

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
Государственное бюджетное образовательное учреждение Российской Федерации
«Кубанский государственный университет»
Прикладная информатика

Предмет: Технология создания и поддержки ПО
Лабораторная работа №_1

Выполнил студен:
Оралов Кирилл Александрович
Группа: ПИ_23

г. Краснодар, 2026

Цель лабораторной работы:

Знакомство с методиками “Пользовательских историй” и “Вариантов использования” получение навыков работы с описаниями вариантов использования и пользовательских историй.

Задачи лабораторной работы:

1. **Теоретическая:** Изучить концепции и отличия пользовательских историй (ПИ) и вариантов использования (ВИ) как инструментов описания функциональных требований.
2. **Аналитическая:** Выявить и систематизировать ключевые функции современного приложения для управления задачами с точки зрения конечного пользователя.
3. **Практическая:**
 - Сформулировать набор из 15 пользовательских историй, отражающих основные, расширенные и специализированные потребности пользователя.
 - Для каждой пользовательской истории разработать подробный вариант использования, включающий цель, актора, начальное состояние, основной сценарий, альтернативный сценарий.

1. Практическая часть

Пользовательские истории (ПИ)

(Приложение для списка задач)

1. Как пользователь, я могу вносить свои задачи, для того, чтобы структурировать информацию
2. Как пользователь я могу объединять задачи на общие темы в один раздел
3. Как пользователь я могу добавлять теги к задачам, чтобы быстро находить связанные задачи через поиск или фильтрацию
4. Как пользователь я могу группировать задачи по степени важности
5. Как пользователь я могу выбирать приоритетные задачи из нескольких равнозначных
6. Как пользователь я могу назначать сроки выполнения задачи
7. Как пользователь я могу получать уведомления о приближающихся сроках выполнения задач, чтобы не пропустить важные дедлайны
8. Как пользователь я могу отслеживать прогресс завершения задачи

9. Как пользователь я могу открывать две вкладки, где в первой вкладке могу отслеживать задачи по разным разделам, а во второй прогресс задач из разных разделов
10. Как пользователь я могу прикреплять файлы (документы, изображения, ссылки) к задачам, чтобы хранить всю необходимую информацию в одном месте
11. Как пользователь я могу архивировать завершенные задачи, чтобы они не загромождали текущий список, но оставались доступными для просмотра истории
12. Как пользователь я могу менять тему приложения на с белой на черную и наоборот
13. Как пользователь я могу использовать умный поиск по задачам с фильтрами по дате, приоритету, тегам и статусу выполнения
14. Как пользователь я могу вести временную статистику по затраченному времени на каждую задачу
15. Как пользователь я могу открыть вкладку "Архив" для того, чтобы отслеживать выпаденные задачи

2. Варианты использования (ВИ)

1. ПИ: Внесение задач

1) Цель "Создание новой задачи"

2) Актор: Пользователь

3) Основной сценарий:

1. Пользователь открывает приложение
2. Нажимает кнопку "+" или "Добавить задачу"
3. Вводит название задачи в текстовое поле
4. Добавляет описание
5. Нажимает "Сохранить"

Результат: Задача добавлена в общий список

4) Альтернативный сценарий №1

Предусловие: на шаге 3 основного сценария пользователь не ввёл название задачи.

1. Система отображает сообщение об ошибке: «Название задачи не может быть пустым».
2. Пользователь вводит название и продолжает с шага 4 основного сценария.

2. ПИ: Объединение задач в разделы

1) Цель : "Создание проекта из нескольких задач"

2) Актор: Пользователь

3) Основной сценарий:

1. Пользователь выбирает несколько задач в списке (через чекбоксы)
2. Нажимает "Объединить в раздел"
3. Вводит название раздела
4. Нажимает "Создать"

Результат: Создан новый раздел с выбранными задачами

4) Альтернативный сценарий №1:

Предусловие: на шаге 1 основного сценария пользователь не выбрал ни одной задачи.

1. Кнопка «Объединить в раздел» неактивна или система отображает подсказку: «Выберите хотя бы одну задачу».
 2. Пользователь выбирает задачи и продолжает с шага 2 основного сценария.
-

3. ПИ: Добавление тегов

1) Цель : "Категоризация рабочей задачи"

2) Актор: Пользователь

3.) Основной сценарий:

1. Пользователь открывает существующую задачу
2. Нажимает на поле "Теги"
3. Выбирает из предложенных или вводит новый тег (например, "Работа", "Срочно")
4. Добавляет несколько тегов через запятую
5. Сохраняет изменения

Результат: Задача помечена тегами для быстрого поиска

4) Альтернативный сценарий №1:

Предусловие: на шаге 3 основного сценария пользователь ввёл недопустимый символ в названии тега.

