

Университет за национално и световно стопанство

Факултет „Приложна информатика и математика“

Катедра „Информационни технологии и комуникации“



РАЗРАБОТКА НА ПРОЕКТ ОТ ТИП БИЗНЕС
ПРИЛОЖЕНИЕ

На тема

**Система за кандидатстване и одобрение на
бързи кредити**

Мириам Риме

Факултетен № 21180011

Специалност „Бизнес информатика и комуникации“

Година на обучение: четвърта

30 май, 2025 г.

| | |
|--|-----------|
| 1. Описание на проекта | 4 |
| 1.1 Цели на проекта | 4 |
| 1.2 Обхват на проекта | 4 |
| 2. Бизнес цели | 5 |
| 3. Заинтересовани страни | 5 |
| 3.1 Клиенти | 5 |
| 3.2 Администратори на системата | 6 |
| 3.3 Служители на фирмата | 6 |
| 3.4 Ръководство на компанията | 6 |
| 3.5 Регулаторни органи | 7 |
| 4. Бизнес процеси | 7 |
| <i>Процес на регистрация на клиент</i> | <i>7</i> |
| <i>Процес по регистриране на служител/администратор</i> | <i>8</i> |
| <i>Процес по кандидатстване за кредит</i> | <i>8</i> |
| <i>Процес на разглеждане и одобрение на кредит</i> | <i>8</i> |
| <i>Процес на преглед на активни кредити, плащане на вноска и приключване на кредит от клиент</i> | <i>9</i> |
| <i>Процес по сторниране на вноска и сторниране на усвояване по кредит</i> | <i>9</i> |
| <i>Приключване на кредит от служител/администратор</i> | <i>10</i> |
| 4.1 Процесна диаграма | 10 |
| 5. Функционалности | 10 |
| 5.1 Регистрация и управление на потребители | 10 |
| 5.2 Кандидатстване за кредит | 11 |
| 5.3 Управление на кредитни заявки | 11 |
| 5.4 Управление на активни кредити | 12 |
| 5.5 Финансови операции | 12 |
| 5.6 Административни функционалности | 13 |

| | | |
|-----|---|----|
| 5.7 | Потребителски интерфейс..... | 13 |
| | <i>Основни екрани</i> | 15 |
| 5.8 | Потоци от данни | 15 |
| 6. | Модел на данните | 15 |
| | <i>Таблица: Clients</i> | 16 |
| | <i>Таблица: ClientFinancials</i> | 16 |
| | <i>Таблица: ClientAddress</i> | 16 |
| | <i>Таблица: Credits</i> | 17 |
| | <i>Таблица: RepaymentPlan</i> | 17 |
| | <i>Таблица: Nomenclature</i> | 18 |
| | <i>Таблица: log_21180011</i> | 18 |
| 6.1 | Връзки между таблиците | 18 |
| 6.2 | Източници на данни | 19 |
| 6.3 | Обработка на данните | 19 |
| 7. | Технологии и средства за реализация | 19 |
| 8. | Тестване | 20 |
| 8.1 | Тестов случай 1 | 20 |
| 8.2 | Тестов случай 2 | 22 |
| | Приложения | 28 |
| | Приложение 1 | 28 |
| | Приложение 2 | 30 |

1. Описание на проекта

„Система за кандидатстване и одобрение на бързи кредити“ е уеб-базирана система за отпускане и управление на краткосрочни заеми, което позволява на потребители (студенти, работещи хора и пенсионери) да кандидатстват онлайн за бързи кредити до заплата. Системата автоматизира процеса на кандидатстване, одобрение и управление на кредити, като предоставя удобен и интуитивен интерфейс.

Приложението решава проблема с продължителния и сложен процес по кандидатстване за кредит в традиционните финансови институции, като предлага по-бърз и достъпен начин за получаване на краткосрочно финансиране. Системата осигурява прозрачност в условията и дава възможност на потребителите да следят статуса на своите кредитни заявки и погасителни планове.

Архитектурата на приложението е изградена на принципа клиент-сървър, по подхода Database-first. За СУБД е използвана програмата MS SQL Server, а като платформа за изпълнение на приложението е избрано Razor Pages с развойна среда Microsoft Visual Studio 2022. Избраната технология позволява бързо зареждане на страници, което оптимизира потребителското изживяване.

1.1 Цели на проекта

Проекта има няколко основни цели. Първата е автоматизация на процеса по кандидатстване за кредит. Намаление на времето и необходимите документи за кандидатстване е от ключово значение за много потребители. Също така се подобрява и ефективността поради дигитализирането на процеса. Бързия и удобен достъп до услугите повишават удовлетвореността на клиентите и подобряват прозрачността на процеса, тъй като клиентите имат достъп до информация за своите кредити по всяко време.

1.2 Обхват на проекта

Системата следните компоненти и функционалности:

- Регистрация и управление на профили на клиенти – клиентите ще имат възможност да създават профили в приложението;
- Онлайн кандидатстване за кредит, който преминава през следните стъпки под формата на форма за кандидатстване:

- Въвеждане на лични данни на клиента като имена, ако клиентът съществува, да се попълват автоматично;
- Въвеждане на адресна информация;
- Въвеждане на финансови данни;
- Избор на параметри на кредита
- Автоматична валидация на въведените данни;
- Процес за разглеждане и одобрение/отхвърляне на кредитни заявки;
- Генериране на погасителен план;
- Проследяване на плащания по кредити;
- Роли и права на достъп: администратор, служител-каса, клиенти;

Извън обхвата ще останат интеграция с банкови системи за автоматично превеждане на средства, система за уведомяване чрез имейл или SMS, и автоматична проверка на кредитна история от външни източници.

2. Бизнес цели

Проектът за онлайн отпускане на бързи кредити допринася за следните стратегически цели:

- Подобряване на конкурентоспособността, тъй като представя опростен процес на кандидатстване и одобрение, което прави платформата атрактивна за нови потребители и подобрява клиентското изживяване;
- Намаляване на оперативните разходи чрез автоматизация на ръчните процеси и намаляване на нуждата от физически офиси;
- Увеличаване на пазарния дял поради достъпността на услугата за по-широк кръг от потребители;

3. Заинтересовани страни

Заинтересованите страни на приложението са клиентите, администратори на системата, служители, ръководство на компанията и регулаторните органи.

3.1 Клиенти

Клиентите са основна част от функционирането на приложението, затова техните нужди и очаквания са главен приоритет.

- Бърз и лесен достъп до краткосрочни кредити при нужда;

- Прозрачни условия по кредита;
- Минимално количество документи за кандидатстване;
- Лесен начин за проследяване на задълженията;
- Защита на личните данни;
- Бърз отговор по заявката за кредит;

Много от потребителите на бързи кредити имат нужда от парите в рамките на същия ден. Бързото кандидатстване с минимални изискуеми документи и бързо одобрение е основна нужда на клиентите. Също така, не може да се пренебреже и нуждата от прозрачни условия по кредита. Всеки клиент очаква да знае по всяко време какво се случва с парите му, което прави не само прозрачните условия, но и лесното проследяването на задълженията основно очакване от страна на потребителите.

