

5.2.2. – Математические, статистические и инструментальные методы в экономике (физико-математические науки, экономические науки)

5.2.2 – Mathematical, statistical and instrumental methods in economics (physical and mathematical sciences, economic sciences)

ПРОЕКТИРОВАНИЕ ПРИЛОЖЕНИЯ ДЛЯ ЭФФЕКТИВНОГО ПОДБОРА КВАРТИР

DESIGNING AN APPLICATION FOR AN EFFECTIVE SELECTION OF APARTMENTS

Авакимян Наталья Николаевна

К.ф.–м.н., доцент

РИНЦ SPIN–код: 6082–4770

email: avnatali@mail.ru

ФГБОУ «Кубанский государственный аграрный университет», 350044, Россия, г. Краснодар, ул. Калинина 13

Avakimyan Natalia Nikolaevna

Cand.Phys.–Math.Sci., associate professor

RSCI SPIN–code: 6082–4770

email: avnatali@mail.ru

Kuban State Agricultural university, 350044, Russia, Krasnodar, Kalinina, 13

Гриднев Максим Александрович

студент группы БИ2102

email: gridnev@yandex.ru

ФГБОУ «Кубанский государственный аграрный университет», 350044, Россия, г. Краснодар, ул. Калинина 13

Gridnev Maksim Aleksandrovich

student of group BI2102

email: gridnev@yandex.ru

Kuban State Agricultural university, 350044, Russia, Krasnodar, Kalinina, 13

Платформа для эффективного подбора квартир предназначена для упрощения и ускорения процесса подбора квартир, обеспечивая пользователям удобный доступ к актуальным предложениям. Система использует алгоритмы, которые анализируют предпочтения клиента и предлагают наиболее подходящие варианты. Это снижает затраты времени на поиск, исключает нерелевантные объекты и повышает вероятность успешной сделки. Для удобства пользователей предусмотрены фильтры, интерактивные карты и рекомендации на основе анализа рынка. В результате клиенты получают персонализированные предложения, а арендаторы и продавцы – более целевую аудиторию. В данной статье для выбранной предметной области выделены основные пользователи системы и роли, бизнес-требования, Epic-задача, детализированы User Story для данного проекта для пользователя и клиента. Расписаны критерии приемки для некоторых Story. Приведены примеры UseCase и шаблон для его описания. Описаны основные сущности и структура сформированной базы данных

The platform for the effective selection of apartments is designed to simplify and accelerate the process of selecting apartments, providing users with convenient access to current offers. The system uses algorithms that analyze the preferences of the client and offer the most suitable options. This reduces the time of time to search, excludes irrelevant objects and increases the likelihood of a successful transaction. For the convenience of users, filters, interactive cards and recommendations based on market analysis are provided. As a result, customers receive personalized offers, and tenants and sellers - a more target audience. In this article, for the selected subject area, the main users of the system and roles, business requirements, an EPIC-product are identified, user Story is detailed for this project for the user and the client. Criteria for acceptance and TASK-goals for some Story are painted. Examples of UseCase and a template are given to describe it. The main entities and structure of the formed database are described

Ключевые слова: БИЗНЕС-ТРЕБОВАНИЯ, USER CASE, USER STORY, СУЩНОСТИ, МОНОЛИТНАЯ АРХИТЕКТУРА

Keywords: BUSINESS REQUIREMENTS, USE CASE, USER STORY, ENTITIES, MONOLITHIC ARCHITECTURE

<http://dx.doi.org/10.21515/1990-4665-209-001>

Платформа для эффективного подбора квартир предназначена для упрощения и ускорения процесса подбора квартир, обеспечивая

<http://ej.kubagro.ru/2025/05/pdf/01.pdf>

пользователям удобный доступ к актуальным предложениям. Система использует алгоритмы, которые анализируют предпочтения клиента и предлагают наиболее подходящие варианты. Это снижает затраты времени на поиск, исключает нерелевантные объекты и повышает вероятность успешной сделки. Для удобства пользователей предусмотрены фильтры, интерактивные карты и рекомендации на основе анализа рынка. В результате клиенты получают персонализированные предложения, а арендаторы и продавцы – более целевую аудиторию.

Целевая аудитория веб-сервиса:

– покупатели и арендаторы – люди, ищущие квартиру для покупки или аренды;

– собственники недвижимости – владельцы квартир, желающие продать или сдать жилье;

– агентства и риелторы – профессиональные посредники, работающие с недвижимостью;

– администраторы платформы – технические специалисты, управляющие системой, модерлирующие контент и обеспечивающие корректную работу сервиса.

