МЕЖДУНАРОДНАЯ ОЛИМПИАДА
ПО ПРОМЫШЛЕННОЙ РАЗРАБОТКЕ

Т БАНК | ЦЕНТРАЛЬНЫЙ УНИВЕРСИТЕТ | ВЫСШАЯ ШКОЛА ЭКОНОМИКИ

Базовые критерии Backend (10 баллов)

A. Инфраструктура и деплой (2 балла)

- Сервис задеплоен. По URL доступен swagger.
- Обновление бэкенда автоматизировано через CI/CD.

B. API и спецификация (2 балла)

- Полная и актуальная OpenAPI/Swagger спецификация
- Структурированные методы, понятные модели, примеры данных

C. Схема данных и миграции (2 балла)

- Явная схема данных (диаграмма/описание)
- Наличие миграции схемы БД

D. Тестирование (3 балла)

- Unit-тесты на ключевые happy-path сценарии бизнес-логики
- Интеграционные тесты, хотя бы один негативный тест
- Отчёт по прогону тестов в CI

E. Понятность бизнес-логики (1 балл)

- Диаграммы/описание ключевых доменных процессов и сущностей

Продвинутые критерии Backend

P1. Архитектура и качество кода (0–2 балла)

Проверяется

- Читаемость и структурность кода.
- Правильное выделение слоёв: контроллеры / сервисы / репозитории.
- Разделение ответственности между сущностями.
- Обоснованность архитектурного подхода.
- Лёгкость добавления новой бизнес-логики.
- Отсутствие бизнес-логики в контроллерах.

Примеры вопросов

- Какие слои выделены в проекте и зачем?
- Почему контроллер X не должен содержать эту логику?
- Как реализована фича X в разрезе слоёв?
- Какие архитектурные альтернативы рассматривались?
- Как вы обеспечиваете расширяемость?

Мини-правка

Перенести логику из контроллера в сервис, выделить новый модуль/класс, улучшить интерфейс слоя данных.

P2. Работа с БД и эффективность данных (0–2 балла)

Проверяется

- Структура таблиц, нормализация, связи.
- Оптимальность запросов (отсутствие N+1, лишних JOIN).
- Умение использовать индексы.
- Правильная работа с транзакциями.
- Соответствие модели данных бизнес-логике.
- Гибкость схемы: как адаптируется при изменениях.
- Умение объяснить выбор формата хранения (JSONB, отдельные таблицы, связи).
- Корректная работа ORM/клиента БД.

Примеры вопросов

- Почему сущности X и Y связаны таким образом?

- Где в проекте возможна проблема N+1?
- Какие поля нужно проиндексировать и почему?
- Зачем здесь транзакция?
- Как оптимизировать запрос X?
- Что изменится в схеме при добавлении нового статуса или правила? **Мини-**

правка

Добавить индекс, оптимизировать SQL-запрос, вынести запрос в репозиторий, заменить неэффективную загрузку данных.

Р3. Эволюция бизнес-логики (0–2 балла)

Проверяется

- Понимание доменной модели и ключевых процессов.
- Умение вносить изменения: правила, статусы, сценарии.
- Согласованность: схема → код → API → тесты → миграции.
- Умение безопасно обновлять правила расчётов.
- Наличие логирования, достаточного для быстрой диагностики после раскатки.
- Логи должны сливаться в одно место, чтобы команда могла быстро понять, что сломалось.
- Архитектура изменений: отсутствие “магических” условий.

Примеры вопросов

- Как добавить новый статус в процесс?
- Какие изменения требуются в модели и миграциях?
- Что сломается, если поменять правило X?
- Как обновится OpenAPI?
- Где смотреть логи после релиза?
- Какие действия вы предпримете, если после раскатки начнётся ошибка 500?
- Как понять, какой шаг процесса сломался?

Мини-правка

Добавить новый статус/переход, обновить схему, обновить тесты + добавить логирование на критический путь или точку принятия решения.

