Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Лабораторная работа № 2.

«Построение функциональной модели IDEF0»

Студент: Пшенко А. Ф.

ФИТ 4 курс 4 группа

Преподаватель: Якубенко К. Д.

Минск 2024

# **Постановка задачи**

# **Тема и цель работы**

Темой данной лабораторной работы является построение функциональной модели IDEF0, необходимой для графического представления бизнес-процессов и функциональных требований системы, представленной в предыдущей лабораторной работе — веб-приложения «ElectroShop» для магазина электроники. Это приложение позволяет пользователям удобно искать, легко приобретать различные товары, такие как смартфоны, ноутбуки и аксессуары, не выходя из дома, а администраторам — управлять данными о товарах и управлять заказами.

Целью лабораторной работы является изучение основ методологии структурного моделирования IDEF, ознакомление с функциональным моделированием на основе методологии IDEF0, а также получение навыков по применению IDEF0 для построения функциональных моделей на основе требований к информационной системе.

Методология IDEF0 представляет собой мощный инструмент для моделирования и анализа сложных систем, таких как веб-приложение «ElectroShop», предоставляя наглядное и структурированное представление функциональных процессов. Построение функциональной модели IDEF0 поможет разработчикам и администраторам не только увидеть общую картину системы, но и выявить возможные узкие места или точки для оптимизации. Кроме того, модель поможет согласовать требования различных заинтересованных сторон — от разработчиков до конечных пользователей и администраторов — обеспечивая единое понимание структуры и функционирования системы.

Методология IDEF0 применяется для описания и анализа бизнес-процессов, предоставляя возможность разбиения системы на функциональные блоки с указанием входных данных, управляющих воздействий, механизмов реализации и выходов. В случае веб-приложения «ElectroShop», построение функциональной модели IDEF0 позволит подробно изучить взаимодействие всех компонентов системы, их связи и зависимости. Это особенно важно для понимания того, как данные перемещаются между разными частями системы, какие действия выполняются, и как они влияют на итоговый результат. В случае с «ElectroShop» это позволит детализировать такие процессы, как управление пользователями, покупка товаров и обработка платежей.

# **Описание функциональных требований**

Функциональные требования к системе веб-приложения «ElectroShop» можно разделить на требования для различных ролей пользователей – клиента, гостя, администратора и пользователя.

Функционально web-приложение должно:

* поддерживать роли гостя, клиента, администратора;

Обеспечивать всем пользователям возможности:

* просмотра каталога товаров;
* просмотра страницы товара;
* фильтрации товаров;
* поиска по товарам;

Обеспечивать гостям возможности:

* зарегистрироваться, аутентифицироваться, авторизоваться;

Обеспечивать клиентам возможности:

* добавления товаров в корзину;
* удаления товаров из корзины;
* покупки товаров;

Обеспечивать администраторам возможности:

* создания и редактирования товаров.

Основные страницы веб-приложения:

* страница регистрации;
* страница авторизации;
* главная страница каталога товаров с фильтрацией и поиском;
* страница карточки товара;
* страница просмотра корзины;
* страница оформления заказа;
* страница создания товара;
* страница редактирования товара;
* страница ошибки.

Основные страницы веб-приложения обеспечивают полное покрытие всех функций, необходимых для эффективного взаимодействия с приложением. Каждая из этих страниц играет ключевую роль в создании удобного и эффективного интерфейса, обеспечивая пользователей и администраторов всем необходимым для успешного взаимодействия с приложением.

1. **Теоретическая часть**

Структурный подход – это методология анализа, проектирования и разработки сложных систем, основанная на декомпозиции системы на составляющие части и выделении основных функций и взаимосвязей между ними. Основная сущность этого подхода заключается в том, что система рассматривается как набор взаимосвязанных элементов, каждый из которых выполняет определённые функции, что позволяет упростить понимание и управление сложностью системы.

DFD (Data Flow Diagram) – диаграмма потоков данных. Это графический инструмент, используемый для описания движения данных внутри системы. DFD моделирует потоки данных между процессами, хранилищами данных и внешними объектами.

IDEF (Integration DEFinition) – семейство методологий для моделирования и анализа систем. Каждая методология IDEF разработана для решения конкретных задач в процессе проектирования и анализа, например, IDEF0 для функционального моделирования, IDEF1 для информационного моделирования и т.д.

SADT (Structured Analysis and Design Technique) – методика структурного анализа и проектирования. Она используется для моделирования систем и процессов с помощью блок-схем, отражающих функции системы и их взаимодействия. Основой методологии является использование контекстных диаграмм, описывающих связи между процессами и данными.

Всего существует 15 IDEF-моделей: начиная IDEF0 и заканчивая IDEF14. Рассмотрим основные:

1. IDEF0 – модели функциональной деятельности. Описывает, как функции системы взаимодействуют между собой и с внешней средой.
2. IDEF1 – информационные модели, используемые для описания структуры данных и их взаимосвязей.
3. IDEF1X – модели для проектирования реляционных баз данных, фокусируются на логической структуре данных.
4. IDEF3 – модели, описывающие процессы и последовательность их выполнения (поток процессов).

