

28. ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО СРЕДСТВА УПРАВЛЕНИЯ ОТДЕЛЕНИЯМИ БАНКА С МОДУЛЕМ АНАЛИЗА ГЕОГРАФИИ ИСПОЛЬЗОВАНИЯ БАНКОВСКИХ УСЛУГ

Юдашкин В. О., студент группы 172303

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Мозоль А. А. – канд. экон. наук, доцент каф. ЭИ

Аннотация. В статье рассматривается процесс проектирования программного средства управления отделениями банка с модулем анализа географии использования банковских услуг. Актуальность проектирования и разработки автоматизированной системы заключается в том, что наличие такой системы закрывает вопрос о необходимости повышения эффективности за счёт минимизации временных затрат на управление отделениями коммерческого банка, регистрации и учёта его клиентов.

Ключевые слова. Проектирование, программное средство, разработка требований, моделирование, IDEF0, BPMN.

В любой стране мира банковская система является неотъемлемой частью её национальной экономики вне зависимости от того, каким образом государство ведёт экономическую политику. Именно на банковской системе лежит ответственность за осуществление денежно-кредитной политики, включающая в себя эмиссию денег, регулирование предложения денег, ставки рефинансирования и поведения национальной валюты государства.

Коммерческие банки в Республике Беларусь составляют второй уровень в двухуровневой банковской системе страны. Коммерческие банки, как и Нацбанк, могут устанавливать курсы валют, но основное их предназначение – это оказание услуг для физических и юридических лиц, таких как предоставление кредитов, открытие банковских счетов, эмиссия банковских карт, проведение расчётов и платежей, консультирование и др.

Для оптимизации деятельности коммерческих банков проектируется и разрабатывается программное обеспечение, которое позволяет автоматизировать основные бизнес-процессы банка. Поэтому целью данного проекта является достижение более эффективного управления отделениями банка, упрощение процесса учёта клиентов банка в виде физических и юридических лиц, а также автоматизация и повышение точности анализа данных о пользовании банковскими услугами.

Для достижения поставленной цели необходимо решить ряд задач. Необходимо выполнить моделирование двух основных процессов: регистрации физических и юридических лиц для дальнейшего взаимодействия с ними в нотации BPMN, а также анализа географии использования банковских услуг в нотации IDEF0. Необходима разработка функциональных требований к программному средству, а также обоснование выбора технологий для дальнейшей разработки системы. Объектом исследования является процесс управления отделениями банка, регистрации и учёта клиентов в виде физических и юридических лиц. Предметом исследования являются методы и средства автоматизации управления отделениями банка, регистрации и учёта клиентов.

В BPMN-модели составляется блок-схема, которая отображает поэтапное выполнение операций от начала до конца процесса. BPMN-схемы наглядно и подробно демонстрируют последовательность рабочих действий и перемещение информационных потоков, необходимых для выполнения процесса, а потому являются одним из ключевых инструментов управления бизнесом.

Основные компоненты BPMN включают действия, которые представляют собой задачи или подзадачи в процессе, события, указывающие на то, что происходит в процессе, и гейты, которые определяют точки развилки, где может происходить выбор пути. Потоки представляют последовательность действий и событий, включая последовательные потоки, сообщения и ассоциации. Пулы и дорожки позволяют разделять участников процесса на отдельные организации и роли или отделы.

На рисунке 1 представлена BPMN-модель процесса управления отделением банка по регистрации новых клиентов банка. Из описанного процесса очевидно, что регистрация физического и юридического лица выполняется по-разному. Физическое лицо располагает такими данными, как фамилия, имя, отчество и номер счёта в банке. Юридическое лицо располагает наименованием, формой собственности, расчётным счётом и юридическим адресом. Ввод банковских счетов

осуществляется согласно стандартам международной структуры номеров счетов (IBAN). Введённые корректно данные сохраняются в хранилище данных банка.

