

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 1 «Платформа для подготовки к олимпиадам»

1. Условия

Учащиеся готовятся к предметным олимпиадам и конкурсам в условиях постоянной информационной перегрузки и ограниченного времени. Традиционные сборники задач не позволяют отслеживать собственный прогресс, а ручной подбор задач занимает много времени. Предлагается разработать платформу, которая сможет решить указанные проблемы за счёт предоставления единого цифрового пространства, в котором пользователи могут последовательно выполнять ключевые учебные и соревновательные активности.

2. Техническое задание

Платформа должна включать следующие основные модули: каталог задач с автоматической проверкой решений, соревновательный режим «1-на-1» (PvP) с синхронной обработкой ответов, административную панель для управления содержимым и аналитический модуль для оценки результатов.

Функциональное задание:

Необходимо разработать клиент-серверную платформу. Клиентская часть должна представлять собой пользовательский интерфейс, а серверная — предоставлять программный интерфейс для обработки запросов, управления базой задач и обеспечения соревнований. Сервер должен поддерживать API для обмена данными с различными клиентами. База данных осуществляет хранение сведений о задачах, пользователях, решениях, результатах соревнований и статистике пользователей. Необходимо предусмотреть использование WebSocket для организации PvP-соревнований и мгновенных уведомлений. Платформа должна быть

МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ ОЛИМПИАДА ШКОЛЬНИКОВ

Профиль «Информационные технологии»

Командный кейс № 1 «Платформа для подготовки к олимпиадам»

масштабируемой и позволять подключать сторонние сервисы (например, открытые банки задач или модели ИИ) для расширения функциональных возможностей. Серверная часть должна обеспечивать стабильную работу при одновременных тренировках, корректно распределять вычислительные ресурсы и обеспечивать безопасность данных.

Обязательная функциональность:

- Регистрация и авторизация. Серверная часть должна хранить данные пользователей в СУБД и использовать шифрование паролей.
- Каталог задач и режим тренировки. Платформа предоставляет банк задач с возможностью фильтрации по предмету, сложности и теме. Пользователь может решать задачи онлайн, получать автоматическую проверку ответов и подсказки. При генерации новых заданий может использоваться открытый банк заданий или модель ИИ для создания вариаций.
- Соревновательный режим (PvP). Платформа должна поддерживать реальные поединки между пользователями, отображать задания, результаты в реальном времени. Должно осуществляться распределение игроков по уровню, а результаты хранятся в базе данных. Требования к соревнованиям:

МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ ОЛИМПИАДА ШКОЛЬНИКОВ

Профиль «Информационные технологии»

Командный кейс № 1 «Платформа для подготовки к олимпиадам»

- Оба игрока отправляют ответы на одну и ту же задачу одновременно. Игроки должны получать корректное отображение текущего счета и статусов задач.
- В случае повторной отправки ответа, если правила матча допускают повторную отправку, платформа корректно заменить предыдущий результат новым, не удваивая очки.
- Платформа должна не позволять игроку влиять на чужой прогресс или изменять состояние матча.
- Проверка корректности подсчета рейтинга для трёх основных исходов матча:
 - победа одного из игроков - рейтинг победителя увеличивается, проигравшего уменьшается;
 - ничья - рейтинги изменяются согласно формуле Ело при фактическом результате 0.5;
 - техническая ошибка, обрыв соединения или отмена матча - рейтинги **не должны изменяться**, а состояние матча фиксируется как «отменён» или «техническая ошибка».

Определение уровня игрока согласно формуле Ело:

1. Начальное значение рейтинга для всех новых игроков
 $R_0 = 1000$;
 2. Коэффициент изменения рейтинга: $K = 32$.
- Отчётность и аналитика. Сбор и анализ данных о ходе обучения должен давать информацию о скорости решения, коэффициенте правильных ответов и прогрессе. Данные визуализируются в личном кабинете в виде графиков и таблиц.

МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ ОЛИМПИАДА ШКОЛЬНИКОВ

Профиль «Информационные технологии»

Командный кейс № 1 «Платформа для подготовки к олимпиадам»

- Модуль администрирования. Администратор может наполнять базу задач, просматривать статистику пользователей, осуществлять импорт/экспорт задач из CSV/JSON.