Базовые принципы моделирования в IDEF0:

1. Декомпозиция – разбивка системы на иерархию функций, каждая из которых описывается отдельно.
2. Модулирование – каждая функция представлена в виде модуля, с определёнными входами, выходами, управляющими воздействиями и механизмами выполнения.
3. Контекстность – каждая диаграмма должна быть связана с контекстом системы, что обеспечивает понимание места функции в общей системе.
4. Связи – связи между функциями и внешними элементами описываются через потоки данных, ресурсов и управлений.

Модель "как есть" (As-Is) используется для описания текущего состояния системы. Применяется, когда нужно зафиксировать существующую ситуацию, например, для анализа или оптимизации текущих процессов.

Модель "как будет" (To-Be) строится для представления будущего состояния системы после изменений или внедрения новой системы. Этот подход целесообразен, когда требуется показать, как будет работать система после улучшений, реорганизации или автоматизации.

Таким образом, применение моделей "как есть" и "как будет" зависит от цели анализа: если необходимо понять и зафиксировать текущее состояние, используется модель "как есть", а если нужно представить будущие изменения, то строится модель "как будет".

# **Описание программных средств**

Для построения моделей было использовано программное средство Draw.io (также известное как diagrams.net). Draw.io — это многофункциональный инструмент, предназначенный для создания разнообразных графических схем, таких как диаграммы классов, диаграммы баз данных, блок-схемы, диаграммы деятельности, диаграммы процессов и многие другие.

Разработчиком Draw.io является компания JGraph Ltd. На момент использования проекта была задействована актуальная веб-версия программного обеспечения. Для доступа и использования данного инструмента можно посетить официальный сайт: <https://app.diagrams.net>.

Draw.io поддерживает интеграцию с облачными сервисами, такими как Google Drive, OneDrive и GitHub, что позволяет легко сохранять и управлять проектами. Кроме того, программное обеспечение поддерживает работу с локальными файлами и экспорт моделей в популярные форматы (PNG, PDF, SVG, XML), что делает его универсальным и удобным в использовании.

Инструмент поддерживает интеграцию с такими популярными сервисами, как Google Workspace, Microsoft Office 365, Confluence и Jira. Это позволяет пользователям бесшовно внедрять диаграммы и схемы в документы.   
Draw.io доступен в двух режимах:

1. Веб-версия — основной способ использования программного средства, работающий через любой современный браузер. Поддерживается на всех платформах, включая Windows, macOS, Linux, а также мобильные устройства под управлением Android и iOS.
2. Десктопная версия — доступна для скачивания и установки на операционные системы Windows, macOS и Linux. Обе версии имеют идентичный функционал. Draw.io предоставляет все необходимые средства для работы с различными моделями и диаграммами, применяемыми в инженерии программного обеспечения, таких как:

* UML диаграммы классов, последовательностей, прецедентов;
* ERD диаграммы (сущность-связь) для моделирования баз данных;
* BPMN диаграммы бизнес-процессов;
* диаграммы архитектуры программных систем и сетевых решений;
* блок-схемы для описания алгоритмов и процессов;
* диаграммы потоков данных и управления;
* схемы организации структуры данных.

Использование данного программного средства значительно упростило процесс проектирования и предоставило возможность создания наглядных схем, что способствует лучшему пониманию логики работы системы всеми заинтересованными сторонами проекта.

# **Описание практического задания**

Для построения функциональной модели IDEF0 информационной системы интернет-магазина электроники «ElectroShop» необходимо выделить основную бизнес-функцию и создать два уровня диаграмм: контекстную диаграмму и диаграмму первого уровня декомпозиции.

Основной бизнес-функцией сервиса интернет-магазина электроники является предоставление возможности покупки товара. Эта функция будет находиться на самом верхнем, абстрактном уровне (A0) модели IDEF0 и представлять собой процесс организации покупки товара. На данном уровне мы рассмотрим общие элементы, такие как входы, выходы, механизмы и управление, связанные с основным процессом.

Контекстная диаграмма (A0) для магазина электроники «ElectroShop» описывает процесс «Предоставлять возможность покупки товара пользователю», представлена на рисунке 3.1.

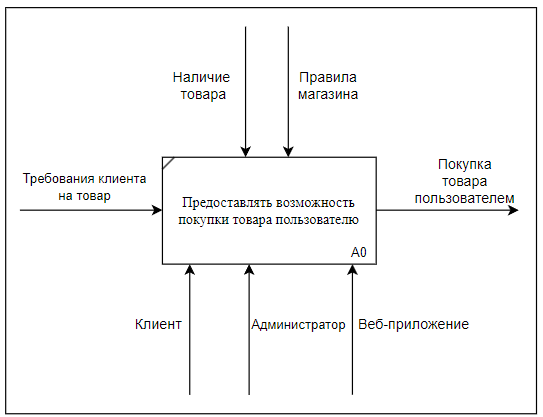


Рисунок 3.1 – Контекстная диаграмма

Входами для этой функции являются данные, которые поступают от клиента: требования на товар. На выходе – покупка товара. Механизмами, которые поддерживают выполнение функции, являются сам клиент, администратор и веб-приложение, через которое происходит взаимодействие. Управляющими элементами для данной бизнес-функции служат правила магазина, наличие товара.

