

МЕЖГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»

Кафедра «Программное обеспечение информационных технологий»

ВЕБ-ТЕХНОЛОГИИ И ПРОГРАММИРОВАНИЕ

*Методические рекомендации к курсовому проектированию
для студентов специальности
6-05-0611-04 «Электронная экономика»
очной и заочной форм обучения*



Могилев 2026

УДК 004.4
ББК 32.9
В26

Рекомендовано к изданию
учебно-методическим отделом
Белорусско-Российского университета

Одобрено кафедрой «Программное обеспечение информационных технологий» «2» марта 2026 г., протокол № 8

Составители: канд. техн. наук, доц. С. К. Крутолевич;
М. А. Шалухова

Рецензент канд. техн. наук, доц. В. М. Ковальчук

Методические рекомендации к курсовому проектированию предназначены для студентов специальности 6-05-0611-04 «Электронная экономика» очной и заочной форм обучения.

Учебное издание

ВЕБ-ТЕХНОЛОГИИ И ПРОГРАММИРОВАНИЕ

Ответственный за выпуск	В. В. Кутузов
Корректор	А. А. Подошевка
Компьютерная верстка	М. М. Дударева

Подписано в печать 23.09.2026. Формат 60×84/16. Бумага офсетная. Гарнитура Таймс.
Печать трафаретная. Усл. печ. л. 0,93. Уч.-изд. л. 0,94. Тираж 21 экз. Заказ № 725.

Издатель и полиграфическое исполнение:
Межгосударственное образовательное учреждение высшего образования
«Белорусско-Российский университет».
Свидетельство о государственной регистрации издателя,
изготовителя, распространителя печатных изданий
№ 1/156 от 07.03.2019.
Пр-т Мира, 43, 212022, г. Могилев.

© Белорусско-Российский
университет, 2026

Содержание

Введение.....	4
1 Цель, организация, содержание и оформление курсового проекта.....	5
2 Методические указания по выполнению курсового проекта.....	5
2.1 Анализ бизнес-процессов пользователей	5
2.2 Моделирование взаимодействия акторов с ИС.....	6
2.3 Разработка дизайна программного продукта.....	11
2.4 Отчёт о разработке программных компонентов и тестировании.....	13
3 Заключение по проекту.....	14
Список литературы.....	14
Приложение А. Типовой бланк задания.....	15

Введение

Методические рекомендации содержат информацию о требованиях к выполнению курсового проекта.

В ходе выполнения курсового проекта студент демонстрирует умения решать задачи профессиональной деятельности на основе применения информационно-коммуникационных технологий и навыки в области поддержки принятия управленческих решений с применением технологий бизнес-аналитики.

Студент подтверждает компетенции:

- осуществлять проектирование информационных систем (ИС) с использованием CASE-средства Enterprise Architect;
- осуществлять документирование требований заказчика с помощью диаграмм вариантов использования и бизнес-процессов;
- документировать поведение проектируемой системы с использованием диаграмм поведения;
- разрабатывать структуру проектируемой системы с использованием диаграммы классов;
- прототипировать программные продукты с использованием редактора Figma.

1 Цель, организация, содержание и оформление курсового проекта

Целью курсового проектирования является изучение технологии проектирования и разработки ИС.

Выполнение курсового проекта осуществляется студентом на основе выданного ему индивидуального задания. Тематика курсового проектирования строится на материале предприятий и организаций, где студенты проходили производственную практику, и на научных работах сотрудников кафедры.

Содержание курсового проекта должно строго соответствовать выданному студенту задания. Типовой бланк задания представлен в приложении А.

Оформление курсового проекта должно соответствовать требованиям ГОСТ 2.105–95. Текстовая часть пояснительной записки выполняется шрифтом Times New Roman с высотой букв 14 пунктов через интервал 1,25.

UML-диаграммы могут печататься на листах большего формата. Высота букв на диаграммах должна быть не меньше 3 мм.

2 Методические указания по выполнению курсового проекта

В разделе «Введение» курсовой работы студент обосновывает начало разработки ИС.

