

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

Постановка задачи и SMART цель. Запуск демоверсии игры на сайте Steam

Дата создания: 16.11.2024

Авторы: Мациевский Илья, Митрофанов Матвей, Могилин Никита,
Поперёков Глеб

Ревизия: 1.0

Москва, 2024

Содержание

1	Постановка задачи по проблеме	3
1.1	План решения	3
2	Постановка задачи по результату	4
2.1	План решения	4
3	Постановка задачи по алгоритму	5
4	Пример неправильной постановки задачи по проблеме	8
4.1	План решения	8
5	Пример неправильной постановки задачи по результату	9
5.1	План решения	9
6	Пример неправильной постановки задачи по алгоритму	10
7	Постановка задачи по SMART	11
7.1	Конкретная (Specific)	11
7.2	Измеримая (Measurable)	11
7.3	Достижимая (Achievable)	11
7.4	Значимая (Relevant)	11
7.5	Ограниченная во времени (Time bound)	12
8	Примеры неправильной постановки задачи по SMART	13
8.1	Пример №1	13
8.2	Пример №2	13
8.3	Пример №3	13
8.4	Пример №4	13
8.5	Пример №5	13

1 Постановка задачи по проблеме

Проблема: Сложность использования демоверсий игр. Текущий процесс ознакомления с играми в Steam через демоверсии требует их предварительной загрузки на устройство пользователя. Это увеличивает время на принятие решения о покупке, снижает вовлеченность и может оттолкнуть часть потенциальных покупателей, особенно тех, кто не готов тратить ресурсы на загрузку.

Решение: Добавить возможность запускать демоверсии на странице игры в Steam.

Ожидаемая выгода: Увеличение продаж на платформе и рост пользовательской лояльности.

1.1 План решения

Необходимо описать архитектуру системы.

1. Проверка прав доступа: требуется организовать проверку прав доступа к демоверсии, проверку авторизации.
2. Интеграция стриминга: создание движка для запуска игр на платформе. На этапе MVP подключение к стороннему движку.
3. Балансировка и отказоустойчивость.
4. Сбор аналитики и отзывов: Реализовать модуль для сбора данных о времени игры, количестве запусков, возникающих ошибках, а также отзывов пользователей о демоверсии.
5. Рекомендательная система: Интегрировать данные из демо-версий в существующую систему рекомендаций для повышения персонализации контента.

Сроки

Дедлайн по выполнению задачи — 3 месяца с момента начала работы.

2 Постановка задачи по результату

Проблема: Сложность использования демоверсий игр. Текущий процесс ознакомления с играми в Steam через демоверсии требует их предварительной загрузки на устройство пользователя. Это увеличивает время на принятие решения о покупке, снижает вовлеченность и может оттолкнуть часть потенциальных покупателей, особенно тех, кто не готов тратить ресурсы на загрузку

Решение: Добавить возможность запускать демоверсии на странице игры в Steam.

Ожидаемая выгода: Увеличение продаж на платформе и рост пользовательской лояльности.

2.1 План решения

(При условии того, что архитектура реализации фичи была оговорена командой, например была нарисована C2 диаграмма уровней)

1. Требуется разработать микросервис авторизации и микросервис обработчика запросов на предоставление доступа к игре.
2. Требуется проксировать пользовательские запросы от Web и Desktop версий Steam на микросервисы авторизации и обработчик запросов на предоставление демо-игры через балансировщик, обеспечивая отказоустойчивость.
3. Интеграция стриминга: создание движка для запуска игр на платформе и его связь с микросервисом обработчика запросов. На этапе MVP подключение к стороннему движку.
4. Создание API-сервиса для дальнейшего взаимодействия с другими ("внешними" для нашей фичи) сервисами Steam'a.
5. Реализовать микросервис для получения обратной связи и осуществить возможность обращения к этому сервису через сервис стриминга игры.
6. Осуществить возможность обращения к API-сервису через микросервисы: авторизации, обработчика запросов, стриминга игры и получения обратной связи.
7. Проксировать запросы от API-сервиса к внешним сервисам системы через балансировщик, обеспечивая отказоустойчивость.

Сроки

Дедлайн по выполнению задачи — 3 месяца с момента начала работы.

3 Постановка задачи по алгоритму

Цель

Создать систему запуска демо-версии игры в Steam, благодаря которой пользователи смогут изучить игру перед покупкой.

MVP (Minimum Viable Product)

- **Основная функция:** Предоставление удаленного доступа к демо-версии игры без необходимости её загрузки.
- **Целевая аудитория:** Пользователи Steam.
- **Приоритеты:**
 - Высокий: предоставление прямого удаленного доступа к демо-версии определенной игры с её страницы, уменьшение затрат на предоставление услуги.
 - Низкий: сбор аналитических данных пользователей, составление рекомендаций для каждого пользователя.

