

Министерство образования Республики Беларусь
Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

КАФЕДРА ИНФОРМАТИКИ

Отчет к проекту
«СЕРВИС ОНЛАЙН-КРЕДИТОВАНИЯ ДЛЯ ФИЗИЧЕСКИХ ЛИЦ»

Выполнил студент гр. 953505
Голубович Ю.И.

Проверил
Гриценко Н.Ю.

Минск, 2022

СОДЕРЖАНИЕ

1 ТЕХНИЧЕСКОЕ ЗАДАНИЕ	3
1.1 Концептуальная модель	3
1.2 Функциональная карта	3
1.3 Путь пользователя	3
1.4 Пользовательский интерфейс	4
1.5 Программные интерфейсы	4
1.6 Нефункциональные требования	5
2 ПРОЕКТИРОВАНИЕ И СОЗДАНИЕ БАЗЫ ДАННЫХ	6
2.1 Разработка логической модели базы данных	6
2.2 Разработка физической модели базы данных	6
2.3 Генерация DDL скриптов	8
2.4 Создание базы данных	8
3 ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ПО НА ЯЗЫКЕ UML	9
3.1 Разработка функциональной модели	9
3.2 Разработка модели данных	10
3.3 Проектирование приложения. Диаграммы	10
4 СОЗДАНИЕ ДИАГРАММ ИНФОРМАЦИОННЫХ ПОТОКОВ	15
4.1 Теоретические сведения	15
4.2 Диаграмма	17
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	18
ПРИЛОЖЕНИЕ А (обязательное) Схематичное представление интерфейса	19

1 ТЕХНИЧЕСКОЕ ЗАДАНИЕ

1.1 Концептуальная модель

Сервис для онлайн-кредитования физических лиц (далее Сервис), предназначенный для выбора подходящего кредита и оформления заявки на рассмотрение в онлайн-режиме без посещения банка. Реализация сервиса в виде сайта для совершеннолетних жителей РБ. В приложении определено две роли: пользователь (человек, желающий приобрести кредит) и администратор (работник банка, регулирующий ставки и виды доступных кредитов). В зависимости от введенных при входе данных будет отображаться пользовательский сайт или страница для администрирования.

1.2 Функциональная карта

На сайте должна быть доступна регистрация номеру телефона или по почте, заполнение заявки на кредит, возможность отредактировать личные данные и сменить пароль, список доступных кредитов, контактные данные банка-кредитора. Функциональная карта представлена на рисунке 1.1.

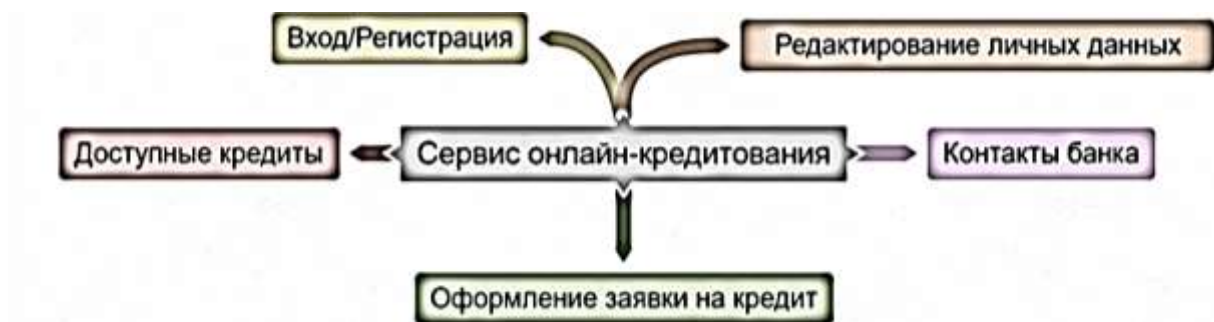


Рисунок 1.1 – Функциональная карта Сервиса

1.3 Путь пользователя

Пользователь регистрируется на сайте онлайн-кредитования или осуществляет вход (при наличии регистрации). На главной странице пользователь просматривает доступные виды кредитов, выбирает необходимый вариант и переходит к оформлению онлайн-заявки. При оформлении заявки пользователь указывает требуемые личные данные и данные о желаемом кредите (сумму, сроки выплаты и т.д.). После обработки заявки пользователь получает от системы сообщение об одобрении/не одобрении запрошенного кредита и ожидает звонка сотрудника банка. У пользователя есть возможность перейти на страницу с персональными данными и на страницу с контактной информацией банка. Схематическое представление пути пользователя представлено на рисунке 1.2.



Рисунок 1.2 – Схема пути пользователя Сервиса

1.4 Пользовательский интерфейс

Оформление сайта должно быть простым, неброским в светло синей цветовой гамме (приложение А). Все элементы должны иметь удобное для пользователя расположения и интуитивно понятные переходы.

1.5 Программные интерфейсы

Возможный вариант сущностей базы данных (ER-модели) представлен в виде диаграммы (см. рисунок 1.3).

