，在同行企业竞争中，助企业一臂之力。随着企业对于智能化办公需求的日益强烈，写字楼的智能化进程也将不断加速，各写字楼间的智能化PK将是一场持久的战役。

智能办公楼是将楼宇自动化管理系统、通信网络系统、办公自动化系统、消防智能化和安保智能化通过总体规划的方法和策略，进行综合和分解，运用系统工程的观点，集成在建筑环境这个支持平台上， 将建筑物的结构、系统、服务和管理四个基本要素进行优化组合， 提供一个投资合理， 具有高效、舒适、安全、方便环境的建筑物，它主要满足以下几个要求和目标。

**1.3产品使用范围**

随着科技水平的发展，智能家居已经逐渐成为现代人装修房屋的刚需了。消费升级，家的功能不再仅仅是遮风挡雨，现在80后、90后渐成购房的刚需群体，和父辈一代千篇一律的家装风格不同，他们在装修时更加注重家装风格的设计感和个性化。在对待“住”这个问题上，他们的需求不仅仅是解决吃饭睡觉的问题，而是更多地考虑怎么样有品质地生活。

智能家居可以解决生活中的诸多痛点，比如出门忘记带钥匙，总担心门窗是否关好？想泡澡时，才记得去放热水？天热时，总觉得空调制冷太慢？家里只有老人孩子，担心他们的安全……智能家居系统，不仅可以随时随地让家就在身边，而且可以大大地提升生活质量。

有非法分子接近您的住宅时，安防监控系统会发出报警声，并可自动拨打电话报警。当屋子有煤气、烟雾泄漏时，燃气、烟雾等感应器会自动报警通知外会自动关闭煤气阀门，不会发生忘记关闭煤气而导致发生危险。

试想一下，是躺在床上动动手就能拉上窗帘、关灯、检查门锁方便，还是下床在屋里转一圈挨个排查方便呢？当然是动动手指就能解决问题更方便，智能家居就能够给你带来这种便捷的生活，在家中任意位置都可控制灯光、家电、窗帘等设备。

工作完下班路上，拿起手机轻轻一按，热水器已经提前启动……任意调节家中灯光的明暗、室内的温湿度，只需轻轻一触，即可实现梦幻般的场景变化。

家庭成员通过智能家居APP即可对住宅进行远程控制。无论你身在何处，想看看家里，你只需要打开手机，便可以远程看到自己的家，还可以对设备进行操作。

外出时，你的住宅定时开关灯，时时像有人居住一样，提供安全感；在家时，通过语音即可控制家里所有的灯光、电器、背景音乐、窗帘、空调、地暖等。

通过对空调，灯光等设备的管理，能够有效的控制家庭电源，在不使用电器时便会及时关闭，减少家庭用电量，可以大大降低家庭能源开支。

智能家居的发展已经成为了未来家居的发展趋势，安装智能家居后，我们的家居生活就会变得轻松自如，做到物为人用，不是为物所累。智能家居让生活更便捷，让家庭更温馨。在科技发达的现在，传统的普通家居将会逐渐逝去，未来必然是智能家居的时代。

**2，产品概述**

目前，智能家居最流行的无线通信技术有蓝牙、WIFI、Z-Wave、Zigbee等。但由于蓝牙技术协议较复杂，功耗高，成本高，传输范围受限，一般有效的范围在10米左右，抗干扰能力不强、信息安全性差，使其并不太适用于要求低成本、低功耗的工业控制和家庭网络，并制约其进一步发展和大规模应用。

而WIFI技术虽然移动性强，但由于采用的是射频技术，通过空气发送和接收数据，使用无线电波传输数据信号，容易受到外界的干扰和受黑客威胁，安全性比较差，因此也不太适合对安全性要求极高的智能家居。与蓝牙和WIFI不同。

但是Z-Wave和Zigbee技术对比并无明显优势，虽然 Z-Wave 具有结构简单，性能可靠的特点，但目前想获得 Z-Wave 芯片的授权还比较困难，更为关键的是 Z-Wave 所用频段(865.2MHz-956MHz)在我国是非民用的。所以对我国智能家居企业来说，ZigBee 协议更适合国内使用的情况，选择 ZigBee 协议一定是经过深思熟也是必然的结果。依然是ZigBee在智能家居中有着更加广泛的应用.