Шаги разработки

1. Проектирование системы

Для визуализации архитектуры системы рекомендуется использовать C2-модель и Sequence-диаграмму. Подробнее о данных моделях написано в RFC-файле.

2. Разработка

Выбор технологий:

- **Backend:** Python (Django/Flask), Java (Spring Boot), GoLang, Ruby.
- **База данных:** PostgreSQL/MySQL, JDBC.
- **Очередь задач:** Kafka.

Реализация API:

Эндпоинты:

- **POST:** для отправки запросов от клиента, запросов от балансировщика соединений, записи в сервис логов.
- **GET/POST:** для авторизации и получения доступа к драйверу демо-игр.

Реализация логики:

- **Балансировщик соединений:**

- Принимает и обрабатывает запросы на доступ к демо-версии.
- Проверяет авторизацию пользователя (идет обращение к Сервису авторизации).
- Возвращает клиенту соединение к Сервису стриминга.

- **Сервис авторизации:**

- Получает от Балансировщика соединений запрос на проверку авторизации пользователя.
- Аутентификация пользователя.
- Возвращает Балансировщику соединений результат аутентификации.

- **Обработчик запросов:**

- В случае успешной авторизации получает от Балансировщика соединений запрос на предоставление доступа к демо-версии игры.
- Получение данных о пользователе.
- Подключение к БД. Проверка доступа демо-версии для пользователя.
- При наличии доступа передает данные о пользователе на Сервер логов.
- Передает запрос на доступ к демо-версии в Сервис стриминга.

- **Сервер логов:**

- Получает от Обработчика запросов данные о пользователе.
- Записывает данные в собственную БД.

- **Сервис стриминга:**

- Получает от Обработчика запросов запрос на доступ к демо-версии игры.
- Подключает движок, на котором происходит воспроизведение демо-версии игры.
- Возвращает Балансировщику соединений соединение к движку стриминга.

3. Тестирование

- **Unit-тесты:** Проверка работы отдельных компонентов (сервисов).
- **Интеграционные тесты:** Проверка взаимодействия компонентов.
- **Нагрузочное тестирование:** Проверка производительности системы при высокой нагрузке.
- **Тестирование на сообществе:** Выбор группы игроков (например, PUBG или Factorio) для тестирования системы и сбора обратной связи.

4. Поддержка и развитие

- **Мониторинг системы:** Отслеживание производительности и ошибок.
- **Внедрение новых функций:**
 - Сбор данных с Сервера логов для формирования рекомендаций на основе пройденных демо-версий игр.

Дополнительные рекомендации

- Документируйте свой код и API для удобства дальнейшей поддержки и развития.
- Используйте Git для контроля версий кода и совместной работы.
- Постоянно поддерживайте связь с командой для обсуждения прогресса и решения возникающих проблем.
- Просите коллег проверять ваш код для повышения качества (Code review).
- Помните о важности безопасности и масштабируемости микросервиса.

Сроки

Дедлайн по выполнению задачи — 3 месяца с момента начала работы.

4 Пример неправильной постановки задачи по проблеме

Проблема: отсутствие возможности играть в игры.

Описание проблемы: Пользователи не могут играть в игры, если у них нет доступа к полным версиям, а демоверсии требуют слишком много шагов для запуска. Это мешает людям знакомиться с играми, и в результате платформа теряет клиентов.

Решение: добавить возможность играть в полные версии игр бесплатно перед покупкой.

Ожидаемая выгода: увеличение времени, которое пользователи проводят на платформе, и рост числа скачиваний игр.

4.1 План решения

Необходимо описать архитектуру системы.

1. Проверка прав доступа не требуется, доступ к играм будет открыт для всех.
2. Интеграция стриминга: разработать собственный стриминговый движок сразу, без использования сторонних решений, и сделать его доступным для всех игр на платформе.
3. Сбор аналитики и отзывов: собрать только отзывы пользователей.
4. Рекомендательная система: не требуется, пользователи сами найдут игры, которые хотят попробовать.

5 Пример неправильной постановки задачи по результату

Проблема: Внедрение фичи "демоверсия игры" (Непонятно, что должна делать фича)

Описание проблемы: У пользователя должна быть возможность запускать демоверсию игры. (Непонятно, по какому принципу должна запускаться демоверсия со скачиванием/без скачивания/еще как-то)

Решение: Добавить возможность играть в полные версии игр бесплатно.

Ожидаемая выгода: Улучшение платформы. (Отсутствует четкая выгода со стороны пользователя, как улучшится его опыт взаимодействия с платформой, как эта фича повлияет на рост продаж различных игр)

5.1 План решения

(Архитектура реализации фичи не была оговорена командой)

1. Фича должна быть реализована через несколько микросервисов, которые связаны между собой. (Непонятно, какие микросервисы нужны, как они должны быть связаны между собой и как они должны взаимодействовать с внешними сервисами системы, отсутствует четкая декомпозиция задачи)

6 Пример неправильной постановки задачи по алгоритму

Цель

Разработка микросервиса "Демоверсия игр" для Steam: Пошаговое руководство для Junior разработчика.

