



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)

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

на разработку распределенной системы

«ИНТЕГРАЦИОННЫЙ ПОРТАЛ БРОНИРОВАНИЯ НАСТОЛЬНЫХ
ИГР»

Инв.№ подл.	Подп. и дата	Взаим.инв.№	Инв.№ дубл.	Подп. и дата

Глоссарий

1. Узел системы – региональный сервер, содержащий данные авторов и читателей указанного региона.
2. «Горячее» переконфигурирование системы – способность системы применять изменения без перезапуска и перекompиляции.
3. WEB-интерфейс – Интерфейс пользователя, предоставляемой системой через Web браузер. В разрабатываемой системе только один веб-интерфейс.
4. Медиана времени отклика – среднее время предоставления данных пользователю.
5. Валидация данных – проверка данных на соответствие заданным условиям и ограничениям.
6. REST (Representational State Transfer) – архитектурный стиль взаимодействия компонентов распределённого приложения в сети.
7. Фронтенд – серверное приложение, принимающее запросы от пользователя. На каждый из типов запросов определяется, как организовать его выполнение. Принимает запросы, анализирует их и в соответствии с заложенным алгоритмом выполняет запросы к бекендам.
8. Бекенд – серверное приложение, выполняющее определенную задачу, например, взаимодействие с СУБД. Бекенды принимают запросы от фронтенда.
9. ПО – программное обеспечение.

Введение

Данное техническое задание составлено для разработки интеграционного портала бронирования настольных игр объединяющего магазины настольных игр одной сетевой компании. Техническое задание выполнено на основе ГОСТ 19.201–78 «ЕСПД. Техническое задание. Требования к содержанию и оформлению».

Краткое описание предметной области

Настольная игра – игра, основанная на манипуляции относительно небольшим набором предметов, которые могут целиком разместиться на столе или в руках играющих.

Целевой аудиторией проекта являются люди от 12 лет, которым нравятся настольные игры. В магазинах сети для которой разрабатывается проект, продаются игры как долгие и сложные игры для группы не больше четырех человек, так и простые игры для большой компании.

Настольные игры подлежат обязательной сертификации согласно стандарту ТР ТС 008/2011 «О безопасности игрушек».

Для того чтобы пользователь не искал по магазинам сети, где интересующая его настольная игра есть в наличии необходимо создать портал который будет агрегировать все настольные игры во всех магазинах данной сети.

Для предотвращения выкупа настольной игры другим клиентом игра может быть забронирована на неделю. В течение этого времени клиент может оставить заявку на отказ от бронирования.

Необходимо предусмотреть, что покупатель может прийти в магазин и купить игру без предварительного бронирования.

Данное техническое задание определяет требования к разработке интеграционного портала для поиска и бронирования настольной игры в магазинах настольных игр.

Существующие аналоги

Среди аналогов можно такой интернет-магазин как «Игровед» (www.igroved.ru). Он позволяет искать и покупать настольные игры в магазинах их сети, но он не реализует функционал рекомендованного поиска игр. Также существуют такие маркетплейсы как Wildberries (www.wildberries.ru) и Ozon (www.ozon.ru), но за пользование их системой они берут комиссию с продавцов.

Основания для разработки

Разработка ведётся в рамках выполнения лабораторных работ по курсу «Методология программной инженерии» на кафедре «Программное обеспечение ЭВМ и информационные технологии» факультета «Информатика и системы управления» МГТУ им. Н.Э. Баумана.

Назначение разработки

Разрабатываемая система должна предоставлять пользователям возможность поиска наличия интересующих их настольных игр в магазинах по продаже настольных игр. Должен быть предусмотрен механизм бронирования настольных игр для предотвращения их покупки другими пользователями

Описание системы

Разрабатываемый сервис должен представлять собой распределённую систему для поиска и бронирования настольных игр. Каждый магазин имеет свою базу данных для учета товара.

Система должна обеспечивать разделение пользователей на три роли:

- **пользователь** — неавторизованный пользователь;
- **клиент** — авторизованный пользователь;
- **администратор** – пользователь ответственный за поддержку Системы.

Если **клиент** хочет оформить бронь, ему необходимо зарегистрироваться, указав информацию: фамилия, имя, электронная почта. В случае, если зарегистрированному ранее пользователю нужно оставить заявку на отмену бронирования или получить информацию о его бронированиях ему нужно авторизоваться.

Для **неавторизованных пользователей** доступен только поиск настольных игр по магазинам.

Для **администратора** должен быть предусмотрен функционал изменения наличия настольных игр в магазинах, а также добавление новых магазинов.

Общие требования к системе

Требования к системе следующие.