Дополнительная функциональность:

- Геймификация и награды. Использование очков, уровней и достижений.
- Персональные рекомендации и адаптивное обучение. Анализируя данные о пользователе, предложение индивидуального плана тренировок.
- Дополнительные виды соревнований. Помимо классического PvP, допускается реализация рейтинговых таблиц, групповых турниров или марафонов с ограничением времени.
- Адаптивный дизайн. Интерфейс должен корректно работать на компьютерах, планшетах и смартфонах;

3. Рекомендации к выполнению

- Клиентская часть платформы должна или работать на мобильных устройствах, или иметь веб-интерфейс.
- Разработку рекомендуется вести с помощью системы контроля версий git.
- Рекомендуется использовать unit-тестирование при разработке продукта.
- Рекомендуется предусмотреть хостинг или автоматизированную установку серверной части на целевой машине.
- Рекомендуется реализовать единый стиль оформления интерфейса.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 1 «Платформа для подготовки к олимпиадам»

4. Требования к документации

- Титульный лист (с указанием названия кейса и перечислением членов команды).
- Обоснование выбора языка программирования и используемых программных средств.
- Структурная и функциональная схемы программного продукта.
- Блок-схема работы основного алгоритма.
- Описание особенностей и аргументация выбранного типа СУБД.
- Схема базы данных.
- Программный код (ссылка на репозиторий), файл README должен включать:
 - краткое описание проекта;
 - инструкцию по установке/развертыванию;
 - ссылку на видеоролик.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 1 «Платформа для подготовки к олимпиадам»

5. Требования к видеоролику

- Видеоролик должен демонстрировать функционирование разработанного программного продукта в соответствии с регламентом испытаний.
- На видео или записи экрана необходимо продемонстрировать выполнение каждого испытания, описанного в регламенте, в соответствии с условиями.
- Видео должно однозначно подтверждать авторство участников (во время записи ролика необходимо четко произнести название команды, ФИО участников, номер школы, ФИО руководителя).
- Видеоролик необходимо разместить на стороннем видеохостинге («ВКонтакте», Rutube и др.)

6. Регламент испытаний

Испытания должны включать проверку всех функциональных требований, указанных в техническом задании, в том числе:

1. Регистрация и вход. Проверяется, что новый пользователь может создать аккаунт и войти, и корректно обрабатываются ошибки (неверный пароль, занятый email и т.п.).
2. Генерация и решение задач. Проверяется, что пользователи могут получать задания из банка или генерировать новые с помощью моделей ИИ. Должен корректно отображаться текст задачи, приниматься ответ, осуществляться проверка и отображаться результат.
3. Отображение меню. Интерфейс должен быть интуитивно понятным: доступ к тренировкам, соревнованиям, аналитике и профилю.
4. PvP-режим. Два пользователя могут подключиться к матчу, увидеть одни и те же задания, отправлять ответы и получать результаты в реальном времени, согласно требованиям.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**

Профиль «Информационные технологии»

Командный кейс № 1 «Платформа для подготовки к олимпиадам»

5. Аналитика. Проверяется корректность построения отчетов и диаграмм, наличие статистики по темам, скорости решения и точности. Данные должны обновляться при решении задач.
6. Административная панель. Проверяется наличие и корректная работа панели администратора, предназначенной только для пользователей с правами администратора. Система должна обеспечивать защиту от несанкционированного доступа. Панель позволяет управлять:
 - a. пользователями (просмотр, блокировка, изменение ролей),
 - b. задачами (добавление, редактирование, удаление),
 - c. соревнованиями и результатами,
 - d. интеграцией с внешними API (если предусмотрено).
7. Дополнительные функции (если реализованы): рекомендуется продемонстрировать геймификацию, адаптивность интерфейса, персональные рекомендации, другие типы соревнований.

7. Методические материалы

Для успешного выполнения задачи рекомендуется ознакомиться с материалами:

- <https://www.python.org/>
- <https://sqlite.org/>
- <https://www.postgresql.org/>
- <https://flask.palletsprojects.com/en/2.0.x/>
- <https://developer.android.com/>
- <https://kotlinlang.org/docs/home.html>
- <https://docs.oracle.com/en/>
- <https://vuejs.org/>

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 2 «Управление столовой»

1. Условия

Современные школы нуждаются в удобных и прозрачных инструментах для организации питания учащихся. Использование бумажных журналов, талонов и устаревших систем учета не позволяет эффективно контролировать процесс, оперативно анализировать данные и обеспечивать безопасность.

Для решения этой задачи предлагается разработать **веб-приложение** – автоматизированную информационную систему школьного питания.

