ЛЕКЦИЯ 7

Проектирование проверок и планирование интеграции

CSE842 Capstone Project
ОП 6B06102 Computer Science



План лекции

1. V&V
2. V-модель
3. Стратегия
4. Классы
5. Покрытие
6. Трассировка
7. Интеграция
8. CI

Верификация и валидация



Верификация

- правильно ли строим?
- по спецификации
- тесты, рецензии



Валидация

- то ли строим?
- задача пользователей
- демонстрация, пилот

V-модель: этап и проверка

Проектирование	Уровень проверки
паспорт, бизнес-цели	приёмочное тестирование
SRS: FR и NFR	системное тестирование
архитектура, контракты	интеграционное
модули	модульное

Тесты проектируются вместе с артефактом

Пирамида тестов



Критерии входа и выхода испытаний

Техники проектирования тестов



Классы

один представитель класса



Границы

0, 1, 240, 241 мин



Таблицы решений

роль × статус × конфликт

Тесты FR-2: классы и границы

№	Новая бронь	Класс	Ожидание
T2	11:00–13:00	частичное	409
T3	10:30–11:30	вложенный	409
T4	12:00–14:00	граница: смежные	201
T5	11:59–14:00	граница: 1 мин	409
T7	12:00–10:00	ошибка	422

Существующая бронь 10:00–12:00

Критерии покрытия: цели

100%

требований Must с тестом

≥ 80%

требований Should

≥ 70%

операторов ядра

≥ 60%

ветвей ядра

Тест без утверждения даёт покрытие, но не находит дефекты

Матрица трассируемости

l07_trace.py

```python
matrix = {  # требование -> (компонент, тесты)
    "FR-1": ("BookingService", ["test_book_ok"]),
    "FR-2": ("BookingService", ["test_overlap_409",
                                "test_touching_ok"]),
    "FR-3": ("Notifier", []),
    "NFR-1": ("API + SQLite", ["perf_p95"]),
}
gaps = [r for r, (_, t) in matrix.items() if not t]
cov = 1 - len(gaps) / len(matrix)
print(f"покрытие требований тестами: {cov:.0%}")
print("без тестов:", ", ".join(gaps))
order = ["SQLite repo", "BookingService", "HTTP API",
         "Notifier", "Web UI"]
for i, c in enumerate(order, 1):
    print(f"шаг {i}: интегрировать {c}")
```

```
$ python l07_trace.py
покрытие требований тестами: 75%
без тестов: FR-3
шаг 1: интегрировать SQLite repo
шаг 2: интегрировать BookingService
шаг 3: интегрировать HTTP API
шаг 4: интегрировать Notifier
шаг 5: интегрировать Web UI
```

- покрытие 75%
- FR-3 без теста
- план: тест с имитацией SMTP

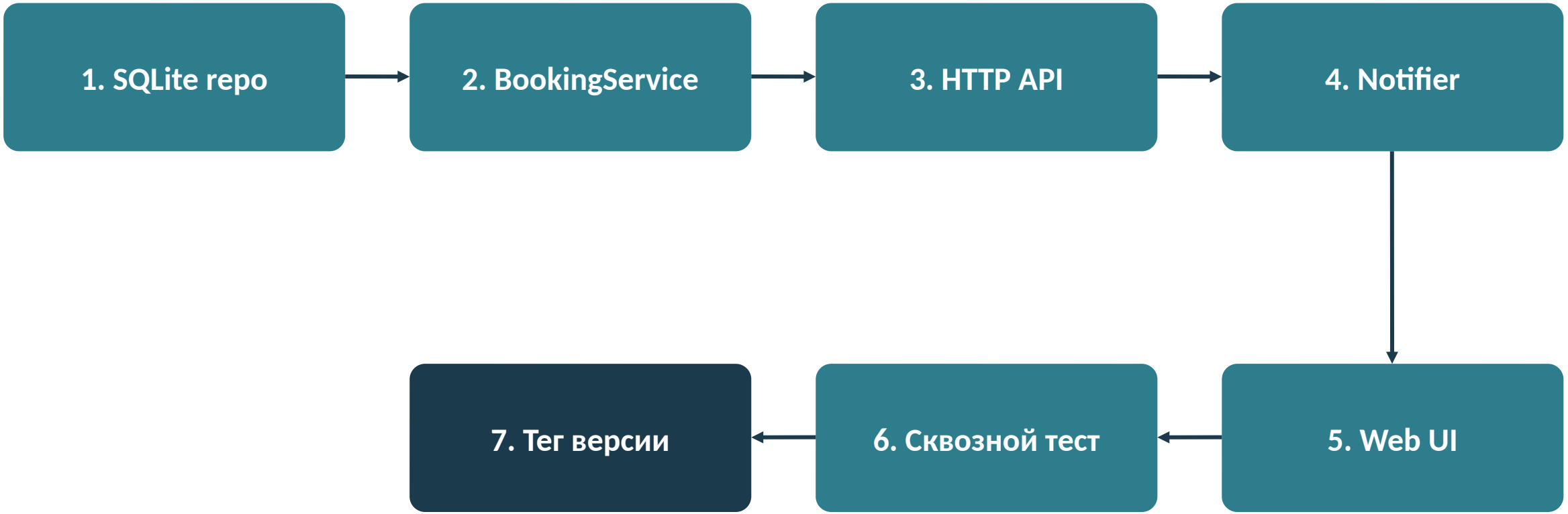
Стратегии интеграции



Большой взрыв	Нисходящая	Восходящая	Скелет
всё сразу в конце — риск	сверху вниз, заглушки	снизу вверх, драйверы	сквозной срез сразу

