**Что такое тестирование?**

**Тестирование программного обеспечения (Software Testing) —** проверка соответствия реальных и ожидаемых результатов поведения программы, проводимая на конечном наборе тестов, выбранном определённым образом.

**В чем цель тестирования?**

**Цель тестирования —** проверка соответствия ПО предъявляемым требованиям, обеспечение уверенности в качестве ПО, поиск очевидных ошибок в программном обеспечении, которые должны быть выявлены до того, как их обнаружат пользователи программы.

**Для чего проводится тестирование ПО?**

* Для проверки соответствия требованиям.
* Для обнаружения проблем на более ранних этапах разработки и предотвращение повышения стоимости продукта.
* Обнаружение вариантов использования, которые не были предусмотрены при разработке. А также взгляд на продукт со стороны пользователя.
* Повышение лояльности к компании и продукту, т. к. любой обнаруженный дефект негативно влияет на доверие пользователей.

**Современные способы тестирования.**

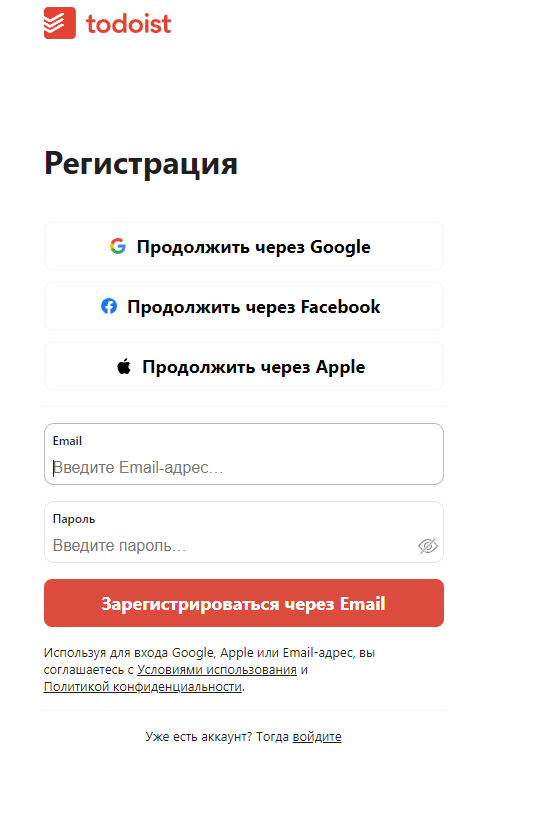
На данный момент существует несколько популярных способов тестирования пользовательских интерфейсов:

1. **Ручное тестирование** - это метод, при котором тестировщики вручную проверяют интерфейс на соответствие требованиям и оценивают его работоспособность. В процессе ручного тестирования тестировщик взаимодействует с интерфейсом, выполняет определенные действия и проверяет, что результаты соответствуют ожиданиям. Ручное тестирование позволяет проверить различные сценарии использования, а также оценить визуальный дизайн и удобность интерфейса.
2. **Автоматизированное тестирование** - это использование специальных инструментов и написание скриптов, которые выполняют определенные тестовые сценарии автоматически. Тестовые скрипты могут симулировать действия пользователя, такие как нажатие кнопок, заполнение форм, навигацию по страницам и т.д. С помощью автоматизированного тестирования можно ускорить процесс тестирования, снизить вероятность человеческой ошибки и повысить надежность результатов.
3. **Тестирование с использованием анализа пользовательских метрик и данных**:
4. В этом виде тестирования данные о пользовательском поведении собираются и анализируются с помощью машинного обучения и статистических методов. Например, можно анализировать данные о времени, проведенном пользователями на страницах, пути продвижения пользователя по сайту, клики на определенные элементы интерфейса и другие метрики. Это может помочь выявить проблемы в интерфейсе, такие как сложности в навигации или непонятные элементы, и предложить улучшения на основе реальных данных.
5. **Тестирование интерфейса пользователя (UI тестирование)—**

это процесс проверки графического интерфейса программного приложения с целью обнаружения ошибок и неполадок. Оно осуществляется путем автоматизации действий пользователя на интерфейсе приложения и последующего анализа результатов.  
UI-тестирование позволяет проверить, что интерфейс приложения соответствует требованиям пользователей и спецификациям. Оно включает в себя проверку внешнего вида, расположения и взаимодействия элементов интерфейса, корректность отображения данных, работу переходов между экранами и другие аспекты, которые влияют на пользовательский опыт.