3.2 Администратори на системата

- Пълен контрол над системата;
- Възможност за управление на потребители, служители и кредити;
- Преглед и редакция на всички данни;

Администраторите трябва да имат пълен достъп и контрол над всички данни и операции в системата.

3.3 Служители на фирмата

Служителите ще имат ограничени права с системата. Техните нужди и очаквания се очаква да бъдат следните:

- Лесен преглед на чакащите заявки за кредит;
- Инструменти за оценка на кредитоспособността;
- Възможност за бързо одобрение/отхвърляне на заявки;
- Управление на съществуващи кредити;
- Преглед на плащания и неплатени вноски;

3.4 Ръководство на компанията

Ръководството на компанията се очаква да има следните нужди и очаквания:

- Намаляване на оперативните разходи;
- Увеличаване на броя клиенти;
- Подобряване на ефективността на процесите;

Очакванията на ръководството съвпадат и с целите на проекта. Всяка компания цели да намали оперативните разходи. Онлайн система за бързи кредити се стреми да намали нуждата от отварянето на нови офиси и наемането на служители на каса като дигитализира процеса по отпускането на кредити. Също така, лесният достъп се очаква да доведе до по-висок приток на клиенти и да подобри ефективността като намали времето за кандидатстване и одобрение.

3.5 Регулаторни органи

Регулаторните органи са основен елемент в разработката на бизнес приложение, поради множеството закони, на които трябва да отговарят системите, които боравят с лична информация за клиентите. Техните очаквания и нужди са следните:

- Спазване на законодателството в областта на потребителските кредити;
- Защита на личните данни на клиентите;
- Прозрачност на условията по кредитите;
- Справедливи практики при кредитиране;

4. Бизнес процеси

Системата ще има следните бизнес процеси, които ще протекат по описаните по-долу стъпки:

Процес на регистрация на клиент

1. Клиентът посещава системата;
2. Избира опция за регистрация;
3. Формата за регистрация се показва;
4. Клиентът попълва необходимите данни;
5. Системата валидира въведените данни
 - а. При успешна валидация, данните на клиента се записват в базата и той получава потвърждение, че вече е регистриран потребител и му се предоставя достъп;
 - б. При неуспешна валидация, клиентът получава съобщение и е върнат, за да коригира данните си;

Имплементираните валидации са относно дали клиентът е въвел имейл адрес в правилен формат (например example@example.com) и дали потребител с такъв имейл вече съществува.

Процес по регистриране на служител/администратор

Регистрирането на служител/администратор се води по същия процес като за регистриране на клиент. Разликата е, че системата е проектирана така, че когато даден потребител се регистрира с корпоративен имейл от вид example@admin.flashpay.bg той автоматично ще бъде регистриран като администратор в системата. Ако потребителя се регистрира с корпоративен имейл от вид example@emp.flashpay.bg той автоматично бива регистриран като служител в системата.

Процес по кандидатстване за кредит

1. Клиентът влиза в системата;
2. Избира менюто за кандидатстване
3. Стартира се процес по кандидатстване и формата за попълване на данни се показва;
4. Клиентът попълва нужните данни в четири стъпки:
 - a. Попълване (или подвърждаване, ако клиентът вече съществува) на личните данни;
 - b. Попълване (или подвърждаване, ако клиентът вече съществува) на адресна информация;
 - c. Попълване (или подвърждаване, ако клиентът вече съществува) на финансови данни;
 - d. Избиране на параметри на желания кредит;
5. Клиентът подвърждава заявката;
6. Показва се подвърждение за успешно кандидатстване;

След всяка стъпка, данните на клиента се записват (ако е нов) или модифицират (ако е извършена редакция). Кредитът автоматично бива записан в статус „В очакване на решение“.

Процес на разглеждане и одобрение на кредит

1. Служител/администратор влиза в системата;
2. Избира опция „Всички кредити“ и избира кредит, който иска да разгледа;
3. Потребителят е препратен към нова страница, в която вижда детайлите на кредита;
4. Взема се решение за одобрение/отхвърляне;

- a. Ако кредитът е одобрен, служител/администратор избира опция „Одобрение“. В базата се генерира погасителен план и статусът на кредита се сменя на „Активен“;
- b. Ако кредитът е отхвърлен, служител/администратор избира опция „Отхвърляне“. Статусът на кредита се сменя на „Отхвърлен“

Процес на преглед на активни кредити, плащане на вноската и приключване на кредит от клиент

1. Клиентът влиза в своя профил;
2. Избира опция за преглед на активни кредити;
3. Показва се информация за кредита и погасителния му план;
4. При преглед, клиентът има право да избере „Плащане на вноската“;
5. Въвежда данни за платежна карта, чийто номер бива валидиран чрез Алгоритъм на Лун. Ако това е последна вноската, статусът на кредита се променя в „Приключен“ и клиентът получава съобщение

Процес по сторниране на вноската и сторниране на усвояване по кредит

1. Потребител от тип служител или администратор влиза в системата;
2. Избира меню „Вноски по кредит“;
3. Ако потребителят е от тип Администратор, той има право да сторнира усвояване по кредит и вноски по кредит:
 - a. Ако това е единствената операция по кредит, администраторът сторнира усвояването, кредитът се връща в статус „Активен“ и погасителния план се изтрива. Кредитът може да бъде одобрен наново.
 - b. Ако кредитът има платени вноски, администраторът получава съобщение, че няма право да сторнира усвояването по кредита.
 - c. При сторниране на вноската, потребителят получава съобщение, че вноската е сторнирана, в базата се генерира финансова операция с обратен знак, датата „Платено на“ се нулира и клиентът може да плати вноската отново.
 - d. При опит за повторно сторниране на същата операция, се получава съобщение, че не може да се извърши повторно сторниране.
4. Ако потребителят е от тип Служител, той има право да сторнира само вноски по кредит:

- a. При опит за сторниране на усвояване по кредит, служителят получава съобщение, че няма права за извършването на тази операция.
- b. При сторниране на вноската, потребителя получава съобщение, че вноската е сторнирана, в базата се генерира финансова операция с обратен знак, датата „Платено на“ се нулира и клиентът може да плати вноската отново.
- c. При опит за повторно сторниране на същата операция, се получава съобщение, че не може да се извърши повторно сторниране.

Приключване на кредит от служител/администратор

1. Служител/администратор влизат в системата;
2. Избира се кредит за преглед и опция „Приключване“;
3. Статусът се променя на „Приключен“.

4.1 Процесна диаграма

Процесните диаграми са представени в Приложение 1.

5. Функционалности

5.1 Регистрация и управление на потребители

Процесът на регистрация е проектиран да бъде интуитивен за потребителите и да осигурява бърз и лесен достъп до системата. Клиентите лесно ще могат да създадат собствен профил като въведат само имейл и парола, с която свързват акаунта си. След създаването на профил, те ще имат възможността да кандидатстват за кредит.