1. Разрабатываемое ПО должно поддерживать функционирование системы в режиме 24 часов, 7 дней в неделю, 365 дней в году (24/7/365) со среднегодовым временем доступности не менее 99.9%. Допустимое время, в течении которого система недоступна, за год должна составлять $24 \cdot 365 \cdot 0.001 = 8.76$ ч.
2. Время восстановления системы после сбоя не должно превышать 15 минут.
3. Каждый узел должен автоматически восстанавливаться после сбоя.
4. Обеспечить безопасность работоспособности за счёт отказоустойчивости узлов.

Требования к функциональным характеристикам

1. По результатам работы модуля сбора статистики медиана времени отклика системы на запросы пользователя на получение информации не должна превышать 3 секунд.
2. Медиана времени отклика системы на действия пользователя должна быть менее 0.8 секунд при условии работы на рекомендованной аппаратной конфигурации, задержках между взаимодействующими сервисами менее 0.2 секунды и одновременном числе работающих пользователей менее 100 на каждый сервер, обслуживающий внешний интерфейс.

3. Система должна обеспечивать возможность запуска в современных браузерах: не менее 85% пользователей Интернета должны пользоваться ей без какой-либо деградации функционала.

Функциональные требования к системе с точки зрения пользователя

Портал должен обеспечивать реализацию следующих функций:

1. Система должна обеспечивать регистрацию пользователей с проверкой правильности вводимых данных.
2. Система должна обеспечивать аутентификацию пользователей.
3. Система должна обеспечивать разделение пользователей на три роли.
4. Система должна предоставлять **администратору** следующие функции:
 - просмотр и изменение списка магазинов, входящих в сеть;
 - просмотр наличия настольных игр в магазинах сети;
 - авторизация в системе;
 - получение и изменение детальной информации по конкретному бронированию;
 - подтверждение оплаты бронирования;
 - отмена оформленного бронирования по заявке;
5. Система должна предоставлять **клиенту** следующие функции:
 - просмотр истории бронирований;
 - бронирование настольной игры в одном из магазинов, где она есть в наличии;
 - получение, изменение детальной информации по конкретному бронированию;
 - составление заявки на отмену оформленного заказа.

6. Система должна предоставлять **пользователю** следующие функции:

- просмотр информации о магазинах: контакты, местоположение;
- просмотр информации о настольных играх: название, описание, правила игры, наличие в магазинах;
- регистрация;
- авторизация.

Входные данные

Входные параметры системы представлены в таблице

1. Клиент / Администратор

- ФИО не более 256 символов каждое поле;
- электронная почта;
- пароль не менее 8 символов и не более 128, как минимум одна заглавная и одна строчная буква, только латинские буквы, без пробелов, как минимум одна цифра;
- номер телефона.

2. Магазин

- идентификатор;
- полный адрес;
- описание;
- список настольных игр в наличии;
- контактный телефон;
- электронная почта.

3. Бронирование

- идентификатор брони;

- список идентификаторов настольной игры;
- имя пользователя;
- идентификатор магазина;
- статус (ЗАБРОНИРОВАНО/ОТМЕНЕНО/ОПЛАЧЕНО);
- сумма для оплаты.

4. Заявка на отмену бронирования

- идентификатор брони;
- статус (СОЗДАНО/ВЫПОЛНЕНО).

5. Авторизация

- электронная почта;
- пароль.

1.7. Выходные данные Выходными параметрами системы являются веб-страницы. Они должны содержать следующую информацию:

- страницу регистрации и авторизации пользователя;
- информацию о магазине (адрес, описание, контактная информация);
- информация о наличии игры в магазинах;
- окно бронирования;
- информацию о бронировании;
- окно для администратора, позволяющее менять статус бронирования.

Топология Системы

На рисунке 1 изображён один из возможных вариантов топологии разрабатываемой распределенной Системы.