Основанием для начала разработки может выступать:

- создание (совершенствование) сайта;
- управление содержимым корпоративного сайта;
- автоматизация процесса учёта товаров в электронном магазине;
- информационная онлайн-система поддержки процесса покупок и продаж;
- автоматизация процесса анализа конкурентоспособности;
- документооборот в маркетинге и автоматизация учета клиентов;
- программная поддержка маркетинговой деятельности (анализа и оптимизации ассортимента, рекламной деятельности и оценки ее эффективности, установления цен, анализа продаж и т. п.);
- автоматизация рабочего места специалиста (маркетолога, менеджера по продажам и т. п.);
- информационная поддержка маркетинговой деятельности на основе веб-технологий;
- система управления маркетинговой деятельностью;
- управление взаимодействием с клиентами и разработка автоматизированной подсистемы технической поддержки и др.

2.1 Анализ бизнес-процессов пользователей

2.1.1 Разработка диаграммы вариантов использования.

Анализ и документирование бизнес-процессов проводится с помощью диаграмм вариантов использования.

Главное назначение диаграммы вариантов использования заключается в формализации функциональных требований к ИС и возможности согласования полученной модели с заказчиком на ранней стадии проектирования. Любой из вариантов использования может быть подвергнут дальнейшей декомпозиции на множество подвариантов использования отдельных элементов, которые образуют исходную сущность. В данной диаграмме пользователи выступают как акторы. Актор – это множество пользователей, выполняющих одинаковые бизнес-процессы. Рекомендуемое общее количество акторов в модели – не более 20, а вариантов использования – не более 50. В противном случае модель теряет свою наглядность и, возможно, заменяет собой одну из некоторых других диаграмм.

Семантика построения диаграммы вариантов использования должна определяться следующими особенностями рассмотренных выше элементов модели. Отдельный экземпляр варианта использования по своему содержанию является выполнением последовательности действий, которая инициализируется посредством экземпляра сообщения от экземпляра актора. В качестве отклика или ответной реакции на сообщение актора экземпляр варианта использования выполняет последовательность действий, установленную для данного варианта использования. Экземпляры акторов могут генерировать новые экземпляры сообщений для экземпляров вариантов использования.

Пример построения диаграммы вариантов использования представлен на рисунке 1.

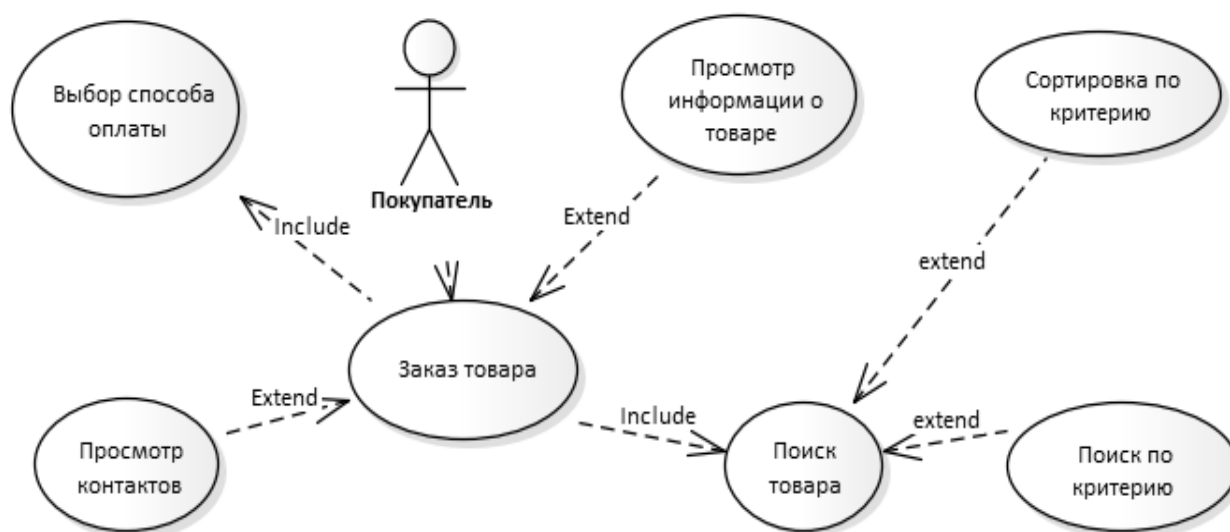


Рисунок 1 – Пример диаграммы вариантов использования

2.1.2 Бизнес-процессы акторов.