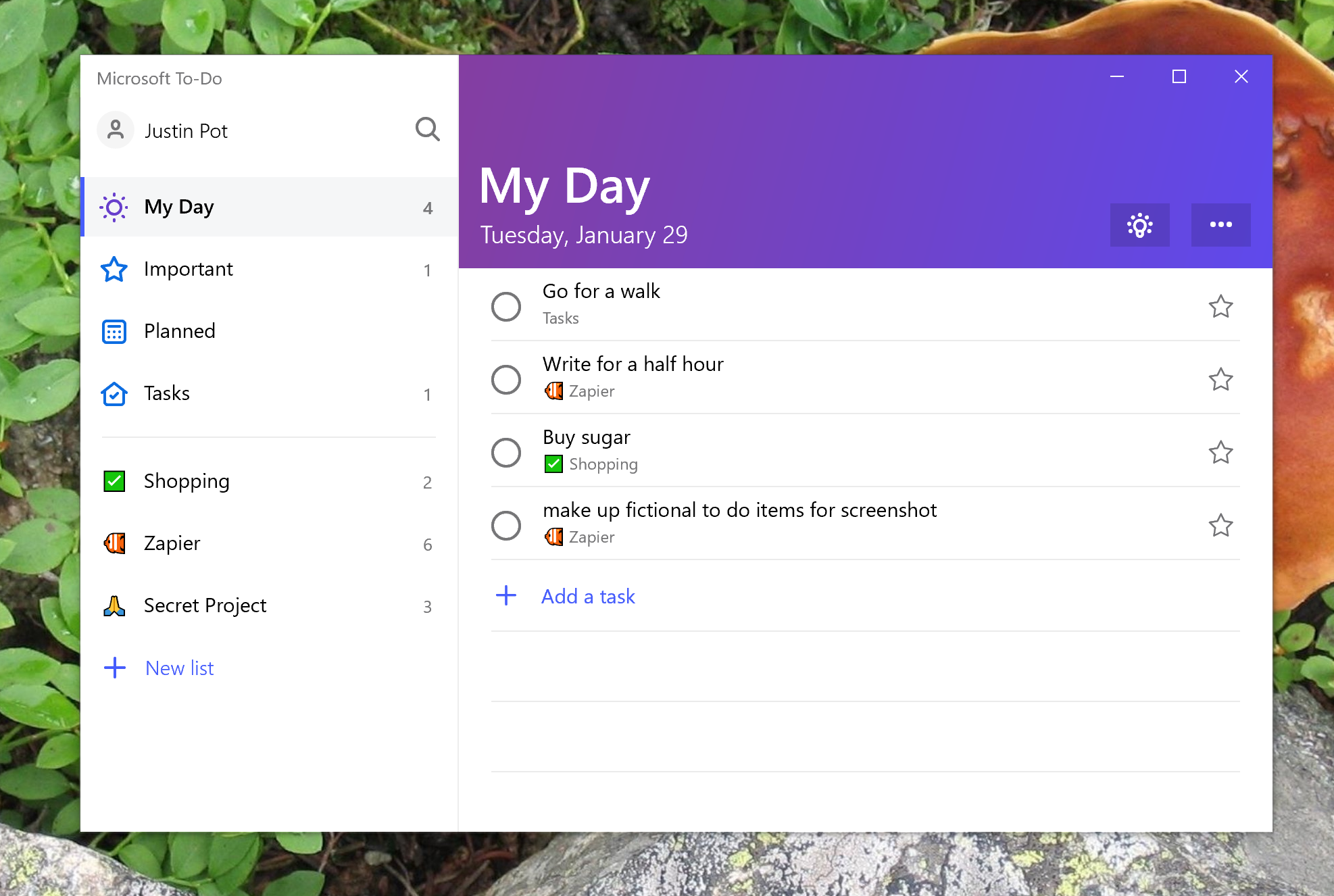
**Пример UI-тестирования на примере приложения ToDoList.**

**Рисунок 1: Пример стартового экрана приложения.**



1. Проверка работоспособности кнопок «Продолжить через Google», «S Продолжить через Facebook», « Продолжить через Apple».
2. Проверка ввода данных в всплывающие окна.
3. Проверка ввода некорректных данных в всплывающие окна.
4. Проверка, что приложение сохраняет данные пользователя после выхода из приложения.
5. Проверка работоспособности гиперссылок «Условия использования» и «Политика конфиденциальности».