След успешна регистрация, потребителят все още няма да разполага с възможност за редактиране или добавяне на данни. Те се добавят чрез формата за кандидатстване, когато клиентът подава заявление за кредит. След успешно подадено заявление, потребителят ще има възможност да редактира личните си данни и да актуализира информацията за контакт и финансовата си информация при необходимост.

Администраторите на системата разполагат с разширени възможности за управление на потребителските профили, включително редактиране на различните роли и деактивиране на профили при необходимост.

Служителите имат ограничен права при оправлението на потребители. Те могат да редактират данни на клиент, но не могат да управляват потребителски профили.

Ако клиентът е подал поне едно заявление за кредит, независимо от неговия статус, то тогава той не може да бъде премахнат от системата.

5.2 Кандидатстване за кредит

Основната функционалност на системата е свързана с процеса на кандидатстване за бърз кредит. Тук е реализиран последователен подход, който разделя процеса на логични стъпки за по-добро потребителско изживяване и по-ефективно събиране на информация.

Процесът на кандидатаване е разделен на четири стъпки. В случай, че това е първи кредит на потребителя, личните му данни, адресната информация и финансова информация ще бъдат запазвани в базата. В противен случай, данните ще бъдат автоматично зареждани при опит за кандидатстване и потребителят само потвърди тяхната коректност.

При всяка се извършва валидация на въведените данни. След завършване на всички стъпки, системата извършва автоматично изчисление на месечната вноса на база избраната сума, лихвения процент и срока за погасяване.

5.3 Управление на кредитни заявки

Служителите и администраторите имат достъп до табло с преглед на всички чакащи заявки за кредит. Списъкът с заявки може да бъде филтриран и сортиран по различни критерии: номер на кредит, сума на кредита, начална и крайна дата на кредит, ЕГН и номер на клиент, статус на кредит., което позволява на служителите да организират работния си процес оптимално.

При избор на конкретна заявка, системата показва подробна информация за клиента, включително личните и финансовите данни, необходими за оценка на кредитоспособността. Интерфейсът е проектиран така, че всички критични данни да бъдат ясно видими - съотношение между доходи и разходи, тип на заетост и наличие на достатъчен остатъчен доход за покриване на месечната вноса. Преди одобрение на кредит, служителя/администратора виждат системно съобщение за платежоспособността на клиента, която се изчислява по формулата:

Месечни доходи – Месечни разходи \geq (Вноса + Месечни доходи * 30%)

Ако резултатът връща *True*, излиза съобщение, че кредитът може да бъде одобрен, тъй като клиентът има достатъчно финансови средства. Ако резултатът връща *False*, излиза съобщение, че е препоръчително кандидатурата да бъде отхвърлена.

При одобрение на заявка, системата автоматично генерира погасителен план, който показва разпределението на плащанията между главница и лихва за всяка вноска. След като заявката бъде обработена, статусът ѝ се променя съответно на "Активен" или "Отхвърлен", което се отразява незабавно в профила на клиента, осигурявайки прозрачност на процеса.

5.4 Управление на активни кредити

Служителите имат достъп до подробен списък на всички активни кредити, който може да бъде филтриран и сортиран по различни критерии: номер на кредит, сума на кредита, начална и крайна дата на кредит, ЕГН и номер на клиент, статус на кредит. Това позволява бързо идентифициране на кредити, които изискват специално внимание, например такива с просрочени плащания.

За всеки активен кредит може да се види погасителния план, кои вноски са платени, кои остават да бъдат платени, дата наплащане на вноската. При постъпване на плащане, системата автоматично обновява статуса на съответната вноската и преизчислява оставащия дълг.

5.5 Финансови операции

Всяко плащане по кредит се регистрира като финансова операция, като се записва датата на плащане, сумата, типа на операцията и връзката с конкретния кредит. Системата автоматично разпределя постъпилата сума между главница и лихва съгласно погасителния план.

За всеки кредит системата поддържа детайлна история на всички свързани финансови операции, което позволява пълно проследяване на паричните потоци. Тази информация е достъпна само за служителите и администраторите.

Всяка финансова операция може да бъде сторнирана при нужда. Само администратор има права да сторнира операция със статус „Усвояване по кредит“. Когато операция по усвояване бъде сторнирана, кредитът се връща в статус „Очакващ решение“ и погасителния план бива автоматично изтрит от базата. При последващо активиране на кредита, погасителния план ще бъде генериран наново.

При сторниране на обикновена операция „Вноска по кредит“, операцията се записва с обратен знак и бутонът за плащане отново става активен. Така клиентът има възможност да опита да плати вноската отново.

5.6 Административни функционалности

Администраторите имат пълен контрол върху потребителските профили и правата за достъп. Те могат да създават нови потребителски акаунти за служители, да определят техните роли и нива на достъп, както и да деактивират акаунти при необходимост. Системата поддържа йерархична структура на ролите, което позволява детайлно дефиниране на позволените действия за всяка потребителска група.

Системата предоставя подробен журнал на всички извършени действия, включително създаване, редактиране и изтриване на записи, както и промени в системните настройки. Този журнал служи като одитна следа и дава възможност за проследяване на всички изменения, което е важно както от гледна точка на сигурността, така и за спазване на регулаторните изисквания.

5.7 Потребителски интерфейс

Когато става въпрос за финансови институции, простотата и яснотата на потребителския интерфейс е от основно значение. Интерфейсът е изчистен от излишни елементи, с фокус върху най-важната информация и действия. Това е особено важно за процеса на кандидатстване за кредит, където потребителите трябва да се чувстват комфортно и уверено при въвеждането на личните и финансовите си данни.

Цветовата схема е подбрана внимателно с цел да създава доверие, но и да привлича вниманието и да стимулира клиентите към действие - използват се предимно жълто, акценти в синьо и тъмно сиво. Жълтото повишава концентрацията и поражда положителни емоции като чувство за топлина, креативност и радост, което кара клиентите да се чувстват уверени и мотивирани. Сините и тъмно-сивите акценти на места придават по-елегантен вид и създават баланс между възбуждане и увереност. Контрастът между цветовете гарантира добра видимост на елементите, което подобрява клиентското изживяване.

Използваната типографията е четлива и последователна в цялото приложение, с ясна йерархия на заглавията и текста. Използван е шрифта *Sofia Sans*, който е създаден от български дизайнери с ясна идея да изглежда добре за кирилица. Шрифтът изглежда

професионално, но не е твърде строго корпоративен, което вдъхва увереност у клиента и същевременно създават чувство за спокойствие и близост.

Навигационната структура е логична и интуитивна, с ясни етикети на менютата и бутоните. Основните действия са лесно разпознаваеми и достъпни. Интерфейсът на системата е организиран в три основни секции, според типа потребители, за които са предназначени: публична част, клиентска част и административна част. Това разделение осигурява целенасочен достъп до функционалностите в зависимост от потребителската роля.

Публична част представлява лицето на услугата за нерегистрирани потребители. Тя е привлекателна и информативна, с цел да привлече нови клиенти и да ги запознае с предлаганите услуги. Началната страница съдържа ясно послание за предимствата на услугата, с акцент върху бързината и удобството. Тук потребителите могат да намерят подробна информация за условията за кредитиране, включително изисквания към кандидатите, необходими документи и процес на одобрение.