Система должна:

- позволять учащимся авторизоваться, просматривать меню завтраков и обедов, оплачивать питание (разово или абонементом), указывать аллергены и другие пищевые особенности, оставлять отзывы;
- предоставлять сотрудникам столовой интерфейс для учета выданных блюд, контроля остатков продуктов и оформления заявок на закупки;
- обеспечивать администраторам доступ к статистике оплат и посещаемости, согласованию заявок на закупку и формированию отчетности.

Реализация веб-приложения позволит повысить эффективность работы школьной столовой и улучшить качество обслуживания.

2. Техническое задание

Требуется разработать веб-приложение для учёта и контроля питания в школьной столовой. Приложение должно предоставлять различные уровни доступа для пользователей и администраторов, поддерживать оплату питания и обратную связь.

Необходимо предусмотреть роли с уровнями доступа:

- ученик;
- повар;

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 2 «Управление столовой»

- администратор.

Обязательная функциональность для ученика:

- регистрация и авторизация;
- просмотр меню завтраков и обедов;
- оплата питания (разовый платеж или абонемент);
- отметка о получении питания;
- указание пищевых аллергий и предпочтений;
- оставление отзывов о блюдах.

Обязательная функциональность для повара:

- авторизация в системе;
- учет выданных завтраков и обедов;
- контроль остатков продуктов и готовых блюд;
- внесение заявок на закупку продуктов.

Обязательная функциональность для администратора:

- авторизация в системе;
- просмотр статистики оплат и посещаемости;
- согласование заявок на закупки;
- формирование отчетов по питанию и затратам.

Дополнительная функциональность:

- модуль уведомлений для учеников и сотрудников.

3. Рекомендации к выполнению

- Использовать систему управления базами данных (СУБД) для хранения данных. Выбор СУБД не регламентируется.
- Рекомендуется применять клиент-серверную архитектуру.
- Использовать систему контроля версий для ведения проекта.
- Предусмотреть автоматизированную установку приложения на целевой машине.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 2 «Управление столовой»

- Уделить внимание вопросам безопасности (персональные данные и платежи).

4. Требования к документации

- Титульный лист (с указанием названия кейса и перечислением членов команды).
- Обоснование выбора языка программирования и используемых программных средств.
- Структурная и функциональная схемы программного продукта.
- Блок-схема работы основного алгоритма.
- Описание особенностей и аргументация выбранного типа СУБД.
- Схема базы данных.
- Программный код (ссылка на репозиторий), файл README должен включать:
 - краткое описание проекта;
 - инструкцию по установке/развертыванию;
 - ссылку на видеоролик.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 2 «Управление столовой»

5. Требования к видеоролику

- Видеоролик должен демонстрировать функционирование разработанного программного продукта в соответствии с регламентом испытаний.
- На видео или записи экрана необходимо продемонстрировать выполнение каждого испытания, описанного в регламенте, в соответствии с условиями.
- Видео должно однозначно подтверждать авторство участников (во время записи ролика необходимо четко произнести название команды, ФИО участников, номер школы, ФИО руководителя).
- Видеоролик необходимо разместить на стороннем видеохостинге («ВКонтакте», Rutube и др.)

6. Регламент испытаний

Испытания должны включать проверку всех функциональных требований, в том числе:

- регистрация и авторизация пользователей;
- оформление оплаты учеником (разовый платеж и абонемент);
- отметка учеником о получении питания;
- учет выданных блюд поваром;
- добавление заявки на закупку продуктов поваром;
- согласование заявки администратором;
- формирование отчета администратором о питании и затратах;
- тестирование обработки исключительных ситуаций (например, повторная отметка питания учеником или отсутствие достаточного количества продуктов).

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 2 «Управление столовой»

7. Методические материалы

Для успешного выполнения задачи рекомендуется ознакомиться с материалами:

Веб-разработка

- Основы HTML, CSS, JavaScript: <https://developer.mozilla.org/ru/>
- React: <https://react.dev/>
- Vue.js: <https://vuejs.org/>

Бэкенд-фреймворки

- Node.js + Express: <https://expressjs.com/>
- Django: <https://docs.djangoproject.com/>
- Spring (Java): <https://spring.io/projects/spring-boot>

Базы данных

- MySQL: <https://dev.mysql.com/doc/>
- PostgreSQL: <https://www.postgresql.org/docs/>
- SQLite: <https://www.sqlite.org/docs.html>