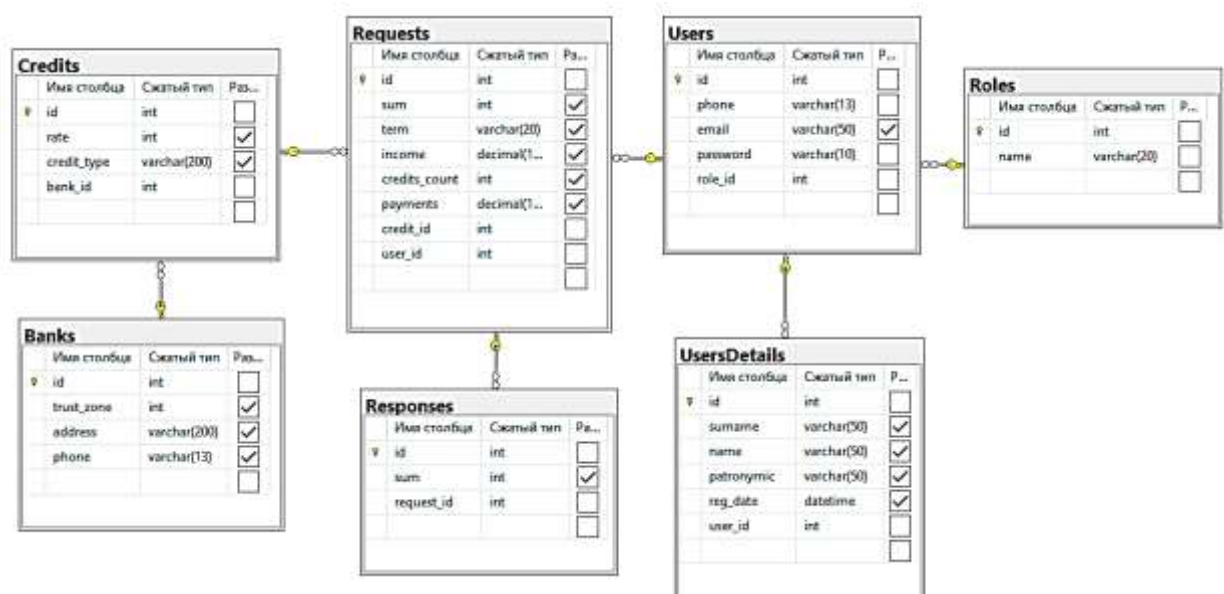


Рисунок 1.3 – ER-модель базы данных Сервиса

1.6 Нефункциональные требования

Интерфейс пользователя должен быть мультязычным, в начале будет два языка: русский и английский.

Сервис должен работать для следующих браузеров последних версий: MS Internet Explorer, Google Chrome. А также НЕ должен позволять НЕ Администраторам физический доступ к интерфейсу администратора. Сервис не должен позволять неавторизованным пользователям доступ к данным.

Исполнитель должен предоставить следующий комплект поставки при сдаче проекта:

- техническое задание;
- исходный код сервиса;
- исполняемые модули сервиса.

Ссылка на github проекта: <https://github.com/YULIYA2001/TOFI-TRPO>

2 ПРОЕКТИРОВАНИЕ И СОЗДАНИЕ БАЗЫ ДАННЫХ

2.1 Разработка логической модели базы данных

Для разработки логической модели базы данных (БД) на языке IDEF1X была использована программа для проектирования и документирования баз данных Erwin Data Modeler [1],[2].

Выделенные при изучении предметной области сущности:

- Пользователь (Users) – зарегистрированный клиент Сервиса, желающий оформить онлайн-заявку на кредит;
- Роль (Roles) – пользовательская роль (администратор или простой пользователь), ограничивающая доступ к данным;
- Детали пользователя (UserDetails) – личные данные пользователя, необходимые для подачи заявки;
- Заявка пользователя (Requests) – запрос об оформлении кредита;
- Ответ на заявку (Responses) – одобрение/отклонение заявки на получение кредита;
- Кредит (Credits) – сущность, описывающая вид кредита, ставку и т.д.
- Банк (Banks) – содержит информацию о банке, предоставляющем кредит.

Логическая модель БД на языке IDEF1X представлена на рисунке 2.1.

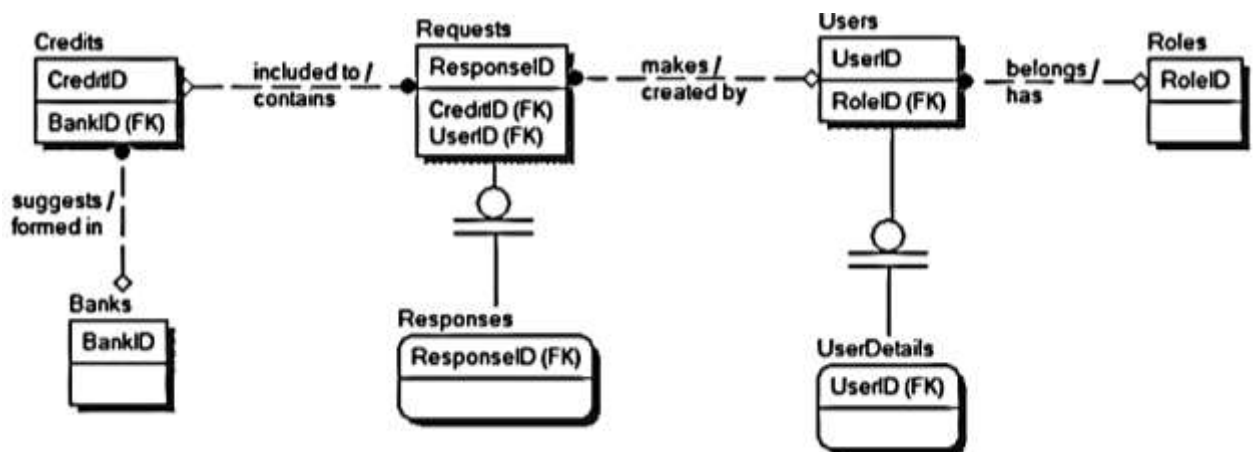


Рисунок 2.1 – Логическая модель БД

2.2 Разработка физической модели базы данных

Для разработки физической модели базы данных (БД) на языке IDEF1X была использована программа для проектирования и документирования баз данных Erwin Data Modeler.

Выделенные при изучении предметной области сущности содержат следующие основные поля:

- а) Пользователь (Users):
 - номер телефона;
 - e-mail адрес;
 - пароль;
- б) Роль (Roles):
 - наименование роли;
- в) Детали пользователя (UserDetails):
 - ФИО пользователя;
 - дата регистрации;
 - идентификационный номер;
- г) Заявка пользователя (Requests):
 - сумма кредита;
 - срок погашения;
 - ежемесячный доход;
 - количество выплачиваемых (на данный момент) кредитов;
 - сумма платежей по текущим кредитам;
- д) Ответ на заявку (Responses):
 - допустимая сумма кредита с учетом указанных в заявке данных;
- е) Кредит (Credits):
 - процентная ставка;
 - тип кредита (наименование назначения кредита);
- ж) Банк (Banks):
 - зона доверия по кредитному скорингу (минимально допустимая сумма баллов (0-400) согласно кредитному отчету пользователя, сформированному в Нацбанке);
 - адрес офиса;
 - телефонные номера.

Физическая модель БД на языке IDEF1X представлена на рисунке 2.2.