MVP (Minimum Viable Product)

- **Основная функция:** Предоставление удаленного доступа к демо-версии игры без необходимости её загрузки.
- **Целевая аудитория:** Пользователи Steam.
- **Приоритеты:**
 - Высокий: предоставление прямого удаленного доступа к демо-версии определенной игры с её страницы, уменьшение затрат на предоставление услуги.
 - Низкий: сбор аналитических данных пользователей, составление рекомендаций для каждого пользователя.

Основные элементы:

1. Балансировщик соединений;
2. Сервис авторизации;
3. Обработчик запросов;
4. Сервер логов;
5. Сервис стриминга;

7 Постановка задачи по SMART

Постановка задачи по системе SMART должна удовлетворять пяти критериям:

- Specific – конкретная;
- Measurable – измеримая;
- Achievable – достижимая;
- Relevant – актуальная;
- Time bound – ограниченная во времени.

7.1 Конкретная (Specific)

Разработать, протестировать и внедрить систему запуска демо-версий игры в Steam. Система должна быть интегрирована со страницей игры, предоставлять удаленный доступ к ее демо-версии без необходимости загрузки.

7.2 Измеримая (Measurable)

Требуемые параметры системы:

- максимальное число одновременных сессий – 100000;
- время приостановки работы сервера (включая запланированное техническое обслуживание) – не более 2 часов в месяц;
- время отклика – не более 200 мс;
- процент покрытия кода модульным тестированием – не менее 80%.

7.3 Достижимая (Achievable)

По результатам анализа опыта команды и описанных требований, был сделан вывод, что реалистичное время разработки системы – 3 месяца.

7.4 Значимая (Relevant)

Ожидаемый результат от внедрения системы:

- уменьшение числа возвратов игр после их покупки на 75%, снижение нагрузки на сервис оплаты игр на 5%;
- улучшение пользовательского опыта при покупке игр;
- привлечение большей аудитории к новым играм.

7.5 Ограниченная во времени (Time bound)

Проект ограничен следующими сроками:

- начало работы над проектом: 1 декабря 2024 г.;
- предварительный показ проекта: 1 февраля 2025 г.;
- финальный показ проекта: 15 февраля 2025 г.;
- запуск работы сервиса: 1 марта 2025 г.

8 Примеры неправильной постановки задачи по SMART

8.1 Пример №1

Формулировка: За 3 месяца разработать систему удаленного запуска демо-версий игр, обеспечить 80% покрытия и не более 2 часов приостановки работы в месяц.

Проблема: Задача не поставлена конкретно, никак не описано взаимодействие с уже существующими системами.

8.2 Пример №2

Формулировка: За 3 месяца разработать систему запуска демо-версий игры в Steam. Система должна быть интегрирована со страницей игры, предоставлять удаленный доступ к ее демо-версии без необходимости загрузки.

Проблема: В постановке задачи никак не описаны показатели и метрики, по которым можно признать результат успешным/удовлетворительным/неудовлетворительным.

8.3 Пример №3

Формулировка: За 2 недели разработать систему запуска демо-версий игры в Steam. Система должна быть интегрирована со страницей игры, предоставлять удаленный доступ к ее демо-версии без необходимости загрузки. Обеспечить 80% покрытия и не более 2 часов приостановки работы в месяц.

Проблема: При отсутствии значительных наработок по этой сфере, срок в 2 недели не представляется реалистичным – задача не достижима.

8.4 Пример №4

Формулировка: За 3 месяца разработать систему запуска демо-версий игры в Steam. Система должна быть развернута в собственном разделе Steam, для каждой демо-версии игры должны быть созданы страницы соответствующей игры, на которых расположены ссылки на страницу полной версии игры. Система должна предоставлять удаленный доступ к ее демо-версии без необходимости загрузки. Обеспечить 80% покрытия и не более 2 часов приостановки работы в месяц.

Проблема: Во-первых, задача не является конкретной, поскольку описаны сразу две разные по своей сути задачи в рамках одной. Во-вторых, задача не актуальна, поскольку в Steam уже реализованы страницы игр, интеграцию с которыми было бы разумно использовать.

8.5 Пример №5

Формулировка: Разработать систему удаленного запуска демо-версий игр, обеспечить 80% покрытия и не более 2 часов приостановки работы в месяц. Система должна быть интегрирована со страницей игры, предоставлять удаленный доступ к ее демо-версии без необходимости загрузки.

Проблема: Сроки проекта никак не ограничены.