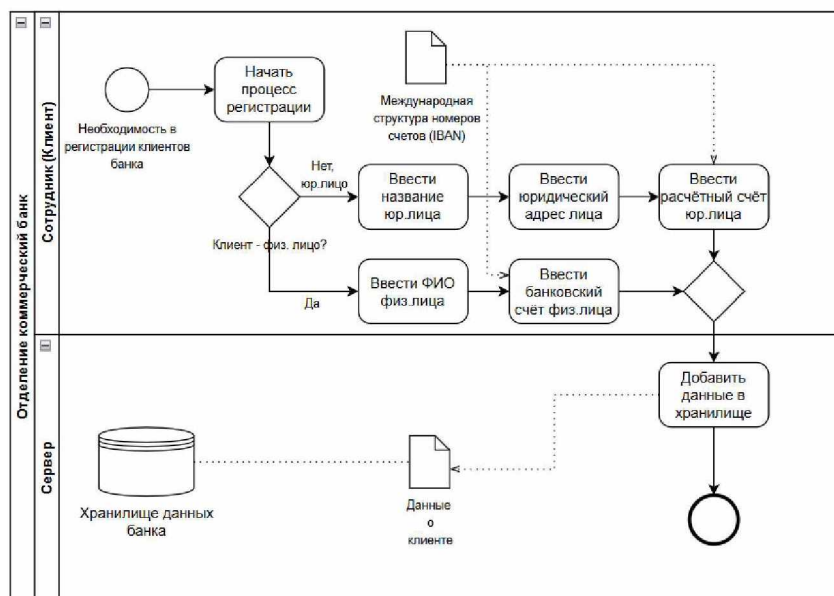


Рисунок 1 – Модель процесса регистрации клиентов в отделении коммерческого банка в нотации BPMN.

Функциональная модель процесса анализа географии использования банковских услуг очень доступно и понятно описывается графически на примере IDEF0-диаграммы. На диаграмме необходимо изобразить, как отдел управления данными, применяя банковское программное обеспечение (ПО) и алгоритмы машинного обучения, руководствуясь локальными нормативно-правовыми актами в области управления данными, документацией к системам хранения данных банка и документацией к библиотекам языка программирования Python, популярный для решения задач машинного обучения, прогнозирует данные об открытии новых отделений банка и установке новых банкоматов.

На рисунке 2 изображена диаграмма декомпозиции первого уровня основного процесса, описанного в контекстной диаграмме «Анализ географии использования банковских услуг». Каждый продемонстрированный на диаграмме декомпозиции подпроцесс подлежит дополнительной декомпозиции, которая описывает алгоритмы сбора и предобработки данных, процесс обучения на основе данных средствами языка программирования Python и алгоритм предсказания данных.

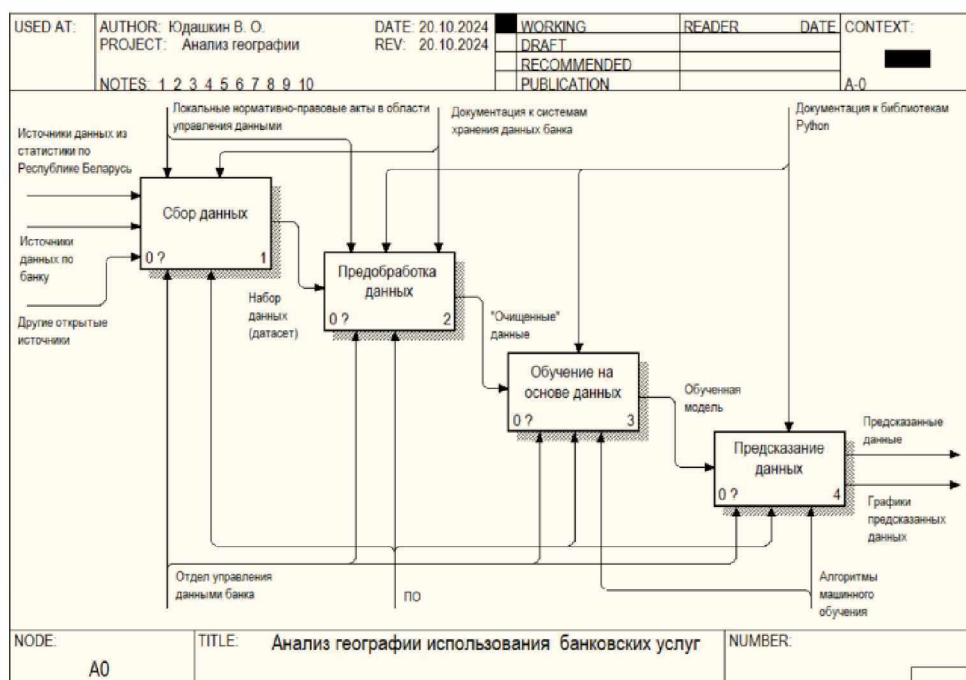


Рисунок 2 – Декомпозиция процесса «Анализ географии использования банковских услуг».