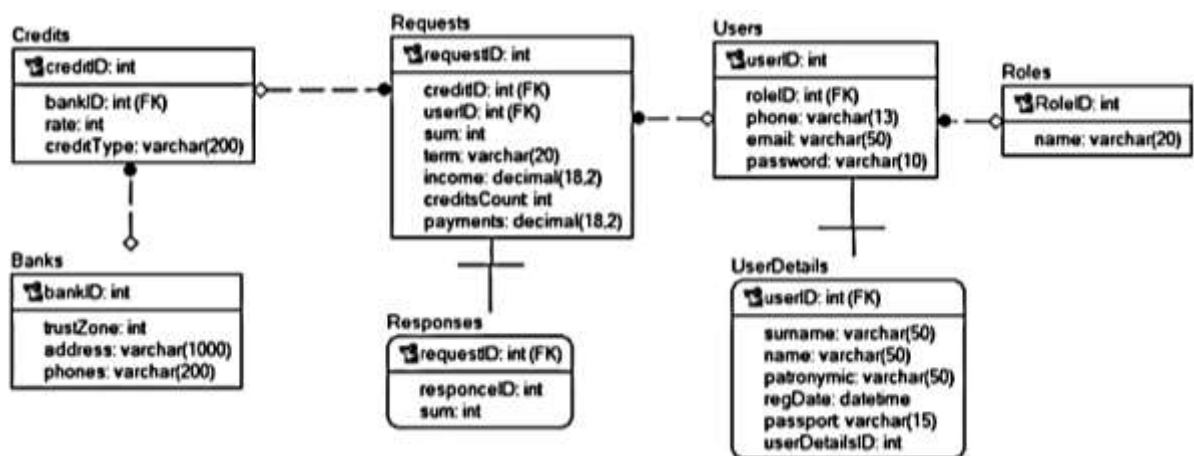


Рисунок 2.2 – Физическая модель БД

2.3 Генерация DDL скриптов

На рисунке 2.3 представлен фрагмент генерации DDL скрипта из физической модели БД в программе Erwin Data Modeler.

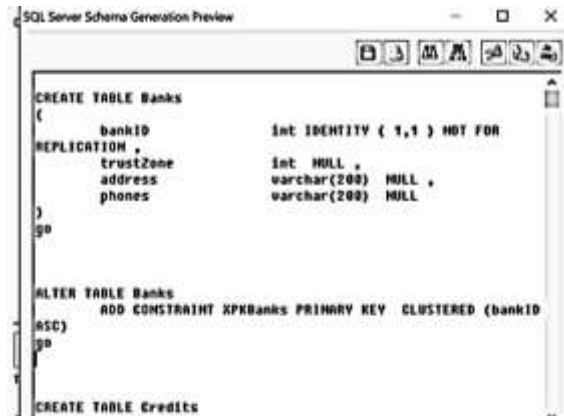


Рисунок 2.3 – Генерация DDL скрипта

2.4 Создание базы данных

Работа с данными основывается на системе управления базами данных SQLServer. Создание локальной БД осуществим с помощью утилиты SQL Server Management Studio и сгенерированного DDL скрипта, предварительно создав пользователя для данной базы и выдав права для работы с базой [3]. Для работы с БД из приложения настройки (имя пользователя, пароль, имя хоста и порт) были записаны в файл application.properties.

Код скрипта создания БД представлен в приложении.

На рисунке 2.4 представлен обозреватель объектов утилиты SQL Server Management Studio, содержащий созданную БД и пользователя.



Рисунок – 2.4 Создание пользователя и БД

3 ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА ПО НА ЯЗЫКЕ UML

3.1 Разработка функциональной модели

Функциональная модель представлена в виде диаграммы использования на языке графического описания UML (см. рис. 3.1) [4].



Рисунок 3.1 – Функциональная модель

В приложении предусматривается две роли: Пользователь и Администратор. Рассмотрим каждую роль подробнее.

Возможные действия Пользователя:

- зарегистрироваться на сайте Сервиса, осуществить вход или выход (при наличии регистрации);
- просмотреть доступные виды кредитов, выбрать необходимый вариант и перейти к оформлению онлайн-заявки;
- при оформлении заявки указать требуемые личные данные и данные о желаемом кредите (сумму, сроки выплаты и т.д.);
- получить от Сервиса сообщение-ответ об одобрении / не одобрении запрошенного кредита;
- перейти на страницу с персональными данными и на страницу с контактной информацией банка.

Возможные действия Администратора:

- осуществить вход (выход) на сайте Сервиса;
- добавить новый вид кредита, изменить или удалить существующий вид кредита;
- изменить зону доверия банка по кредитному скорингу (минимально допустимая сумма баллов (0-400) согласно кредитному отчету Пользователя, сформированному в Нацбанке);
- добавить / поменять контактную информацию о банке.

3.2 Разработка модели данных

Модель данных — это концептуальное представление для выражения и передачи бизнес-требований. Она наглядно показывает характер данных, бизнес-правила, управляющие данными, и то, как данные будут организованы в базе данных.

На рисунке 3.2 представлена физическая модель БД на языке графического описания UML.