Основные роли в системе:

- Пользователь (арендатор/покупатель) – ищет квартиру, использует фильтры, просматривает объекты, связывается с владельцами.

- Владелец недвижимости – размещает объявления, управляет предложениями, общается с потенциальными клиентами.

- Риелтор/Агентство – размещает и управляет объявлениями, ведет переговоры с клиентами.

- Администратор – модерлирует объявления, контролирует соблюдение правил платформы, следит за техническим состоянием сервиса.

Бизнес-требования:

– автоматизированный подбор недвижимости. Платформа должна предоставлять пользователям интеллектуальный механизм поиска и рекомендаций, основанный на заданных фильтрах, предпочтениях и анализе рынка;

– эффективное управление объявлениями. Владельцы недвижимости и риелторы должны иметь возможность быстро размещать, редактировать и управлять своими объявлениями, включая загрузку фото, описание и параметры объекта;

– удобная коммуникация между пользователями. Необходимо реализовать встроенный чат или систему обмена контактами для оперативного взаимодействия между покупателями, арендаторами и владельцами недвижимости;

– прозрачность и безопасность сделок. Система должна обеспечивать верификацию пользователей, защиту от мошенничества и предоставлять надежные инструменты оценки объектов, такие как отзывы, рейтинги и проверенные документы;

– аналитика и персонализация. Платформа должна собирать и анализировать пользовательские предпочтения, предоставляя персонализированные рекомендации и статистику по рынку недвижимости.

В рамках данного проекта будет сформирован один Epic «RPS-001 Разработка платформы для подбора недвижимости (Real Estate Property Selection Platform Development). Создание веб-сервиса, который обеспечивает удобный поиск, покупку и аренду недвижимости, управление объявлениями, безопасное взаимодействие между пользователями и персонализированные рекомендации.

Для проекта можно выделить следующие Story:

1. RPS-002 Регистрация пользователя
2. RPS-003 Авторизация пользователя
3. RPS-004 Поиск недвижимости
4. RPS-005 Размещение объявления
5. RPS-006 Управление объявлениями
6. RPS-007 Коммуникация между пользователями
7. RPS-008 Верификация пользователей
8. RPS-009 Аналитика и рекомендации
9. RPS-010 Оставление отзывов и рейтингов
10. RPS-011 Управление пользователями
11. RPS-012 Модерация объявлений
12. RPS-013 Обработка жалоб
13. RPS-014 Управление статистикой и аналитикой
14. RPS-015 Управление рекламой и монетизацией

Опишем выделенные Story по технике UserStory. UserStory для незарегистрированного пользователя приведены в таблице 1.

Таблица 1 – Незарегистрированный пользователь

Id	As a <type of user>	I can <functionality>	so that <rationale>
User	Незарегистрированный пользователь	Зарегистрироваться в системе	Для получения доступа к возможностям платформы
User	Авторизация на портале	Произвести вход по логину и паролю	Для возможности работы с платформой

UserStories для клиента (покупателя/ арендатора) приведены в таблице 2.

Таблица 2 - Клиент (покупатель/арендатор)

Id	As a <type of user>	I can <functionality>	so that <rationale>
Client	Зарегистрированный клиент	Авторизоваться в системе	Для получения доступа к сайту
Client	Зарегистрированный клиент	Искать недвижимость по заданным параметрам	Для подбора подходящих вариантов
Client	Зарегистрированный клиент	Просматривать объявления с детальной информацией	Чтобы выбрать наиболее подходящий объект
Client	Зарегистрированный клиент	Общаться с владельцами недвижимости	Чтобы обсудить условия сделки
Client	Зарегистрированный клиент	Добавлять объявления в избранное	Для удобного доступа к понравившимся вариантам
Id	As a <type of user>	I can <functionality>	so that <rationale>
Client	Зарегистрированный клиент	Оставлять отзывы и оценки владельцам недвижимости	Чтобы делиться опытом с другими пользователями

UserStories для владельца недвижимости (продавец / арендодатель) приведены в таблице 3.

Таблица 3 – Владелец недвижимости (продавец/арендодатель)

Id	As a <type of user>	I can <functionality>	so that <rationale>
Owner	Зарегистрированный владелец недвижимости	Авторизоваться в системе	Для управления своими объявлениями
Owner	Зарегистрированный владелец недвижимости	Размещать объявления о продаже или аренде	Для поиска потенциальных покупателей или арендаторов
Owner	Зарегистрированный владелец недвижимости	Управлять своими объявлениями (редактировать, удалять)	Чтобы поддерживать актуальность информации
Owner	Зарегистрированный владелец недвижимости	Отвечать на запросы пользователей	Для обсуждения условий сделки
Owner	Зарегистрированный владелец недвижимости	Просматривать аналитику по своим объявлениям	Чтобы повышать их эффективность
Owner	Зарегистрированный владелец недвижимости	Получать отзывы и оценки от клиентов	Для улучшения репутации и привлечения новых клиентов

UserStories для риелтора (агентства недвижимости) приведены в таблице 4.