Диаграммы бизнес-процессов фиксируют функциональные требования заказчиков на этапе проектирования ИС. Для каждого варианта использования разрабатывается диаграмма бизнес-процесса. Каждый бизнес-процесс должен содержать набор полей с информацией. На диаграммах бизнес-процессов приводится структура информационных полей.

Примерный вид диаграмм бизнес-процессов представлен на рисунках 2 и 3.

Для каждой страницы сайта (мобильного приложения) в записке приводится таблица следующего вида (таблица 1).

Таблица 1 – Поля на странице «Имя страницы»

Поле на странице	Обозначение

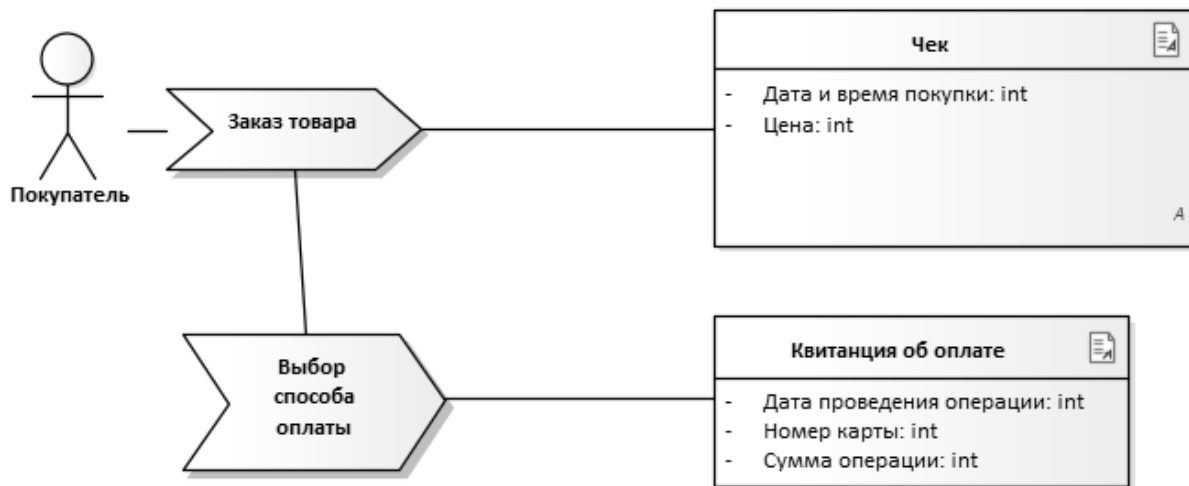


Рисунок 2 – Пример бизнес-процесса «Модель процесса «Заказ товара»



Рисунок 3 – Пример бизнес-процесса «Модель процесса «Поиск товара»

2.2 Моделирование взаимодействия акторов с ИС

Моделирование этого процесса производится с помощью диаграммы взаимодействия.

С помощью диаграммы последовательности описывают полный контекст взаимодействий между объектами для выполнения одного варианта использования.

На диаграмме последовательности изображаются объекты, которые непосредственно участвуют во взаимодействии. При этом никакие статические связи с другими объектами не визуализируются. Для диаграммы последовательности ключевым моментом является именно динамика взаимодействия объектов во времени. При этом диаграмма последовательности имеет как бы два измерения. Одно – слева направо в виде вертикальных линий, каждая из которых изображает линию жизни отдельного объекта, участвующего во взаимодействии. При этом каждый объект графически изображается в форме прямоугольника и располагается в верхней части своей линии жизни. Внутри прямоугольника записываются собственное имя объекта со строчной буквы и имя класса, разделенные двоеточием.

Крайним слева на диаграмме изображается объект, который является инициатором взаимодействия. Правее изображается другой объект, который непосредственно взаимодействует с первым. Таким образом, порядок расположения объектов на диаграмме последовательности определяется исключительно соображениями удобства визуализации их взаимодействия друг с другом.

Второе измерение диаграммы последовательности – вертикальная временная ось, направленная сверху вниз. Начальному моменту соответствует самая верхняя часть диаграммы. При этом процесс взаимодействия объектов реализуется посредством сообщений, которые посылаются одними объектами другим. Сообщения изображаются в виде горизонтальных стрелок с именем сообщения и образуют некоторый порядок относительно времени своей инициализации; сообщения, расположенные на диаграмме последовательности выше, передаются раньше тех, которые расположены ниже.