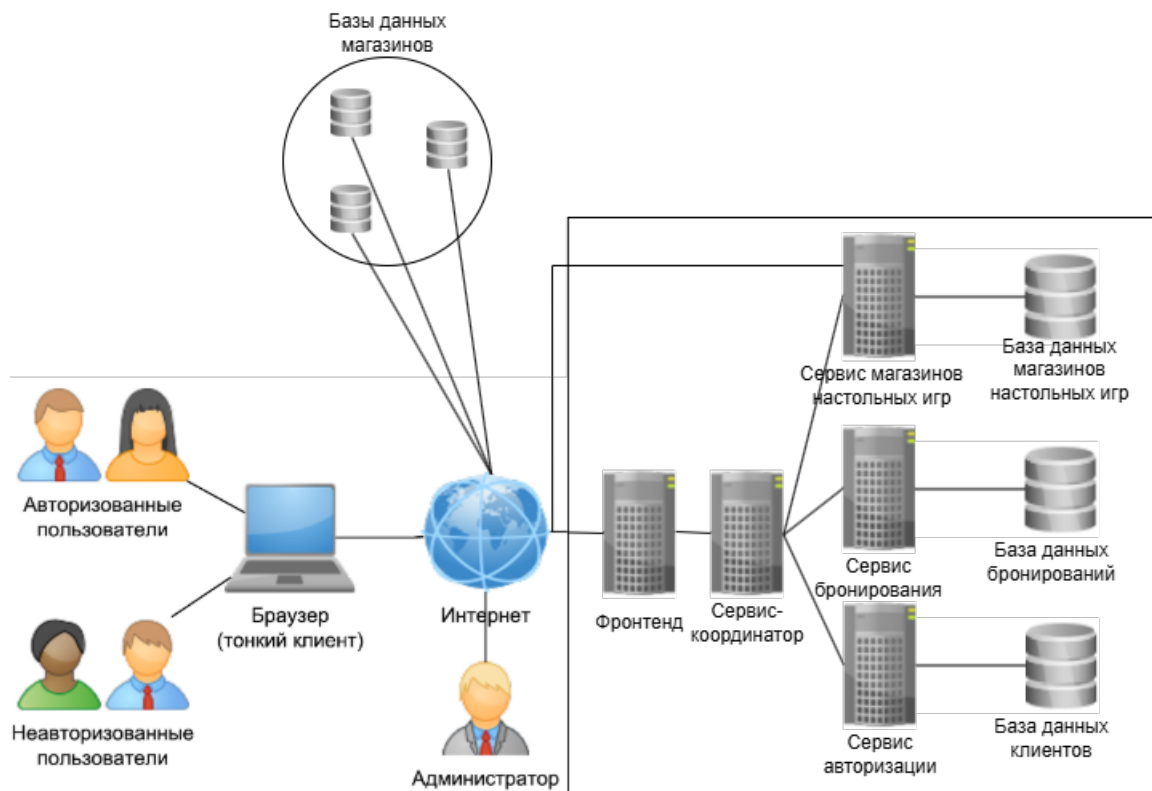


Рисунок 1 – Топология системы.

Система будет состоять из фронтенда и 5 подсистем:

- сервис-координатор;
- сервис регистрации и авторизации;
- сервис бронирования;
- сервис оплаты;
- сервис настольных игр.

Фронтенд принимает запросы от пользователей по протоколу HTTP и анализирует их. На основе проведенного анализа выполняет запросы к микросервисам бекенда, агрегирует ответы и отправляет их пользователю.

Сервис-координатор – единая точка входа и межсервисной коммуникации.

Сервис-регистрации и авторизации отвечает за:

- возможность регистрации нового клиента;

- аутентификацию пользователя (клиента и администратора);
- авторизацию пользователя;
- выход из сессии.

Сервис бронирования отвечает за:

- получение, создание бронирования;
- создание заявки на отмену бронирования;
- изменение статуса заявки бронирования;
- изменение статуса бронирования.

Сервис настольных игр реализует функции:

- получение списка игр которые есть в наличии в конкретном магазине;
- получение списка магазинов;
- добавление нового магазина.

Требования к программной реализации

1. Требуется использовать СОА (сервис-ориентированную архитектуру) для реализации системы.
2. Система состоит из микросервисов. Каждый микросервис отвечает за свою область логики работы приложения и должны быть запущены изолированно друг от друга.
3. При необходимости, каждый сервис имеет своё собственное хранилище, запросы между базами запрещены.
4. При разработке базы данных необходимо учитывать, что доступ к ней должен осуществляться по протоколу ТСП.
5. Необходимо реализовать один web-интерфейс для фронтенда. Интерфейс должен быть доступен через тонкий клиент (браузер).

6. Для межсервисного взаимодействия использовать HTTP (придерживаться RESTful).
7. Выделить Gateway Service как единую точку входа и межсервисной коммуникации. В системе не должно осуществляться горизонтальных запросов.
8. При недоступности систем портала должна осуществляться деградация функционала или выдача пользователю сообщения об ошибке.
9. Необходимо предусмотреть авторизацию пользователей, как через интерфейс приложения, так и через популярные социальные сети.
10. Проверку правильности входных данных необходимо проводить и на стороне пользователя, и на стороне фронтенда. Микросервисы бекенда не должны валидировать входные данные, поскольку пользователь не может к ним обращаться напрямую, они должны получать уже отфильтрованные входные данные.
11. Для запросов, выполняющих обновление данных на нескольких узлах распределенной системы, в случае недоступности одной из систем, необходимо выполнять полный откат транзакции.
12. Приложение должно поддерживать возможность горизонтального и вертикального масштабирования за счет увеличения количества функционирующих узлов и совершенствования технологий реализации компонентов и всей архитектуры системы.

