**Спецификация проекта**

**«Создание системы удаленного мониторинга растений»**

**Раздел 2. Создание UI**

**Содержание**

Введение 2

Описание проблемы 3

Формулировка задачи и описание концепции решения 4

Описание реализации («как это будет сделано») 4

Функциональность решения («как это будет выглядеть») 6

Порядок взаимодействий («как это будет работать») 7

Ключевые характеристики 8

High-level проектирование пользовательского интерфейса 10

Список функций приложения (feature list) 10

Сценарии использования приложения (central user journey) 10

Mid-level проектирование пользовательского интерфейса 10

Общая структура 10

Разметка страницы 11

Low-level проектирование пользовательского интерфейса 11

Определение стилей и «состояний» (Style Definitions & State Definitions) 11

Описание виджетов, которые будут использоваться 12

# Введение

Цель данного документа –сформулировать задание на исполнение для UI/UX дизайнера.

# Описание проблемы

Наши клиенты – люди, которые ухаживают за декоративными растениями: дома, на даче, в зимнем саду и т.д. Это могут быть те, кто сами выращивают растения или же те, на ком временно лежит ответственность по уходу, например, родственники, которых попросили присмотреть. Итак, наши клиенты – это люди, которые несут ответственность за то, чтобы растения всегда были в хорошем состоянии (чтобы всегда было достаточно влаги и все окружающие условия были оптимальными).

Декоративные растения требуют внимательного присмотра и регулярного ухода, поэтому тому, кто за ними ухаживает, нужно всегда помнить, поливал он растения или нет, если не поливал, то когда какое растение нужно полить, сколько поливать и т.п. Если он не фанат-растениевод, то это становится проблемой. Можно легко что-то забыть, полить какое-то растение недостаточно или, наоборот, с избытком. Особенно проблема актуальна, если среди растений есть дорогие/экзотические/прихотливые экземпляры.

# Формулировка задачи и описание концепции решения

Цель данного документа – зафиксировать концепцию решения предложенной задачи для согласования с заказчиком.

Цель данного проекта – создать систему, которая 1) будет информировать людей, которые ухаживают за декоративными растениями, о состоянии этих растений; 2) будет выдавать рекомендации для помощи клиентам в принятии решений по уходу.

Для реализации предлагается использовать технологии Интернета вещей. В составе системы:

1. «датчики» – «умные» устройства, подключенные к Интернету, которые собирают данные о влажности почвы растений и отправляют их в «облако»;
2. приложение Интернета вещей, которое получает данных с датчиков, обрабатывает их, выводит необходимую клиенту информацию в удобном виде через веб-интерфейс, а также информирует клиента по СМС/Email.



Приложение в контексте Интернета вещей - это, по сути, набор "должностных инструкций" для "умных устройств" и средств контроля их исполнения.

Решение, предлагаемое в данном проекте, будет реализовано командой, состоящей из двух стейкхолдеров – IoT-разработчика и UI/UX-дизайнера.

## Описание реализации («как это будет сделано»)

В качестве "умной вещи" для сбора данных может быть выбрана любая система, которая способна измерять значение влажности почвы и отправлять их в Интернет.

В качестве такой системы предлагается использовать комплект SmartKits AGRO.



В состав комплекта входит:

* плата с ардуино-совместимым микроконтроллером и Wi-Fi модулем ESP8266;
* датчик влажности почвы;
* два световых индикатора;
* кнопка;
* «шилд» для удобного подключения к плате дополнительных датчиков;
* соединительные провода.



Комплект измеряет влажность почвы и отправляет данные Интернет через REST HTTP-запрос. Данные отправляются каждые 10 секунд по беспроводному каналу связи с помощью Wi-Fi модуля в составе комплекта.

Собранные данные в итоге приходят в облачное приложение. Приложение позволяет пользователю взаимодействовать с этими данными двумя способами: через веб-интерфейс и через оповещения на электронной почте.

Приложение будет реализовано на платформе Интернета вещей ThingWorx. Для моделирования решения будет использована среда разработки ThingWorx Composer. В ThingWorx Composer будут созданы "цифровые двойники" "умных вещей" и организовано взаимодействие между ними. Затем, физические вещи будут связаны с их цифровыми двойниками, в результате чего поведение физических вещей будет подчиняться поведению их моделей в ThingWorx Composer.