Линия жизни объекта (object lifeline) изображается пунктирной вертикальной линией. Линия жизни служит для обозначения периода, в течение которого объект существует в системе и, следовательно, может потенциально участвовать во всех ее взаимодействиях. Если объект существует в системе постоянно, то его линия жизни должна продолжаться по всей плоскости диаграммы последовательности от самой верхней ее части до самой нижней. С другой стороны, отдельные объекты, выполнив свою роль в системе, могут быть уничтожены (разрушены), чтобы освободить занимаемые ими ресурсы. Для таких объектов линия жизни обрывается в момент его уничтожения. Для обозначения момента уничтожения объекта в языке UML используется специальный символ в форме латинской буквы «X». Ниже этого

символа пунктирная линия не изображается, поскольку соответствующего объекта в системе уже нет.

В процессе функционирования объектно-ориентированных систем одни объекты могут находиться в активном состоянии, непосредственно выполняя определенные действия, или в состоянии пассивного ожидания сообщений от других объектов. Чтобы явно выделить подобную активность объектов, на диаграммах последовательности применяется специальное понятие, получившее название «фокус управления» (focus of control). Фокус управления изображается в форме вытянутого узкого прямоугольника, верхняя сторона которого обозначает начало получения фокуса управления объекта (начало активности), а ее нижняя сторона – окончание фокуса управления (окончание активности).

С другой стороны, периоды активности объекта могут чередоваться с периодами его пассивности или ожидания. В этом случае у такого объекта фокусы управления изменяют свое изображение на линию жизни и наоборот.

В отдельных случаях инициатором взаимодействия в системе может быть актор или внешний пользователь. В этом случае актор изображается на диаграмме последовательности самым первым объектом слева со своим фокусом управления. Наиболее часто актор и его фокус управления будут существовать в системе постоянно, отмечая характерную для пользователя активность в инициировании взаимодействий с системой. При этом сам актор может иметь собственное имя либо оставаться анонимным.

На диаграммах последовательности могут присутствовать три разновидности сообщений. Первая разновидность сообщения является наиболее распространенной и используется для вызова процедур, выполнения операций или обозначения отдельных вложенных потоков управления. Начало этой стрелки, как правило, соприкасается с фокусом управления того объекта-клиента, который иницирует это сообщение. Конец стрелки соприкасается с линией жизни того объекта, который принимает это сообщение и выполняет в ответ определенные действия.

При этом принимающий объект получает фокус управления, становясь в этом случае активным. Передающий объект может потерять фокус управления или остаться активным. Вторая разновидность сообщения используется для обозначения простого асинхронного сообщения, которое передается в произвольный момент.

Передача такого сообщения обычно не сопровождается получением фокуса управления объектом-получателем. Третья разновидность сообщения используется для возврата из вызова процедуры. Примером может служить простое сообщение о завершении некоторых вычислений без предоставления результата расчетов объекту-клиенту. В процедурных потоках управления эта стрелка может быть опущена, поскольку ее наличие неявно предполагается в конце активизации объекта. В то же время считается, что каждый вызов процедуры имеет свою пару – возврат вызова.

Таким образом, каждое сообщение ассоциируется с некоторым действием, которое должно быть выполнено принявшим его объектом. При этом действие может иметь некоторые аргументы или параметры, в зависимости от конкретных значений которых может быть получен различный результат. Соответствующие параметры будет иметь и вызывающее это действие сообщение. Более того, значения параметров отдельных сообщений могут содержать условные выражения, образуя ветвление или альтернативные пути основного потока управления.

Одной из особенностей диаграммы последовательности является возможность визуализировать простое ветвление процесса. Для изображения ветвления изображаются две или более стрелки, выходящие из одной точки фокуса управления объекта. При этом рядом с каждой из стрелок должно быть явно указано соответствующее условие ветви в форме булевского выражения. Хотя количество ветвей может быть произвольным, следует помнить, что наличие ветвлений может существенно усложнить интерпретацию диаграммы последовательности. Напомним, что предложение-условие может быть записано в форме обычного текста, псевдокода или выражения на языке программирования и должно представлять собой некоторое булевское выражение. Запись этих условий должна исключать одновременную передачу альтернативных сообщений по двум и более ветвям. В этом случае принято говорить об отсутствии конфликта для условий ветвления.

