智能花园

软件需求规约

版本 <1.0>

修订历史记录

|  |  |  |  |
| --- | --- | --- | --- |
| **日期** | **版本** | **说明** | **作者** |
| <03/06/2018> | <1.0> | 起草 | 吴怜颐，吴幸融 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

1. 简介 4

1.1 目的 4

1.2 定义、首字母缩写词和缩略语 4

1.3 参考资料 4

2. 整体说明 4

3. 具体需求 5

3.1 功能 5

3.1.1 Use case 图 5

3.1.2 Log in/Register 6

3.1.3 Create Garden 7

3.1.4 Control Garden 8

3.1.5 Create Sensor or Controller 9

3.1.6 Manage User State 10

3.1.7 Manage Garden State 11

3.1.8 Get Data 11

3.1.9 Handle Request 12

3.1.10 Manage Garden 12

3.1.11 Manage Sensor or Controller 13

3.1.12 Send Data 14

3.1.13 Change State 14

3.2 易用性 15

3.2.1 用户培训时间 15

3.2.2 图形标准 15

3.3 可靠性 15

3.3.1 系统可用性 15

3.3.2 平均故障间隔时间 (MTBF) 15

3.3.3 平均修复时间 (MTTR) 15

3.3.4 精确度 15

3.3.5 最高错误或缺陷率 15

3.3.6 错误或缺陷率 15

3.4 性能 15

3.4.1 事务响应时间 15

3.4.2 吞吐量 16

3.4.3 可容纳量 16

3.4.4 系统降级模式 16

3.4.5 资源利用情况 16

3.5 可支持性 16

3.5.1 编码标准 16

3.5.2 命名约定 16

3.5.3 类库 16

3.5.4 维护访问权 16

3.6 设计约束 16

3.6.1 编程语言 16

3.6.2 开发工具 16

3.6.3 系统开发流程 16

3.6.4 构架及设计约束 16

3.6.5 兼容性约束 17

3.6.6 测试约束 17

3.6.7 购买的构件、类库 17

3.7 联机用户文档和帮助系统需求 17

3.8 接口 17

3.8.1 用户界面 17

3.8.2 通信接口 17

软件需求规约 (简化版)

# 简介

## 目的

本文档目的在于提供关于智能花园管理系统的需求功能的详述以及系统性能参数的说明。

## 定义、首字母缩写词和缩略语

暂无

## 参考资料

具体资料暂无，参考资料的大致方面：

1. 农生方面的资料（关于不同季节和天气情况下，不同植物不同生长阶段需要的湿度、温度、光照和肥料条件，进行灌溉、施肥、人工光照的时机、种类和用量）。

2. Unity、Android SDK等框架的官方文档，查询API。

3. 使用的无线传感器/控制器模块的说明书，查询串口等信息用。

4. GUI 国际标准，指导人机界面的开发。

# 整体说明

产品总体效果：实现温度和湿度自动监测，并根据预设值进行自动浇灌和温度调节。

产品功能：

1. 账户管理：用户通过登陆或注册访问系统。
2. 显示并控制智能花园的传感器和监控器状态：提供用户当前所管理的花园状态，可以远程人工控制，调节温度和湿度。
3. 创建花园：用户可以新创建花园，需要给出花园的名字、尺寸、理想的温度和湿度等。
4. 添加控制器和传感器：用户可以在花园中添加传感器或控制器，及其类型。
5. 手机APP开发：用户通过访问服务器，可以获得花园温度和湿度分布图，通过遥控来改变控制参数和进行人工干预。

