**Задание 1**

**1. Обзор проекта**

**Цель:** Обеспечение качества альфа-версии интернет-магазина перед релизом.  
**Сроки:**

* Альфа-версия (2 недели) → Тестирование MVP
* Бета-версия (2 месяца) → Полноценное тестирование
* Релиз (3 месяца) → Финализация
* Поддержка (1 месяц) → Мониторинг и исправление критических багов

**2. Этапы тестирования**

**2.1. Подготовительный этап (1-я неделя)**

**Действия:**  
 **Анализ требований**

* Проверка документации (SRS, user stories, макеты интерфейса).
* Формирование чек-листа ключевых функций.

**Планирование тестирования**

* Составление тест-плана (стратегия, приоритеты, ресурсы).
* Оценка трудоемкости (разбивка по модулям).

**Подготовка тестовой среды**

* Развертывание тестового стенда (DEV, STAGING).
* Настройка CI/CD (если есть).

**Тест-дизайн**

* Разработка тест-кейсов (Smoke, Functional, UI, API).
* Подготовка тестовых данных (валидные/невалидные сценарии).

**Метрики:**

* Покрытие требований тест-кейсами ≥80%.

**2.2. Альфа-тестирование (2-3 недели)**

**Цель:** Проверка базовой функциональности (MVP).  
**Виды тестирования:**  
 **Функциональное тестирование** – проверка:

* Просмотра каталога (фильтры, поиск, пагинация).
* Добавления товаров в корзину.
* Оформления заказа (валидация полей, логика).
* Выбора способа доставки.
* Интеграции с платежным шлюзом (мок-тесты).

**UI-тестирование** – проверка вёрстки, адаптивности, accessibility.

**API-тестирование** (Postman/Swagger) – проверка эндпоинтов.

**Риски:**

* Нестабильность альфа-версии → Приоритет на Smoke-тесты.
* Ограниченное время → Фокус на критический функционал.

**Результат:**

* Отчет о багах (Jira/YouTrack).
* Рекомендации по доработке.

**2.3. Бета-тестирование (4-10 недели)**

**Цель:** Глубокое тестирование + нефункциональные проверки.  
**Виды тестирования:**

* **Регрессионное тестирование** – проверка исправленных багов.
* **Интеграционное тестирование** – взаимодействие модулей.
* **Нагрузочное тестирование** (JMeter) – проверка отказоустойчивости.
* **Юзабилити-тестирование** – удобство интерфейса.
* **Кросс-браузерное/кросс-девайс тестирование** (BrowserStack).

**Метрики:**

* Количество открытых/закрытых багов.
* % автоматизированных тестов (цель – 30-50%).

**2.4. Релиз и поддержка (11-16 недели)**

**Действия:**  
🔸**Финал-тестирование** – проверка всех критических сценариев.  
🔸 **Мониторинг после релиза** – сбор багов от пользователей.  
🔸 **Анализ метрик** – оценка стабильности (например, % успешных платежей).

**Документация:**

* Итоговый отчёт о тестировании.
* Рекомендации по улучшению.

**3. Ресурсы**

**Внутренние:**

* Тест-кейсы (TestRail/Qase).
* Баг-трекер (Jira/YouTrack).
* Автотесты (PyTest + Selenium).

**Сторонние:**

* BrowserStack (кросс-браузерное тестирование).
* JMeter (нагрузка).
* Postman (API).

**4. Заключение**

**Ключевые принципы:**  
✔ **Раннее тестирование** – вовлечение QA с этапа проектирования.  
✔ **Приоритизация** – фокус на критических сценариях.  
✔ **Гибкость** – адаптация к изменениям требований.