После того, как выполнено моделирование основных бизнес-процессов проектируемого программного средства, необходимо приступить к разработке функциональных требований к программному средству. Итак, для достижения цели создания современной, масштабируемой и интуитивно понятной системы, позволяющей эффективно управлять основными процессами коммерческого банка необходимо решить следующие задачи: разработать гибкую и расширяемую архитектуру, реализовать модули управления сотрудниками, клиентами, отделениями банка и банкоматами, внедрить модуль анализа географии использования банковских услуг, высокий уровень безопасности и надежности системы, разработать интуитивно понятный и отзывчивый пользовательский интерфейс, провести тестирование системы и подготовить всю необходимую техническую документацию.

Далее необходимо разработать функциональные требования к разрабатываемому программному продукту. Система управления отделениями банка должна обеспечивать следующие ключевые функциональные возможности: *работа с сотрудниками (для администраторов)*: добавление сотрудников, редактирование данных о сотрудниках, удаление сотрудников, поиск сотрудников; *работа с отделениями банка (для администраторов)*: добавление отделений банка, редактирование данных об отделениях банка, удаление отделений банка, поиск отделений банка; *работа с банкоматами (для администраторов)*: добавление банкоматов, редактирование данных о банкоматах, удаление банкоматов, поиск банкоматов; *работа с клиентами*: добавление клиентов, редактирование данных о клиентах, удаление клиентов, поиск клиентов; *аналитика и отчетность*: обучение модуля анализа географии использования банковских услуг на актуальных данных, визуализация срезов данных в виде графиков, сбор данных об использовании банковских услуг.

Для того чтобы получить права доступа к системе необходимо авторизоваться в системе. Авторизовавшись, пользователь получает один из двух типов пользователей: сотрудник банка или администратор. Все функциональные требования к программному средству можно изложить в виде диаграммы вариантов использования (use-case diagram) (рисунок 3). Это UML-диаграмма, на которой отображаются различные процессы (прецеденты), происходящие внутри бизнес-процесса, и взаимосвязи между акторами и данными процессами.