Блок общее впечатление (0–2 балла)

PROD

МЕЖДУНАРОДНАЯ ОЛИМПИАДА
ПО ПРОМЫШЛЕННОЙ РАЗРАБОТКЕ

Т БАНК | ЦЕНТРАЛЬНЫЙ УНИВЕРСИТЕТ | ВЫСШАЯ ШКОЛА ЭКОНОМИКИ

Базовые критерии Mobile (10 баллов)

A. Деплой (1 балл)

- Есть APK/IPA. Приложение запускается

B. Функционал (2 балла)

Весь функционал завершен и везде, где необходимо, есть связь с бэкендом. Отсутствует недоделанный функционал

Есть Readme с описанием функционала — указаны все экраны со статусом разработки и связью с бэкендом. Для каждого экрана указано его продуктовое назначение (к какому сценарию клиента относится экран и какую потребность закрывает)

C. UX (3 балла)

- Навигация работает корректно и соответствует гайдлайнам платформы
- Поведение у схожих UI компонентов предсказуемо и консистентно
- Дизайн и интерактивные элементы (кнопки, слайдеры, меню и т.д.) не выбиваются из подходов на платформе

D. Производительность UI (4 балл)

- Отсутствуют: блокировка UI, подтормаживания отрисовки и реакции на пользовательский ввод
- Отсутствуют краши
- Ошибки обрабатываются
- Все поля имеют валидацию вводимых данных

Итого: 10 баллов базы.

Продвинутые критерии Mobile

Каждый блок описан по структуре:

Проверяется → Примеры вопросов → Мини-правка

P1. Качество кода (0–2 баллов)

Проверяется

- Декомпозиция и читаемость кода
- Выделение компонентов, интерфейсы
- Архитектура в целом
- Насколько легко добавлять новый функционал

Примеры вопросов

- Из каких логических компонентов состоит экран X?
- За счёт каких сущностей реализована фича X?
- Как компонент X взаимодействует с компонентом Y?
- Почему именно так организован проект? В чем преимущества и недостатки использованной архитектуры

Мини-правка

Добавить новую логику

P2. Дизайн-система (0–2 баллов)

Проверяется

- Умение выделять UI компоненты и создавать единую компонентную базу
- Умение формирования единого дизайна приложения. Работа с константами (дизайн-токенами)
- Как переиспользуется UI. Какие конфигурации и кастомизации заложены
- Формирование отдельного DesignKit модуля

Примеры вопросов

- Из каких дизайн компонентов состоит экран X?
- Какие дизайн компоненты переиспользуются в приложении?
- Как выглядят константы дизайна? Почему они именно такие и хранятся выбранном способом?
- Как конфигурируется компонент X?
- Как можно кастомизировать и переиспользовать компонент X?

- Почему UI-элементы X и Y сделаны отдельными компонентами?
- Есть ли отдельный модуль для дизайн-системы?

Мини-правка

- Добавить новую кастомизацию для какого-то общего компонента. Например, возможность указать для общей кнопки её цвет в конкретном месте использования.
- Изменить дизайн константу

P3. Тестирование и CI/CD (0–2 баллов)

Проверяется

- Умение выбора инструментов и уровня тестирования
- Тестирование критических сценариев и корнер-кейсов. Понимание, что такое критический путь
- Умение добавления запуска тестов на CI (обязательно для 5 баллов)

Примеры вопросов

- Почему были выбраны эти инструменты для тестов?
- Какие были альтернативы, чем они хуже?
- Как протестирован сценарий X?
- Почему для тестирования фичи X был выбран уровень тестирования Y?
- Как работает проверка в тесте X? Почему она перестанет работать, если внести ломающую правку в проект?

Мини-правка

Добавить новый тест-кейс и выполнить его.

Блок общее впечатление (0–2 баллов)

PROD

МЕЖДУНАРОДНАЯ ОЛИМПИАДА
ПО ПРОМЫШЛЕННОЙ РАЗРАБОТКЕ

Т БАНК | ЦЕНТРАЛЬНЫЙ УНИВЕРСИТЕТ | ВЫСШАЯ ШКОЛА ЭКОНОМИКИ

Базовые критерии ML (10 баллов)

A. EDA и Качество данных (2 балла)

- Проведен разведочный анализ данных с визуализацией и выполнена базовая статистика по признакам
- Обработаны пропуски, выбросы и аномалии

B. Воспроизводимость (2 балла)

- Код версионируется в Git
- Зафиксированы зависимости requirements.txt (или другим способом)

C. Качество модели (3 балла)

- Выбраны корректные метрики для типа задачи
- Реализовано разделение на train/validation/test
- Модель показывает адекватное качество на тестовых данных

D. Serving (2 балла)

- Модель обернута в API (FastAPI/Flask и т.п.)
- Доступен эндпоинт для получения предсказаний и API корректно обрабатывает запросы

E. Деплой (1 балл)

- Модель задеплоена и доступна

Итого: 10 баллов базы.