**ZigBee 协议的特性：**

ZigBee无线通信芯片,领先传统ZigBee芯片CC253.三十年，芯片架构好，空间大，处理速度快，携带节点设备更多，ZigBee协议的优势在于自组网能力，最多支持 65000 个设备组网;对于智能家居来说，ZigBee 协议还有一个宝贵的优点就是它的安全性很高，至今全球尚未出现一起破解先例。

①**低功耗**。

在低耗电待机模式下，2节5号干电池可支持1个节点工作6～24个月，甚至更长。这是ZigBee的突出优势。相比较，蓝牙能工作数周、WiFi可工作数小时。

**②低成本。**

通过大幅简化协议(不到蓝牙的1/10)，降低了对通信控制器的要求，按预测分析，以8051的8位微控制器测算，全功能的主节点需要32KB代码，子功能节点少至4KB代码，而且ZigBee免协议专利费。每块芯片的价格大约为2美元。

**③低速率。**

ZigBee工作在20～250kbps的速率，分别提供250 kbps(2.4GHz)、40kbps(915 MHz)和20kbps(868 MHz)的原始数据吞吐率，满足低速率传输数据的应用需求。

**④近距离。**

传输范围一般介于10～100m之间，在增加发射功率后，亦可增加到1～3km。这指的是相邻节点间的距离。如果通过路由和节点间通信的接力，传输距离将可以更远。

**⑤短时延。**

ZigBee的响应速度较快，一般从睡眠转入工作状态只需15ms，节点连接进入网络只需30ms，进一步节省了电能。相比较，蓝牙需要3～10s、WiFi 需要3 s。

**⑥高容量。**

ZigBee可采用星状、片状和网状网络结构，由一个主节点管理若干子节点，最多一个主节点可管理254个子节点;同时主节点还可由上一层网络节点管理，最多可组成65000 个节点的大网。

**⑦高安全。**

ZigBee提供了三级安全模式，包括无安全设定、使用访问控制清单(Access Control List， ACL) 防止非法获取数据以及采用高级加密标准(AES 128)的对称密码，以灵活确定其安全属性。

综上所述，ZigBee技术在智能家居中，加密高，安全性好的优点ZigBee技术在智能家居产品中的应用和推广。灵活运用ZigBee技术的优点，才能够更好的提供高性价比，高可靠性的智能家居产品，引爆智能家居市场！

以下是智能家居中ZigBee的一些使用大用户：亚马逊，霍尼韦尔，南京物联，松下等等。也是三次元智能家居选择采用ZigBee技术的理由。

**2.1.1RFID技术**

RFID（射频识别）技术是一种无接触的自动识别技术，利用射频信号及其空间耦合传输特性，实现对静态或移动待识别物体的自动识别，用于对采集点的信息进行“标准化”标识。鉴于RFID技术可实现无接触的自动识别，全天候、识别穿透能力强、无接触磨损，可同时实现对多个物品的自动识别等诸多特点，将这一技术应用到物联网领域，使其与互联网、通信技术相结合，可实现全球范围内物品的跟踪与信息的共享，在物联网“识别”信息和近程通讯的层面中，起着至关重要的作用。另一方面，产品电子代码(EPC)采用RFID电子标签技术作为载体，大大推动了物联网发展和应用。

**2.1.2传感器技术**

有价值的信息不仅需要射频识别技术，还要有传感技术。物联网经常处在自然环境中，传感器会受到环境恶劣的考验。所以，对于传感器技术的要求就会更加严格、更加苛刻。

传感器可以采集大量信息，它是许多装备和信息系统必备的信息摄取器件。若无传感器对最初信息的检测、交替和捕获，所有控制与测试都不能实现。即使是最先进的计算机，若是没有信息和可靠数据，都不能有效地发挥传感器本身作用。传感器技术的突破和发展有3个方面：网络化、感知信息、智能化。

**2.1.3网络通信技术**

网络通信技术包含很多重要技术，其中M2M技术最为关键，该技术范围应用广泛，不仅能与远距离，而且还能与近距离技术相衔接。现在的M2M技术以机器对机器通信为核心，对于建筑学、航空航天、医学、农业等行业是专业人士未来要努力实现的。

**2.1.4嵌入式系统技术**

嵌入式系统技术是综合了计算机软硬件、传感器技术、集成电路技术、电子应用技术为一体的复杂技术。经过几十年的演变，以嵌入式系统为特征的智能终端产品随处可见；小到人们身边的MP3，大到航天航空的卫星系统。嵌入式系统正在改变着人们的生活，推动着工业生产以及国防工业的发展。如果把物联网用人体做一个简单比喻，传感器相当于人的眼睛、鼻子、皮肤等感官，网络就是神经系统用来传递信息，嵌入式系统则是人的大脑，在接收到信息后要进行分类处理。这个例子很形象的描述了传感器、嵌入式系统在物联网中的位置与作用。