Сообщения могут иметь собственное имя, в качестве которого выступает имя операции, вызов которой инициируют эти сообщения у принимающего объекта. В этом случае рядом со стрелкой записывается имя операции с круглыми скобками, в которых могут указываться параметры или аргументы соответствующей операции. Если параметры отсутствуют, то скобки все равно должны быть изображены после имени операции.

Построение диаграммы последовательности целесообразно начинать с выделения из всей совокупности тех классов, объекты которых участвуют в моделируемом взаимодействии. После этого все объекты наносятся на диаграмму с соблюдением некоторого порядка инициализации сообщений. Здесь необходимо установить, какие объекты будут существовать постоянно, а какие временно – только на период выполнения требуемых действий.

Когда объекты визуализированы, можно приступать к спецификации сообщений. При этом необходимо учитывать те роли, которые играют сообщения в системе. При необходимости уточнения этих ролей следует использовать их разновидности и стереотипы. Для уничтожения объектов, которые создаются на время выполнения своих действий, нужно предусмотреть явное сообщение.

Наиболее простые случаи ветвления процесса взаимодействия можно изобразить на одной диаграмме с использованием соответствующих графических примитивов. Однако следует помнить, что каждый альтернативный поток управления может существенно затруднить понимание построенной модели. Поэтому общим правилом является визуализация каждого

потока управления на отдельной диаграмме последовательности. В этой ситуации такие отдельные диаграммы должны рассматриваться совместно как одна модель взаимодействия.

Примерный вид диаграммы взаимодействия представлен на рисунке 4.