Системы контроля версий

- Git: <https://git-scm.com/doc>
- GitHub Guides: <https://docs.github.com/>

Безопасность и работа с данными

- Основы работы с JWT (JSON Web Tokens): <https://jwt.io/introduction>
- OWASP Top 10 (основные угрозы безопасности веб-приложений): <https://owasp.org/Top10/ru/>

Управление проектами

- Git + GitHub flow: <https://guides.github.com/introduction/flow/>
- Agile и Scrum основы: <https://www.scrumguides.org/>

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 3 «Анализ поступления»

1. Условия

Работа современных приёмных комиссий в вузах сопровождается большими объемами данных: десятки тысяч абитуриентов подают заявления на разные образовательные программы с различными приоритетами и баллами. Отслеживание динамики проходных баллов и вероятности поступления вручную становится весьма затрудненным. Для оптимизации работы приёмных комиссий и информирования абитуриентов предлагается разработать программное обеспечение, позволяющее автоматически рассчитывать вероятность поступления на основе конкурсных списков и приоритетов заявлений. Система обеспечивает анализ данных в динамике, формирование отчётности и визуализацию результатов.

2. Техническое задание

Главной задачей является разработка программного обеспечения (ПО), способного в автоматизированном режиме анализировать конкурсные ситуации при поступлении абитуриентов в вуз по их заявленным приоритетам и текущим конкурсным данным.

Предлагается разработать ПО в виде веб-приложения или десктоп приложения, отвечающее следующим требованиям:

1. Должна быть организована база данных (БД) для хранения информации об абитуриентах, баллах, приоритетах и конкурсных списках. Особых требований к технологиям и типу БД не предъявляется.
2. Должна быть реализована функциональность, позволяющая обеспечить загрузку конкурсных списков абитуриентов в БД из внешних источников в табличном формате.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 3 «Анализ поступления»

3. Необходимо предусмотреть возможность обновления БД абитуриентов при загрузке последней версии конкурсных списков. В случае, если БД на момент обновления пуста, информация из конкурсных списков полностью переносится в БД. Время загрузки не должно превышать 5 секунд.
4. Обновление конкурсных списков заключается в следующих действиях:
 - a. Удаление информации об абитуриенте (в случае, если в последней версии конкурсных списков информация об абитуриенте отсутствует, но присутствует в БД).
 - b. Добавление информации об абитуриенте (в случае, если в последней версии конкурсных списков информация об абитуриенте присутствует, но отсутствует в БД).
 - c. Обновление информации об абитуриенте (в случае, если информация об абитуриенте есть и в последней версии конкурсных списков и в БД). Приоритет в актуальности информации отдается последнему загруженному конкурсному списку.
5. Для разработки и тестирования ПО необходимо использовать 4 списка для каждой из 4 образовательных программ (ОП), соответствующие разным дням приемной кампании (1 августа, 2 августа, 3 августа, 4 августа).
6. Перечень образовательных программ и количество бюджетных мест на ОП, которые необходимо заполнить:
 - a. Прикладная математика (ПМ): 40 мест.
 - b. Информатика и вычислительная техника (ИВТ): 50 мест.
 - c. Инфокоммуникационные технологии и системы связи (ИТСС): 30 мест.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ
Профиль «Информационные технологии»
Командный кейс № 3 «Анализ поступления»**

- d. Информационная безопасность (ИБ): 20 мест.
7. Конкурсные списки должны быть сформированы и сгенерированы с учетом следующей структуры конкурсных списков:
- a. Уникальный идентификатор абитуриента (ID): int
 - b. Наличие согласия о зачислении в ВУЗе: boolean
 - c. Приоритет ОП (от 1 до 4): int
 - d. Балл Физика/ИКТ: int
 - e. Балл Русский язык: int
 - f. Балл Математика: int
 - g. Балл за индивидуальные достижения: int
 - h. Сумма баллов: int
8. Количество абитуриентов в конкурсных списках относительно ОП и дней приемной кампании:

Дата	Количество всех абитуриентов			
	ПМ	ИВТ	ИТСС	ИБ
01.08	60	100	50	70
02.08	380	370	350	260
03.08	1000	1150	1050	800
04.08	1240	1390	1240	1190

9. Списки должны быть сгенерированы таким образом, чтобы соблюдалось количество пересечений множеств абитуриентов всех ОП в каждый день приемной кампании (один абитуриент может участвовать в конкурсе от 1 до 4 программ):

Дата	Количество пересечений для множества абитуриентов только двух ОП					
	ПМ-ИВТ	ПМ-ИТСС	ПМ-ИБ	ИВТ-ИТСС	ИВТ-ИБ	ИТСС-ИБ
01.08	22	17	20	19	22	17
02.08	190	190	150	190	140	120

