ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 1
ПОСТАНОВКА ИНЖЕНЕРНОЙ ЗАДАЧИ

Дисциплина: CSE842 «Capstone Project». **Образовательная программа:** 6B06102 «Computer Science», 4 курс, 8 семестр. **Связь с лекцией:** лекция 1 «Цикл исследований и разработок в программном проекте». **Результаты обучения:** РО1. **Объём:** 3 часа практики, 7 часов самостоятельной работы, включая СРСП. **Максимальный балл:** 2.

Цель занятия: сформулировать инженерную задачу проекта, определить пользователей, ограничения и основной сквозной сценарий, подготовить паспорт проекта.

Задачи:

1. Сформировать команду, распределить роли и создать репозиторий проекта.
2. Описать проблему пользователей и сформулировать измеримый проверяемый результат.
3. Определить границы проекта («входит / не входит»), ограничения и допущения.
4. Выбрать основной сквозной сценарий, проходящий через несколько подсистем.
5. Составить и сдать паспорт проекта.

Программное обеспечение и инструменты: Git и GitHub/GitLab, трекер задач (GitHub Projects, Jira или Trello), Markdown, средство диаграмм (draw.io, Mermaid).

1. Краткие теоретические сведения

Инженерная задача описывает проблему конкретных пользователей, способ её решения в заданных ограничениях и признак, по которому результат будет признан достигнутым. Проверяемый результат содержит число, условие и метод измерения (критерии SMART). Сквозной сценарий — последовательность действий пользователя, проходящая через все основные подсистемы; именно он будет продемонстрирован на РК2.

Паспорт проекта (1–2 страницы, файл docs/passport.md) содержит: название и команду с ролями; инженерную задачу; пользователей и заинтересованных лиц; сквозной сценарий; границы; ограничения и допущения; проверяемый результат; предварительный стек; 3–5 ключевых рисков; план контрольных точек.

Таблица 1 — Пример паспорта (сквозной пример курса LabBook)

Раздел	Содержание
Задача	40 приборов лаборатории бронируются через бумажный журнал, 15–20 конфликтов за семестр
Пользователи	студент, лаборант, преподаватель

Раздел	Содержание
Сквозной сценарий	найти прибор → забронировать → проверка конфликта → уведомление лаборанту → подтверждение
Границы	входит: бронь, конфликты, уведомления; не входит: оплата, мобильное приложение
Ограничения	15 недель, 3 человека, бесплатный хостинг
Результат	0 двойных броней на 1000 тестовых заявок, p95 ≤ 300 мс

2. Порядок выполнения работы

1. Изучите вариант (таблица 2) и при необходимости согласуйте с преподавателем уточнение темы или собственную тему.
2. Создайте репозиторий со структурой src/, tests/, docs/, scripts/ и файлом README.md; пригласите участников и руководителя.
3. Проведите 2–3 коротких интервью с представителями пользователей; зафиксируйте проблему в фактах и числах.
4. Сформулируйте инженерную задачу и проверяемый результат; проверьте формулировку по SMART.
5. Определите границы, ограничения и допущения; для каждого допущения укажите, как и когда оно будет проверено.
6. Опишите сквозной сценарий по шагам с указанием подсистем на каждом шаге.
7. Заполните паспорт проекта и создайте в трекере задачи на недели 2–3.

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: темы проектов

№	Тема проекта	Пользователи	Сквозной сценарий (уточнить)
1	Электронная очередь в поликлинику	пациент, регистратор, врач	выбрать врача → выбрать слот → подтвердить запись → SMS пациенту → отметка о приёме
2	Учёт посещаемости по динамическому QR-коду	студент, преподаватель, эдвайзер	преподаватель открывает занятие → динамический QR → студент сканирует → отметка → отчёт эдвайзеру

№	Тема проекта	Пользователи	Сквозной сценарий (уточнить)
3	Мониторинг качества воздуха (IoT)	житель города, эколог	датчик измеряет PM2.5 → публикация в MQTT → запись в БД → график на дашборде → оповещение при превышении
4	Автоматическая проверка программных заданий	студент, преподаватель	студент загружает решение → очередь → запуск тестов в песочнице → оценка → отчёт
5	Интернет-магазин с рекомендательной подсистемой	покупатель, менеджер	поиск товара → рекомендации → корзина → заказ → уведомление
6	Информационная система университетской библиотеки	читатель, библиотекарь	найти книгу → забронировать → выдача по штрихкоду → напоминание → возврат
7	Чат-бот поддержки студентов	студент, сотрудник деканата	вопрос студента → поиск ответа → ответ или передача сотруднику → оценка ответа
8	Мониторинг свободных мест на парковке	водитель, оператор парковки	кадр камеры → статус мест → карта свободных мест → водитель выбирает место
9	Складской учёт со сканером штрихкодов	кладовщик, менеджер	приёмка → сканирование → размещение в ячейку → отгрузка → отчёт об остатках
10	Доставка еды с трекингом курьеров	клиент, курьер, ресторан	заказ → ресторан принял → курьер назначен → трекинг на карте → доставлено
11	Обнаружение аномалий в журналах сервера	системный администратор	журнал → парсинг → признаки окна → детектор → оповещение администратору
12	Мобильный трекер личных расходов с распознаванием чеков	пользователь	фото чека → распознавание суммы → категория → операция в бюджет → отчёт за месяц
13	Электронный документооборот деканата	студент, методист, декан	заявление → согласование методистом → подпись декана → PDF → уведомление студенту
14	Бронирование спортивных площадок	студент, администратор, тренер	выбор площадки → слот → временная блокировка → оплата → QR-пропуск → вход

№	Тема проекта	Пользователи	Сквозной сценарий (уточнить)
15	Умный дом на протоколе MQTT	житель, администратор	датчик движения → правило → включить свет → уведомление в приложении
16	Распознавание автомобильных номеров на КПП	охранник, администратор	кадр → номер → проверка пропуска → открыть шлагбаум → запись в журнал
17	Мониторинг энергопотребления здания	инженер, хозяйственная служба	счётчик → сбор → хранение → прогноз на сутки → оповещение о перерасходе
18	Платформа онлайн-тестирования	студент, преподаватель	преподаватель создаёт тест → студент проходит → автосохранение → автопроверка → результаты
19	Планировщик маршрутов общественного транспорта	пассажир	откуда и куда → маршрут с пересадками → время прибытия → показ на карте
20	Запись на консультации преподавателей	студент, преподаватель	выбрать преподавателя → слот → подтвердить → ссылка на встречу → напоминание

4. Содержание отчёта

1. Паспорт проекта (docs/passport.md).
2. Протоколы интервью (кратко).
3. Ссылка на репозиторий и трекер, распределение ролей.

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 — **2** (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

Критерий	Доля	Что проверяется
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. Чем инженерная задача отличается от темы-технологии?
2. Как проверить, что результат проекта измерим?
3. Чем ограничения отличаются от допущений?
4. Почему сквозной сценарий выбирают на первой неделе?
5. Какие разделы содержит паспорт проекта?

Литература

1. Sommerville I. Software Engineering. — 10th ed. — Pearson, 2016.
2. IEEE Computer Society. SWEBOK Guide v4.0a.
3. NASA. Systems Engineering Handbook. NASA/SP-2016-6105 Rev2.