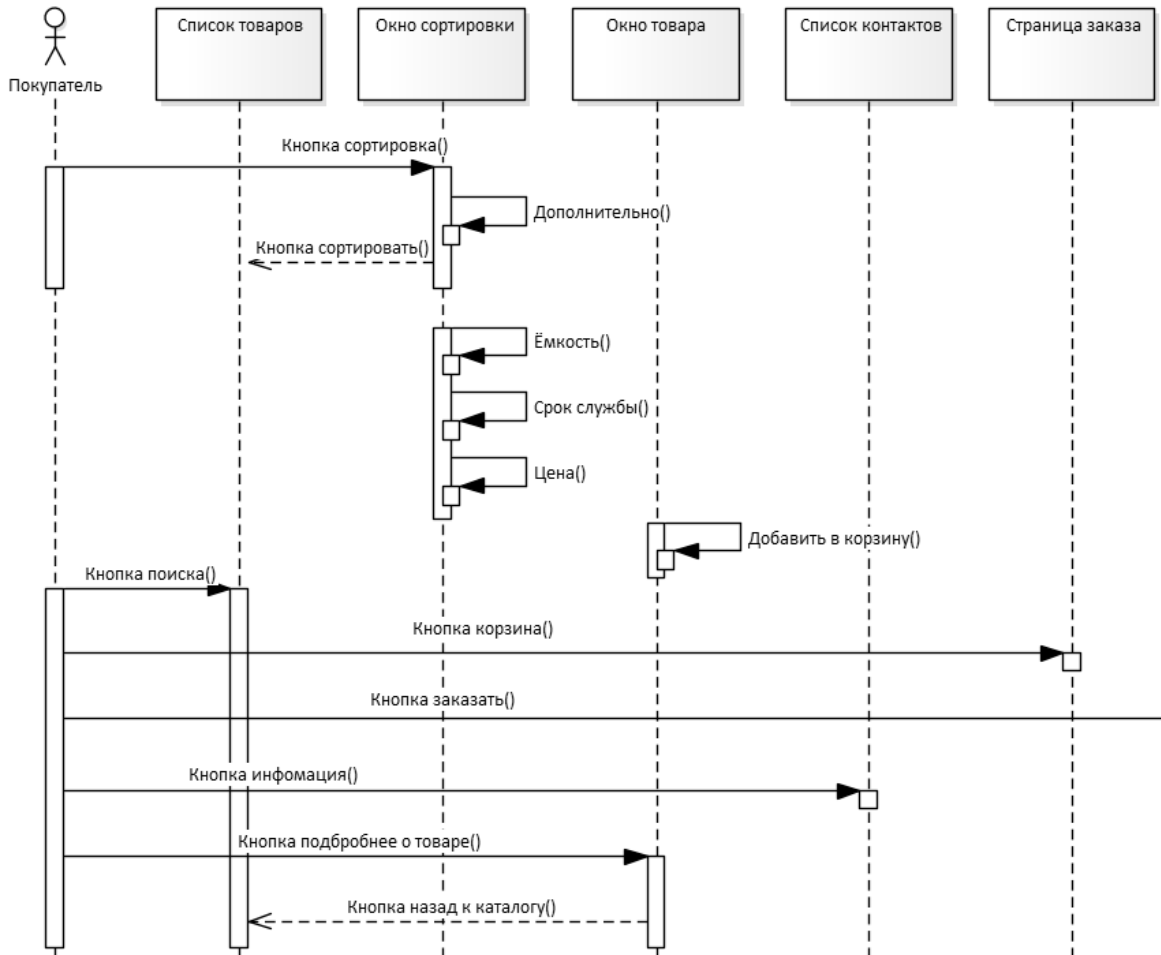


Рисунок 4 – Пример диаграммы последовательности

2.3 Разработка дизайна программного продукта

2.3.1 Разработка прототипа сайта.

Различают несколько стадий прорисовки прототипа сайта.

Wireflow – это последовательность набросков экранов. Эта схема включает грубые наброски экранов с важными элементами интерфейса (кнопки, поля ввода, места для картинок) и действия. Все упрощено, но главные элементы интерфейса на каждом экране чётко выделены.

Screenflow – последовательность экранов с готовой UI-частью. Этот вид используется на более поздних этапах разработки. Здесь дизайн каждого экрана выполнен с максимальной (финальной) детализацией. Выполняется только в графических редакторах и специализированных приложениях.

Wireframe – это статический набросок конкретного экрана, который определяет расположение и размер UI-элементов. Визуальная часть всегда ограничивается черно-белой палитрой и простыми фигурами.

По степени детализации Wireframe можно разделить на два вида.

1 Low-fidelity wireframes – очень грубые и быстро выполненные наброски экранов. Обычно содержат минимум текста, в большинстве случаев он заменен горизонтальными линиями, все кнопки и изображения показаны прямоугольниками (иногда используются круги). Часто расположение и размер элементов приблизительны.

2 High-fidelity wireframes – точно выполненные наброски экранов. Содержат весь текст; расположение, размеры кнопок и остальных элементов максимально приближаются к финальному дизайну. Требуют намного больше времени на отрисовку, содержат больше деталей. Пример прототипа на данной стадии представлен на рисунке 5.