**2.1.5云计算**

云计算是把一些相关网络技术和电脑融合在一起的产物。它是利用分布式计算机计算出的信息和运行数据中心改成与互联网相近，使资源能够运用到有用的技术上，对存储系统和电脑做必要的咨询。目的是把各种消费进行低成本处理并融合为功能完整的实体，还可以运用MSP、SAAS等模式分布并计算到终端用户。云计算是以加强改善其处理能力为重点，用户终端的负担也相应降低，I/O设备也能够简化，还可以对它的计算功能进行合理的享受并运用。例如百度等搜索功能就是它的应用之一。

**2.2，产品种类**

**1.智能门锁**

三次元智能门锁采用高端锁体，在锁体植入ZigBee芯片来实现远程管理和控制，不仅具备开锁信息和低电量提醒信息上报，最大亮点就是ZigBee无线技术应用到门锁保障了安全性，和远程开锁采用随机密码，当需要远程开锁时系统就会随机生成一个开锁密码，这个密码在有效时间内可以通过这个密码正常开锁，超过时间该密码即失效无法开锁。

**主要技术亮点**：

1、指纹开锁识别上报：不同的人开门自动上报，主人就会知道什么人在什么时间进的家门（便于知晓小孩或老人每天有没有按时回家）。

2、胁迫开锁报警：当有歹徒胁迫你开锁时，也可以用其他手指将门打开，在门被打开的同时，报警信息也会悄无声息的向指定电话号码发出。

3、联动功能：门开启时自动开启相应的灯光、电器等迎接回家。

4、远程手机开锁：采用国际目前最先进的ZigBee无线通讯技术远程开锁，杜绝串码误开等不安全因数，确保家门安全。

5、低电量提醒：当指纹锁电量较低时，会提前在手机APP上提醒。

**2.智能网关**

用户操作流程:

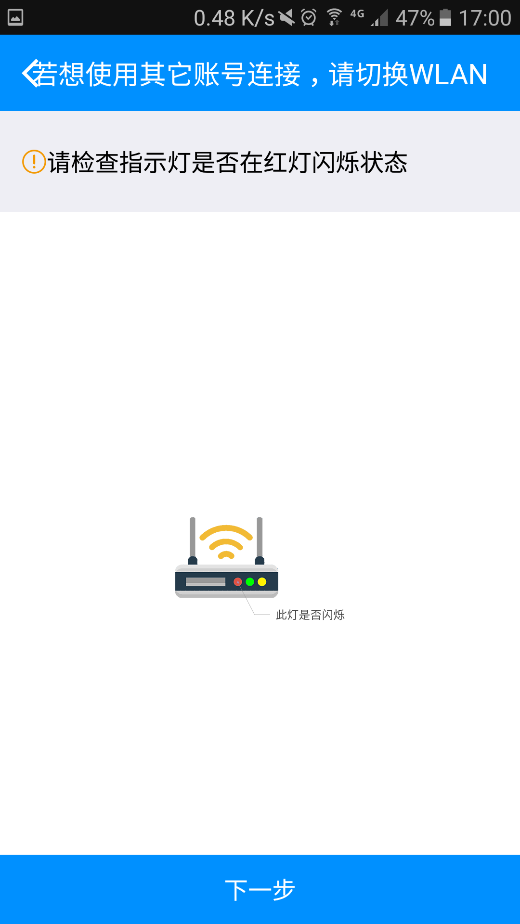
打开Cislunar智能软件:



点击“我的”页面进行网关添加:



点击切换网关进行切换网关添加网关进行配置:



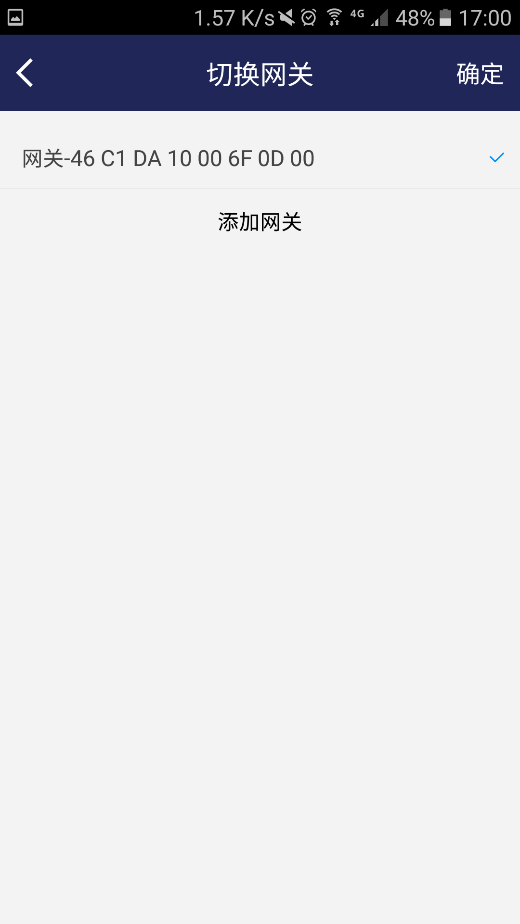
按住网关8秒进行网关配置：



点击下一步进行配置:



填写WLAN密码和端口号进行配置:



添加成功配置完成。

产品特点：内置天线，无线WiFil链接，无需网线，组网规模大,可带负载300个设备以上，抗干扰能力强,多网多负载共存无干扰。

**3.智能情景面板**

****

使用场景：用户可以安装在客厅，卧室，洗手间等任意场景，进行用户操作

产品特点：反应灵敏操作简单内置ZigBee模块芯片与网关无线链接，可联动回家场景，就餐场景，离家场景。

**4.智能语音背景音乐**

****

产品特点：跨空间的智能家居控制设备，场景控制趣味人机对话，精准语音点歌，音乐播放控制，生活类信息服务(百度百科,股票信息，天气信息随问随答)日程闹钟设置，标配两个25W功率高保真音响

**5.智能电动卷帘/智能开合窗帘**

****

产品特点：支持手机远程控制，实现场景联动设置，采用高精度交流电机，静音设计，可应用于中小型卷帘，百叶窗等场景。适用于老年人和小孩的场景，通过手机和遥控控制实现智能升降。

**6.智能室内云台摄像头**

****

产品特点：内置ZigBee模块芯片，水平360度全景，智能追踪，全程记录异常动态，实时推送报警信息，为家人的生命财产安全保驾护航。

**7.智能开关控制**

****

产品特点：多联智能开关，支持灵活多样的组合控制模式，全系1开-3开,内置ZigBee模块，手机利用无线WiFi信号或4G信息对开关进行远程开关控制，面板开关信息和状态及时上报手机，智能节能，低功耗，静态功耗小于0.1W

**8.智能调色LED灯带**

****

产品特点：内置ZigBee模块芯片与开关无线连接，手机通过APP对智能调色LED灯进行自定义颜色，同时可通过手机远程调节LED的高度

**9. 智能移动插座/计量插座**

****

产品特点：智能移动插座，内置ZigBee模块芯片与网关无线连接，可通过手机远程管理定时开关功能，起到安全防护的作用，节能环保，省心省钱，用电功率和电流等信息实时上报至家人手机，随时随地掌握安全用电情况

**10.智能平移式开窗器**

****

产品特点：风雨感应器与平移开窗器配套一起使用，可根据天气的变化情况，智能调节开窗的大小和角度，也可以通过手机经行远程自定义控制，实现真正智能话管理

**11.智能红外控制/红外人体感应器**

****

产品特点：手机一键控制家中电器(空调/电视/机顶盒/音响等),通过场景智能联动，预设场景模式，一键完成多个遥控指令，通过家中的红外转发实时控制家中的红外设备。当有陌生人进入家中时，将信息及时发送到家人的手机上，及时处理安防事件，真正起到了智能安全的控制理念。

**12．智能水浸探测器**

****

产品特点：智能水浸探测器漏水，溢水，水位变化第一时间知晓，凭借高精度和高敏感度，让地板，墙面，桌面，每处角落意外都得到实时感知，发现并处理可能存在的安全隐患。从而避免了因水侵造成更大的损失，同时通过远程发送到手机实现随时监控。

**13.智能燃气探测器**

****

产品特点：当有天然气，煤气泄露时,燃气感应器检测发出信号给声光报警和机械手，机械手自动关闭机械阀门，同时向用户发送报警信息以及泄漏时间和地点，让用户及时处理安全隐患，实现了智能安防的作用。