Интерактивният калкулатор за кредити позволява на посетителите да изчислят примерен размер на месечните вноски според желаната сума и срок, което им дава предварителна представа за финансовите параметри на кредита. Страницата за регистрация и вход е опростена, с минимално необходими полета, за да се насърчи създаването на профил.

Клиентската част е достъпна след влизане в системата и е проектирана да предостави на клиента пълен контрол над неговите кредити и лични данни. Централен елемент в тази част е персонализираното табло за управление, което показва обобщена информация за кредитите на клиента във всички фази – кандидатури, активни кредити, приключени кредити и отхвърлени кредити, лична информация. От таблото клиентът има бърз достъп до всички основни функционалности чрез интуитивни икони и бутони.

Процесът на кандидатстване за нов кредит е структуриран в четири ясно очертани стъпки, с индикатор за прогрес, който показва на клиента къде се намира в процеса. При преглед на погасителен план на активните кредити, клиентът вижда ключови параметри като обща сума, главница и лихва по вноската, оставащ дълг и предстоящи плащания. Погасителният план е представен в табличен вид с ясно разграничение между платени и неплатени вноски.

Административната част е предназначена за служители и администратори и предоставя инструменти за управление на дейността на системата. Таблото за управление в тази част съдържа списък със заявки за кредити, който е централен елемент в работата на служителите. Той е проектиран да предоставя бърз преглед на основната информация с възможност за филтриране и сортиране по различни критерии. При избор на конкретна заявка, системата показва подробна информация в добре структуриран формат, който позволява бърза оценка на кредитоспособността на клиента.

Основни екрани

- Начална страница - информация за услугите, кредитен калкулатор, бутони за кандидатстване и регистрация;
- Табло за управление (клиенти) – информация за активни и приключени кредити, информация за лични данни и възможност за тяхната редакция;
- Форма за кандидатстване;
- Табло за управление (служители и администратор) – информация за клиенти и кредити, възможност за добавяне, редакция и изтриване

5.8 Потоци от данни

Диаграма на потоците от данни е представена в Приложение 2.

6. Модел на данните

Системата използва релационна база данни SQL Server. Структурата на данните е изградена, така че да дава възможност за по-нататъчно развитие на програмата с цел минимална загуба на данни. Основните таблици и връзките между тях са следните:

Таблица: Accounts

В тази таблица се съхранява информацията за потребителските профили

- ID (int, primary key) - Уникален идентификатор
- Username (nvarchar(256)) – Имейл адрес, с който се регистрира потребителя
- ClientID (int, foreign key) – Връзка към таблица Clients
- PasswordHash (varbinary(64)) – Парола (Хеш)
- PasswordSalt (varbinary(128)) – Парола (Сол)
- Role (tinyint) – Роля на потребителя в системата

- IsActive – Флаг за активност на потребител
- CreatedAt (datetime) – Дата и час на създаване
- ModifiedOn_21180011 (datetime) – Дата и час на последна промяна

Таблица: Clients

Съхранява информация за клиентите.

- ID (int, primary key) - Уникален идентификатор
- FirstName (nvarchar(50)) - Име на клиента
- MiddleName (nvarchar(50)) - Бащино име на клиента
- LastName (nvarchar(50)) - Фамилия на клиента
- EGN (nvarchar(10)) - ЕГН
- Email (nvarchar(20)) - Емейл адрес
- PhoneNumber (nvarchar(10)) - Телефонен номер
- IDCardNumber (nvarchar(20)) - Номер на лична карта
- IDValidityDate (date) - Дата на валидност на личната карта
- IDIssueDate (date) - Дата на издаване на личната карта
- IDIssuer (nvarchar(50)) - Издател на личната карта
- CreatedOn (datetime) - Дата и час на създаване
- ModifiedOn_21180011 (datetime) - Дата и час на последна промяна

Таблица: ClientFinancials

Съхранява финансова информация за клиентите.

- ID (int, primary key) - Уникален идентификатор
- ClientID (int, foreign key) - Връзка към таблица Clients
- MonthlyIncome (decimal) - Месечен доход
- MonthlyExpenses (decimal) - Месечни разходи
- EmploymentType (int, foreign key) - Тип на заетост (връзка към Nomenclature)
- CreatedOn (datetime) - Дата и час на създаване
- ModifiedOn_21180011 (datetime) - Дата и час на последна промяна

Таблица: ClientAddress

Съхранява адресна информация за клиентите.

- ID (int, primary key) - Уникален идентификатор
- ClientID (int, foreign key) - Връзка към таблица Clients
- City (nvarchar(50)) - Град
- StreetNeighbourhood (nvarchar(50)) - Улица/квартал
- Number (nvarchar(5)) - Номер
- PostCode (nvarchar(4)) - Пощенски код
- CreatedOn (datetime) - Дата и час на създаване
- ModifiedOn_21180011 (datetime) - Дата и час на последна промяна

Таблица: Credits

Съхранява информация за кредитите.

- ID (int, primary key) - Уникален идентификатор
- ClientID (int, foreign key) - Връзка към таблица Clients
- CreditAmount (decimal) - Сума на кредита
- CreditPeriod (int) – Период на кредита
- TotalCreditAmount (decimal) – Сума на кредит в края на периода
- MonthlyInstallment (decimal) – Месечна вноска по кредит
- CreditBeginDate (date) - Начална дата на кредита
- CreditEndDate (date) - Крайна дата на кредита
- InterestRate (decimal) - Лихвен процент
- Status (int, foreign key) - Статус на кредита (връзка към Nomenclature)
- CreatedOn (datetime) - Дата и час на създаване
- ModifiedOn_21180011 (datetime) - Дата и час на последна промяна

Таблица: RepaymentPlan

Съхранява погасителния план по кредитите.

- ID (int, primary key) - Уникален идентификатор
- CreditID (int, foreign key) - Връзка към таблица Credits
- InstallmentNumber (int) - Номер на вноската
- InstallmentDate (date) - Дата на вноската
- InstallmentAmount (decimal) - Сума на вноската
- Principal (decimal) - Главница
- Interest (decimal) - Лихва

- PayedOnDate (date) - Дата на плащане
- CreatedOn (datetime) - Дата и час на създаване
- ModifiedOn_21180011 (datetime) - Дата и час на последна промяна

Таблица: FinancialOperations

Съхранява финансови операции по кредитите.

- ID (int, primary key) - Уникален идентификатор
- CreditID (int, foreign key) - Връзка към таблица Credits
- PayedOnDate (date) - Дата на плащане
- PayedAmount (decimal) - Платена сума
- OperationType (int, foreign key) - Тип на операцията (връзка към Nomenclature)
- RepaymentPlanID (int, foreign key) – Връзка към Погасителен план
- CreatedOn (datetime) - Дата и час на създаване
- ModifiedOn_21180011 (datetime) - Дата и час на последна промяна

Таблица: Nomenclature

Съхранява номенклатури за различни категории.