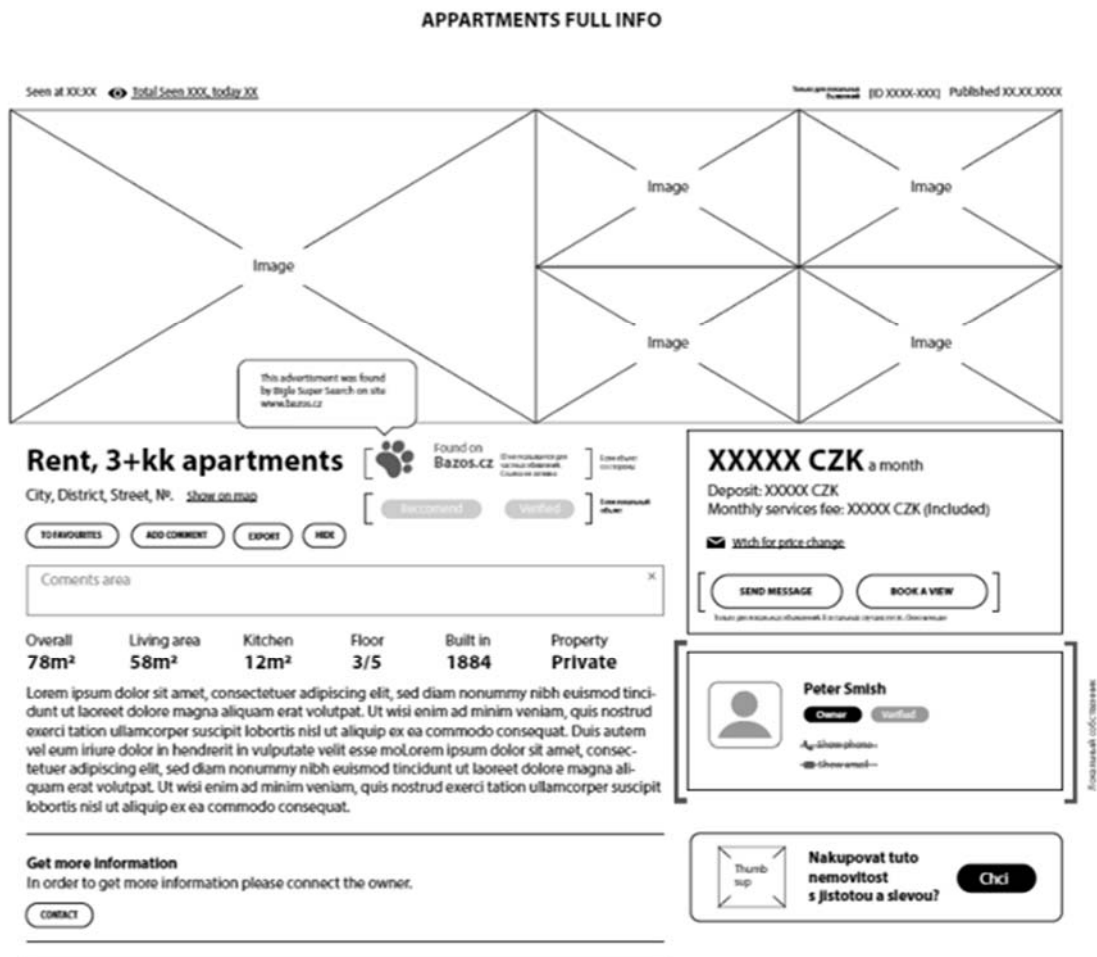


Рисунок 5 – Пример прототипа на стадии High-fidelity wireframes

Prototype или интерактивный прототип – это дизайн экрана, который реагирует на действия пользователей, т. е. это не статическая картинка, как в случае с wirefram или userflow. Прототипы позволяют понять, как будет работать веб-сайт до начала программирования, выявить проблемные

и неудобные экраны. Часто используются для тестирования на реальных пользователях.

Интерактивные прототипы запускаются в браузере на компьютере или в специальных приложениях на смартфоне и могут выглядеть и вести себя, как настоящие законченные продукты. Но на самом деле это лишь подделки, никаких серьезных вычислений или обработки данных там выполняться не может.

Интерактивный прототип на основе предыдущих набросков может быть использован для дальнейшего тестирования. Для одной функции иногда может быть разработано несколько прототипов и уже на основе отчетов Usability-тестирования принимается решение, какой вариант утверждать. Пример интерактивного прототипа представлен на рисунке 6.