Продвинутые критерии ML

Каждый блок описан по структуре:

Проверяется → Примеры вопросов → Мини-правка

P1. EDA, Feature Engineering и Качество модели (0–2 баллов)

Проверяется:

- Глубина разведочного анализа и формулирование гипотез
- Создание информативных признаков на основе выводов EDA
- Обоснование выбора метрик через бизнес-задачу
- Сравнение с baseline и интерпретация результатов

Примеры вопросов:

- Какие закономерности обнаружены в данных? Какие гипотезы сформулированы и как проверялись?
- Как выводы из EDA повлияли на создание новых признаков?
- Есть ли выбросы/аномалии/проблемы качества данных? Как они обрабатываются и почему?
- Почему выбраны именно эти метрики? Как они связаны с бизнес-целью?
- С чем сравнивается модель? Что означает полученное значение для продукта?
- Есть ли trade-off между метриками? Какая ошибка более критична (FP vs FN)?

Мини-правка:

- На основе показанного графика из EDA создать новый признак и добавить в модель
- ИЛИ изменить threshold классификации и объяснить влияние на бизнес-метрики

P2. ML Pipeline: Воспроизводимость и Эксперименты (0–2 баллов)

Проверяется:

- Структурированность кода и автоматизация пайплайна
- Версионирование кода, данных и моделей (DVC или альтернатива)
- Систематическое отслеживание экспериментов
- Возможность воспроизвести результаты

Примеры вопросов:

- Как устроен ML-пайплайн? Из каких этапов состоит?
- Как версионируются данные и модели? Где хранятся артефакты?

- Можно ли воспроизвести процесс обучения одной командой, например через DVC?
- Какие эксперименты проводились? Как логируются результаты?
- Какой инструмент для tracking используется? (MLflow, W&B, таблицы)
- Можно ли посмотреть историю экспериментов и сравнить их?
- Как принималось решение, какая модель лучше?

Мини-правка:

- Запустить пайплайн обучения с нуля и воспроизвести метрики
- ИЛИ запустить новый эксперимент с измененными параметрами и показать его в системе tracking

P3. Production Readiness: Serving и Observability (0–2 баллов) Проверяется:

- Логирование запросов, ответов и ошибок API
- Метрики производительности (latency, throughput)
- Оптимизация модели для инференса
- Возможность отладки и мониторинга

Примеры вопросов:

- Какие события логируются при работе API? Где хранятся логи?
- Логируются ли входные данные, предсказания и ошибки?
- Какое время отклика у API? Как оно измерялось?
- Как можно отследить неожиданный результат модели и воспроизвести его?
- Была ли оптимизирована модель для инференса? (ONNX, quantization)
- Как API справится с большим количеством запросов?
- Есть ли батчинг запросов или другие оптимизации?

Мини-правка:

- Отправить несколько запросов к API и показать их в логах с временем обработки
- ИЛИ провести простое нагрузочное тестирование или показать оптимизацию модели (например, ONNX)

Блок общее впечатление (0-2 баллов)

PROD

МЕЖДУНАРОДНАЯ ОЛИМПИАДА
ПО ПРОМЫШЛЕННОЙ РАЗРАБОТКЕ

Т БАНК | ЦЕНТРАЛЬНЫЙ УНИВЕРСИТЕТ | ВЫСШАЯ ШКОЛА ЭКОНОМИКИ

Базовые критерии Frontend (10 баллов)

A. Деплой (2 балла)

- По URL доступен фронтенд с осмысленным функционалом.
- Обновление фронтенда автоматизировано через CI/CD.

B. Работа с бекендом (2 балла)

- Фронтенд отправляет запросы к бекенду.
- Данные от бекенда синхронизированы с интерфейсом.