**Рисунок 2: Пример главного экрана приложения.**



1. Проверка работоспособности кнопки «Добавить новый список» или другой аналогичной кнопки для выполнения этой команды.
2. Проверка наличия поля для ввода новых задач.
3. Ввод новой задачи в поле ввода.
4. Проверка работоспособности кнопки «Добавить» или другой аналогичной кнопки для добавления задачи.
5. Проверка появления задачи в списке задач.
6. Проверка возможности пометить задачу как выполненную.
7. Проверка, что выполненная задача отображается соответственно или отображается отдельно.
8. Проверка возможности удалить задачу.
9. Проверка, что задача успешно удаляется и исчезает из списка задач.
10. Проверка, что приложение обрабатывает ситуации с некорректными вводными данными, например с пустым полем.
11. Проверка, что введенная задача сохраняется после перезагрузки приложения.
12. Проверка редактирования задачи.
13. Проверка сохранения и отображения задачи в списке.
14. Проверка возможности фильтрации задач по разным категориям или приоритетам.
15. Проверка сортировки задач по датам и другим параметрам.
16. Проверка, что приложение корректно отображается на разных устройствах и разрешениях экрана.
17. Проверка возможности импорта\экспорта списка задач в различных форматах(например, CSV или JSON).
18. Проверка, что приложение корректно обрабатывает ошибки и выводит соответствующие сообщения об ошибках.
19. Проверка работоспособности разных вкладок с задачами.
20. Проверка возможности сменить аккаунт.

После тестирования и вывода продукта на рынок необходима обратная связь от пользователей.

**Зачем нужна обратная связь?**

Сбор обратной связи, анализ отзывов и изменения в услуге на основе полученной информации помогают удерживать старых и привлекать новых клиентов, в конечном счете отражаясь на прибыльности бизнеса.

Существует несколько способов получения и интерпретации обратной связи.

Они могут быть разными в зависимости от контекста и цели обратной связи.

1. **Анкеты и опросы**: Этот метод включает создание структурированных вопросов и их распространение среди целевой аудитории. Анкеты могут быть распространены как в печатном формате, так и в электронном виде через электронную почту, социальные сети или специализированные платформы для проведения опросов. Анкеты могут быть как открытыми (с возможностью свободного ответа), так и закрытыми (с предложенными вариантами ответов). После получения ответов, результаты анкеты анализируются для выявления общих тенденций и взглядов аудитории.
2. **Фокус-группы**:Этот метод включает организацию групповых дискуссий с небольшой группой представителей целевой аудитории. Участники фокус-группы могут обсуждать определенную тему, задавать вопросы друг другу и делиться своими мнениями и отзывами. Фокус-группы могут проводиться лично, по телефону или онлайн через видеоконференции. Результаты дискуссий могут быть записаны и проанализированы для получения ценных выводов.
3. **Открытые форумы:** Организация открытых форумов или платформ, где люди могут свободно высказывать свое мнение и делиться отзывами и идеями. Это может быть веб-сайт, специальный онлайн-форум, чат-бот или социальная сеть. Открытые форумы позволяют людям общаться между собой, задавать вопросы, отвечать на них и обмениваться идеями. Результаты открытых форумов могут быть анализированы для выявления общих мнений и предпочтений.
4. **Обратная связь через социальные сети:** Отслеживание и анализ отзывов и комментариев, оставленных людьми на платформах социальных сетей, таких как Вконтакте, Telegram, Twitter и др. Мнения и отзывы, оставленные публично, могут быть анализированы для определения общего мнения пользователей о продукте, бренде или услуге. Это также может помочь в выявлении проблем и сильных сторон продукта или услуги.
5. **Клиентские обзоры и оценки:** Изучение обзоров, оставленных клиентами на различных платформах, таких как Google, Yelp, Tripadvisor и другие. Клиенты имеют возможность оставить отзывы и оценки после использования продукта или услуги. Это дает возможность получить представление о том, что клиенты думают о вашей компании или продукте.

Интерпретация обратной связи требует анализа полученных данных и выделения ключевых выводов. Важно обратить внимание не только на конкретные мнения, но и на общие тенденции и паттерны. Кроме того, автоматизация обработки обратной связи с помощью машинного обучения и анализа данных может существенно упростить этот процесс.