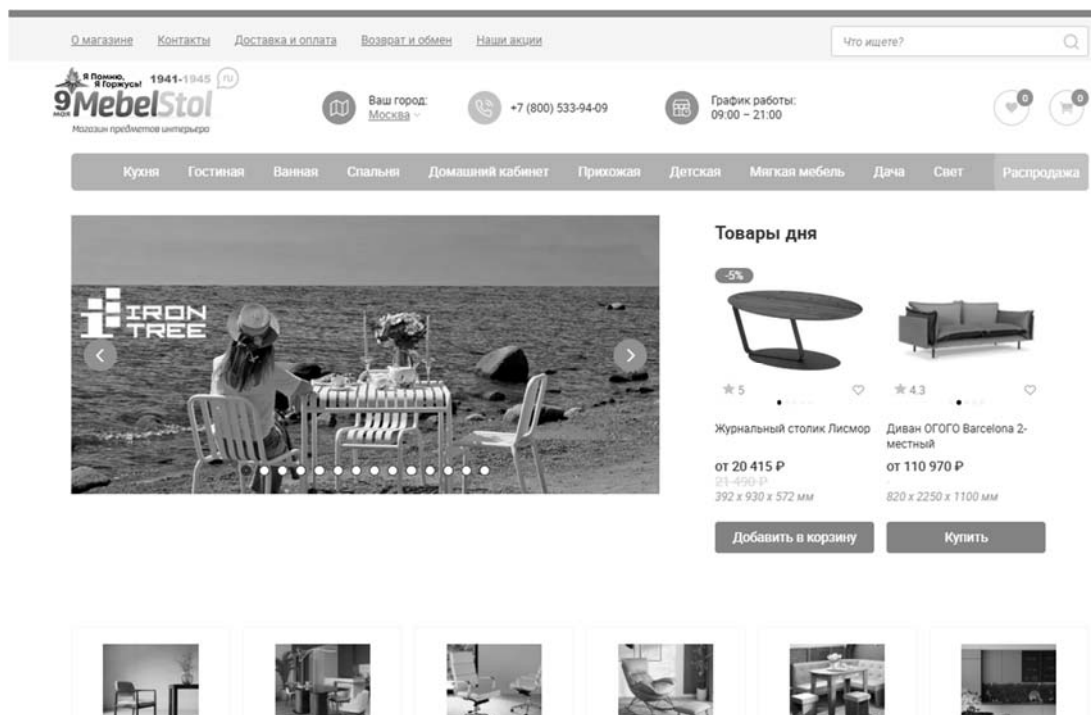


Рисунок 6 – Пример интерактивного прототипа

2.4 Отчёт о разработке программных компонентов и тестировании

2.4.1 Отчёт о разработке программного обеспечения.

В разделе представлены копии экранов разработанных страниц ИС. В приложении к курсовой работе нужно представить фрагменты кода для формирования страниц ИС.

2.4.2 Отчет о тестировании.

В разделе приводятся результаты Usability-тестирования разработанной ИС.

Usability-тестирование или проверка эргономичности – это исследование, чтобы проверить, насколько удобно использовать интерфейс для выполнения конкретной задачи. Такие тесты выполняются на реальных людях. В большинстве случаев UX-дизайнер вручает устройство с запущенной ИС потенциальным пользователям, объясняет, что нужно сделать и наблюдает за их работой. В результате тестирования фиксируются время выполнения бизнес-процессов и количество совершенных ошибок.

3 Заключение по проекту

В заключении по проекту описываются применённые в курсовом проекте информационные технологии и степень реализации ИС. Делается вывод, что разработанная ИС соответствует разработанному прототипу и способна осуществлять бизнес-процессы пользователей.

Список литературы

- 1 **Полуэктова, Н. Р.** Разработка веб-приложений: учебник / Н. Р. Полуэктова. – 2-е изд. – М. : Юрайт, 2024. – 204 с.
- 2 **Гвоздева, В. А.** Базовые и прикладные информационные технологии: учебник / В. А. Гвоздева. – М. : Форум, 2023. – 386 с.
- 3 **Новиков, В. А.** Web-программирование: учеб. пособие / В. А. Новиков. – Мн. : Адукацыя і выхаванне, 2024. – 351 с.: ил.
- 4 **Немцова, Т. И.** Компьютерная графика и web-дизайн: учеб. пособие / Т. И. Немцова, Т. В. Казанкова, А. В. Шнякин. – М. : Форум, 2023. – 400 с.

Приложение А (рекомендуемое) Типовой бланк задания

Межгосударственное образовательное учреждение высшего образования
«Белорусско-Российский университет»

Факультет электротехнический
УТВЕРЖДАЮ
Зав. кафедрой ПОИТ Кутузов В. В.

4 сентября 20__ г.

ЗАДАНИЕ по курсовому проектированию

Студенту гр. _____

1. Тема проекта «Разработка ИС для маркетинговой деятельности» (по вариантам).

2. Срок сдачи студентом законченного проекта: согласно графику учебного процесса.

3. Исходные данные к проекту:

- материалы производственной практики;
- методические указания по КП, материалы курса.

4. Содержание расчетно-пояснительной записки:

Введение.

1 Анализ бизнес-процессов пользователей.

1.1 Обоснование начала разработки.

1.2 Функциональные требования к АСОИ.

1.3 Прочие требования к АСОИ.

2 Моделирование бизнес-процессов пользователей.

2.1 Диаграммы вариантов использования.

2.2 Диаграммы последовательности.

3 Программная реализация ИС.

3.1 Отчёт о разработке программных компонентов.

3.2 Отчёт о тестировании.

Заключение по проекту.

Список литературы.

5. Дата выдачи задания: согласно графику учебного процесса.

6. Календарный график работы над проектом: согласно графику учебного процесса.

Руководитель проекта _____
(подпись, ФИО)

Задание принял к исполнению _____ 5.09.20__
(подпись, ФИО)