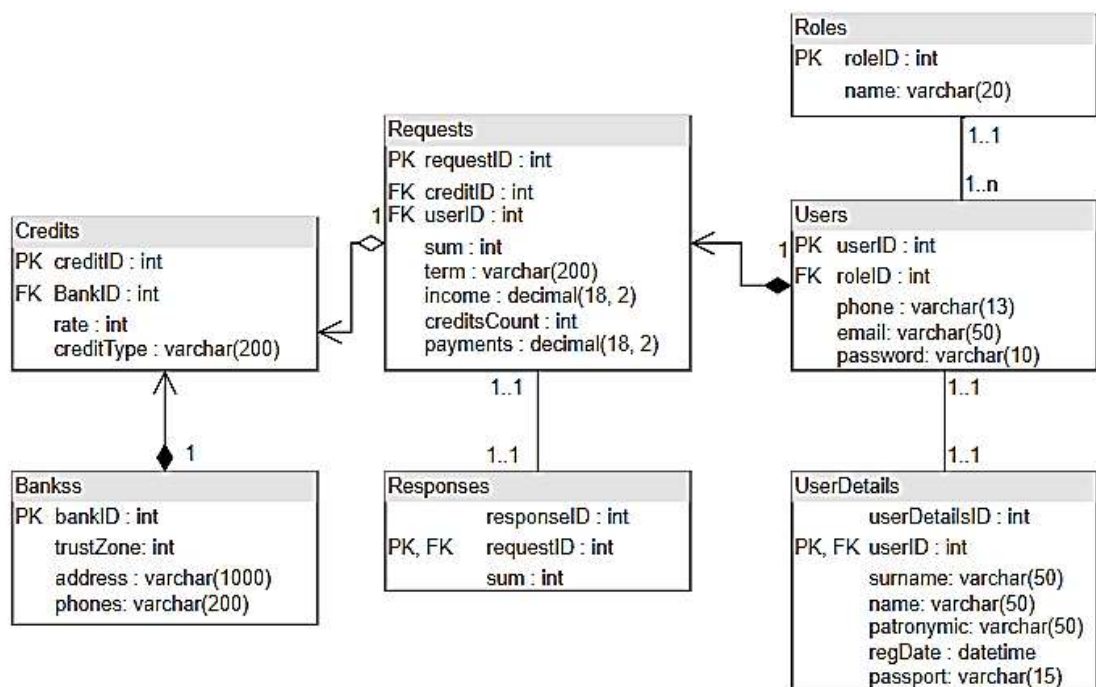


Рисунок 3.2 – Физическая модель БД

3.3 Проектирование приложения. Диаграммы

3.3.1 Диаграмма классов

Диаграмма классов — это центральная методика моделирования, которая используется практически во всех объектно-ориентированных

методах [5]. Эта диаграмма описывает типы объектов в системе и различные виды статических отношений, которые существуют между ними.

Три наиболее важных типа отношений в диаграммах:

- 1 Ассоциация, которая представляет отношения между экземплярами типов.
- 2 Наследование, которое имеет непосредственное соответствие наследованию в Объектно-Ориентированном дизайне.
- 3 Агрегация, которая представляет из себя форму композиции объектов в объектно-ориентированном дизайне.

Диаграмма классов для Сервиса на языке графического описания UML представлена на рисунке 3.3.

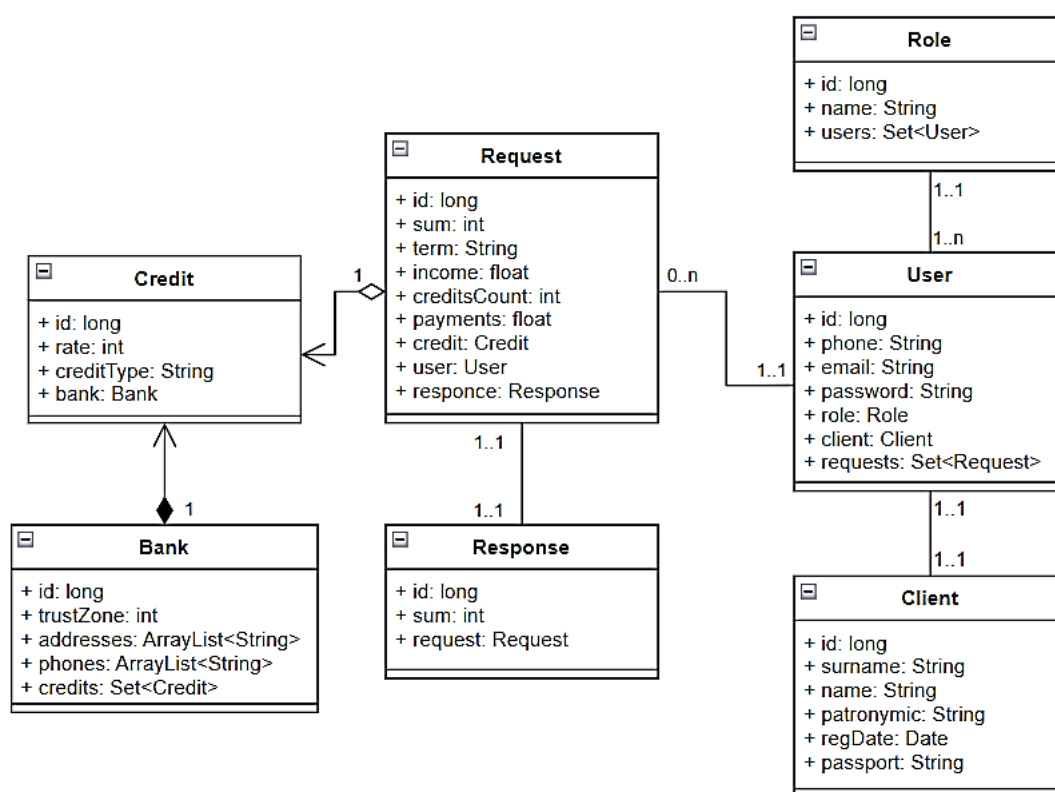


Рисунок 3.3 – Диаграмма классов

3.3.2 Диаграмма состояний

Диаграмма состояний — это тип диаграммы, используемый в UML для описания поведения систем, который основан на концепции диаграмм состояний Дэвида Харела. Диаграммы состояний отображают разрешенные состояния и переходы, а также события, которые влияют на эти переходы. Она помогает визуализировать весь жизненный цикл объектов и, таким образом, помогает лучше понять системы, основанные на состояниях.

На рисунке 3.4 представлена диаграмма состояний для Сервиса на языке графического описания UML.