**14．智能晾衣系统**

****

产品特点：智能控制器可以控制晾衣架，通过手机远程控制衣架的收起，同时可以联动风雨感应器，当刮风下雨时，智能衣架自动收回防止衣服被雨水淋湿。

**15．完整的智能生态体系设计逻辑**

****

****

1、逻辑设计：用户在开门进入，因室内红外均无法探测有用户在房间，门磁将会联动打开灯光，实现人来灯亮，同时打开空调（可以先把窗帘关闭，20s后自动打开窗帘，告诉住户我们这儿是有高科技产品的）【管家设置一次联动打开灯，条件是室内红外无报警；两次窗帘，一次关闭，一次延时打开窗帘】

2、用户到卫生间时，卫生间红外人体探测，将会探测到人进入，卫生间将会自动打开灯光和排风扇。可设置延时5分钟关闭。若能探测到人则“取消此设备等待执行的延时任务”，直到探测不到人体活动时，才会延时5分钟关闭。【管家设置增加两次开关控制，第一次是开灯，第二次是取消延时任务。新起关闭任务，延时关闭灯光开关】

3、卫生间和床头加装紧急按钮，用户不舒服时，可按下紧急按钮，通过BMS管理平台或PMS酒店管理平台接收报警信息，客房中心将安排人员上门进行协助。

4、红外安装角度斜对着窗帘，当探测人到窗户时，联动打开窗帘（也可轻拉窗帘自动打开）。当探测到用户不在窗帘附近是，可延时5分钟关闭窗帘，以保护用户的隐私。【窗帘可绑定到场景开关手动打开关闭，切换场景排除自动关闭】【管家设置红外联动和红外无人关闭】

5、室内红外正对房间内，正常情况下联动开关保持现状不变，当住户打开灯时，室内的灯光保持打开，；当用户关闭灯光时，室内灯光保持关闭。（睡眠模式时，探测到人起床，则会打开玄关小筒灯轻微照亮往卫生间的通道。）

当室内红外、卫生间红外、窗帘红外均不能探测用户活动，则延时5分钟关闭卧室灯（2卫生间灯已延时关闭了）。

6、人在客房开门外出，门磁打开，但因室内红外有探测人，故门磁不会联动开灯。

两人时一人外出一人在客房，室内红外、卫生间红外探测室内有人，则门磁不会执行开灯动作，保持室内设置不变。

人开门外出后，红外均无法探测人存在，延时5分钟将会自动关闭室内灯光。【门磁消警，红外无人，延时5分关灯；红外有人，设置不变（可不设置）】

7、可增加炫彩灯，启动浪漫模式，炫彩灯进行炫彩变换，变换时间间隔为2~300ms。灯光色彩流水般缓慢的变换，营造温柔浪漫的气氛环境。如搭配轻柔背景音乐，可营造优雅舒适的惬意生活。

8、背景音乐，门磁打开联动背景音乐响起，自动播放轻柔舒适的音乐。在睡眠模式时，联动不同的睡眠曲，并设置定时关闭。

9，语音控制窗帘、灯光照明、空调温度、电视。设置绑定物联网关系统。通过情景面板只能以控制家中的各个灯光和智能设备的控制，当早上起来时实现家中相应的设备工作运转，当晚上回到家中通过手机或者情景控制面板为用户运行自定义相应的灯光和智能设备的运转，实现了自定义的场景控制。

**2.3通讯协议示例**

**通过UDP模拟C/S通讯过程V1.0**

package com.sanciyuan.Demo1;  
import java.io.\*;  
import java.net.Inet4Address;  
import java.net.InetSocketAddress;  
import java.net.Socket;  
//客户端的发送  
public class Client {

public static void main(String[] args) throws IOException {

//创建Socket的对象  
Socket socket=new Socket( );  
//设置超时时间  
socket.setSoTimeout( 3000 );  
//链接本地,端口2000,超时时间3000ms  
socket.connect(new InetSocketAddress( Inet4Address.*getLocalHost*(),2000 ),3000);  
System.*out*.println("已发起服务器链接,并进入后续流程!");  
//客户端信息  
System.*out*.println("客户端信息："+"地址("+socket.getLocalAddress()+")端口号("+socket.getLocalPort()+")");  
//服务器端信息  
System.*out*.println("服务器信息："+"地址("+socket.getInetAddress()+")端口号("+socket.getPort()+")");  
try {  
//发送接收数据  
*todo*(socket);  
}catch (Exception e){  
System.*out*.println("客户端已退出!");  
}  
//释放资源  
socket.close();  
System.*out*.println("客户端已退出!");  
}  
//客户端的创建实现  
public static void todo(Socket client) throws IOException {  
//构建键盘流输入流  
InputStream in=System.*in*;  
//创建缓冲读取流对象  
BufferedReader input=new BufferedReader( new InputStreamReader( in ) );  
//得到Sockt输出流,并转换为打印流(客户端)  
OutputStream outputStream=client.getOutputStream();  
PrintStream printStream = new PrintStream( outputStream );