用户特征：针对拥有花园，并希望对其进行智能化管理的个人、企业或机构等用户。

约束：要求用户群具备连接因特网的条件和基本的使用检验以及基本的安全信息防卫意识。

假设与依赖关系：假设开发经验不足，功能设计不够完善，都会影响本项目的开发流畅性。项目时间设计不足都会影响到后续开发进程。

# 具体需求

## 功能

### Use case 图

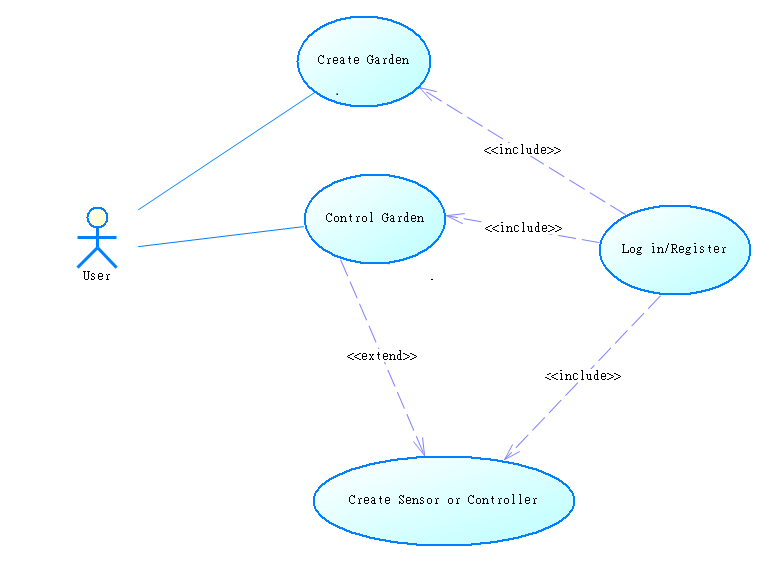


图1 Use Case图（User）

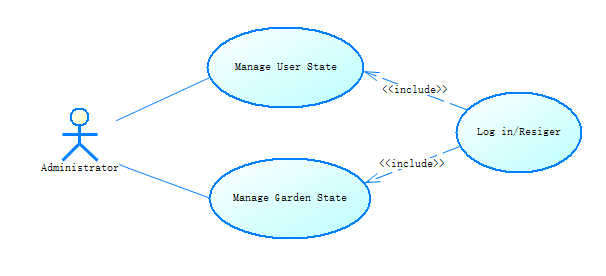


图2 Use Case图（Administrator）

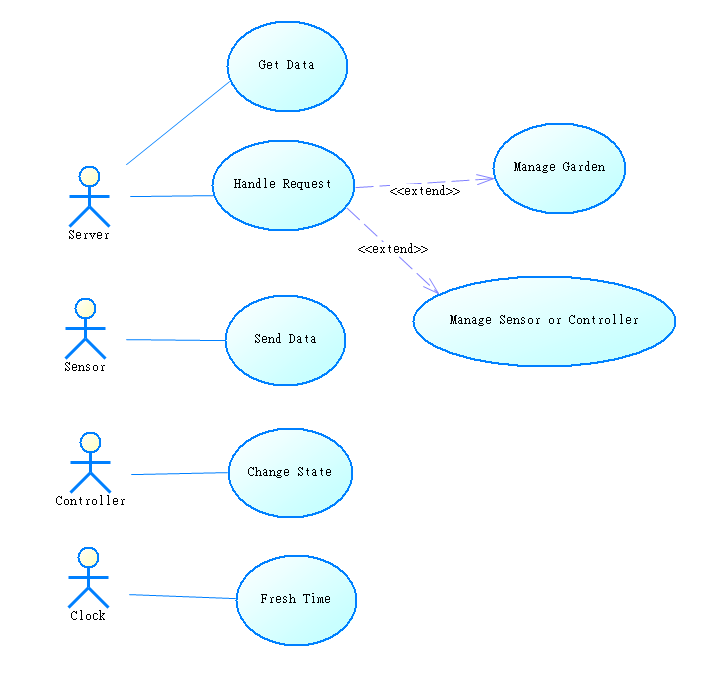


图3 User Case图（外部设备）

### Log in/Register

#### **描述**

用户使用账户信息登陆系统或者注册系统账户。

#### **事件流**

##### 基本流：

1.系统请求用户输入用户名和密码登陆或者进行用户注册。若用户选择注册账户，则执行“注册账户”子流。否则执行“登陆账户”子流。

1）注册账户：

A.用户点击Register按钮。

B.用户输入用户名、密码、姓名、性别、邮件地址。

C.用户点击submit按钮。

D.系统验证用户信息。

E.系统向用户邮箱发送激活邮件，激活邮件中包含激活地址。

F.用户点击激活邮件中的激活地址，激活账户。

2）登陆账户：

A.用户输入用户名和密码。

B.用户点击submit按钮。

C.系统验证用户名和密码。

##### 备选流：

1.1.E.a 用户名与已有用户名重复

停留在本页面，并提示用户更改用户名或者放弃注册。

1.1.F.a 用户未激活账户

该账户未被成功注册，不能使用。

1.2.C.a 无效的用户名与密码

停留在本页，并提示用户错误信息。

#### **非功能需求**

系统响应用户时间不超过1秒。

#### **前置条件**

无

#### **后置条件**

如果用户登陆或注册成功，则跳转至新建花园的地图界面。如果失败，则停留在本页。

#### **扩展点**

无

### Create Garden

#### **描述**

用户给出花园的名字、尺寸、理想的温度和湿度等属性，以创建花园。

#### **事件流**

##### 基本流：

1. 用户点击create a new garden按钮。
2. 用户输入新建花园的名字、长度、宽度、理想温度和理想湿度。
3. 用户点击submit按钮。
4. 系统验证新建花园信息。
5. 系统将请求发送给服务器端。
6. 服务器端返回“创建成功”信息。
7. 系统在用户花园列表中添加该新建花园。

##### 备选流：

4a 新建花园名字与现有花园名字重复

停留在本页面，并提示用户更改名字或放弃创建花园。