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ
Профиль «Информационные технологии»
Командный кейс № 3 «Анализ поступления»**

03.08	760	600	410	750	460	500
04.08	1090	1110	1070	1050	1040	1090

Количество пересечений для множества абитуриентов только трех или четырех ОП					
Дата	ПМ-ИВТ- ИТСС	ПМ-ИВТ -ИБ	ИВТ-ИТС С-ИБ	ПМ-ИТСС- ИБ	ПМ-ИВТ-И ТСС-ИБ
01.08	5	5	5	5	3
02.08	70	70	70	70	50
03.08	500	260	300	250	200
04.08	1020	1020	1000	1040	1000

10. При обновлении конкурсных списков (при смене дня) должно удаляться от 5 до 10% процентов записей, появляться не менее 10% процентов новых записей об абитуриентах, остальные записи должны обновляться.
11. Количество абитуриентов, которые предоставили согласия на зачисления в финальном списке 04.08 для каждой ОП, должно быть больше, чем количество мест на ОП. Для других дней это условие может не соблюдаться с целью прохождения испытания №2 регламента испытаний.
12. Программное обеспечение должно предоставлять возможность визуализации конкурсных списков, находящихся в БД в графическом интерфейсе (как по каждой программе в отдельности, так и по всей БД в виде единого списка с указанием каскада приоритетов для каждого абитуриента). Должны быть реализованы возможности сортировки, фильтрации списков по признакам. Операция изменения списков в графическом интерфейсе должна быть исключена. Время перестроения визуализаций не должно превышать 3 секунды.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 3 «Анализ поступления»

13. После обновления всех списков разрабатываемое программное обеспечение должно рассчитывать проходной балл на каждую ОП в соответствии с конкурсной ситуацией. При расчете проходного балла учитываются приоритеты ОП, выбранных абитуриентами, но учитываются только абитуриенты, предоставившие согласие о зачислении в ВУЗ. В графическом интерфейсе должно быть отображено количество мест на ОП и проходной балл на определенную дату и время.
14. Разрабатываемое программное обеспечение должно иметь возможность формирования отчета в формате .pdf, в котором будет указываться следующая информация:
- a. Дата и время формирования отчета.
 - b. Проходные баллы на ОП – сумма баллов последнего абитуриента, предоставившего согласие на зачисление, при котором заполнены все места на ОП (если количество таких абитуриентов меньше, чем количество мест, вместо суммы баллов необходимо вывести – НЕДОБОР).
 - c. Динамика проходного балла на ОП по всем дням в виде графиков (за 4 дня).
 - d. Списки абитуриентов, которые будут зачислены на каждую образовательную программу с указанием их ID и суммы баллов (4 списка).
 - e. Статистику по каждой ОП в виде таблицы, в которой указана следующая информация:

	ПМ	ИВТ	ИТСС	ИБ
Общее кол-во заявлений				
Количество мест на ОП	40	50	30	20
Кол-во заявлений 1-го приоритета				

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 3 «Анализ поступления»

Кол-во заявлений 2-го приоритета				
Кол-во заявлений 3-го приоритета				
Кол-во заявлений 4-го приоритета				
Кол-во зачисленных 1-го приоритета				
Кол-во зачисленных 2-го приоритета				
Кол-во зачисленных 3-го приоритета				
Кол-во зачисленных 4-го приоритета				

3. Рекомендации к выполнению

- Разработку рекомендуется вести с помощью системы контроля версий git
- Рекомендуется на первом этапе работы генерировать списки при помощи разработки программного кода в соответствии с требованиями, и затем приступать к разработке ПО для обновления списков и расчета проходного балла.

4. Требования к документации

- Титульный лист (с указанием названия кейса и перечислением членов команды).
- Обоснование выбора языка программирования и используемых программных средств.
- Структурная и функциональная схемы программного продукта.
- Блок-схема работы основного алгоритма.
- Описание особенностей и аргументация выбранного типа СУБД.
- Схема базы данных.
- Скриншоты графического интерфейса и примеры отчетов.
- Списки абитуриентов.
- Программный код (ссылка на репозиторий), файл README должен включать:
 - краткое описание проекта;
 - инструкцию по установке/развертыванию;
 - ссылку на видеоролик.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 3 «Анализ поступления»

5. Требования к видеоролику

- Видеоролик должен демонстрировать функционирование разработанного программного продукта в соответствии с регламентом испытаний.
- На видео или записи экрана необходимо продемонстрировать выполнение каждого испытания, описанного в регламенте, в соответствии с условиями.
- Видео должно однозначно подтверждать авторство участников (во время записи ролика необходимо четко произнести название команды, ФИО участников, номер школы, ФИО руководителя).
- Видеоролик необходимо разместить на стороннем видеохостинге («ВКонтакте», Rutube и др.)