//得到Socket输入流  
InputStream inputStream=client.getInputStream();  
BufferedReader bufferedReader=new BufferedReader( new InputStreamReader( inputStream ) );  
//布尔类型判断循环退出  
boolean flag=false;  
do {  
//键盘读取一行  
String str=input.readLine();  
//发送到服务器端  
printStream.println( str );  
//从服务器读取一行  
String echo=bufferedReader.readLine();  
//字节流的比较  
if("bye".equalsIgnoreCase(echo)){  
flag=false;  
}else {  
System.*out*.println(echo);  
}  
}while (flag);  
//资源释放  
printStream.close();  
bufferedReader.close();  
}

}

**服务器端的发送过程V1.0**

package com.sanciyuan.Demo1;  
import java.io.\*;  
import java.net.ServerSocket;  
import java.net.Socket;  
//服务器端的发送  
public class Server {  
public static void main(String[] args) throws IOException {  
//初始化服务器端口号为2000  
ServerSocket serverSocket=new ServerSocket( 2000 );  
System.*out*.println("服务器准备就绪!");  
//客户端信息  
System.*out*.println("服务器信息："+"地址("+serverSocket.getInetAddress()+")端口号：("+serverSocket.getLocalPort()+")");  
//等待客户端链接  
for (;;){  
//得到客户端  
Socket socketclient=serverSocket.accept();  
//客户端构建异步线程  
ClientHandler clientHandler = new ClientHandler(socketclient);  
//启动线程  
clientHandler.start();  
}  
}  
//客户端消息处理  
private static class ClientHandler extends Thread{  
private Socket socket;  
private boolean flag=true;  
//构造方法  
ClientHandler(Socket socket){  
this.socket=socket;  
}  
@Override  
//重写run方法  
public void run(){  
super.run();  
System.*out*.println("新客户端链接:"+socket.getLocalPort()+"端口号:"+socket.getLocalPort());  
try{  
//得到打印流,用于输出;服务器回送数据使用  
PrintStream printStream=new PrintStream( socket.getOutputStream() );  
//得到输入流对象，用于接收数据  
BufferedReader socketInput=new BufferedReader( new InputStreamReader( socket.getInputStream() ) );  
do{  
//通过客户端拿到一条数据  
String str=socketInput.readLine();  
if("bye".equalsIgnoreCase( str )){  
flag=false;  
//回送  
printStream.println( "bye" );  
}else{  
//打印到屏幕并回送数据长度  
System.*out*.println(str);  
printStream.println( "回送数据长度" +str.length());  
}  
}while (flag);  
//释放资源  
socketInput.close();  
printStream.close();  
}catch (Exception e){  
System.*out*.println("异常断开链接!");  
}finally {  
//链接关闭  
try {  
socket.close();  
} catch (IOException e) {  
e.printStackTrace();  
}  
}  
//输出端口号:  
System.*out*.println("客户端已退出:"+socket.getInetAddress()+"P:"+socket.getLocalPort());  
}  
}  
}

**3.移动端功能模块 V1.0**

3.1智能家居控制终端的登录功能展示

UI界面视图:



**main\_activity.xml**

<LinearLayout

android:id="@+id/number"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/iv"

android:layout\_centerVertical="true"

android:layout\_marginBottom="5dp"

android:layout\_marginLeft="10dp"

android:layout\_marginRight="10dp"

android:layout\_marginTop="15dp"

android:background="#ffffff">

<TextView

android:id="@+id/tv\_number"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:padding="10dp"

android:text="账号"

android:textColor="#000"

android:textSize="20dp" />

<EditText

android:id="@+id/et\_username"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="5dp"

android:background="@null"

android:padding="10dp" />

</LinearLayout>

<LinearLayout

android:id="@+id/password"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/number"

android:layout\_centerVertical="true"

android:layout\_marginLeft="10dp"

android:layout\_marginRight="10dp"

android:background="#ffffff">

<TextView

android:id="@+id/tv\_password"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:padding="10dp"

android:text="密码"

android:textSize="20dp"

android:textColor="#000" />

<EditText

android:id="@+id/et\_password"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="5dp"