5a 系统请求发送失败

停留在本页面，并将错误原因报告给用户。

6a 服务器返回“创建失败”信息

停留在本页面，并将错误信息报告给用户。

#### **非功能需求**

系统响应用户时间不超过1秒。

#### **前置条件**

用户需要登陆系统。

#### **后置条件**

如果用户创建花园成功，则跳转至新建花园的地图界面。如果创建失败，停留在本页面，并告知用户失败原因。

#### **扩展点**

无

### Control Garden

#### **描述**

用户查看花园的传感器状态，并控制各个控制器的开关。

#### **事件流**

##### 基本流：

1. 系统显示该用户的智能花园列表。
2. 用户选择进行操作的花园。
3. 系统显示选中花园的地图，及各个传感器和控制器的位置，其中数据超出预设范围的传感器，用红色表示。地图右侧给出各个控制器的控制界面。

如果用户点击map按钮，执行“显示热力图”子流。

如果用户点击地图中传感器图标，执行“显示传感器折线图”子流。

1. 用户通过控制界面来控制各种控制器的开关。
2. 系统发送控制信息给服务器端。
3. 服务器端返回“控制成功”信息。
4. 显示热力图
5. 系统发送请求给服务器端。
6. 服务器端返回该用户端最新的数据包。

C. 系统将花园地图切换显示成整个花园温度热力图的界面。

D. 用户点击Humidity按钮，系统显示湿度热力图界面。

2）显示传感器折线图

A. 系统不断地发送请求给服务器端。

B. 服务器端返回该传感器最新的数据。

C. 系统不断地以折线图的方式显示该传感器的数据

##### 备选流：

3.1.A.a 系统发送请求失败

停留在原界面，并报告错误原因给用户。

3.1.B.a 服务器返回数据包失败

停留在原界面，并报告错误原因给用户。

3.2.A.a 系统发送请求失败

停留在原界面，并报告错误原因给用户。

3.2.B.a 服务器返回数据包失败

停留在原界面，并报告错误原因给用户。

5a 系统发送请求失败

停留在原界面，并报告错误原因给用户。

6a 服务器返回“控制失败”信息

停留在原界面，并报告错误原因给用户。

#### **非功能需求**

系统响应用户时间不超过5秒。

#### **前置条件**

用户需要登陆系统，并已选择需要进行操作花园。

#### **后置条件**

无

#### **扩展点**

无

### Create Sensor or Controller

#### **3.1.5.1 描述**

用户给出地图上的位置和传感器类型，以在花园中添加传感器或控制器。

#### **3.1.5.2 事件流**

##### 基本流：

1. 用户点击add a sensor or controller按钮。
2. 如果用户选择添加传感器，执行“添加传感器”子流。如果用户选择添加控制器，执行“添加控制器”子流。
3. 添加传感器