Таблица 4 – Риелтор (агентство недвижимости)

Id	As a <type of user>	I can <functionality>	so that <rationale>
Realtor	Зарегистрированный риелтор	Авторизоваться в системе	Для работы с объектами недвижимости
Realtor	Зарегистрированный риелтор	Размещать объявления от имени клиентов	Чтобы находить покупателей и арендаторов
Id	As a <type of user>	I can <functionality>	so that <rationale>
Realtor	Зарегистрированный риелтор	Управлять портфелем объектов	Для эффективной работы с клиентами
Realtor	Зарегистрированный риелтор	Использовать систему аналитики	Чтобы предлагать клиентам выгодные решения
Realtor	Зарегистрированный риелтор	Оставлять и получать отзывы	Чтобы повысить доверие пользователей

UserStories для администратора приведены в таблице 5.

Таблица 5 – Администратор

Id	As a <type of user>	I can <functionality>	so that <rationale>
Admin	Администратор	Авторизоваться в систему	Для управления платформой
Admin	Администратор	Управлять пользователями (блокировать, верифицировать)	Чтобы обеспечивать безопасность и исключать мошенников
Admin	Администратор	Модерировать объявления (одобрять, отклонять, редактировать)	Чтобы поддерживать качество и актуальность контента
Admin	Администратор	Просматривать статистику аналитику активности пользователей	Для контроля эффективности платформы
Admin	Администратор	Обрабатывать жалобы от пользователей	Чтобы оперативно решать конфликты и улучшать пользовательский опыт
Admin	Администратор	Управлять рекламными платными функциями платформы	Для монетизации сервиса

Критерии приемки для Scenario «Регистрация пользователя на портале» приведены на рисунке 1.

Scenario: Регистрация пользователя на портале Given пользователь находится на странице регистрации When пользователь вводит "Фамилию" And пользователь вводит "Имя" And пользователь вводит "Email" And пользователь вводит "Номер телефона" And пользователь выбирает "Роль (Клиент, Владелец, Риэлтор)" And пользователь вводит "Пароль" And пользователь подтверждает "Пароль" Then система должна отобразить сообщение об успешной регистрации And пользователь должен быть перенаправлен на страницу входа

Рисунок 1 – Критерии приемки для Scenario «Регистрация пользователя на портале»

Критерии приемки для Scenario «Авторизация» приведены на рисунке 2.

Scenario: Авторизация Given пользователь находится на странице входа When пользователь вводит корректный "Email" And пользователь вводит корректный "Пароль" And пользователь нажимает кнопку "Войти" Then система должна авторизовать пользователя And пользователь должен быть перенаправлен в личный кабинет

Рисунок 2 – Критерии приемки для Scenario «Авторизация»

Критерии приемки для Scenario «Поиск недвижимости» приведены на рисунке 3.

Scenario: Поиск недвижимости Given пользователь находится на странице поиска When пользователь вводит "Локацию" And пользователь выбирает "Ценовой диапазон" And пользователь выбирает "Тип недвижимости" And пользователь нажимает кнопку "Поиск" Then система должна отобразить список недвижимости, соответствующей критериям
--

Рисунок 3 – Критерии приемки для Scenario «Поиск недвижимости»

Критерии приемки для Scenario «Размещение объявления» приведены на рисунке 4.

Scenario: Размещение объявления Given пользователь авторизован как "Владелец" или "Риэлтор" When пользователь переходит на страницу "Разместить объявление" And пользователь вводит "Детали недвижимости" (заголовок, описание, цена, локация) And пользователь загружает "Фотографии" And пользователь нажимает кнопку "Отправить" Then система должна сохранить объявление And объявление должно отображаться в результатах поиска

Рисунок 4 – Критерии приемки для Scenario «Размещение объявления»

Критерии приемки для Scenario «Управление объявлениями» приведены на рисунке 5.

Scenario: Управление объявлениями Given пользователь авторизован как "Владелец" или "Риэлтор" And у пользователя есть хотя бы одно активное объявление When пользователь переходит в "Мои объявления" And пользователь выбирает объявление для редактирования And пользователь обновляет детали объявления And пользователь нажимает кнопку "Сохранить" Then система должна успешно обновить объявление
--

Рисунок 5 – Критерии приемки для Scenario «Управление объявлениями»

Критерии приемки для Scenario «Коммуникация между пользователями» приведены на рисунке 6.