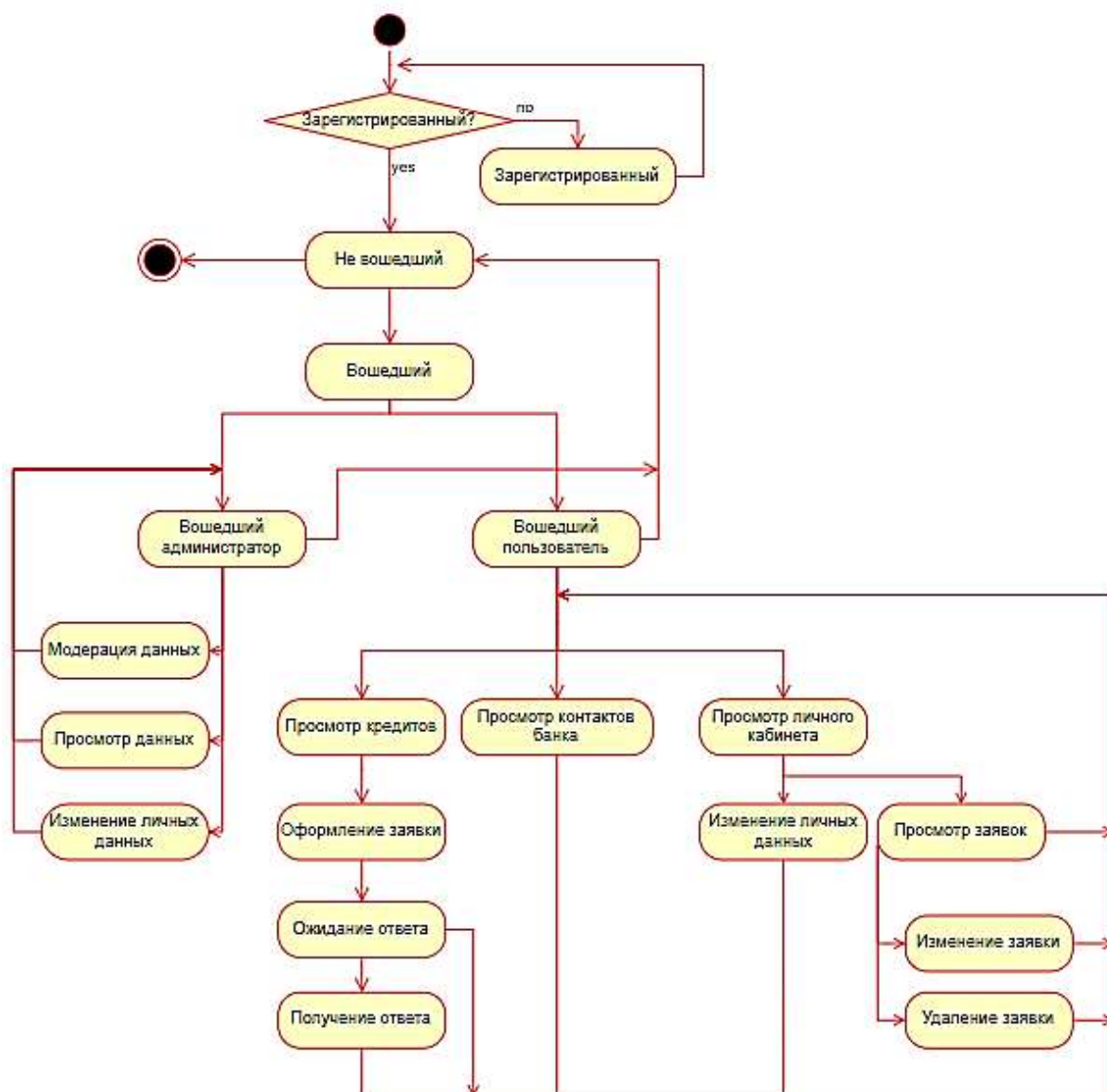


Рисунок 3.4 – Диаграмма состояний

3.3.3 Диаграмма активности

Диаграммы активности представляют собой графическое представление рабочих процессов поэтапных действий и действий с поддержкой выбора, итерации и параллелизма [6]. Они описывают поток управления целевой системой, такой как исследование сложных бизнес-правил и операций, а также описание прецедентов и бизнес-процессов. В UML диаграммы активности предназначены для моделирования как вычислительных, так и организационных процессов.

На рисунке 3.5 представлена диаграмма активности для регистрации, входа и оформления онлайн-заявки пользователем на языке графического описания UML.