C. UX (3 балла)

- Интерактивные элементы (кнопки, ссылки, выпадающие меню и т.д.) выглядят и ведут себя предсказуемо.
- Все интерактивные элементы работают. Нет элементов, которые идейно должны быть интерактивными, но по факту ничего не делают (например, нерабочие кнопки).
- Понятен формат ввода данных, на клиенте есть необходимая и достаточная валидация. Интерфейс ограничивает ввод невалидных данных (например, в поле ввода «возраст» нельзя ввести отрицательное число).

D. Консистентность UI (1 балл)

- Соблюдается единый гармоничный стиль оформления — цвета, шрифты, отступы и т.д.

E. Адаптивность (2 балл)

- Интерфейс корректно отображается на мобильном телефоне.
- Интерфейс корректно отображается на десктопе (в том числе очень широком).

Итого: 10 баллов базы.

2.1. Продвинутые критерии Frontend

Каждый блок описан по структуре:

Проверяется → Примеры вопросов → Мини-правка

P1. Стек и архитектура (0–2 балла)

Проверяется

- Обоснование выбора фреймворков и библиотек.
- Понимание процесса сборки и доставки интерфейса.

Примеры вопросов

- Почему была выбрана технология X? (фреймворк, библиотека и т.д.) Какие рассматривались альтернативы?
- Какие действия по сборке, подготовке статики выполняются перед деплоем?
- Где и как задается конфигурация сборки и деплоя?

Мини-правка

Внести правку в конфигурацию или скрипт сборки.

P2. Структура и декомпозиция (0–2 балла)

Проверяется

- Умение описать взаимодействие между компонентами.
- Соблюдение зон ответственности между сущностями.
- Обоснованность организации файлов и папок.

Примеры вопросов

- Как компонент X взаимодействует с компонентом Y?
- За счёт каких сущностей реализована фича X?
- Из каких компонентов состоит страница X?
- Почему компонент X имеет такой контракт или использует такую часть стейта?
- Почему у проекта такая структура директорий? Какие рассматривались альтернативы?
- Я хочу найти код элемента X. Как навигировать до него по папкам проекта?

Мини-правка

Добавить новую настройку для какого-то компонента, чтобы расширить его кастомизируемость.

Р3. Эволюция бизнес-логики (0–2 балла)

Проверяется

- Понимание доменной модели.
- Способность адаптировать интерфейс к изменениям.
- Гибкость и расширяемость интерфейса.

Примеры вопросов

- Как добавить отображение новых данных на форме или в таблице?
- Как добавить новый раздел приложения?
- Как интерфейс отреагирует на добавление нового элемента в список?

Мини-правка

Добавить новый статус, раздел или расширить форму.

Блок общее впечатление (0–2 балла)

PROD

МЕЖДУНАРОДНАЯ ОЛИМПИАДА
ПО ПРОМЫШЛЕННОЙ РАЗРАБОТКЕ

Т БАНК | ЦЕНТРАЛЬНЫЙ УНИВЕРСИТЕТ | ВЫСШАЯ ШКОЛА ЭКОНОМИКИ

Базовые критерии Product (10 баллов)

A. Проблема и пользователь (2 балла)

A1. Проблема, которую продукт решает для пользователя сформулирована ясно и конкретно. (Да/Нет)

A2. Участники конкретно понимают кто пользователь продукта. Понимают как продукт решает проблему пользователя. (Да/Нет)

B. Сценарии и соответствие реализации (2 балла)

B1. Участники отрисовали и презентовали пользовательский путь. Проработали основные User-Story. (Да/Нет)

B2. Реализованный продукт полностью поддерживает основные сценарии. (Да/Нет)

C. Ценность и приоритизация (3 балла)

C1. Посчитана и обоснована бизнес-ценность реализованного продукта (Да/Нет)

C2. Реализованная функциональность позволяет пройти основной пользовательский путь. (Да/Нет)

D. Понятность UX/UI (2 балла)

D1. Реализованный продукт очевиден и понятен пользователю. Не требуется дополнительных пояснений, чтобы пройти основной пользовательский путь. (Да/Нет)

D2. Визуальный стиль и продукт соответствует времени и выполняемым задачам (Да/Нет)

E. Метрики и проверка гипотез (2 балла)

E1. Озвучена и посчитана NordStar (главная) метрика продукта. (Да/Нет)

E2. Команда назвала и объяснила хотя бы одну метрику успеха реализованного продукта и способ проверки (MVP/тест/интервью). (Да/Нет)

Блок общее впечатление (0-2 баллов)