6. Регламент испытаний

Работоспособность проверяется в процессе 3-х испытаний:

1. В случае если испытание не может быть полностью пройдено разработанным ПО, участникам необходимо продемонстрировать частичное выполнение данного испытания. На видео они в явном виде поясняют, какая функциональность ПО в рамках испытаний реализована, а что отсутствует.
2. Испытание №1. Производится проверка корректности сформированных конкурсных списков абитуриентов и возможность обновления списков в БД:
 - a. Участники демонстрируют наличие списков для каждого дня и каждой ОП (всего 16 списков).
 - b. Участники демонстрируют соответствие общего количества абитуриентов в списках с требованиями ТЗ.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 3 «Анализ поступления»

- c. Участники демонстрируют реализацию в списках множеств абитуриентов с пересечением по разным ОП и в явном виде показывают соответствие их количества требованиям ТЗ.
 - d. Участники демонстрируют различия конкурсных списков разных дней для каждой ОП в соответствии с требованием пункта 2.10.
 - e. Участники демонстрируют структуру БД (количество таблиц и связи).
 - f. Участники демонстрируют функционал возможности загрузки конкурсных списков в пустую БД.
 - g. Участники демонстрируют функционал обновления списков в БД (удаление, добавление, обновление данных) для разных дней.
3. Испытание №2. Участники демонстрируют работу функционала по расчету проходного балла на ОП:
- a. Участники демонстрируют процесс очистки БД.
 - b. Затем участники загружают списки от 01.08 и рассчитывают проходной балл по всем ОП. Конкурсные списки должны быть сформированы таким образом, чтобы разработанное ПО выдавало для ОП результат – НЕДОБОР. При этом участники демонстрируют конкурсные списки абитуриентов по каждой ОП, производят сортировку и фильтрацию по наличию согласия о зачислении, и сравнивают количество абитуриентов, предоставивших согласие о зачислении, с количеством мест на программе (количество мест на программе должно быть больше, чем количество абитуриентов с согласиями).

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 3 «Анализ поступления»

- c. Затем участники загружают список от 02.08. По каждой ОП должен быть произведен расчет проходного балла. Конкурсные списки 02.08 должны гарантировать, что проходной балл для каждой ОП может быть рассчитан. Участники поясняют, каким образом вычисляется проходной балл для каждой ОП с учетом того, что в списках есть абитуриенты с различными приоритетами (от 1 до 4 ОП).
 - d. Затем участники загружают список от 03.08. По каждой ОП также производится расчет проходного балла. Однако, для ОП ПМ и ИВТ этот балл должен повыситься относительно 02.08, а для ИТСС и ИБ упасть (конкурсные списки должны обеспечивать подобное поведение балла).
 - e. Затем участники загружают список от 04.08. По каждой ОП производится расчет проходного балла. Для всех образовательных программ происходит рост балла относительно 03.08. При этом конкурсные списки должны обеспечивать следующий результат по убыванию проходного балла на всех ОП: ПМ – ИБ – ИВТ – ИТСС (где на ПМ самый высокий проходной балл, а на ИТСС самый низкий из 4 заявленных ОП). Участники подробно объясняют, как происходит формирование проходных баллов по итоговым спискам, демонстрируют перемещение абитуриентов с одной ОП на другую из-за вытеснения их по приоритетам.
4. Испытание №3. Участники демонстрируют возможность формирования отчетов в соответствии с пунктом 2.14:
- a. Участники демонстрируют возможность формирования отчетов в формате pdf для каждого дня.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
Командный кейс № 3 «Анализ поступления»

- b. Участники демонстрируют наличие в отчете даты и времени, проходного балла на ОП, списки зачисленных абитуриентов (которые должны изменяться в течении хода приемной кампании и в соответствии с испытанием №2), графики динамики проходных баллов по ОП, и общую статистику.
- c. Отдельно участники демонстрируют общую статистику и сравнивают ее с конкурсными списками, доказывая, что алгоритм вычисления проходного балла работает верно.
- d. Статистика из пункта 2.14.е. должна быть заполнена ненулевыми значениями в соответствии с конкурсными списками.