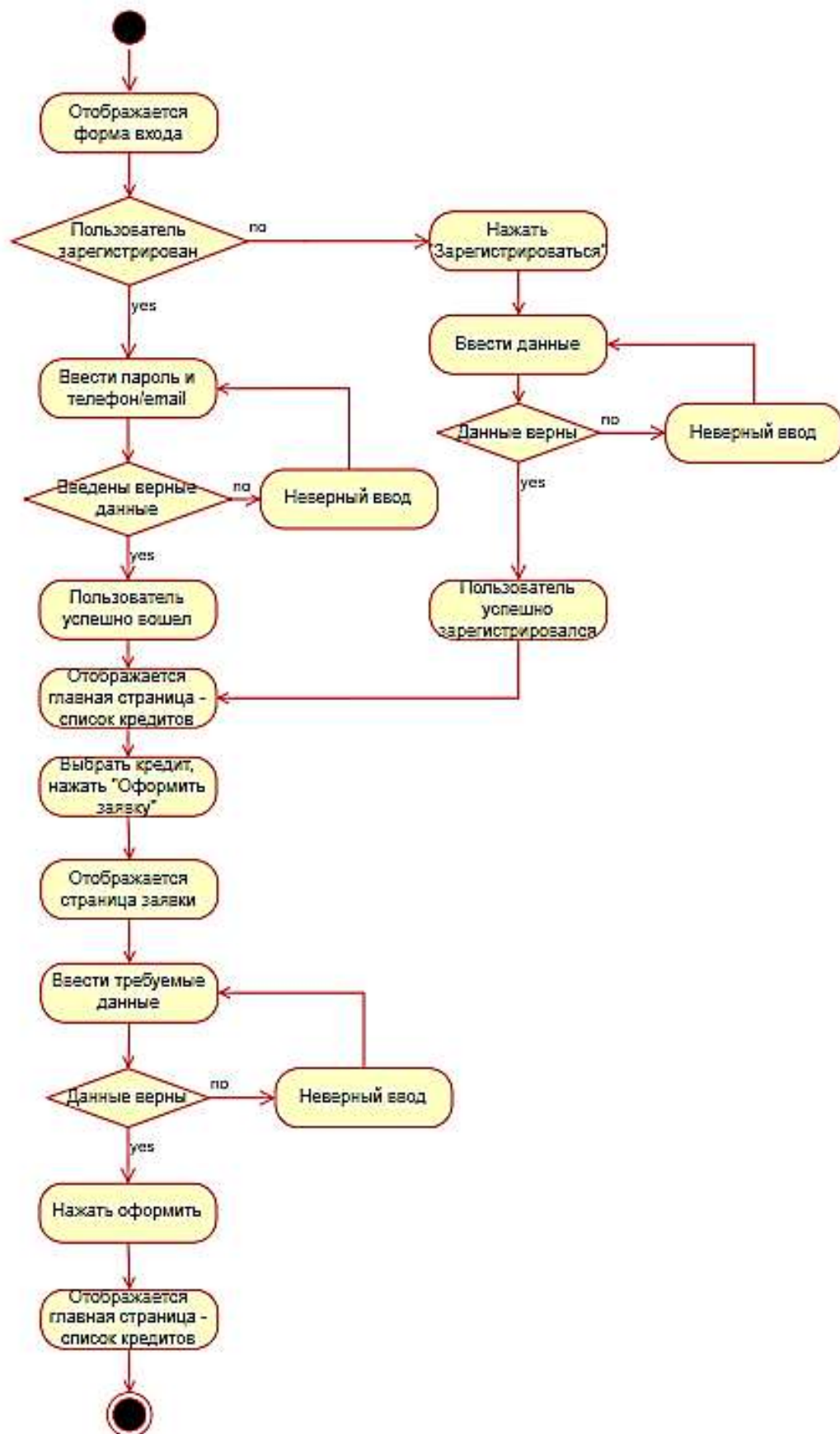


Рисунок 3.5 – Диаграмма активности

3.3.4 Диаграмма компонентов

На языке унифицированного моделирования диаграмма компонентов показывает, как компоненты соединяются вместе для формирования более крупных компонентов или программных систем [7]. Она иллюстрирует архитектуры компонентов программного обеспечения и зависимости между ними. Эти программные компоненты включают в себя компоненты времени выполнения, исполняемые компоненты, а также компоненты исходного кода.

Диаграмма компонентов Сервиса на языке графического описания UML представлена на рисунке 3.6.

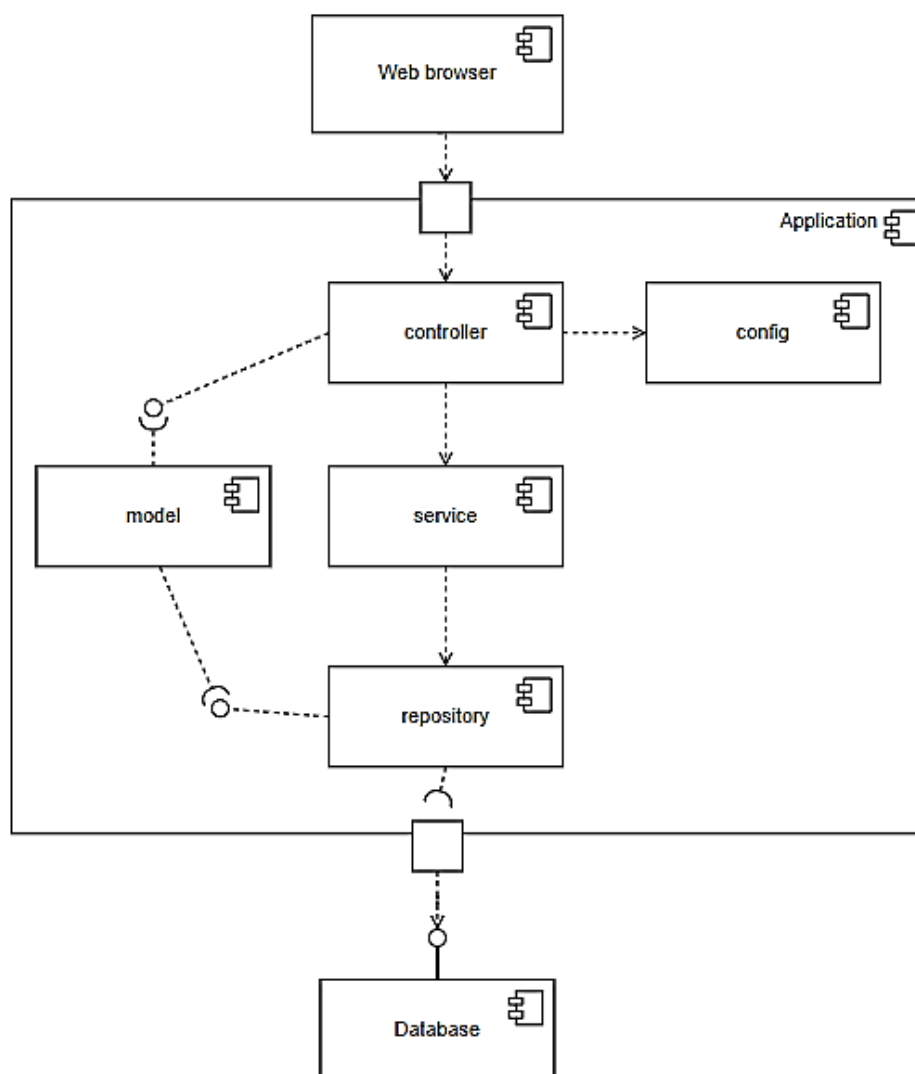


Рисунок 3.6 – Диаграмма компонентов

4 СОЗДАНИЕ ДИАГРАММ ИНФОРМАЦИОННЫХ ПОТОКОВ

4.1 Теоретические сведения

Диаграммы IDEF3 также называют WorkFlow diagramming – методологией моделирования, использующей графическое описание информационных потоков, взаимоотношений между процессами обработки информации и объектов, являющихся частью этих процессов [8]. Диаграммы WorkFlow используются для анализа процедур обработки информации.

Цель IDEF3 – дать аналитикам описание последовательности выполнения процессов, а также объектов, участвующих совместно в одном процессе.

Диаграммы IDEF3 отображают *действие* (единицу работы – Unit of Work (UOW)) в виде прямоугольника. Как уже отмечалось, действия именуются с использованием глаголов или отглагольных существительных, каждому из действий присваивается уникальный идентификационный номер. Этот номер не используется вновь даже в том случае, если в процессе построения модели действие удаляется.

Связи выделяют существенные взаимоотношения между действиями (см. рис. 4.1). Все связи в IDEF3 являются однонаправленными, и, хотя стрелка может начинаться или заканчиваться на любой стороне блока, обозначающего действие, диаграммы IDEF3 обычно организуются слева направо таким образом, что стрелки начинаются на правой и заканчиваются на левой стороне блоков.