- NomCode (int, primary key) - Уникален код
- Description (nvarchar(255)) - Описание
- CreatedOn (datetime) - Дата и час на създаване
- ModifiedOn_21180011 (datetime) - Дата и час на последна промяна

Таблица: log_21180011

Съхранява логове за действия в системата.

- ID (int, primary key) - Уникален идентификатор
- TableName (nvarchar(50)) - Име на таблицата
- ActionType (nvarchar(50)) - Тип на действието
- ActionDate (datetime) - Дата и час на действието

6.1 Връзки между таблиците

1. Clients -> ClientFinancials: One-to-One (един клиент има една финансова информация)

2. Clients -> ClientAddress: One-to-One (един клиент има един адрес)
3. Clients -> Credits: One-to-Many (един клиент може да има много кредити)
4. Credits -> RepaymentPlan: One-to-Many (един кредит има много вноски)
5. Credits -> FinancialOperations: One-to-Many (един кредит има много финансови операции)
6. Nomenclature -> ClientFinancials: One-to-Many (тип на заетост)
7. Nomenclature -> Credits: One-to-Many (статус на кредит)
8. Nomenclature -> FinancialOperations: One-to-Many (тип на операция)

6.2 Източници на данни

1. Потребителски вход - данни въведени от клиентите при регистрация и кандидатстване за кредит;
2. Административен вход - данни въведени от служители и администратори
3. Системно генерирани данни - автоматично създадени от системата (например погасителни планове)
4. Номенклатури - предварително дефинирани данни за типове заетост, статуси на кредити и др.

6.3 Обработка на данните

1. Валидация на входни данни - проверка за коректност на въведената информация
2. Изчисление на кредитни параметри - изчисляване на вноски, лихви и главници
3. Генериране на погасителен план - автоматично създаване на план за погасяване
4. Проследяване на плащания - обновяване на статуси на вноски и кредити
5. Журналиране на промени - запис на всички промени в системата за одит

7. Технологии и средства за реализация

Проектът е разработен на Razor Pages, част от ASP.NET Core за изграждане на интерфейса на приложението. Razor синтаксисът позволява враждане на C# код директно в HTML, което улеснява работата с данни и динамично генериране на съдържание. Бизнес логиката, обработката на заявки и комуникация с базата данни са реализирани на C#.

За контрол на достъпа и разделяне на права, използвам вградените механизми на ASP.NET Core Authorization, за да гарантирам, че само потребители с подходящи права мога да достигнат до определени страници в приложението. За регистрация и логин, използвам ASP.NET Identity, който предоставя готови модели за потребители, роли и claims, които могат да се разширяват с допълнителни полета. Интеграция с Entity Framework Core позволява лесно съхранение на всички данни за аутентикация и авторизация в базата.

За бази от данни използвам Microsoft SQL Server, като използвам database-first подход - съществуващата база данни се scaffold-ира в EF Core модели чрез Scaffold-DbContext, което позволява бързо генериране на класове и миграции, съобразени с реалната схема. Достъпът до данните и ORM функционалностите се управляват чрез Entity Framework Core. Относно администрирането на базата, използвам SQL Server Management Studio (SSMS) за писане на SQL скриптове и преглед на данни.

Вътрешните данни и бизнес операции са достъпни чрез RESTful API контролери (ASP.NET Core Web API), а за външни интеграции се използват JSON/HTTP заявки с аутентикация чрез JWT токени и автоматично генерирана документация с OpenAPI (Swagger).

Експортирането на данни в Excel файл става чрез използването на ClosedXML библиотеката. Това е wrapper върху Open XML SDK, който позволява генериране и форматиране на .xlsx файлове без инсталиран Office на сървъра.

Основното IDE, на което е разработен проекта е Visual Studio 2022. За управлението на версиите на проекта и контрол на кода е използван Git (GitHub).

8. Тестване

8.1 Тестов случай 1

В Тестов случай 1 ще тестваме Регистрация и Вход на потребител с примерни входни данни:

- **Имейл:** ivan.ivanov@mail.com
- **Парола:** ivan_ivanov1234

| Use case 1: Регистрация и вход на потребител | | | | |
|--|--|--|---|---|
| Test Case ID | Предварителни Условия | Заглавие/Описание | Стъпки | Очакван резултат |
| REG01 | Потребителят не е логнат; Потребителят не съществува в системата; | Проверка за наличие на всички полета във формата | Кликнете върху бутон „Вземи сега“ или „Вход“; Кликнете върху „Нямаш акаунт? Регистрация“ | Налични са следните етикети: - Имейл - Парола - Повторете парола Налични са следните бутони: - Регистрация - Вече имаш акаунт? Вход |
| REG02 | Потребителят не е логнат; Потребителят не съществува в системата; | Проверка на валидация за имейл | Въведете невалиден имейл (например client@), парола и повторете паролата | Появява се съобщение „Въведете валиден имейл адрес. Пример: example@example.com“ |
| REG03 | Потребителят не е логнат; Потребителят не съществува в системата; | Проверка на валидация за парола | Въведете валиден имейл адрес, парола и различна парола в полето „Повтори парола“ | Появява се съобщение „Паролата не съвпада“ |
| REG04 | Потребителят не е логнат; Потребителят не съществува в системата; | Проверка на валидация за празни полета | Кликнете „Регистрирай се“ без да попълвате нищо | Появяват се следните съобщения - Въведете имейл адрес. - Въведете парола. - Въведете парола. |
| REG05 | Потребителят не е логнат; Потребителят не съществува в системата; | Проверка на успешно регистриране с валидни данни | Въведете валидни данни | Формата за регистрация е предадена успешно; Пренасочени сте към формата за Вход в системата |
| LOG01 | Потребителят не е логнат; Потребителят съществува в системата; | Проверка на валидация за празни полета | Кликнете „Вход“ без да попълвате нищо | Появяват се следните съобщения - Въведете имейл адрес. - Въведете парола. |
| LOG02 | Потребителят не е логнат; | Проверка на валидация за имейл | Въведете невалиден имейл (например client@) и парола | Появява се съобщение „Въведете валиден имейл адрес. Пример: example@example.com“ |

| | | | | |
|-------|---|--|---|---|
| | Потребителят съществува в системата; | | | |
| LOG03 | Потребителят не е логнат; Потребителят съществува в системата; | Проверка на валидация за съществуващ имейл | Въведете несъществуващ имейл (например myemail@myemail.com), и парола | Появява се съобщение „Този имейл не съществува“ |
| LOG04 | Потребителят не е логнат; Потребителят съществува в системата; | Проверка на успешен вход с валидни данни | Въведете данните, с които сте се регистрирали | Системата ще ви пренасочи към началната страница. |

8.2 Тестов случай 2

В Тестов случай 2 ще тестваме Процеса по кандидатстване за кредит.