Кроме того, будет создан **веб-интерфейс** для контроля поведения "умных вещей". В данном случае, на интерфейсе будет выводиться информация о состоянии растения и рекомендации к действиям по уходу за ним.

Для создания веб-интерфейса будет использован “drag-and-drop” инструмент Mashup Builder, встроенный в среду ThingWorx Foundation. Взаимодействие между ThingWorx Composer и Mashup Builder реализуется простыми средствами, встроенными в платформу ThingWorx.

Наконец, будет реализована возможность получения **оповещений на электронную почту**. Оповещения позволяют пользователю незамедлительно узнавать о критических ситуациях с растением. Это позволяет ему избежать последствий, которые могут возникнуть из-за длительного пребывания растения в плохих условиях.

## Функциональность решения («как это будет выглядеть»)

Основной функционал веб-интерфейса:

1. вывод общей информации о растении: название, фото, комментарии;
2. вывод текущего состояния почвы растения: значение влажности, его положение относительно критических значений, сообщение о состоянии влажности,
3. вывод текущего состояния растения;
4. вывод истории изменения влажности почвы за последний месяц;
5. метеоданные: текущая влажность и температура воздуха.

Прототип итоговой веб-страницы:



Пример оповещения на электронную почту.

 

## Порядок взаимодействий («как это будет работать»)

**Взаимодействующие объекты (устройства/люди/системы)**

|  |  |
| --- | --- |
| **Объект** | **Свойства объекта, важные для данной задачи** |
| Пользователь | Идентификатор, номер телефона/адрес электронной почты, логин-пароль |
| Контроллируемые растения | Наименование, критические значения влажности, текущее значение влажности, общее состояние, история изменения влажности за последний месяц.Возможно: вид (из «словаря»), категория, местоположение |
| Датчики влажности | Идентификатор, значение влажности, наличие связи с Интернетом, Возможно: заряд батареи |
| Веб-интерфейс | Адрес веб-страницы в Интернете |
| Система оповещения пользователя | Текст для отправки в случае критической ситуации |

**Порядок получения и передачи данных**

* Получение информации с подключенных датчиков - каждые 4 часа.
* Задание идентификатора пользователя, его номера телефона/адреса электронной почты - в любое время через веб-интерфейс.

**Примечание**: на данном этапе эти характеристики будут задаваться командой разработчиков, на следующем шаге будет добавлена эта возможность.

* Задание свойств растений: фото, вид, стадия вегетации, оптимальные значения влажности почвы и пр. - в любое время через веб-интерфейс.

**Примечание**: на данном этапе эти характеристики будут задаваться командой разработчиков, на следующем шаге будет добавлена эта возможность.

* Просмотр информации о растениях – в любое время через веб-интерфейс.
* Получение СМС/Email-оповещений – при наступлении критических условий.

**Основные сценарии взаимодействия объектов**

1. Вывод данных через веб-интерфейс.



1. Занесение свойств растения пользователем.



1. Оповещение о критических ситуациях.



## Ключевые характеристики

В целях обеспечения безопасности, для отправки данных в приложение нужно использовать ключ – т.н. Application Key. Ключ предоставим заказчику.

Веб-интерфейс будет расположен на сервере компании. Доступ к интерфейсу будет по ссылке. Ссылку предоставим заказчику в комплекте с готовым решением. Посмотреть прототип приложения можно уже сейчас по ссылке <https://academic-educatorsextension.portal.ptc.io/Thingworx/Runtime/index.html#mashup=OneDeviceMashupMain_SKAgro&__fromBuilder=7894c781-86c7-4dc2-87f1-8d9f6d4856b0>

Описанные ниже характеристики пока будут задаваться командой разработчиков:

1. изображение и название растения;
2. критические значения влажности;
3. местоположение для получения метеоданных.

На следующем шаге будет добавлена возможность задания вышеуказанных характеристик пользователем приложения.

Кроме того, на следующем шаге будет реализовано два дополнительных интерфейса для взаимодействия с пользоваетем: получение оповещений на электронную почту и вывод состояния растения через интерфейс дополненной реальности.

**Оповещения на электронную почту** позволят пользователю незамедлительно узнавать о критических ситуациях с растением. Это позволит ему избегать негативных последствий, которые могут возникнуть из-за длительного пребывания растения в плохих условиях.