1. Система отображает сообщение об ошибке: «Недопустимый символ в названии тега».
2. Пользователь исправляет ввод и продолжает с шага 4 основного сценария.

4. ПИ: Группировка по важности

1) Цель : "Сортировка задач перед совещанием"

2) Актор: Пользователь

3) Основной сценарий:

1. Пользователь открывает настройки отображения
2. Выбирает "Группировать по важности"
3. Назначает каждой задаче уровень важности:
Высокая/Средняя/Низкая
4. Приложение автоматически сортирует задачи по группам

Результат: Задачи отображаются в трех колонках по важности

4) Альтернативный сценарий №1:

Предусловие: на шаге 3 основного сценария пользователь отменил действие до завершения.

1. Система сохраняет предыдущие настройки отображения.
 2. Группировка по важности не применяется.
-

5. ПИ: Выбор приоритетных задач

1) Цель : "Определение задач на сегодня"

2) Актор: Пользователь

3) Основной сценарий:

1. Пользователь видит список из 10 задач с одинаковым сроком
2. Нажимает "Выбрать приоритетные"
3. Отмечает звездочкой 3 самые важные задачи
4. Фильтрует список по "Только приоритетные"

Результат: Отображаются только выбранные задачи для фокусировки

4) Альтернативный сценарий №1:

Предусловие: на шаге 4 основного сценария пользователь снимает фильтр «Только приоритетные».

1. Система отображает полный список задач.
 2. Приоритетные задачи остаются отмеченными звездочкой.
-

6. ПИ: Назначение сроков

1) Цель : "Планирование дедлайна проекта"

2) Актор: Пользователь

3) Основной сценарий:

- Пользователь создает задачу "Сдать отчет"
- Нажимает на иконку календаря рядом с задачей
- Выбирает дату 15.03.2024 из виджета календаря
- Устанавливает время 18:00
- Подтверждает выбор

Результат: К задаче привязан срок выполнения

4) Альтернативный сценарий №1:

Предусловие: на шаге 3 основного сценария пользователь выбрал прошедшую дату.

- Система отображает предупреждение: «Выбранная дата уже прошла».
- Пользователь выбирает корректную дату и продолжает с шага 4 основного сценария.

7. ПИ: Уведомления о сроках

1) Цель: "Напоминание о встрече"

2) Актор: Система (автоматически)

3) Основной сценарий:

- За 2 часа до дедлайна задачи система проверяет ее статус
- Если задача не завершена, отправляет push-уведомление
- Пользователь получает сообщение: "Через 2 часа: Встреча с клиентом"
- Пользователь может отложить напоминание на 15 минут

Результат: Пользователь вовремя предупрежден о дедлайне

4) Альтернативный сценарий №1:

Предусловие: на шаге 2 основного сценария задача уже помечена как выполненная.

- Система не отправляет уведомление.
- Сценарий завершается.

8. ПИ: Отслеживание прогресса

1) Цель: "Отметка этапов выполнения"

2) Актор: Пользователь

3) Основной сценарий:

- Пользователь работает над задачей "Написать код модуля"
- После завершения первого этапа нажимает "Обновить прогресс"
- Перетаскивает ползунок прогресса с 0% на 30%
- Добавляет комментарий: "Завершена архитектура"

5. Сохраняет изменения

Результат: Прогресс задачи визуализирован в списке

4) Альтернативный сценарий №1:

Предусловие: на шаге 5 основного сценария пользователь отменяет сохранение.

1. Изменения не сохраняются, прогресс остаётся прежним.
 2. Система возвращает пользователя к карточке задачи.
-

9. ПИ: Две вкладки интерфейса

1) Цель: "Параллельный просмотр задач и прогресса"

2) Актор: Пользователь

3) Основной сценарий:

1. Пользователь открывает приложение
2. Видит две вкладки: "Все задачи" и "Прогресс"
3. В первой вкладке сортирует задачи по разделам
4. Переключается на вторую вкладку
5. Видит три раздела: Ожидают выполнения, В процессе, Выполнены

Результат: Единый обзор структуры и статуса всех проектов

10. ПИ: Прикрепление файлов

1) Цель: "Добавление документа к задаче"

2) Актор: Пользователь

3) Основной сценарий:

1. Пользователь открывает задачу
2. Нажимает "Прикрепить файл"
3. Выбирает "Загрузить с устройства"
4. Выбирает файл presentation.docx

Результат: Файл доступен в карточке задачи

11. ПИ: Архивация задач

1) Цель: "Очистка списка после завершения проекта"

2) Актор: Пользователь

3) Основной сценарий:

1. Пользователь завершает все задачи в разделе "Квартальный отчет"

2. Выбирает раздел и нажимает "Архивировать"
3. Подтверждает действие
4. Раздел исчезает из основного списка
5. Пользователь может найти его в меню "Архив"

Результат: Актуальный список очищен, история сохранена

1. 4) Альтернативный сценарий №1:

Предусловие: на шаге 3 основного сценария пользователь отменяет подтверждение.