Функциональные требования к подсистемам

Подсистемы: фронтенд, бекенд-координатор, бекенд регистрации и авторизации, бекенд бронирования, бекенд оплаты, бекенд настольных игр.

Фронтенд – серверное приложение, предоставляет пользовательский интерфейс и внешний API системы, при разработке которого нужно учитывать следующее:

- должен принимать запросы по протоколу HTTP и формировать ответы пользователям в формате HTML;

- в зависимости от типа запроса должен отправлять последовательные запросы в соответствующие микросервисы;
- запросы к микросервисам необходимо осуществлять по протоколу HTTP;
- данные необходимо передавать в формате JSON.

Сервис-координатор – серверное приложение, через которое проходит весь поток запросов и ответов, должен соответствовать следующим требованиям разработки:

- принимать и возвращать данные в формате JSON по протоколу HTTP;
- накапливать статистику запросов, в случае, если система не ответила N раз, то в $N + 1$ раз вместо запроса сразу отдавать fallback. Через некоторое время выполнить запрос к реальной системе, чтобы проверить её состояние;
- выполнять проверку существования клиента, также регистрацию и аутентификацию пользователей;
- получение списка всех магазинов с интересующей пользователя игрой в наличии;
- получение информации и обновление данных о зарегистрированном пользователе;
- оформление и изменение бронирования;
- получение данных о бронированиях пользователя;
- получение, добавление, изменение, удаление детальной информации о магазине;
- оформление заявки на отмену бронирования.

Сервис-регистрации и авторизации должен реализовывать следующие функциональные возможности:

- принимать и возвращать данные в формате JSON по протоколу HTTP;

- возможность регистрации нового клиента и обновление данных уже существующего;
- проверка существования клиента;
- обеспечение авторизации пользователя через аккаунт в системе.

Хранимая в базе данных сущность **аккаунта**, ассоциированная с сервисом:

- *фамилия*, не более 256 символов;
- *имя*, не более 256 символов;
- *отчество*, не более 256 символов;
- *захешированный пароль*;
- *номер телефона*;
- *электронная почта*, первичный ключ.

Сервис бронирования реализует следующие функции:

- получение и отправка данных в формате JSON по протоколу HTTP;
- получение таких данных, как:
 - информация о забронированных играх в конкретном магазине;
 - все бронирования, зарегистрированные на конкретного клиента;
 - информация о конкретном бронировании по его идентификатору;
- составление заявки на отмену бронирования;
- создание, редактирование и отмена бронирования.

Соответствующая база данных содержит следующую сущность **бронирования**.

- *идентификатор*, является первичным ключом;
- *электронная почта клиента*;

- *идентификатор настольной игры;*
- *идентификатор магазина;*
- *статус, RESERVED/CANCELED/PAID;*
- *дата создания.*

Сущность **заявки на отмену бронирования.**

- *идентификатор, является первичным ключом;*
- *идентификатор бронирования;*
- *статус, CREATED/COMPLETED.*

Сервис настольных игр должен реализовывать представленные такие функциональные возможности, как:

- получение и отправка ответов на запросы в формате JSON по протоколу HTTP;
- получение списка магазинов в которых есть в наличии настольная игра;
- получение наличия конкретных игр в конкретных магазинах;
- получение информации о конкретной настольной игре;
- получение, добавление, изменение, удаление информации о конкретном магазине.

Соответствующая база данных содержит следующие сущности.

1. **Магазин.**

- *идентификатор, первичный ключ;*
- *адрес;*
- *описание;*
- *контактная информация.*

Требования к составу и параметрам технических средств

Все серверные приложения должны потреблять суммарно не более 4 Гбайт оперативной памяти и работать на сервере с процессором AMD Ryzen 5 4600H 3.00 GHz.

Требования к надёжности

Система должна работать в соответствии с данным техническим заданием без рестарта. Необходимо использовать «зеркалируемые серверы» для всех подсистем, которые будут держать нагрузку в случае сбоя до тех пор, пока основной сервер не восстановится.

Требования к документации

Исполнитель должен подготовить и передать заказчику руководство:

- для Администратора Системы;
- для Пользователя Системы;
- для Клиента Системы;
- по развёртыванию Системы.

ПРОФИЛЬ СТАНДАРТОВ

1. ГОСТ 19.201–78 «ЕСПД. Техническое задание. Требования к содержанию и оформлению»
2. ТР ТС 008/2011 «О безопасности игрушек»