Рекомендуется «ходячий скелет»

Порядок интеграции LabBook



Сначала то, от чего зависят другие, и самое рискованное

Заглушки, драйверы, имитации



Заглушка

фиксированный ответ вместо модуля



Драйвер

тестовый код вызывает модуль

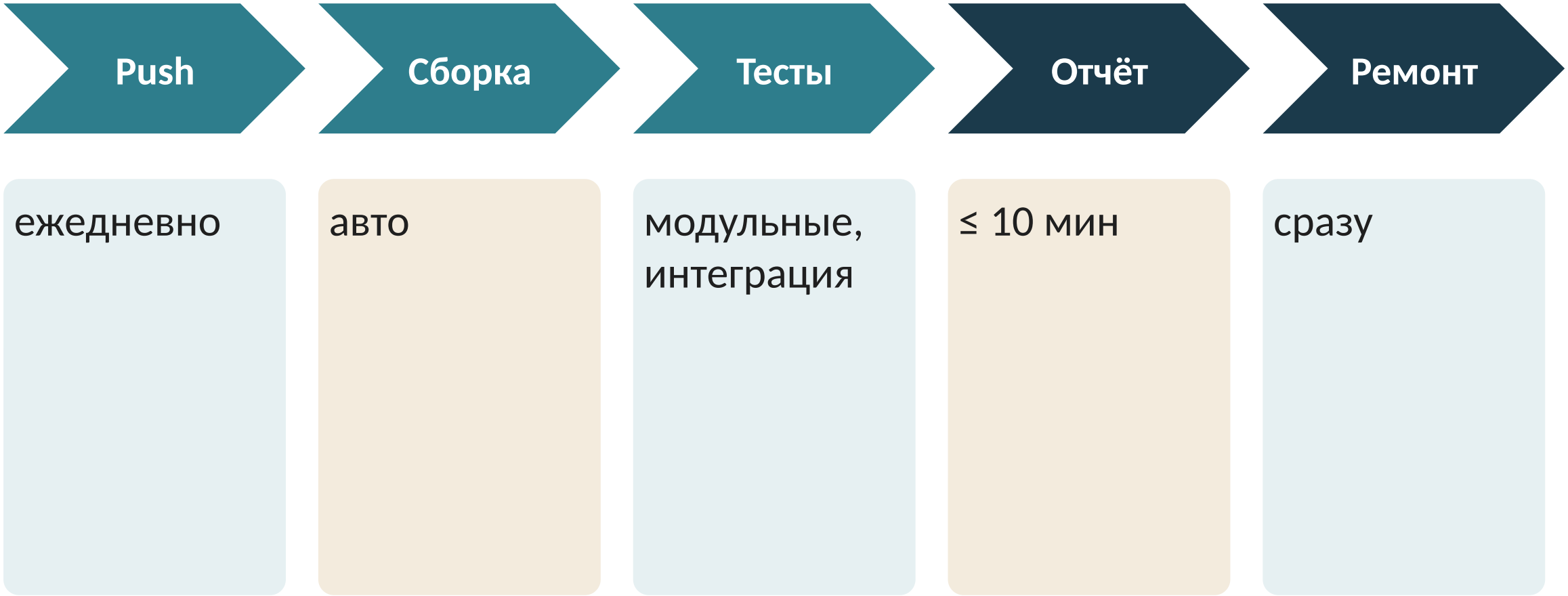


Имитация

проверяет, как её вызвали

Перед защитой заглушки на пути сценария заменить

Непрерывная интеграция



М. Фаулер: сломанная сборка — главный приоритет

План недель 9–15

Неделя	Результат	Проверка
9	ядро и API	модульные тесты
10	сквозной сценарий	e2e в CI
11	тесты и протокол	отчёт CI
12	исправления	регрессия
13	NFR измерены	нагрузка

Результат недели 7



Стратегия

уровни, инструменты, критерии



Тесты Must

классы и границы



Трассировка

требование — компонент — тест



Интеграция

порядок, заглушки, план

Вопросы для самопроверки

1. Верификация и валидация
2. V-модель
3. Пирамида тестов
4. Классы и границы
5. Покрытие
6. Трассируемость
7. Стратегии интеграции
8. Принципы CI



Неделя 7: план реализации и проверки (4 балла)