A.用户输入传感器的坐标位置。

B.用户点击submit按钮。

C.系统验证传感器坐标位置。

D.系统将请求发送给服务器端。

E.服务器返回“创建成功”信息。

D.系统在已选花园中相对坐标位置添加传感器。

1. 添加控制器

A.用户输入控制器的坐标位置。

B.用户选择控制器的类型。

C.用户点击submit按钮。

D.系统验证控制器坐标位置。

E.系统将请求发送给服务器端。

F.服务器返回“创建成功”信息。

G.系统在已选花园中相对坐标位置添加控制器。

##### 备选流：

1.1.C.a 添加传感器坐标位置与已有传感器或控制器、监视器重复

停留在本页面，并提示用户更改坐标或放弃增加传感器。

1.1.D.a 系统发送请求失败

停留在本页面，并报告用户错误原因。

1.1.E.a 服务器返回信息失败

停留在本页面，并报告用户错误原因。

1.2.D.a 添加控制器坐标位置与已有控制器或控制器、监视器重复

停留在本页面，并提示用户更改坐标或放弃增加控制器。

1.2.E.a 系统发送请求失败

停留在本页面，并报告用户错误原因。

1.2.F.a 服务器返回“创建失败”信息

停留在本页面，并报告用户错误原因。

#### **3.1.5.3 非功能需求**

系统响应用户时间不超过3秒。

#### **3.1.5.4 前置条件**

用户需要登陆系统，并已选择需要进行操作花园。

#### **3.1.5.5 后置条件**

如果传感器或控制器设置成功，返回花园控制页面。如果设置失败，则停留在本页面。

#### **3.1.5.6 扩展点**

无

### Manage User State

#### **3.1.6.1 描述**

#### 管理员更改用户的状态。

#### **3.1.6.2 事件流**

##### 基本流：

1. 管理员选择“User”界面。
2. 管理员选择用户。
3. 管理员点击“禁用/启用”按钮。
4. 若用户原状态为“启用”，系统提示管理员填写“禁用原因”。若用户原状态为“禁用”，直接执行5。
5. 系统修改用户状态。

备选流：

#### **3.1.6.3 非功能需求**

系统响应用户时间不超过1秒。

#### **3.1.6.4 前置条件**

管理员需登陆系统。

#### **3.1.6.5 后置条件**

用户状态切换成功后，回到管理用户页面。

#### **3.1.6.6 扩展点**

无

### Manage Garden State

#### **3.1.7.1 描述**

#### 管理员更改花园状态。

#### **3.1.7.2 事件流**

##### 基本流：

1. 管理员选择“Garden”界面。
2. 管理员选择花园。
3. 管理员点击“禁用/启用”按钮。
4. 若用户原状态为“启用”，系统提示管理员填写“禁用原因”。若用户原状态为“禁用”，直接执行5。
5. 系统修改花园状态。

备选流：

#### **3.1.7.3 非功能需求**

系统响应用户时间不超过1秒。

#### **3.1.7.4 前置条件**

管理员需登陆系统。

#### **3.1.7.5 后置条件**

花园状态切换成功后，回到管理花园页面。

#### **3.1.7.6 扩展点**

无

### Get Data

#### **3.1.8.1 描述**

服务器端定时从传感器获取数据。

#### **3.1.8.2 事件流**

##### 基本流：

1. 服务器检查时间，若时间符合要求，进入2，否则重复1。
2. 服务器从传感器获取数据。
3. 服务器将获取到的数据打包，并打上上时间戳。
4. 服务器将数据包导入数据库。