7. Методические материалы

- Django documentation [Электронный ресурс]. — Режим доступа: <https://django.fun/docs/django/dev/>
- Express.js — Fast, unopinionated, minimalist web framework for Node.js [Электронный ресурс]. — Режим доступа: <https://expressjs.com/>
- Flask Framework Documentation [Электронный ресурс]. — Режим доступа: <https://flask.palletsprojects.com/en/stable/>
(см. также версия 2.0.x: <https://flask.palletsprojects.com/en/2.0.x/>)
- Git Documentation [Электронный ресурс]. — Режим доступа: <https://git-scm.com/doc>
- GitVerse. Система контроля версий [Электронный ресурс]. — Режим доступа: <http://gitverse.ru>
- MDN Web Docs. HTML, CSS и JavaScript [Электронный ресурс]. — Режим доступа: <https://developer.mozilla.org/en-US/>
- MySQL Documentation [Электронный ресурс]. — Режим доступа: <https://dev.mysql.com/doc/>
- Node.js Documentation [Электронный ресурс]. — Режим доступа: <https://nodejs.org/>

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**

**Профиль «Информационные технологии»
Командный кейс № 3 «Анализ поступления»**

- Oracle Documentation [Электронный ресурс]. — Режим доступа: <https://docs.oracle.com/en/>
- PostgreSQL Documentation [Электронный ресурс]. — Режим доступа: <https://www.postgresql.org/>
- Python Official Website [Электронный ресурс]. — Режим доступа: <https://www.python.org/>
- SQLite Documentation [Электронный ресурс]. — Режим доступа: <https://sqlite.org/>
- Vue.js Documentation [Электронный ресурс]. — Режим доступа: <https://vuejs.org/>

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 4 «Интерактивная платформа для освоения
цифровой грамотности»**

1. Условия

В современном мире цифровые технологии стали неотъемлемой частью повседневной жизни. Однако многие люди, особенно старшего поколения или начинающие пользователи, сталкиваются с трудностями при использовании смартфонов, компьютеров, популярных сайтов (Госуслуги, онлайн-банки), сервисов (такси, доставка) и мессенджеров. Существующие текстовые инструкции и видеоуроки часто оказываются сложными для восприятия, не учитывают психологические барьеры и отсутствие базовых навыков.

Для решения этой проблемы требуется интерактивная веб-платформа с изменением состояния в ответ на действия пользователей, которая обеспечивает безопасную среду для практического обучения через симуляции реальных интерфейсов. Т.е. она содержит не статичные «демо»-видео и фото, а полноценный симулятор. Такой подход позволяет снизить страх перед ошибками, дает возможность отработать действия без риска и повышает уверенность пользователей в использовании цифровых инструментов.

2. Техническое задание

Функциональное задание:

1. Обязательная функциональность:

1.1. Адаптивный веб-интерфейс, оптимизированный для различных устройств (ПК, планшеты, смартфоны) с учетом особенностей целевой аудитории:

- Крупные элементы управления (кнопки, иконки, шрифты).
- Голосовое сопровождение инструкций.
- Визуальные подсказки: анимация курсора, подсветка активных зон, интеграция скриншотов/скринкастов.
- Упрощенная навигация с линейным прохождением курсов и кнопкой «Помощь».
- Возможность выбора уровня сложности (Базовый/Расширенный).

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 4 «Интерактивная платформа для освоения
цифровой грамотности»**

1.2. Система управления курсами с интерактивными модулями:

- Темы: «Основы смартфона», «Работа с мессенджером МАХ», «Онлайн-покупки» «Госуслуги».
- Сценарий для темы «Основы смартфона»:
 - Базовый: включение и выключение смартфона, регулировка громкости
 - Расширенный: то же, что и в базовом с добавлением следующих пунктов: подключение к Wi-Fi (поиск сети → ввод пароля → успешное подключение), установка приложения из RuStore (поиск → выбор → установка).
- Сценарий для темы «Работа с мессенджером МАХ»:
 - Базовый: отправка текстового сообщения, отправка фото, создание нового чата с одним человеком (поиск контакта → создание чата → отправка приветственного сообщения)
 - Расширенный: то же, что и в базовом с добавлением следующих пунктов: создание группового чата (выбор контактов → создание → отправка приветственного сообщения), отправка эмоджи и стикеров.
- Сценарий для темы «Онлайн покупки»:
 - Базовый: поиск товара по каталогу, добавление в корзину,
 - Расширенный: то же, что и в базовом с добавлением следующего пункта: оформление доставки (выбор способа доставки курьер/пункт выдачи → выбор адреса доставки → выбор способа оплаты карта/сбп/при получении → подтверждение/создание заказа).
- Сценарий для темы «Госуслуги»:
 - Базовый: регистрация/вход (ввод номера телефона/почты/СНИЛС → ввод пароля), запись на приём к врачу (ввод полиса → выбор специальности → выбор врача → выбор поликлиники → выбор даты и времени → запись к врачу)
 - Расширенный: то же, что и в базовом с добавлением следующего пункта: получение электронного свидетельства пенсионера.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 4 «Интерактивная платформа для освоения
цифровой грамотности»**