За стъпка 1 въвеждаме:

- **Име:** Иван
- **Бащино име:** Иванов (незадължително)
- **Фамилия:** Иванов
- **ЕГН:** 0044076528
- **Имейл:** ivan.ivanov@mail.com
- **Телефон:** 0887383956
- **№ на лична карта:** 678002117
- **Дата на издаване:** 12/08/2020
- **Дата на валидност:** 12/08/2030
- **Издадена от (МВР):** София

За стъпка 2 въвеждаме:

- **Населено място:** София
- **Улица/жилищен комплекс:** ул. 8-ми декември
- **Номер:** 19
- **Пощенски код:** 1700

За стъпка 3 въвеждаме:

- **Размер на месечни доходи:** 3200 лв
- **Размер на месечни разходи:** 1400 лв
- **Вид заетост:** На пълен работен ден

За стъпка 4 въвеждаме:

- **Желана сума на кредит:** 2500 лв
- **Период на кредита:** 15 месеца

| Use case 2: Процес по кандидатстване за кредит | | | | |
|--|---|--|--|---|
| Test Case ID | Предварителни Условия | Заглавие/Описание | Стъпки | Очакван резултат |
| FIRST1 | Потребителят е логнат; | Проверка за наличие на всички полета във формата | Кликнете върху бутон „Вземи сега“ или „Кандидатствай“; | Отваря се форма за попълване на лични данни. Налични са следните етикети: <ul style="list-style-type: none"> - Име - Бащино име - Фамилия - ЕГН - Имейл - Телефон - № на лична карта - Дата на издаване - Дата на валидност - Издадена от (МВР) |
| FIRST2 | Потребителят е логнат; Намира се във формата: Стъпка 1 „Лични данни“ | Проверка на валидация за празни полета | Кликнете „Напред“ без да попълвате нищо | Появява се съобщение за грешка на задължителните полета: <ul style="list-style-type: none"> - Моля, въведете име. - Моля, въведете фамилия. - Моля, въведете ЕГН. - Моля, въведете телефонен номер. - Моля, въведете № на лична карта. - Моля, въведете дата на издаване на лична карта. |

| | | | | |
|--------|--|---|---|---|
| | | | | <p>- Моля, въведете дата на валидност на лична карта.</p> <p>- Моля, въведете място на издаване на лична карта.</p> |
| FIRST3 | <p>Потребителят е логнат;</p> <p>Намира се във формата: Стъпка 1 „Лични данни“</p> | Проверка на валидация дали клиентът е навършил 18 години | Въведете ЕГН на човек, който не е навършил 18 години към текущия момент. Например: 0944076528 | Появява се съобщение „Клиентът трябва да е навършил поне 18 години.“ |
| FIRST4 | <p>Потребителят е логнат;</p> <p>Намира се във формата: Стъпка 1 „Лични данни“</p> | Проверка за валидност на месец в ЕГН | Въведете ЕГН с невалиден месец. Например: 0054076528 | Появява се съобщение „Невалиден месец в ЕГН.“ |
| FIRST5 | <p>Потребителят е логнат;</p> <p>Намира се във формата: Стъпка 1 „Лични данни“</p> | Проверка за валидност на дата в ЕГН | Въведете ЕГН с невалидна дата. Например: 0044326528 | Появява се съобщение „Невалидна дата в ЕГН.“ |
| FIRST6 | <p>Потребителят е логнат;</p> <p>Намира се във формата: Стъпка 1 „Лични данни“</p> <p>В системата съществува потребител с ЕГН, което потребителят се опитва да въведе.</p> | Проверка за наличност на клиент със същото ЕГН | Въведете ЕГН на потребител, който вече съществува. Например: 0044074417 | Появява се съобщение „Вече съществува клиент с това ЕГН.“ |
| FIRST7 | <p>Потребителят е логнат;</p> <p>Намира се във формата: Стъпка 1 „Лични данни“</p> | Проверка за валидност на дата на валидност на лична карта | Въведете Дата на валидност на лична карта по-рано от 10 години след посочената дата на издаване. Например: 12/07/2030 | Появява се съобщение „Дата на валидност трябва да бъде 10 години след датата на издаване.“ |

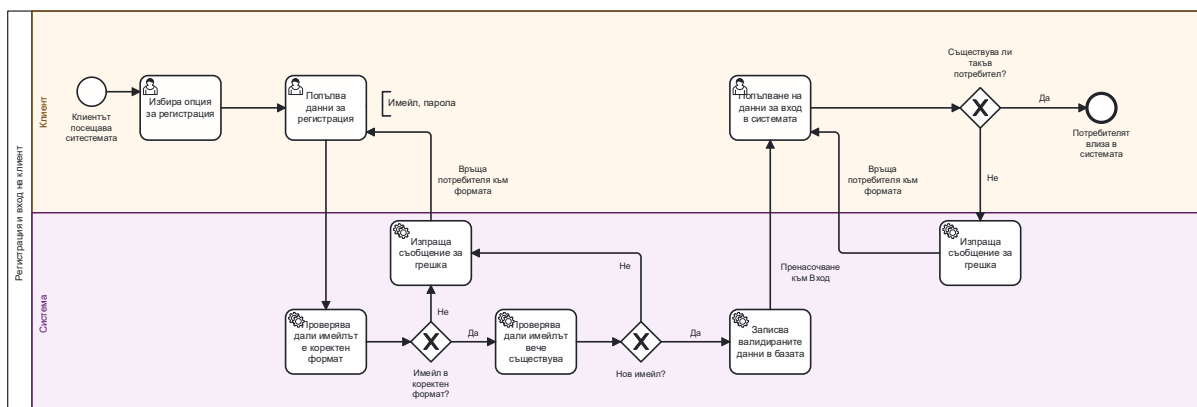
| | | | | |
|--------|--|---|---|--|
| FIRST8 | <p>Потребителят е логнат;</p> <p>Намира се във формата: Стъпка 1 „Лични данни“</p> <p>Дата на издаване е 10 години преди Дата на валидност</p> | Проверка за валидност на Дата на валидност на лична карта | <p>Въведете Дата на валидност на лична карта по-рано от днешна дата. Например:</p> <p>Дата на издаване: 12/08/2012</p> <p>Дата на валидност: 12/08/2022</p> | Появява се съобщение „Личната Ви карта е изтекла.“ |
| FIRST9 | <p>Потребителят е логнат;</p> <p>Намира се във формата: Стъпка 1 „Лични данни“</p> | Проверка за успешно предаване на формата | <p>Въведете валидни данни;</p> <p>Натиснете „Напред“</p> | Преминавате към Стъпка 2 „Данни за адрес“ |
| SECST1 | <p>Потребителят е логнат;</p> <p>Намира се във формата: Стъпка 2 „Данни за адрес“</p> | Проверка на валидация за празни полета | Кликнете „Напред“ без да попълвате нищо | <p>Появява се съобщение за грешка на задължителните полета:</p> <ul style="list-style-type: none"> - Моля, въведете населено място. - Моля, въведете улица/жилищен комплекс. - Моля, въведете пощенски код. |
| SECST2 | <p>Потребителят е логнат;</p> <p>Намира се във формата: Стъпка 2 „Данни за адрес“</p> | Проверка за успешно предаване на формата | <p>Въведете валидни данни;</p> <p>Натиснете „Напред“</p> | <p>Появява се съобщение за грешка на задължителните полета:</p> <ul style="list-style-type: none"> - Моля, въведете населено място. - Моля, въведете улица/жилищен комплекс. - Моля, въведете пощенски код. |
| SECST2 | <p>Потребителят е логнат;</p> <p>Намира се във формата:</p> | Проверка за успешно предаване на формата | <p>Въведете валидни данни;</p> <p>Натиснете „Напред“</p> | Преминавате към Стъпка 3 „Финансови данни“ |

