

Архитектура ПО

Принципы тестирования приложений

Задача:

Разработать тесты под проект системы управления роботом пылесосом (достаточно один протокол):

- 1) Предложить и разработать E2E тесты*
- 2) Предложить и разработать приёмочные тесты (UAT) для сценариев Use case*

Краткая информация о предложенных подходах тестирования.

E2E (End-to-End) сквозное тестирование — это тип тестирования, который имитирует работу системы в реальных условиях. Проверяет, как система работает в целом, включая все ее компоненты и интерфейсы. Тесты имитируют действия пользователя и проверяют, как система реагирует на эти действия.

UAT (User Acceptance Testing) тестирование — это финальный этап тестирования продукта или системы, который проводится перед тем, как продукт передается в эксплуатацию. Тесты проводятся с целью убедиться, что продукт удовлетворяет требованиям и ожиданиям пользователей.

Реализация. E2E тесты.

Выдержка из протокола тестирования при использовании сквозного тестирования.

Тестирование облачного сервиса авторизации		
Наименование сущности	Наименование метода	Описание планируемого взаимодействия
AccessController	checkUserAuth	Проверяет активную авторизацию пользователя. Принимает User, возвращает Boolean.
	createUserAuth	Создает активную авторизацию пользователя. Принимает User, ничего не возвращает.
	removeToken	Удаляет токен авторизации. Принимает Token, ничего не возвращает.
TokenAdapter	getToken	Запрашивает токен авторизации для текущего пользователя. Принимает User, возвращает Token.
	checkToken	Проверяет токен авторизации текущего пользователя. Принимает Token, ничего не возвращает.
	deleteToken	Удаляет токен авторизации. Принимает Token, ничего не возвращает.
Repo*	createToken	Создает токен авторизации и помещает его в базу данных. Принимает String, возвращает Boolean.
	readToken	Ищет и читает токен авторизации из базы данных. Принимает String, возвращает Token.
	updateToken	Обновляет токен авторизации в базе данных. Принимает (Token, String), возвращает Boolean.
	deleteToken	Удаляет токен авторизации из базы данных. Принимает Token, ничего не возвращает.

* Используется вариант реализации высоконагруженного сервиса. Токены хранятся в БД.


Реализация. UAT тесты.

Выдержка из протокола тестирования при использовании функционального тестирования.

Тестирование регистрации и входа в систему		
Задача теста	Визуализация	Описание взаимодействия и проверок
Создание нового аккаунта через опцию «ЗАРЕГИСТРИРУЙТЕСЬ»	 	<ul style="list-style-type: none"> ❖ Переход на новый экран для регистрации: <ul style="list-style-type: none"> -> на экране «Новое устройство» тапнуть по ссылке «ЗАРЕГИСТРИРУЙТЕСЬ» -> осуществляется переход на новый экран ❖ Вводимая информация на экране регистрации: <ul style="list-style-type: none"> -> обязательное заполнение всех полей -> валидность ввода телефона, почты и пароля -> сообщение об ошибке в случае провала проверки валидности -> отправка регистрационных данных на сервер -> получение приложением токена и переход на домашний экран -> смена кнопки помощи на аватар пользователя
Ввод неверного пароля и проверка на наличие сообщения об ошибке	 	<ul style="list-style-type: none"> ❖ Переход по кнопке «ВЫПОЛНИТЕ ВХОД»: <ul style="list-style-type: none"> -> появление дополнительного слоя с вводом данных учетной записи -> обязательное заполнение всех полей -> валидность ввода телефона, почты и пароля -> сообщение об ошибке в случае провала проверки валидности -> отправка регистрационных данных на сервер -> получение приложением токена и переход на домашний экран -> смена кнопки помощи на аватар пользователя
Проверка работы опции "НЕ ПОМНЮ ПАРОЛЬ"	 	<ul style="list-style-type: none"> ❖ Переход по ссылке «НЕ ПОМНЮ ПАРОЛЬ» дополнительного слоя для ввода учетных данных: <ul style="list-style-type: none"> -> появление дополнительного слоя с полями для восстановления пароля по номеру телефона или почте -> отправка данных для восстановления на сервер -> запрос одноразового кода для восстановления пароля -> получение приложением токена и переход на домашний экран -> смена кнопки помощи на аватар пользователя

Выдержка из протокола тестирования при использовании функционального тестирования.

Тестирование основных функций приложения

Задача теста	Визуализация	Описание взаимодействия, проверок и результата
Добавление нового устройства в систему		<ul style="list-style-type: none"> ❖ Переход по кнопке «+» на новый экран для добавления устройств: -> на экране «Добавление нового устройства» происходит поиск устройств – приложение посредством широкого вещания по Bluetooth v5.1 сети рассылает приглашение на присоединение -> найденные устройства поочередно отображаются на экране – проверить работу вариантов взаимодействия: «Добавить», «Вернуться», «Продолжить» ❖ Появление устройства на домашнем экране после нажатия кнопки «Добавить» на странице добавления устройств: -> добавленное устройство отображается на домашнем экране в поле «Мои устройства»
Обновление статуса устройства	 	<ul style="list-style-type: none"> ❖ После добавления устройства приложение обновляет статус устройства и добавляет пиктограммы в соответствии со статусом: <ul style="list-style-type: none"> ● -> выбранные в данный момент устройства для управления ■ -> уровень заполненности контейнера 📶 -> статус подключения устройства к приложению 🔄 -> доступность обновлений 📅 -> отсутствие расписаний 🗺️ -> отсутствие маршрутов 🔋 -> уровень заряда батареи
Обновление микропрограммы устройства	 	<ul style="list-style-type: none"> ❖ Переход по нажатию на пиктограмму устройства на главном экране на экран «Страница устройства» -> на экране «Страница устройства» нажать на кнопку «Обновление», приложение переходит на домашний экран ❖ ИЛИ нажатие на кнопку «Обновление» напрямую на домашнем экране: -> статус устройства меняется на обновление, при этом для выбранного обновляемого устройства становятся недоступны кнопки управления -> кнопки управления становятся доступны для обновляемого устройства сразу после обновления, статус устройства возвращается в обычный вид, пиктограмма с доступными обновлениями деактивируется