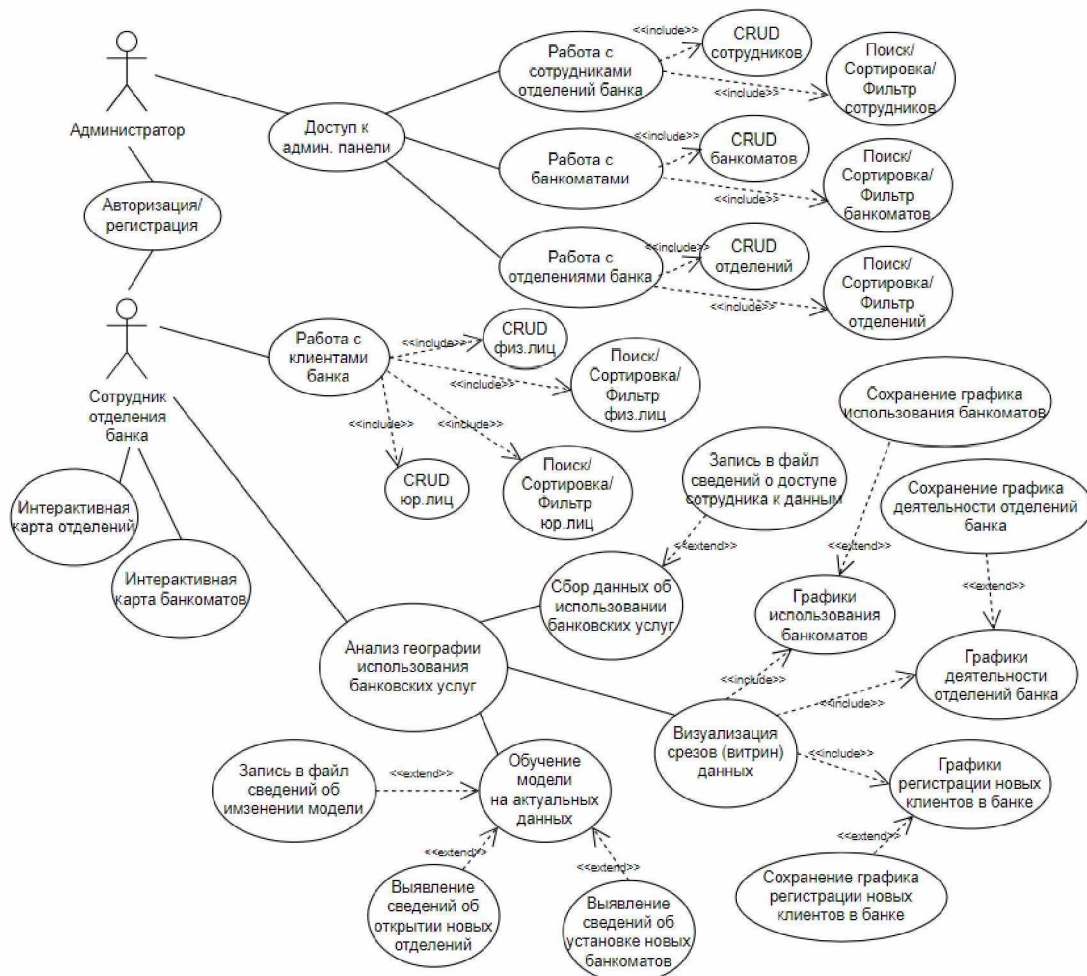


Рисунок 3 – Декомпозиция процесса «Анализ географии использования банковских услуг».

Очередным важным этапом проектирования и разработки программного средства является выбор компонентов и технологий для его реализации. Для начала необходимо определиться, какого типа будет разрабатываемое в рамках курсовой работы приложение. Всего существует три популярных типа приложений: веб-приложение, мобильное приложение и десктопное (настольное) приложение. Для разработки мобильного приложения для операционных систем Android и IOS требуется углубленное изучение языков программирования Kotlin и Swift, что представляет собой значительные временные затраты. К тому же, так как в первую очередь система разрабатывается для ноутбуков и персональных компьютеров, то вариант с мобильным приложением был отвергнут.

В таблице 1 прилагается сравнение Web- и Desktop-приложений.

Таблица 1 – Сравнительный анализ веб- и десктоп-приложений.

Критерий	Веб-приложение	Десктоп-приложение
Кроссплатформенность	Обычно работает на всех платформах без изменений (благодаря браузерам)	Часто необходимо создавать отдельные версии для каждой ОС
Обновления	Автоматически обновляется на сервере	Требует ручной установки обновлений, либо автоматизации
Производительность	Ограничена мощностью браузера и интернет-соединением	Обычно быстрее, так как использует ресурсы локального устройства напрямую
Доступность оффлайн	Как правило, требует постоянного интернет-соединения	Может работать полностью оффлайн
Уровень безопасности	Завязан на защите серверов и соединения (HTTPS, шифрование). Уязвимо для атак через браузер или сеть	Меньше уязвимостей при условии использования на защищенном устройстве
Интеграция с системой	Ограниченные возможности: доступ к файловой системе, устройствам (камера, микрофон) через браузерные API	Глубокая интеграция с системой: управление файлами, периферийными устройствами, ресурсами
Масштабируемость	Легко масштабируется, пользователи подключаются через браузер	Масштабируемость ограничена установкой и обновлением на каждом устройстве
Поддержка интерфейса	Ограничена браузером: интерфейс может быть менее гибким или отзывчивым	Большие возможности кастомизации интерфейса с учетом платформы

Исходя из проведенного сравнительного анализа, можно выделить основные плюсы и минусы для каждого типа приложения.

Основные преимущества веб-приложения заключается в том, что оно обновляется автоматически, а также легко масштабируется. Однако главным недостатком является обязательное интернет-соединение, а также низкая производительность в случае слабого интернет-соединения.

Главное преимущество десктопного приложения заключается в том, что для него не требуется доступ в Интернет, так как зачастую все данные находятся на локальной машине пользователя. Такие приложения предоставляют большие возможности кастомизации пользовательского интерфейса, что позволяет проектировать гибкий и отзывчивый интерфейс, в то время как в веб-приложении поддержка интерфейса ограничена браузером. Десктопные приложения легко интегрируются с системой организации, в то время как для веб-приложения необходимо для этого браузерное API.

В банковских организациях крайне важна высокая производительность систем, так как часто требуется работа с большими объемами данных. В таких организациях доступ к хранилищам данных осуществляется, как правило, по локальной компьютерной сети, то есть не требуется выход в Интернет. Таким образом было принято решение о проектировании и разработке десктопного приложения.

Далее необходимо обосновать выбор языка программирования для разработки десктопного приложения. Самые популярные языки программирования, C++, Python и Java, представляют мощные библиотеки для разработки настольных приложений и проектирования пользовательского интерфейса.

Прилагается сравнительный анализ библиотек вышеперечисленных языков программирования (таблица 2).