Изображение	Название	Назначение
	Временное предшествование (Temporal precedence)	Исходное действие должно завершиться, прежде чем конечное действие сможет начаться
	Объектный поток (Object flow)	Выход исходного действия является входом конечного действия. Из этого, в частности, следует, что исходное действие должно завершиться, прежде чем конечное действие сможет начаться
	Нечеткое отношение (Relationship)	Вид взаимодействия между исходным и конечным действиями задается аналитиком отдельно для каждого случая использования такого отношения

Рисунок 4.1 – Типы связей в методологии IDEF3

Завершение одного действия может инициировать начало выполнения сразу нескольких других действий или, наоборот, определенное действие может требовать завершения нескольких других действий до начала своего выполнения. *Соединения* разбивают или соединяют внутренние потоки и используются для описания ветвления процесса:

- разворачивающие соединения используются для разбиения потока;
- сворачивающие соединения объединяют потоки.

На рисунке 4.2 представлены три типа соединений в IDEF3.

Графическое обозначение	Название	Вид	Правила инициации
	Соединение «и»	Разворачивающее	Каждое конечное действие обязательно иницируется
		Сворачивающее	Каждое исходное действие обязательно должно завершиться
	Соединение «эксклюзивное "или"»	Разворачивающее	Одно и только одно конечное действие иницируется
		Сворачивающее	Одно и только одно исходное действие должно завершиться
	Соединение «или»	Разворачивающее	Одно или несколько конечных действий иницируются
		Сворачивающее	Одно или несколько исходных действий должны завершиться

Рисунок 4.2 – Типы соединений в методологии IDEF3

Указатели — это специальные символы, которые ссылаются на другие разделы описания процесса. Они используются при построении диаграммы для привлечения внимания пользователя к каким-либо важным аспектам модели. Указатель изображается на диаграмме в виде прямоугольника. Имя указателя обычно включает его тип и идентификатор (см. рис. 4.3).

Тип указателя	Назначение
ОБЪЕКТ (OBJECT)	Для описания того, что в действии принимает участие какой-либо заслуживающий отдельного внимания объект
ССЫЛКА (GOTO)	Для реализации цикличности выполнения действий. Указатель ССЫЛКА может относиться и к соединению
ЕДИНИЦА ДЕЙСТВИЯ (Unit of Behavior — UOB)	Для многократного отображения на диаграмме одного и того же действия. Например, если действие «Подсчет наличных» выполняется несколько раз, в первый раз оно создается как действие, а последующие его появления на диаграмме оформляются указателями UOB
ЗАМЕТКА (NOTE)	Для документирования любой важной информации общего характера, относящейся к изображенному на диаграммах. В этом смысле ССЫЛКА служит альтернативой методу помещения текстовых заметок непосредственно на диаграммах
УТОЧНЕНИЕ (Elaboration — ELAB)	Для уточнения или более подробного описания изображенного на диаграмме. Указатель УТОЧНЕНИЕ обычно используется для описания логики ветвления у соединений

Рисунок 4.3 – Типы указателей в методологии IDEF3

4.2 Диаграмма

На рисунке 4.5 представлена диаграмма информационных потоков Сервиса на языке IDEF3.