| | | | | |
|--------|--|---|--|---|
| | Стъпка 2 „Данни за адрес“ | | | |
| THRST1 | Потребителят е логнат; Намира се във формата: Стъпка 3 „Финансови данни“ | Проверка на валидация за празно поле „Размер на месечни доходи“ | Кликнете „Напред“ без да попълвате полето за месечни доходи | Появява се съобщение за грешка на задължителните полета: - Моля, въведете размер на доходи. |
| THRST2 | Потребителят е логнат; Намира се във формата: Стъпка 3 „Финансови данни“ | Проверка на валидация за празно поле „Размер на месечни разходи“ | Кликнете „Напред“ без да попълвате полето за месечни разходи | Появява се съобщение за грешка на задължителните полета: - Моля, въведете размер на разходи. |
| THRST3 | Потребителят е логнат; Намира се във формата: Стъпка 3 „Финансови данни“ | Проверка за успешно предаване на формата | Въведете валидни данни; Натиснете „Напред“ | Преминавате към Стъпка 4 „Данни за кредит“ |
| FORST1 | Потребителят е логнат; Намира се във формата: Стъпка 4 „Данни за кредит“ | Проверка на валидация за празни полета | Кликнете „Напред“ без да попълвате нищо | Появява се съобщение за грешка на задължителните полета |
| FORST2 | Потребителят е логнат; Намира се във формата: Стъпка 4 „Данни за кредит“ | Проверка на валидация на Желана сума на кредит | Въведете сума по- малка от 300 лв или по- голяма от 5000 лв. | Появява се съобщение за грешка „Сумата трябва да е между 300 и 5000 лв.“ |

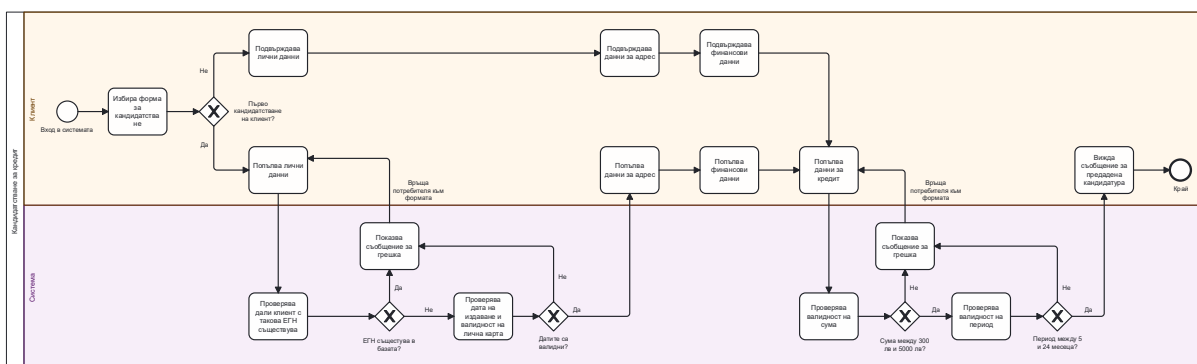
| | | | | |
|--------|--|---|---|---|
| FORST3 | <p>Потребителят е логнат;</p> <p>Намира се във формата: Стъпка 4 „Данни за кредит“</p> | <p>Проверка на валидация на Период на кредита</p> | <p>Въведете период по-малък от 5 месеца или по-голям от 24 месеца</p> | <p>Появява се съобщение за грешка „Периодът трябва да бъде между 5 и 24 месеца.“</p> |
| FORST4 | <p>Потребителят е логнат;</p> <p>Намира се във формата: Стъпка 4 „Данни за кредит“</p> | <p>Проверка на валидация на Период на кредита</p> | <p>Въведете валидни данни; Натиснете „Напред“</p> | <p>Преминавате към финалната страница; Показва се съобщение за успешно записана кандидатура</p> |