android:layout\_toRightOf="@id/tv\_password"

android:background="@null"

android:inputType="textPassword"

android:padding="10dp" />

</LinearLayout>

<Button

android:id="@+id/button\_login"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_below="@id/password"

android:layout\_marginLeft="10dp"

android:layout\_marginRight="10dp"

android:layout\_marginTop="60dp"

android:background="#3c8dc4"

android:text="登录"

android:textColor="#ffffff"

android:textSize="20dp" />

<Button

android:id="@+id/button\_register"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_below="@id/button\_login"

android:layout\_marginLeft="10dp"

android:layout\_marginRight="10dp"

android:layout\_marginTop="30dp"

android:background="#b7585556"

android:text="注册"

android:textColor="#ffffff"

android:textSize="20dp" />

<CheckBox

android:checked="true"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="记住密码"

android:id="@+id/checkBox"

android:layout\_below="@+id/password"

android:layout\_marginLeft="10dp"

android:layout\_marginTop="5dp"/>

3.2智能家居控制终端的注册功能展示

确定注册------按钮实现注册，判断以上四个注册信息是否符合规则，判断两次输入密码是否一样，并且不为空。并且显示提示信息  
返回登录------按钮实现跳转到刚才的登录界面



main\_activity.java类

public class MainActivity extends AppCompatActivity {

private EditText et\_username;

private EditText et\_password;

private EditText et\_password2;

private EditText et\_mail;

private Button btn\_login;

private Button btn\_register;

private CheckBox checkbox;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

Map<String, String> userInfo = SaveInfo.getSaveInformation(this);

if (userInfo != null) {

et\_username.setText(userInfo.get("username"));

et\_password.setText(userInfo.get("password"));

}

et\_username =(EditText) findViewById(R.id.et\_username);

et\_password =(EditText) findViewById(R.id.et\_password);

et\_password2 =(EditText) findViewById(R.id.reg\_password2);

et\_mail = (EditText) findViewById(R.id.reg\_mail);

checkbox = (CheckBox) findViewById(R.id.checkBox);

btn\_login =(Button) findViewById(R.id.button\_login);

btn\_register =(Button) findViewById(R.id.button\_register);

btn\_login.setOnClickListener(new MyButton());

btn\_register.setOnClickListener(new MyButton());

}

public class MyButton implements View.OnClickListener{

@Override

public void onClick(View view){

String username =et\_username.getText().toString().trim();

String password =et\_password.getText().toString().trim();

switch (view.getId()) {

//当点击登录按钮时

case R.id.button\_login:

if(TextUtils.isEmpty(username) || TextUtils.isEmpty(password)){

Toast.makeText(MainActivity.this,"密码或账号不能为空",Toast.LENGTH\_SHORT).show();

} else {

if(checkbox.isChecked()){

//保存密码的操作

}

Toast.makeText(MainActivity.this,"登录成功",Toast.LENGTH\_SHORT).show();

Intent intent = new Intent(MainActivity.this, LoginActivity.class);

startActivity(intent);

}

break;

//当点击注册按钮事件时

case R.id.button\_register:

Intent intent = new Intent(MainActivity.this,RegisterActivity.class);

startActivity(intent);

break;

}

}

}

}

register\_activity

<TextView

android:layout\_marginTop="60dp"

android:id="@+id/reg\_number1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:padding="10dp"

android:text="账号:"

android:textColor="#000"

android:textSize="20dp" />

<EditText

android:layout\_alignBottom="@+id/reg\_number1"

android:layout\_toRightOf="@+id/reg\_number1"

android:id="@+id/reg\_username"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:padding="10dp" />

<TextView

android:id="@+id/reg\_number2"

android:layout\_marginTop="5dp"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/reg\_number1"

android:padding="10dp"

android:text="密码:"

android:textColor="#000"

android:textSize="20dp" />

<EditText

android:layout\_alignBottom="@id/reg\_number2"

android:layout\_toRightOf="@+id/reg\_number2"

android:id="@+id/reg\_password"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:padding="10dp" />

<TextView

android:id="@+id/reg\_number3"

android:layout\_marginTop="5dp"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/reg\_number2"

android:padding="10dp"

android:text="密码:"

android:textColor="#000"

android:textSize="20dp" />

<EditText

android:layout\_alignBottom="@id/reg\_number3"

android:layout\_toRightOf="@+id/reg\_number3"

android:id="@+id/reg\_password2"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:padding="10dp" />