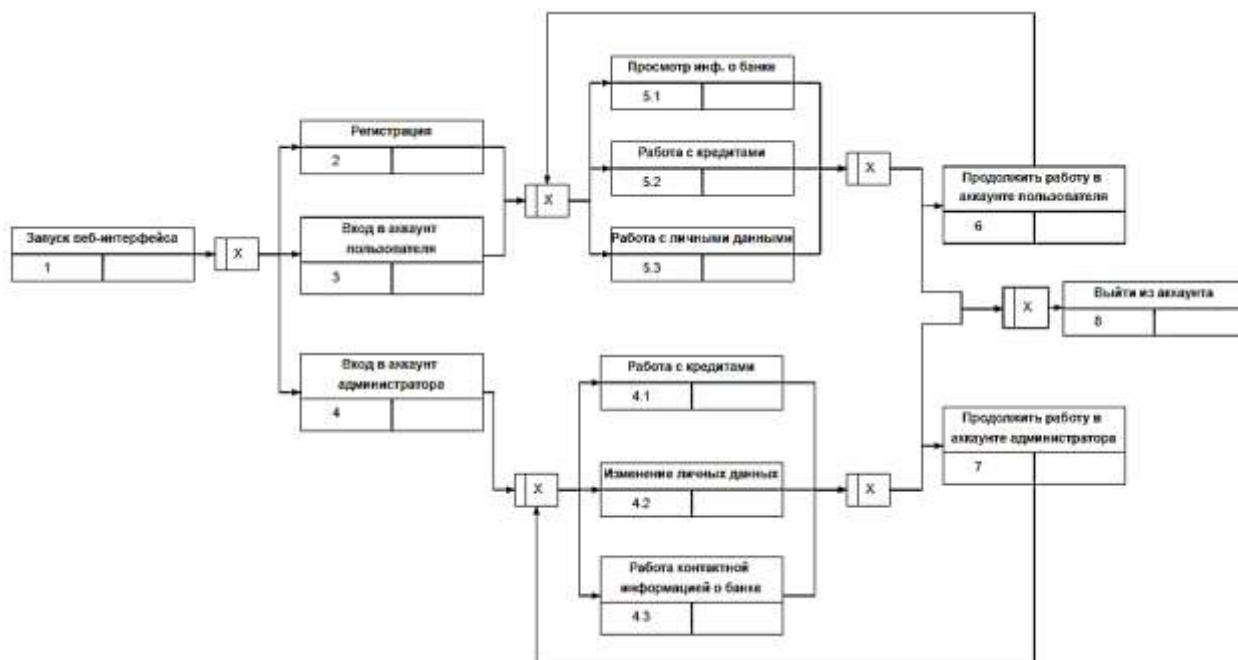


Рисунок 4.5 – Диаграмма информационных потоков

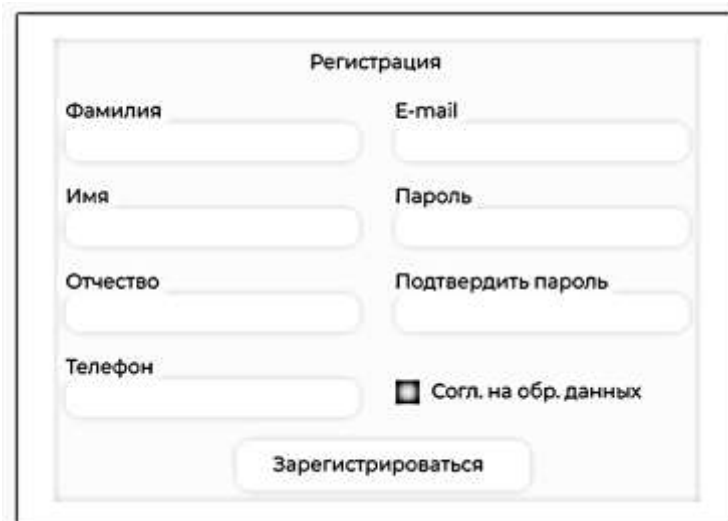
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- [1] Документация Erwin [Электронный ресурс]. – Режим доступа: <http://surl.li/drnog>.
- [2] IDEF1X [Электронный ресурс]. – Режим доступа: <https://ru.wikipedia.org/wiki/IDEF1X>.
- [3] Transact-SQL [Электронный ресурс]. – Режим доступа: <https://learn.microsoft.com/en-us/sql/t-sql/>.
- [4] UML-диаграммы классов [Электронный ресурс]. – Режим доступа: <https://prog-cpp.ru/uml-classes/>.
- [5] Что находится между идеей и кодом? Обзор 14 диаграмм UML [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/post/508710/>.
- [6] Простое руководство по диаграммам активности UML [Электронный ресурс]. – Режим доступа: <http://surl.li/dtqei>.
- [7] Простое руководство по диаграммы компонентов [Электронный ресурс]. – Режим доступа: <http://surl.li/cdbwv>.
- [8] Методология IDEF3 [Электронный ресурс]. – Режим доступа: <https://itteach.ru/bpwin/metodologiya-idef3#toc2>.

ПРИЛОЖЕНИЕ А


(обязательное)

Схематичное представление интерфейса



The registration form is titled "Регистрация" and is contained within a light gray box. It features two columns of input fields. The left column contains fields for "Фамилия", "Имя", "Отчество", and "Телефон". The right column contains fields for "E-mail", "Пароль", and "Подтвердить пароль". Below the "Телефон" field is a checkbox labeled "Согл. на обр. данных". At the bottom center of the form is a button labeled "Зарегистрироваться".

Рисунок А.1 – Схема страницы регистрации



The login form is titled "Вход" and is contained within a light gray box. It features two input fields: "Телефон или E-mail" and "Пароль". Below these fields is a button labeled "Войти".

Рисунок А.2 – Схема страницы входа

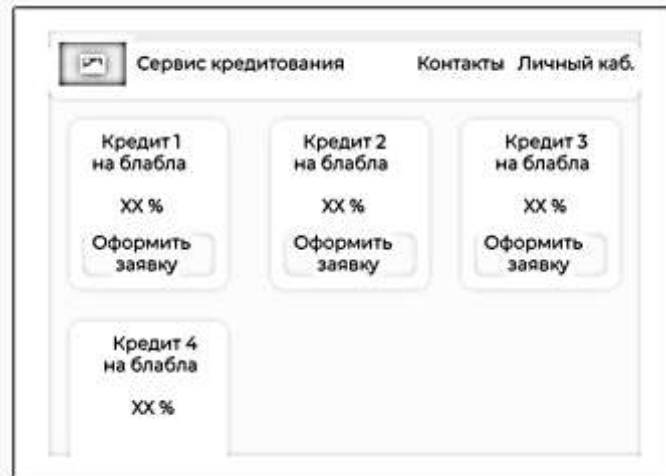


Рисунок А.3 – Схема главной страницы

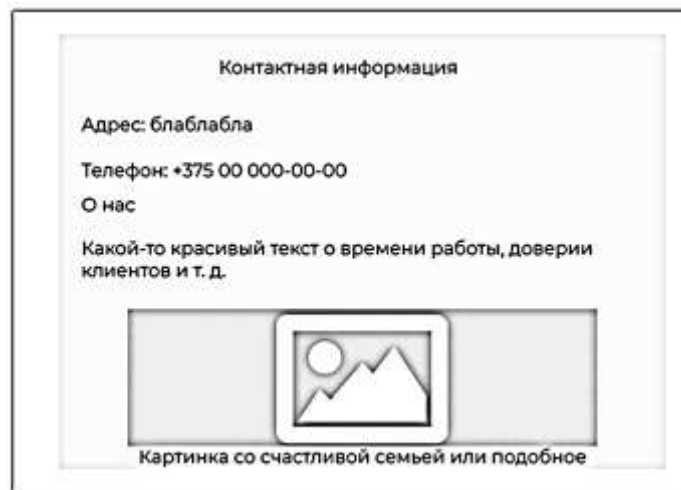


Рисунок А.4 – Схема страницы контактов банка



Рисунок А.5 – Схема страницы личного кабинета

Оформление онлайн-заявки

Кредит 1 на блабла	Наличие кредитов
Ставка - XX%	<input type="text"/>
Сумма <input type="text"/>	Платежи по кредитам (мес.) <input type="text"/>
Срок <input type="text"/>	Идентификационный № <input type="text"/>
Доходы (мес.) <input type="text"/>	<input type="checkbox"/> Согл. на обр. данных
<input type="button" value="Отправить заявку"/>	

Рисунок А.6 – Схема страницы оформления заявки

Благодарим Вас за обращение

Одобрено (не одобрено) XXXX бел. руб.
 Ставка - XX%, срок - XX мес.(лет), платеж - XXX бел. руб.

В ближайшее время с Вами
 свяжется специалист от банка

Рисунок А.7 – Схема сообщения-ответа на заявку от банка

Администратор

Ставка <input type="text"/>	Вид кредита <input type="text"/>	<input style="border: 1px solid black; border-radius: 50%; width: 20px; height: 20px; text-align: center; line-height: 20px; vertical-align: middle;" type="button" value="+"/>
Зона доверия (1.400) <input type="text"/>		
Адрес банка <input type="text"/>		
Контактный телефон <input type="text"/>		
<input type="button" value="Сохранить"/>		

Рисунок А.8 – Схема страницы администрирования