Таблица 2 – Сравнительный анализ библиотек языков программирования C++, Python, Java для разработки десктоп-приложений

Критерий	C++ (Qt)	Python (PyQt, PySide, Tkinter)	Java (JavaFX, Swing)
Кроссплатформенность	Высокая	Высокая	Высокая
Производительность	Очень высокая	Средняя	Высокая
Простота разработки	Сложная	Простая	Умеренная
UI-дизайн	Гибкий и мощный инструмент Qt Designer для визуального проектирования интерфейсов	PyQt/PySide имеют Qt Designer, Tkinter ограничен по дизайну	JavaFX поддерживает FXML для визуального проектирования интерфейсов, встроенные темы и стили
Возможности интеграции	Легкая интеграция с C++ библиотеками и системными API	Ограниченная, но возможна интеграция с C/C++ через ctypes или Cython	Легко интегрируется с серверной частью Java, базами данных и веб-сервисами

Исходя из сравнительного анализа, у C++ был выделен главный недостаток, несмотря на то, что приложения на данном языке, как правило, обладают высокой производительностью: язык C++ требует ручное управление памятью, что повышает временные затраты, связанные с разработкой. Язык Python уступает конкурентам из-за своей относительно низкой производительности, что явно не подойдёт для системы управления отделениями банка, которая может требовать работы с большими объёмами данных, а также ограниченной интеграцией.

В то же время выбор языка Java можно обосновать следующими фактами: библиотека JavaFX использует FXML для декларативного описания интерфейсов, что является альтернативой привычному стеку HTML/CSS; библиотека JavaFX поддерживает анимации, мультимедиа, графики и 3D-стили; десктопное приложение на языке Java легко интегрируется с серверной частью, особенно если она так же написана на Java; при разработке приложения на языке Java у программиста нет необходимости заботиться об управлении памятью, как в C++. Java располагает сборщиком мусора (Garbage Collection), который обеспечивает автоматическое управление памятью.

Таким образом, в процессе проектирования программного средства управления отделениями банка с модулем анализа географии использования банковских услуг было осуществлено моделирование в нотации BPMN процесса регистрации клиента банка, являющийся одним из основных процессов, который подлежит программной реализации. Была спроектирована функциональная модель в нотации IDEF0, описывающая работу модуля анализа географии потребления банковских услуг клиентами. Разработаны требования к автоматизированной системе и были обоснован выбор стека, на котором будет разрабатываться программное средство. Было принято решение разрабатывать настольное (десктопное) приложение, так как оно обеспечивает наиболее высокую производительность при работе с большим объёмом данных за счёт отсутствия необходимости доступа в Интернет. Спроектированное программное средство позволит решать задачу автоматизации и повышения эффективности процесса управления отделениями банка, сокращая время на управление как самими отделениями банка, так и банкоматами, принадлежащие данному банку, регистрацию и учёт клиентов банка. Desktop-приложение будет полезно для всех сотрудников банка, непосредственно участвующие в основной деятельности банка.

Список использованных источников:

1. Коммерческие банки Беларуси [Электронный ресурс]. – Режим доступа: <https://myfin.by/wiki/term/kommercheskie-banki-belarusi/>.
2. Счёт IBAN [Электронный ресурс]. – Режим доступа: <https://myfin.by/wiki/term/iban>
3. Бабич, А. В. UML: Первое знакомство / А. В. Бабич. – Москва. : БИНОМ. Лаборатория знаний, 2008. – 342 с.

4. Qt Framework [Электронный ресурс]. – Режим доступа: <https://www.qt.io/product/framework>
5. PyQt Documentation [Электронный ресурс]. – Режим доступа: <https://www.riverbankcomputing.com/static/Docs/PyQt5/>
6. Tkinter – Python interface to Tcl/Tk [Электронный ресурс]. – Режим доступа: <https://docs.python.org/3/library/tkinter.html>
7. Getting started with JavaFX [Электронный ресурс]. – Режим доступа: <https://openjfx.io/openjfx-docs/>

UDC 004.415.2 : 004.414.38

DESIGNING A SOFTWARE FOR BANK BRANCHES' MANAGEMENT WITH A MODULE FOR ANALYZING THE GEOGRAPHY OF BANK SERVICES' USING

Judashkin V. O.

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

Mozol A. A. – PhD in Economics, Associate Professor

Annotation. The process of designing a software for bank branches' management with a module for analyzing the geography of bank services' using is dealt in the article. The relevance of designing and development of an automated system lies in the fact that the availability of such system settles the issue about the necessity of improving the efficiency by minimizing time, which is spent on the commercial bank branches' management, registration and accounting the customers.

Keywords. Design, software, requirements development, modeling, IDEF0, BPMN.