<TextView

android:id="@+id/reg\_number4"

android:layout\_marginTop="5dp"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/reg\_number3"

android:padding="10dp"

android:text="邮箱:"

android:textColor="#000"

android:textSize="20dp" />

<EditText

android:layout\_alignBottom="@id/reg\_number4"

android:layout\_toRightOf="@+id/reg\_number4"

android:id="@+id/reg\_mail"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:padding="10dp" />

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="确定注册"

android:background="#74e674"

android:id="@+id/reg\_btn\_sure"

android:layout\_marginTop="38dp"

android:layout\_below="@+id/reg\_mail"

android:layout\_marginLeft="50dp" />

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="返回登录"

android:background="#f27758"

android:id="@+id/reg\_btn\_login"

android:layout\_alignBottom="@id/reg\_btn\_sure"

android:layout\_toRightOf="@id/reg\_btn\_sure"

android:layout\_marginLeft="60dp"

/>

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="账号注册"

android:textSize="30dp"

android:layout\_marginTop="5dp"

android:layout\_alignParentLeft="true"

android:layout\_alignParentStart="true" />

</RelativeLayout>

registeractivity.java类

public class RegisterActivity extends AppCompatActivity {

private EditText reg\_username;

private EditText reg\_password;

private EditText reg\_password2;

private EditText reg\_mail;

private Button reg\_btn\_sure;

private Button reg\_btn\_login;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_register);

reg\_username = (EditText) findViewById(R.id.reg\_username);

reg\_password = (EditText) findViewById(R.id.reg\_password);

reg\_password2 = (EditText) findViewById(R.id.reg\_password2);

reg\_mail = (EditText) findViewById(R.id.reg\_mail);

reg\_btn\_sure = (Button) findViewById(R.id.reg\_btn\_sure);

reg\_btn\_login = (Button) findViewById(R.id.reg\_btn\_login);

reg\_btn\_sure.setOnClickListener(new RegisterButton());

reg\_btn\_login.setOnClickListener(new RegisterButton());

}

public class RegisterButton implements View.OnClickListener {

@Override

public void onClick(View v) {

String username = reg\_username.getText().toString().trim();

String password = reg\_password.getText().toString().trim();

String password2 = reg\_password2.getText().toString().trim();

String mail = reg\_mail.getText().toString().trim();

switch (v.getId()) {

//注册开始，判断注册条件

case R.id.reg\_btn\_sure:

if (TextUtils.isEmpty(username) || TextUtils.isEmpty(password) || TextUtils.isEmpty(password2) || TextUtils.isEmpty(mail)) {

Toast.makeText(RegisterActivity.this, "各项均不能为空", Toast.LENGTH\_SHORT).show();

} else {

if (TextUtils.equals(password, password2)) {

//执行注册操作

SaveInfo.SaveInformation(RegisterActivity.this,username,password,password2,mail);

Toast.makeText(RegisterActivity.this,"注册成功,请返回登录",Toast.LENGTH\_SHORT).show();

} else {

Toast.makeText(RegisterActivity.this, "两次输入的密码不一样", Toast.LENGTH\_SHORT).show();

}

}

break;

case R.id.reg\_btn\_login:

Intent intent = new Intent(RegisterActivity.this, MainActivity.class);

startActivity(intent);

break;

}

}

}

}

登录成功界面创建一个布局文件就可以了，写上你想要的东西，我自己就是创建了一个布局，在这里因为要做一个保存操作，所以创建了一个java工具类，其中定义了两个方法，一个保存登录名和密码，一个负责调用保存的登录名和密码  
saveinfo

public class SaveInfo {

public static boolean SaveInformation(Context context, String username, String password, String password2, String mail) {

try {

FileOutputStream fos = context.openFileOutput("data.txt", Context.MODE\_APPEND);

fos.write(("用户名:" + username + " 密码:" + password + "邮箱：" + mail).getBytes());

fos.close();

return true;

} catch (Exception e) {

e.printStackTrace();

return false;

}

}

public static Map<String, String> getSaveInformation(Context context) {

try {

FileInputStream fis = context.openFileInput("data.txt");

BufferedReader br = new BufferedReader(new InputStreamReader(fis));

String str = br.readLine();

String[] infos = str.split("用户名:"+"密码:"+"邮箱:");

Map<String, String> map = new HashMap<String, String>();

map.put("username", infos[0]);

map.put("password", infos[1]);

fis.close();

return map;

} catch (Exception e) {

e.printStackTrace();

return null;

}

}

}

示例图片：





