ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 11
ТЕСТИРОВАНИЕ ИНТЕГРИРОВАННОГО ПРОТОТИПА

Дисциплина: CSE842 «Capstone Project». **Образовательная программа:** 6B06102 «Computer Science», 4 курс, 8 семестр. **Связь с лекцией:** лекция 11 «Верификация и валидация программной системы». **Результаты обучения:** РО2, РО6. **Объём:** 3 часа практики, 7 часов самостоятельной работы, включая СРСП. **Максимальный балл:** 4.

Цель занятия: проверить нормальные, граничные и ошибочные сценарии интегрированного прототипа, провести системные и приёмочные испытания и оформить протокол.

Задачи:

1. Дополнить модульные и интеграционные тесты до покрытия требований Must и Should.
2. Применить тестовые двойники и измерить покрытие ветвей.
3. Провести системные испытания NFR и приёмочные испытания с пользователями.
4. Оформить протокол испытаний и зарегистрировать дефекты.

Программное обеспечение и инструменты: unittest/pytest, unittest.mock, coverage.py, Locust или k6, pip-audit, bandit, behave (по желанию).

1. Краткие теоретические сведения

Тест обнаруживает дефект только при выполнении условий RIPR: достижимость, заражение состояния, распространение, обнаружение оракулом. Граничных и ошибочных сценариев должно быть больше, чем нормальных. Внешние зависимости заменяются двойниками (stub, mock, fake, spy), время — фиксированными часами. Испытания завершаются по критериям выхода.

Пример теста границы:

```
def test_boundary_touching(self):
    # 10-12 и 12-14 не пересекаются
    self.assertFalse(overlaps((10, 12), (12, 14)))
```

Таблица 1 — Шаблон сводки протокола испытаний

Требование	Норм.	Гран.	Ошиб.	Итог	Дефекты
FR-1	4	3	5	12/12	—
FR-2	2	4	2	7/8	BUG-14
NFR-1	1	—	—	1/1	—

2. Порядок выполнения работы

1. Дополните тесты по матрице трассируемости; назовите тесты по виду сценария (normal, boundary, error).
2. Реализуйте сценарии варианта (таблица 2) как автоматические тесты.
3. Замените внешние зависимости двойниками; внедрите фиксированные часы.

4. Измерьте покрытие операторов и ветвей ядра; добавьте тесты для непокрытых ветвей.
5. Выполните системные испытания NFR и проверки безопасности (403, инъекции, pip-audit).
6. Проведите юзабилити-тест с 3–5 пользователями по сквозному сценарию.
7. Оформите протокол испытаний и зарегистрируйте дефекты в трекере.

3. Индивидуальные задания

Номер варианта совпадает с номером темы проекта, выбранной на практическом занятии № 1, и сохраняется до конца семестра. Команда (до 4 человек) выполняет задание своего варианта; распределение работы между участниками указывается в отчёте.

Таблица 2 — Индивидуальные задания: сценарии испытаний

№	Граничные сценарии	Ошибочные сценарии	Системное испытание NFR
1	слот, начинающийся сейчас; последний слот дня	двойное нажатие «Записаться»; отмена чужой записи	подтверждение записи ≤ 2 с при 100 одновременных пользователях
2	токен на 29-й и 31-й секунде; последний студент группы	повторное сканирование; чужой токен	отметка 200 студентов за 2 мин
3	значение ровно на пороге; пустой интервал	отрицательная концентрация; дубликат измерения	задержка от датчика до дашборда ≤ 5 с
4	решение ровно на лимите времени; пустой файл	fork-бомба; попытка сетевого запроса	50 решений в минуту, отчёт ≤ 30 с
5	покупка последней единицы; корзина из 1 товара	два заказа на последнюю единицу; отрицательное количество	страница каталога ≤ 1 с; рекомендации ≤ 300 мс
6	возврат в последний день срока; лимит 5 книг	выдача уже выданного экземпляра; неизвестный штрихкод	поиск ≤ 200 мс на 100 тыс. записей
7	вопрос из 1 слова; 4000 символов	эмодзи и стикеры; вопрос на казахском	ответ ≤ 3 с
8	тень на месте; частично заехавший автомобиль	камера отключена; пустой кадр	обновление карты ≤ 5 с