##### 备选流：

2a 服务器获取数据失败

该传感器该时间数据记为null，继续执行3。

#### **3.1.8.3 非功能需求**

系统响应时间不超过0.5秒。

#### **3.1.8.4 前置条件**

无

#### **3.1.8.5 后置条件**

无

#### **3.1.8.6 扩展点**

无

### Handle Request

#### **3.1.9.1 描述**

服务器端处理用户的请求，并将请求发送给控制器端。

#### **3.1.9.2 事件流**

##### 基本流：

1. 服务器端获取用户的请求，并将用户请求打上时间戳，导入数据库。

##### 备选流：

1a 请求数据导入数据库失败

将失败原因报告给管理员，并将请求数据暂存于服务器端。

#### **3.1.9.3 非功能需求**

系统响应时间不超过0.5秒。

#### **3.1.9.4 前置条件**

无

#### **3.1.9.5 后置条件**

无

#### **3.1.9.6 扩展点**

无

### Manage Garden

#### **3.1.10.1 描述**

服务器处理用户“更改花园”请求。

#### **3.1.10.2 事件流**

##### 基本流：

1. 服务器接收用户请求。若请求为创建花园，执行“创建花园”子流。若请求为显示地图，执行“显示地图”子流。若请求为显示折线图，执行“显示折线图”子流。
2. 创建花园
3. 服务器端将新建花园信息导入数据库。
4. 服务器端返回“创建成功”信息。
5. 显示地图
6. 服务器端根据花园信息，从数据库导出该花园最新的数据包。
7. 服务器端返回数据包。
8. 显示折线图
9. 服务器端根据传感器信息，从数据库导出该数据库近期的数据。
10. 服务器端返回数据。