**Интерфейс дополненной реальности** позволит видеть текущее состояние растения на экране мобильного устройства или через очки дополненной реальности. Контент дополненной реальности появляется при наведении камеры устройства (или AR-очков) на специальную метку, которая приклеевается к горшку растения. Для вывода информации может быть использовано мобильное приложение ThingWorx View.

В дальнейшем в приложение может быть добавлен следующий функционал (список отсортирован по увеличению времени реализации):

1. Задание характеристик через веб-интерфейс: изображение и название растения, критические значения влажности, местоположение для получения метеоданных (следующая версия решения).
2. Получение оповещений на электронную почту.
3. Просмотр состояния растения через интерфейс дополненной реальности.

# High-level проектирование пользовательского интерфейса

## Список функций приложения (feature list)

1. Вывод общей информации о растении: название, фото, комментарии и т.д.
2. Вывод текущего состояния растения и его окружающих условий.
3. Вывод истории изменения состояния окружающих условий растения.
4. Задание оптимальных окружающих условий.

## Сценарии использования приложения (central user journey)

1. **Сценарий «основной»**

|  |  |
| --- | --- |
| **Actor** | **Action** |
| Пользователь | Заходит на страницу авторизации |
| Приложение | Выводит страницу авторизации |
| Пользователь | Авторизуется |
| Приложение | Переходит на главную страницу приложения для мониторинга состояния растения |
| Пользователь |  |

# Mid-level проектирование пользовательского интерфейса

*Структура/разметка UI*

## Общая структура

|  |
| --- |
| Header |
| Основной контент |

## Разметка страницы

|  |  |
| --- | --- |
| Лого | Заголовок: Система мониторинга состояния |
| Общая информация о растении и о его состоянии | История изменения состояния окружающих условий |
| Вывод и установка диапазона влажности |
| Вывод местной погоды |

# Low-level проектирование пользовательского интерфейса

## Определение стилей и «состояний» (Style Definitions & State Definitions)

|  |  |  |  |
| --- | --- | --- | --- |
| **Название** | **Тип объекта** | **Описание** | **Параметры** |
| PlantMonitoringTextStyle | Style Definition | Стиль основного текста |  |
| PlantMonitoringHeaderTextStyle | Style Definition | Стиль текста в заголовках |  |
| PlantMonitoringSoilMoistureState | State Definition | Стиль дисплея для отображения влажности почвы. Зависит от текущего состояния почвы: если влажность почвы внутри оптимального диапазона – значение зеленое, если выходит за пределы – красное  |  |

Подробное описание стилей

## Описание виджетов, которые будут использоваться

* Организационные виджеты

|  |  |  |
| --- | --- | --- |
| **Название** | **Назначение** | **Свойства** |
| Layout | Задание общей разметки страницы | Соответствие разметке страницы, описанной в Mid-level |
| Panel | Подложка под другие виджеты области |

* Виджеты для отображения данных

|  |  |  |
| --- | --- | --- |
| **Название** | **Назначение** | **Свойства** |
| Image | Изображение растения | Растянуть на целый слой Layout, без добавления панели |
| Label | Название растения | Стиль:  |
| Состояние растения | Стиль: |
| Заголовки | Стиль: PlantMonitoringHeaderTextStyle |
| Подписи | Стиль: PlantMonitoringTextStyle |
| Slider | Показывает положение текущего значения влажности по отношению к критическим значениям |  |
| Label Chart | График влажности почвы от времени | ChartType: Line/MarkerXAxisField: timestampXAxisFormat: hh:mm:ss aXAxisLabelType: Date/TimeDataField1: SoilMoistureDataLabel1: Влажность почвы |
| LED Display | Текущее значение влажности почвы | LEDSize: SmallDigits: 3Decimals: 0LEDFormatter: PlantMonitoringSoilMoistureState |

* Интерактивные виджеты

|  |  |  |
| --- | --- | --- |
| **Название** | **Назначение** | **Свойства** |
| Auto Refresh | Обновление данных при нажатии на виджет | Интервал между автообновлениями: 30 секундНадпись на виджете: Обновить |
| Numeric Entry | Отображение и изменение критических значений влажности почвы |  |
| Button | Кнопка для изменения критических значений влажности почвы на введенные | Label: Установить новый диапазон |

Примерный вид готовой веб-страницы.

