

Тест-план для мобильного приложения «Punctualis»

1. Цель тестирования

Цель тестирования — обеспечить всестороннюю проверку корректности, устойчивости и безопасности работы мобильного приложения «Punctualis», предназначенного для планирования задач, анализа продуктивности и взаимодействия с ИИ-ассистентом. Особое внимание уделяется пользовательским сценариям, отображению данных, логике взаимодействия и защите информации.

2. Объект тестирования

Объект тестирования — мобильное приложение «Punctualis», разработанное на Flutter (Dart) с серверной частью на FastAPI (Python) и базой данных PostgreSQL. Приложение доступно на устройствах под управлением Android версии 8.0 и выше.

3. Среда тестирования

- Устройства: смартфоны под управлением Android 8.0 и выше (например, Xiaomi, Samsung, Realme)
- Разрешение экранов: от 720×1280
- Сетевое соединение: стабильное подключение через Wi-Fi или мобильную сеть (4G)
- База данных: PostgreSQL (тестовая копия)
- Серверное окружение: Docker, Redis, REST API с HTTPS
- Клиентская часть: Flutter SDK 3.29.2, язык Dart
- Серверная часть: Python 3.12, FastAPI

4. Типы тестирования

- Функциональное тестирование: проверка заявленного функционала и сценариев взаимодействия пользователя с приложением
- Интерфейсное тестирование (UI/UX): оценка удобства, логики и доступности пользовательского интерфейса
- Нагрузочное тестирование: оценка стабильности и отзывчивости приложения при большом числе запросов

- Регрессионное тестирование: повторная проверка уже протестированных компонентов после внесения изменений
- Кроссплатформенное тестирование: оценка корректной работы приложения на разных версиях Android и устройствах
- Тестирование доступности: проверка наличия необходимых подсказок, читаемости текста, масштабируемости элементов интерфейса

5. Методы тестирования

- Ручное тестирование: используется для проверки пользовательского интерфейса, навигации и визуального отображения
- Автоматизированное тестирование: применяется для REST API, проверки повторяющихся сценариев (например, регистрация, логин, CRUD-операции)
- Тестирование безопасности: включает проверку HTTPS, JWT и хеширования паролей

6. Тестовые роли и учетные записи

- Гость (неавторизованный пользователь)
- Пользователь (авторизованный пользователь)
- Администратор (пользователь с правами управления системой)

Для каждой роли создаются отдельные тестовые учетные записи. Тестовая база данных содержит демонстрационные данные: задачи, теги, профили, события.

7. Критерии входа в тестирование

- Завершена реализация функциональных модулей
- Успешно собрана тестовая сборка приложения
- Осуществлена настройка тестовой базы данных
- Осуществлён деплой серверной части

8. Критерии выхода из тестирования

- Пройдены все тест-кейсы с результатами "успешно"
- Устранены все критические и высокоприоритетные дефекты

- Сформированы и представлены итоговые отчеты
- Приложение стабильно работает при базовой и средней нагрузке
- Пользовательские сценарии выполняются без сбоев

9. Инструменты тестирования

- Android Studio (эмуляторы)
- Swagger (API-тестирование)
- VS Code (логика backend, логирование)
- Excel (ведение тест-кейсов)
- GitHub (репозиторий, CI/CD)
- Jira или Notion (таск-трекинг, отчётность)