Scenario: Коммуникация между пользователями Given пользователь авторизован And пользователь находится на странице объявления недвижимости When пользователь нажимает кнопку "Отправить сообщение" And пользователь вводит текст сообщения And пользователь нажимает "Отправить" Then система должна отправить сообщение владельцу недвижимости And владелец должен получить уведомление
--

Рисунок 6 – Критерии приемки для Scenario «Коммуникация между пользователями»

Критерии приемки для остальных Scenario отражены в таблице 6.

Таблица 6 – Критерии приемки

Scenario	Описание
Верификация пользователей	<p>Given пользователь авторизован</p> <p>When пользователь загружает "Документы для верификации"</p> <p>And пользователь отправляет их на проверку</p> <p>Then система должна отметить профиль как "Ожидает верификации"</p> <p>And администратор должен проверить документы</p> <p>And после одобрения система должна отметить пользователя как "Верифицирован"</p>
Оставление отзывов и рейтингов	<p>Given пользователь завершил сделку</p> <p>When пользователь переходит в раздел "Оставить отзыв"</p> <p>And пользователь выбирает рейтинг от 1 до 5 звезд</p> <p>And пользователь вводит текст отзыва</p> <p>And пользователь нажимает "Отправить"</p> <p>Then система должна сохранить отзыв</p> <p>And отзыв должен быть видим на профиле оцениваемого пользователя</p>
Управление пользователями	<p>Given администратор авторизован</p> <p>And администратор находится на странице "Управление пользователями"</p> <p>When администратор выбирает пользователя</p> <p>And нажимает кнопку "Заблокировать"</p> <p>Then система должна отметить пользователя как "Заблокирован"</p> <p>And пользователь больше не должен иметь возможность входить в систему</p>
Модерация объявлений	<p>Given администратор авторизован</p> <p>And администратор находится на странице "Ожидающие объявления"</p> <p>When администратор выбирает объявление</p> <p>And нажимает кнопку "Одобрить"</p> <p>Then объявление должно стать доступным для всех пользователей</p>
Обработка жалоб	<p>Given пользователь авторизован</p> <p>When пользователь переходит в раздел "Подать жалобу"</p> <p>And пользователь выбирает категорию жалобы</p> <p>And пользователь вводит описание жалобы</p> <p>And пользователь нажимает "Отправить"</p> <p>Then система должна сохранить жалобу</p> <p>And администратор должен получить уведомление о новой жалобе</p>
Управление статистикой и аналитикой	<p>Given администратор авторизован</p> <p>When администратор переходит в раздел "Статистика и аналитика"</p> <p>And администратор выбирает период отчета</p> <p>And администратор анализирует графики и показатели</p> <p>Then система должна отобразить актуальные аналитические данные</p>

Описание UseCase «Пользователь.Регистрация пользователя» приведено в таблице 7.

Таблица 7 – Описание UseCase «Пользователь.Регистрация пользователя»

Код и наименование ВИ	Пользователь.Регистрация пользователя
Краткое описание	Пользователь заполняет форму регистрации для создания аккаунта
Актеры Главный (инициирует ВИ) Второстепенный	Пользователь
Ссылки на другие ВИ	-
Предусловия	Открыта страница регистрации
Основной поток	1) Пользователь вводит данные 2) Нажимает кнопку "Зарегистрироваться" 3) Система отправляет код подтверждения 4) Пользователь вводит код 5) Аккаунт создается
Постусловия	Пользователь получает подтверждение о регистрации
Альтернативные потоки	1) Введены некорректные данные 2) Код подтверждения не введен
Альтернативный поток 1	Код подтверждения не совпадает
	1) АП начинается после шага 3 основного потока 2) Система выдает сообщение, о формировании нового кода
Специальные требования	Не быть ранее авторизованным
Точки расширения	Нет
Дополнительная информация	Нет

Платформа для эффективного подбора квартир состоит из следующих сущностей: «Пользователи», «Район», «Квартиры», «Заявки», «Статус заявки», «Администратор», «Собственник недвижимости», «Агентство» (рисунок 7).

Для старта веб-сервиса по поиску и аренде недвижимости целесообразно использовать монолитную архитектуру. Это позволит сократить время на разработку и развертывание, а также упростит поддержку системы.