1. Архивация отменяется.
 2. Раздел остаётся в основном списке.
-

12. ПИ: Смена темы

1) Цель: "Переключение на темную тему вечером"

2) Актор: Пользователь

3) Основной сценарий:

1. Пользователь открывает настройки приложения
2. Находит раздел "Внешний вид"
3. Переключает тумблер с "Светлая" на "Темная"
4. Интерфейс мгновенно меняет цвета

4) Результат: Приложение отображается в темной цветовой схеме

13. ПИ: Умный поиск

1) Цель: "Поиск всех срочных задач на этой неделе"

2) Актор: Пользователь

3) Основной сценарий:

1. Пользователь открывает поисковую строку
2. Вводит "срочно"
3. Открывает фильтры
4. Устанавливает "Дата: Эта неделя"
5. Добавляет фильтр "Статус: В процессе"
6. Нажимает "Найти"

Результат: Отображены все срочные незавершенные задачи на текущей неделе

4) Альтернативный сценарий №1:

Предусловие: на шаге 6 основного сценария не найдено ни одной задачи.

1. Система отображает сообщение: «По вашему запросу ничего не найдено».
2. Пользователь может изменить критерии поиска.

14. ПИ: Временная статистика

1) Цель: "Анализ времени за неделю"

2) Актор: Пользователь

3) Основной сценарий:

1. Пользователь заходит в раздел "Аналитика"
2. Выбирает период "Прошлая неделя"
3. Нажимает на задачу "Разработка"
4. Видит график: Пн - 4ч, Вт - 6ч, Ср - 3ч...

Результат: Наглядная статистика распределения времени

4) Альтернативный сценарий №1:

Предусловие: на шаге 2 основного сценария выбран период, за который нет данных.

1. Система отображает сообщение: «За выбранный период данных нет».
2. Пользователь выбирает другой период.

15. ПИ: Экспорт задач

1) Цель: "Подготовка отчета для руководителя"

2) Актор: Пользователь

3) Основной сценарий:

1. Пользователь фильтрует задачи по разделу "Проект X"
2. Нажимает "Экспорт"
3. Выбирает формат "PDF"
4. Нажимает "Скачать"
5. Получает файл project_x_15_03_2024.pdf

Результат: Структурированный отчет в удобном формате

4) Альтернативный сценарий №1:

Предусловие: на шаге 4 основного сценария произошла ошибка генерации PDF.

1. Система отображает сообщение: «Не удалось создать файл.
Попробуйте ещё раз».
2. Пользователь может повторить экспорт или выбрать другой формат.

3. Контрольные вопросы

6. Для чего используются ПИ и ВИ?

ПИ - быстрый способ документировать требования, без необходимости разрабатывать обширные формализованные документы и впоследствии тратить ресурсы на их поддержание

ВИ - формальное описание взаимодействия системы и пользователя при решении конкретной задачи. Каждый ВИ нацелен на конкретную задачу и описывает некоторое функциональное требование.

ВИ описывает, «кто» и «что» может сделать с рассматриваемой системой, или что система может сделать с «кем» или «чем».

7. Какие обстоятельства затрудняют применение ПИ?

- Зыбкость. Без определенных приемочных испытаний ПИ являются открытыми для различных интерпретаций, что усложняет их использование как основу для соглашения.
- Затратность. Требуют близкого контакта с клиентом на протяжении всего проекта, что в некоторых случаях может быть сложно либо приводить к накладным затратам.
- Плохая масштабируемость. ПИ могут плохо масштабироваться на больших проектах.
- Полагаются на компетентность разработчиков.
- Отсутствие понятной формализации. Используются для начала дискуссии. К сожалению, ПИ могут не фиксировать окончание дискуссии и таким образом не в состоянии служить надежным методом документации системы.

3. Какой уровень детализации традиционно применяется при описании ВИ?

Детализированный. На этом уровне ВИ представляет собой формальный документ, основанный на подробном шаблоне с различными разделами. Именно этот вариант (детализированный) подразумевается в большинстве случаев под понятием варианта использования.

4. Заключение

Познакомились с методиками “Пользовательских историй” и “Вариантов использования”. Получили навыки работы с описаниями вариантов использования и пользовательских историй.