Для большей детализации основной бизнес-функции построим диаграмму первого уровня декомпозиции, представленную на рисунке 3.2. На данной диаграмме более подробно раскрывается основная бизнес-функция, которая состоит из трех этапов: выбор товара, покупка и уведомление пользователя о покупке.

Клиент является механизмом функциональных блоков А0.1 и А0.2, а веб-приложение – механизмом блоков А0.1 и А0.2.

Наличие товара является управлением блоков А0.1 и А0.2, а правила магазина – блоков А0.2 и А0.3.

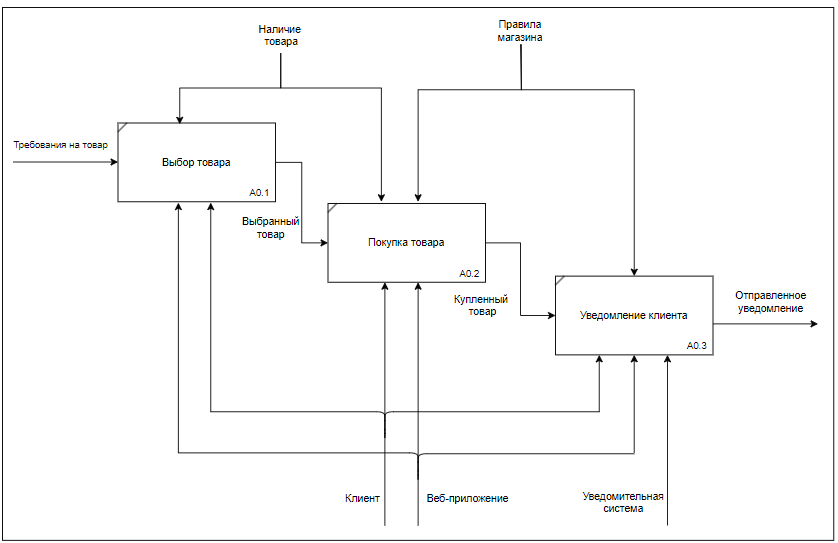


Рисунок 3.2 – Диаграмма первого уровня декомпозиции

В блоке «Выбор товара» (A0.1) входом служат требования на товар, а механизмами являются веб-приложение и клиент. Управляющими факторами для этого блока выступает наличие товара. На выходе получается выбранный товар.

Следующий блок «Покупка товара» (A0.2) предполагает, что клиент выбирает товар, используя веб-приложение, путем добавления его в корзину. Входом для этого блока является выбранный товар, а выходом — купленный товар. Механизмами выступают клиент, веб-приложение. Управление здесь обеспечивается правилами магазина.

После подтверждения покупки в блоке «Покупка товара» (A0.2), процесс переходит к следующему блоку «Уведомление клиента» (A0.3). Этот блок отвечает за информирование клиента о покупке товара. Входом для данного блока является купленный товар, а на выходе — отправленное уведомление. Механизмами здесь выступают веб-приложение и уведомительные системы (e-mail, SMS или push-уведомления). Управляющими факторами выступают настройки уведомлений, заданные в системе, такие как частота и способ уведомлений.

# **Заключение**

В ходе выполнения данной лабораторной работы была достигнута основная цель – изучение методологии IDEF0 для построения функциональных моделей. В процессе работы было создано графическое представление бизнес-процессов и функциональных требований системы веб-приложения интернет-магазина «ElectroShop». Это позволило глубже понять структуру системы и её ключевые компоненты, а также проанализировать взаимодействие между различными элементами приложения.

Использование методологии IDEF0 продемонстрировало свою высокую эффективность для моделирования сложных систем, таких как веб-приложения для покупки электроники. Разделение системы на функциональные блоки с указанием входов, выходов, управляющих воздействий и механизмов реализации позволяет разработчикам и аналитикам получить чёткое представление о том, как работают отдельные процессы, и как они взаимодействуют друг с другом. Такой подход помогает выявить возможные узкие места, потенциальные риски и зависимости, которые могут повлиять на эффективность работы системы.

Построенная модель включает два уровня: контекстную диаграмму и декомпозицию основной функции покупки товара. Контекстная диаграмма предоставляет общее представление о работе системы в целом, включая ключевые входы, выходы и механизмы.

Лабораторная работа также позволила закрепить навыки работы с инструментами моделирования, такими как Draw.io, и углубить понимание структурного моделирования. Работа с моделями, основанными на IDEF0, является важным этапом в разработке и внедрении сложных информационных систем. Эти модели помогают не только разработчикам, но и другим заинтересованным сторонам — бизнес-аналитикам, менеджерам и владельцам бизнеса — получить ясное представление о том, как работает система и как её можно усовершенствовать.

Таким образом, выполнение данной лабораторной работы подтвердило важность применения методологии IDEF0 для структурного анализа и проектирования сложных систем. Полученные результаты и созданные модели могут быть использованы для дальнейшего совершенствования и масштабирования системы «ElectroShop», а также для внедрения новых функций и улучшений.