№	Граничные сценарии	Ошибочные сценарии	Системное испытание NFR
9	отгрузка ровно остатка; пустая ячейка	повторная отправка операции; отгрузка сверх остатка	сканирование и запись ≤ 1 с
10	курьер ровно на границе зоны; последний свободный курьер	два диспетчера назначают одновременно; GPS-скачок	положение курьера на карте с задержкой ≤ 2 с
11	окно на границе суток; пустое окно	повреждённая строка журнала; неизвестный шаблон	обработка ≥ 1000 строк/с
12	чек с суммой 0; очень длинный чек	размытое фото; повторный чек	обработка чека ≤ 3 с
13	согласование последним участником маршрута; пустое поле шаблона	изменение после подписи; повторное согласование	300 заявлений в день, PDF ≤ 2 с
14	оплата на 10-й минуте блокировки; последний слот	повторный webhook; отказ оплаты	p95 ≤ 500 мс при 100 RPS
15	два правила на одно событие; датчик на границе расписания	устройство офлайн; повтор события	реакция «датчик → свет» ≤ 300 мс
16	пропуск истекает сегодня; номер с похожими символами 0/O	грязный номер; два автомобиля в кадре	решение ≤ 1 с от кадра
17	переход на летнее время; ровно порог перерасхода	счётчик сбросился в 0; пропуск 6 ч	дашборд ≤ 2 с за год данных
18	ответ за 1 с до конца; последний вопрос	две вкладки с одной попыткой; разрыв сети	p95 ≤ 300 мс при 300 одновременных студентах
19	отправление ровно в момент запроса; последняя пересадка дня	одинаковые точки; повреждённый фид	построение маршрута ≤ 1 с
20	слот вплотную к паре; изменение во время записи	удаление события в Google; истёк токен	синхронизация изменений ≤ 1 мин

4. Содержание отчёта

1. Протокол испытаний (объект, тесты, результаты, дефекты, покрытие, выводы).
2. Отчёт CI и отчёт coverage.
3. Результаты системных и приёмочных испытаний.
4. Список зарегистрированных дефектов.

5. Форма сдачи

Артефакты занятия размещаются в репозитории проекта (каталог docs/ или указанный в задании), в трекере задач создаются и закрываются соответствующие задачи. Преподавателю сдаётся ссылка на версию (коммит или тег) и краткий отчёт.

6. Критерии оценивания

Максимальный балл за работу — **4** (календарный план силлабуса). Баллы распределяются по трём критериям силлабуса.

Таблица 3 — Критерии оценивания текущего задания

Критерий	Доля	Что проверяется
Полнота и соответствие требованиям задания	40%	все пункты задания и индивидуального варианта выполнены
Техническая корректность	40%	решения обоснованы, расчёты и код верны, терминология точна
Проверяемость материалов	20%	артефакты в репозитории, ссылки, команды запуска, данные

Работа сдаётся через канал, указанный преподавателем; в командном проекте каждый участник указывает свой вклад. Использование генеративного ИИ раскрывается по правилам университета.

7. Контрольные вопросы

1. Что такое условия RIPR?
2. Какие двойники вы использовали и зачем?
3. Каково покрытие ветвей ядра?
4. Какие NFR подтверждены испытаниями?
5. Что показал юзабилити-тест?

Литература

1. Ammann P., Offutt J. Introduction to Software Testing. — 2nd ed. — Cambridge University Press, 2016.
2. IEEE Computer Society. SWEBOK Guide v4.0a.
3. NASA. Systems Engineering Handbook. NASA/SP-2016-6105 Rev2.