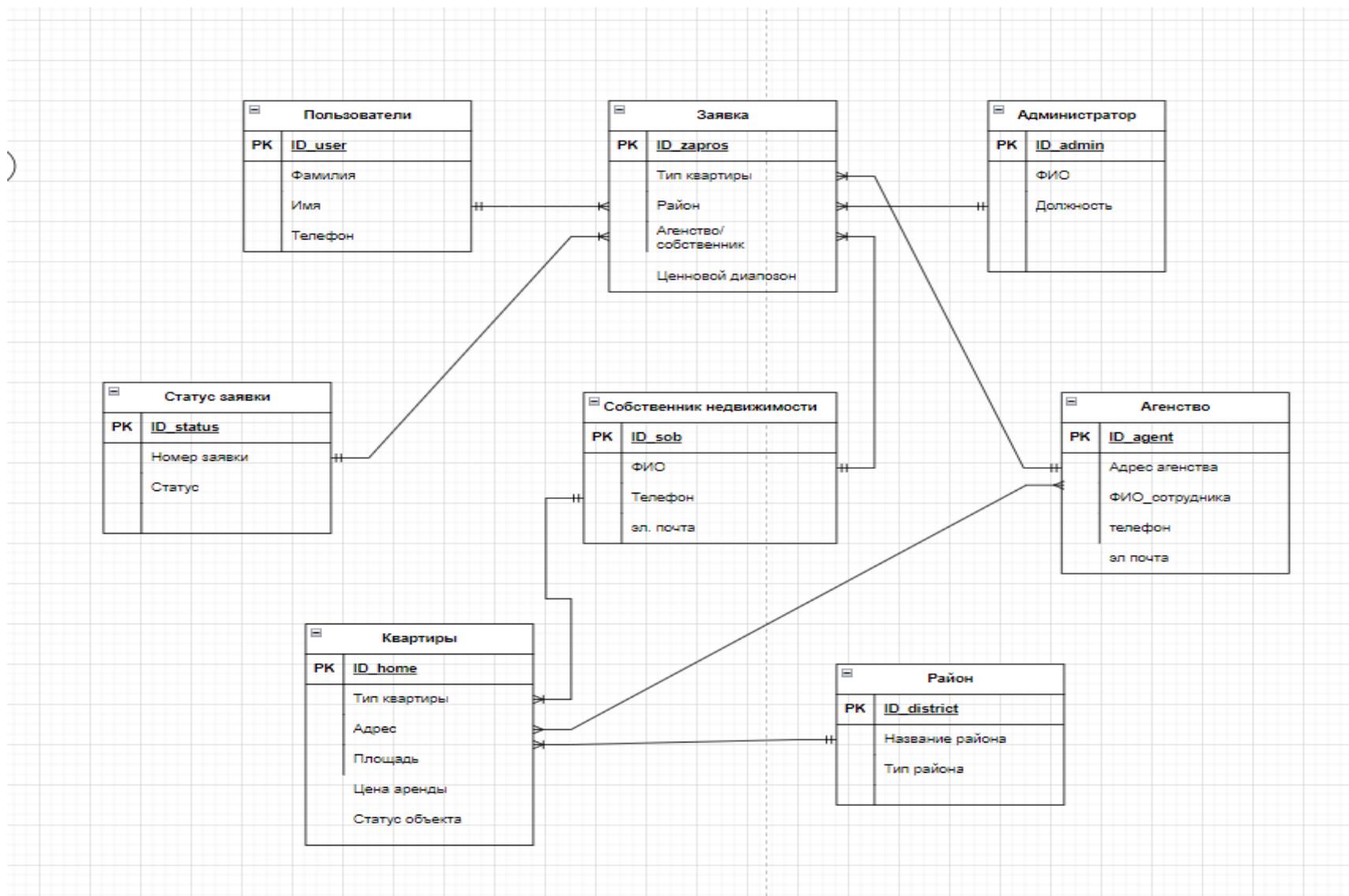


Рисунок 7 – Сущности платформы

Преимущества монолитной архитектуры:

- простота разработки и развертывания;
- единая база данных, что облегчает работу с данными;
- упрощенное тестирование, так как все компоненты находятся в одном месте;
- минимальные затраты на начальном этапе;
- перспективы масштабирования.

При увеличении нагрузки и расширении функциональности сервиса может возникнуть необходимость в переходе на микросервисную архитектуру. Это позволит разделить систему на независимые сервисы, что повысит отказоустойчивость, упростит масштабирование отдельных компонентов и ускорит разработку новых функций.

Список использованной литературы:

Разработка требований к программному обеспечению / К. Вигерс, Д. Битти. – СПб : БХВ, 2024. – 736 с.

References:

Razrabotka trebovanij k programmnomu obespecheniju / K. Vigers, D. Bitti. – SPb : BHV, 2024. – 736 s.