##### 备选流：

1.1.A.a 信息导入数据库失败

服务器返回“创建失败”信息，和错误原因。

1.2.A.a 导出数据包失败

服务器返回错误原因。

1.3.A.a 导出数据失败

服务器返回错误原因。

#### **3.1.10.3 非功能需求**

系统响应时间不超过0.5秒。

#### **3.1.10.4 前置条件**

无

#### **3.1.10.5 后置条件**

无

#### **3.1.10.6 扩展点**

无

### Manage Sensor or Controller

#### **3.1.11.1 描述**

服务器端处理用户“更改传感器和控制器”请求。

#### **3.1.11.2 事件流**

##### 基本流：

1. 服务器接收用户请求。若请求为创建传感器或控制器，执行“创建传感器或控制器”子流。若请求为开关控制器，执行“开关控制器”子流。

1）创建传感器或控制器

A. 服务器端将新建传感器或控制器信息导入数据库。

B. 服务器端返回“创建成功”信息。

2）开关控制器

A. 服务器端根据控制器信息，发送请求给该控制器。

##### 备选流：

1.1.A.a 信息导入数据库失败

服务器返回“创建失败”信息，和错误原因。

1.2.A.a 请求发送失败

服务器返回错误原因。

#### **3.1.11.3 非功能需求**

系统响应时间不超过0.5秒。

#### **3.1.11.4 前置条件**

无

#### **3.1.11.5 后置条件**

无

#### **3.1.11.6 扩展点**

无

### Send Data

#### **3.1.12.1 描述**

传感器传递数据给服务器端。

#### **3.1.12.2 事件流**

##### 基本流：

1. 传感器接受服务器端的请求，发送传感器当前信息。

##### 备选流：

1a 若发送信息失败

传感器忽略本次请求。

#### **3.1.12.3 非功能需求**

系统响应时间不超过0.5秒。

#### **3.1.12.4 前置条件**

无

#### **3.1.12.5 后置条件**

无

#### **3.1.12.6 扩展点**

无

### Change State

#### **3.1.13.1 描述**

控制器切换状态。

#### **3.1.13.2 事件流**

##### 基本流：

1. 控制器接受服务器端的请求，关闭或开启控制器。

##### 备选流：

1a 若关闭或开启控制器失败

返回错误信息给服务器。

#### **3.1.13.3 非功能需求**

系统响应时间不超过1秒。

#### **3.1.13.4 前置条件**

无

#### **3.1.13.5 后置条件**

无

#### **3.1.13.6 扩展点**

无

## 易用性

### 用户培训时间

智能花园系统的一般用户培训时间大约是1小时。

### 图形标准

符合Microsoft 的GUI标准要求。

## 可靠性

### 系统可用性

系统可用时间达到了98％以上，持续可运行时间可达1500小时。

### 平均故障间隔时间 (MTBF)

智能花园系统平均故障间隔时间为六个月。

### 平均修复时间 (MTTR)

智能花园系统平均修复时间要小于六小时。

### 精确度

对于时间计量，精确到秒。

对于温度计量，精确到小数点后一位。

对于湿度计量，精确到小数点后一位。

### 最高错误或缺陷率

智能花园系统要求每千行代码的错误数目少于10个。

### 错误或缺陷率

小错误：画面显示异常，部分传感器和控制器未显示

错误率：小于10%

大错误：对于网站用户的输入或命令，服务器无响应

错误率：小于3%

严重错误：系统停止工作，处于崩溃状态，数据库运行失常造成数据存储及读取命令无效的情况

缺陷率： 避免发生

## 性能

### 事务响应时间

网站使用者在不受网速制约情形下，平均响应时间是3s，最长响应时间是8s。系统管理员最长相应时间不超过2s。

### 吞吐量

智能花园系统最多可容纳的同时在线客户数不超过5000。

### 可容纳量

智能花园系统最多可接受500000用户的注册。

### 系统降级模式

系统降级模式时，系统只提供用户登陆及控制花园已有传感器、控制器功能。不提供用户注册、创建花园和创建传感器、控制器、监视器功能。

### 资源利用情况

智能花园系统服务器要求至少1GB内存、80GB硬盘空间并已连接Internet。网络连接速度应大于1MB/s。

客户机端要求客户端：64 MB SDRAM和200MB 硬盘空间并支持Internet网络连接。

## 可支持性

### 编码标准

代码一律采用Unicode进行编码。

### 命名约定

名字中每个单词的首字母(比如类型或变量)要大写，单词之间通常不要使用下划线。

类型名称有一个额外的大写字母前缀，以便将类型名称和变量名称区分开来。

类型和变量名称必须是名词。

方法名称是动词，该动词描述了该方法的作用或者描述了无效的方法返回值。

变量、方法及类的名称应该清晰、明确且具有描述性。名称的作用域越大，取一个符合标准的具有描述性的名称的重要性便越强。避免过度缩写。

所有变量都应该一次仅声明一个，以便可以提供有关这个变量的含义的注释。

### 类库

Best HTTP (Pro)、HttpClient、Chart and Graph、SpringUnity。

### 维护访问权

由开发团队持有。

## 设计约束

### 编程语言

系统使用C#和Java编写（由所使用的库决定），数据库使用MySQL语句，网页使用HTML5构建。

### 开发工具

在Microsoft Visual Studio 2017 环境下进行C++代码编写，在IntelliJ IDEA 2018.1环境下进行Java代码和HTML5代码编写，在Unity 2018.1.1f1 (64-bit)环境下进行Unity框架相关代码编写，在MySQL Workbench 6.3 CE环境下进行数据库语句编写。

### 系统开发流程

使用C#和Unity框架进行前端开发，使用Java和MySQL进行后端开发。最后将两者进行整合调试。

### 构架及设计约束

采用分布式系统的分层架构风格，分为表示层、应用层、服务层、领域层、持久性层。

### 兼容性约束

服务器操作系统要求为Windows版本，推荐为Windows 10。

浏览器版本支持Internet Explorer 9以上，或其他任何支持HTML5的浏览器。

移动客户端操作系统要求为Android版本，推荐为Android 4.0以上。

### 测试约束

测试在Microsoft Visual Studio 2017 环境下进行完后，于MySQL语句结合后必须在服务器上进行调试，测试。

### 购买的构件、类库

Arduino UNO R3

## 联机用户文档和帮助系统需求

暂无，根据开发实际情况填写。

## 接口

### 用户界面

注册界面、登录界面、创建花园界面、管理花园界面、显示花园温度/湿度热力图界面、显示传感器实时数据界面、管理智能方案界面、创建智能方案界面。

### 通信接口

暂无，根据开发实际情况填写。