- Пошаговые симуляции интерфейсов.
- Практические задания в виртуальной среде.
- Тесты для проверки усвоения материала.

1.3. Личный кабинет пользователя с отслеживанием прогресса:

- Статистика пройденных уроков.
- Система достижений и наград.
- Автосохранение прогресса.

2. Дополнительная функциональность:

- Виртуальный помощник (чат-бот) с контекстными подсказками на основе текущего шага курса.

3. Рекомендации к выполнению

- Реализовать проект по методологии Agile с разбивкой на спринты.
- Использовать компонентный подход на фронтенде (React/Vue.js).
- Организовать RESTful API на бэкенде (Node.js/Express или Python/Django) для взаимодействия с клиентской частью.
- Применить реляционную СУБД (PostgreSQL или MongoDB) для хранения данных пользователей и прогресса.
- Настроить автоматизированное тестирование (unit-тесты для ключевых модулей).
- Разработку рекомендуется вести с помощью системы контроля версий git.

4. Требования к документации

- Титульный лист (с указанием названия кейса и перечислением членов команды).
- Обоснование выбора языка программирования и используемых программных средств.
- Структурная и функциональная схемы программного продукта.
- Блок-схема работы основного алгоритма.
- Описание особенностей и аргументация выбранного типа СУБД.
- Схема базы данных.
- Программный код (ссылка на репозиторий), файл README должен включать:

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 4 «Интерактивная платформа для освоения
цифровой грамотности»**

- краткое описание проекта;
- инструкцию по установке/развертыванию;
- ссылку на видеоролик.

5. Требования к видеоролику

- Видеоролик должен демонстрировать функционирование разработанного программного продукта в соответствии с регламентом испытаний.
- На видео или записи экрана необходимо продемонстрировать выполнение каждого испытания, описанного в регламенте, в соответствии с условиями.
- Видео должно однозначно подтверждать авторство участников (во время записи ролика необходимо четко произнести название команды, ФИО участников, номер школы, ФИО руководителя).
- Видеоролик необходимо разместить на стороннем видеохостинге («ВКонтакте», Rutube и др.)

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ
ОЛИМПИАДА ШКОЛЬНИКОВ**
Профиль «Информационные технологии»
**Командный кейс № 4 «Интерактивная платформа для освоения
цифровой грамотности»**

6. Регламент испытаний

Испытания должны включать проверку всех функциональных требований, указанных в техническом задании, в том числе:

- Тестирование базового функционала:
 - Регистрация и авторизация пользователя.
 - Корректное отображение интерфейса на разных устройствах (мобильная версия и версия для ПК).
- Проверка системы обучения:
 - Запуск симуляции интерфейса (например, экрана мессенджера).
 - Выполнение практического задания (например, обязательного сценария «Работа с мессенджером МАХ»). Пользователь должен выполнить последовательность действий без ошибок.
 - Работа кнопки «Помощь».
- Оценка личного кабинета:
 - Отображение статистики прогресса.
 - Корректность сохранения данных после перезагрузки страницы.
 - Проверка дополнительных функций:
 - Работа чат-бота при запросе подсказки.
 - Адаптация контента под выбранный уровень сложности.

7. Методические материалы

Рекомендуется ознакомиться с материалами:

- Frontend:
 - React: <https://react.dev/>
 - Vue.js: <https://vuejs.org/>
- Backend:
 - Express.js: <https://expressjs.com/>
 - Node.js: <https://nodejs.org/en>
 - Django: <https://www.djangoproject.com/>
- Базы данных:
 - PostgreSQL: <https://www.postgresql.org/>
 - MongoDB: <https://www.mongodb.com/>