Приложения

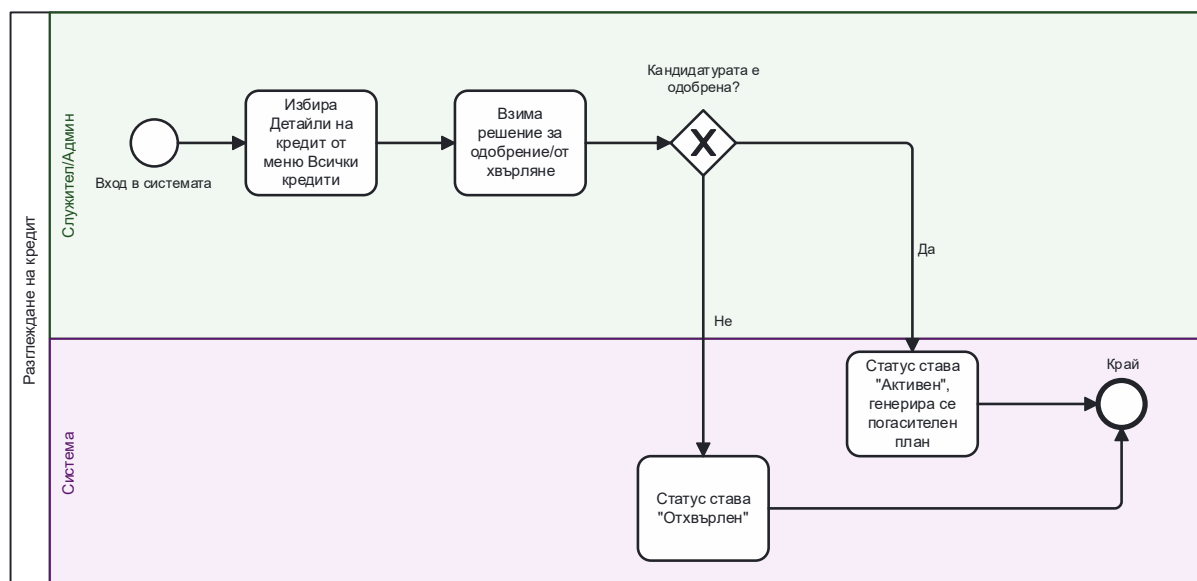
Приложение 1



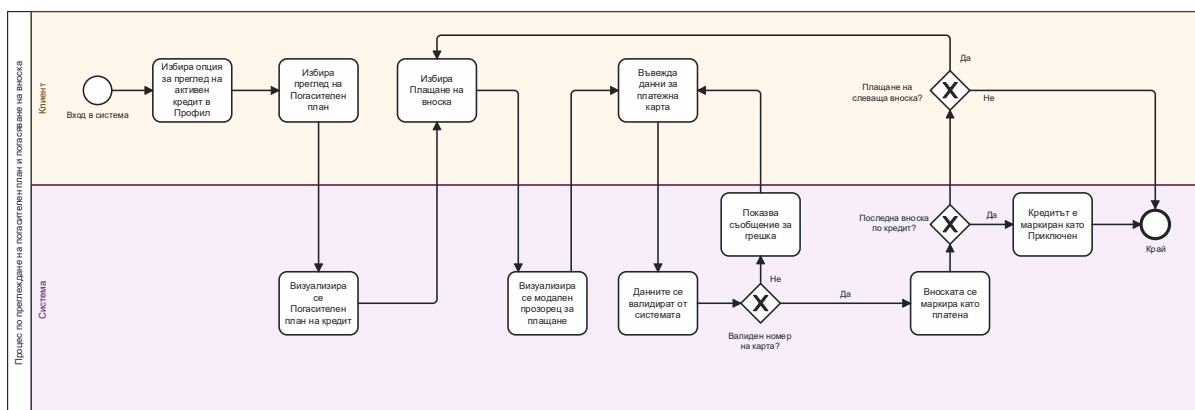
Фигура 1. Процес по регистрация и вход на клиент



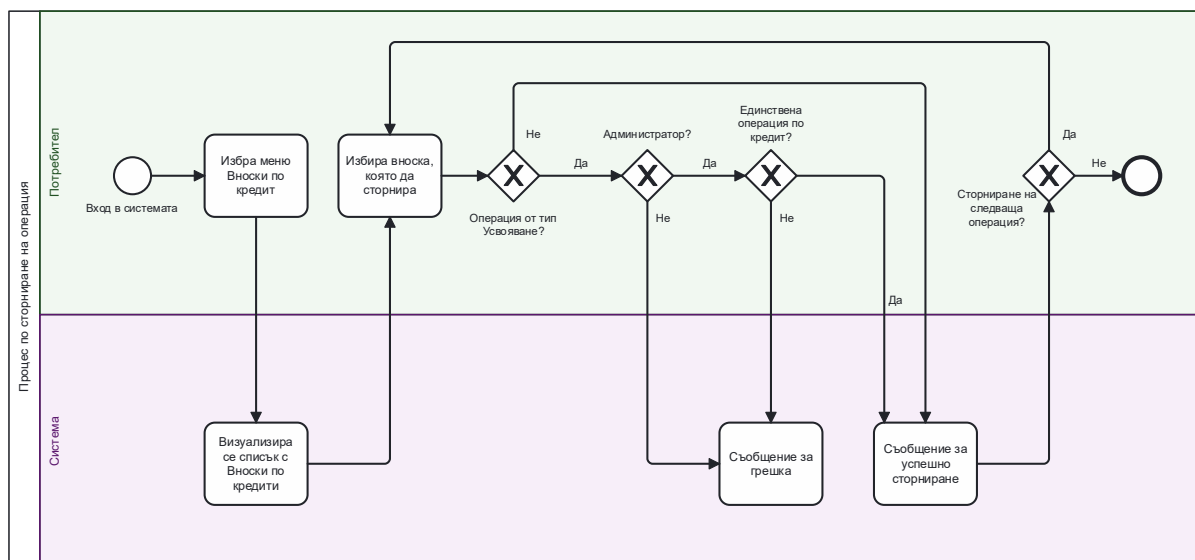
Фигура 2. Кандаставане за кредит



Фигура 3. Процес по разглеждане на кредит

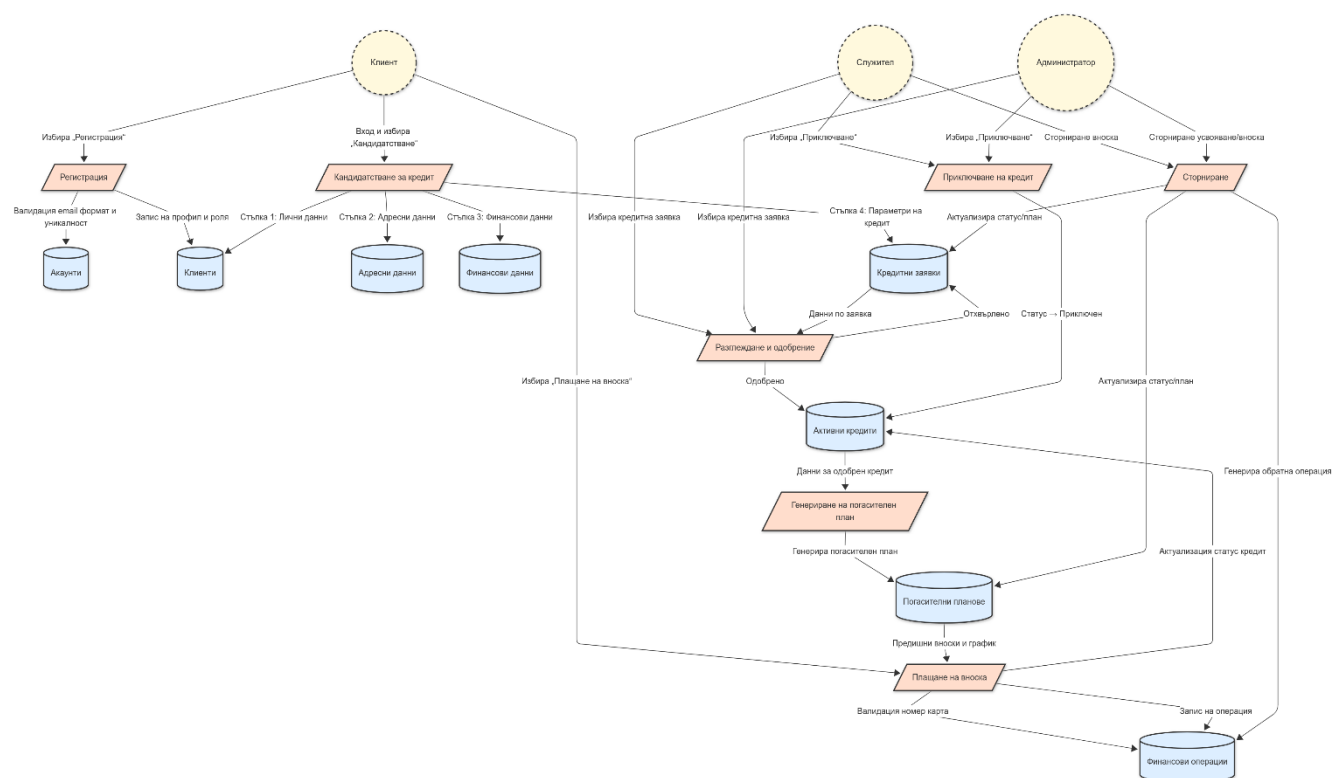


Фигура 4. Процес по плащане на вноски



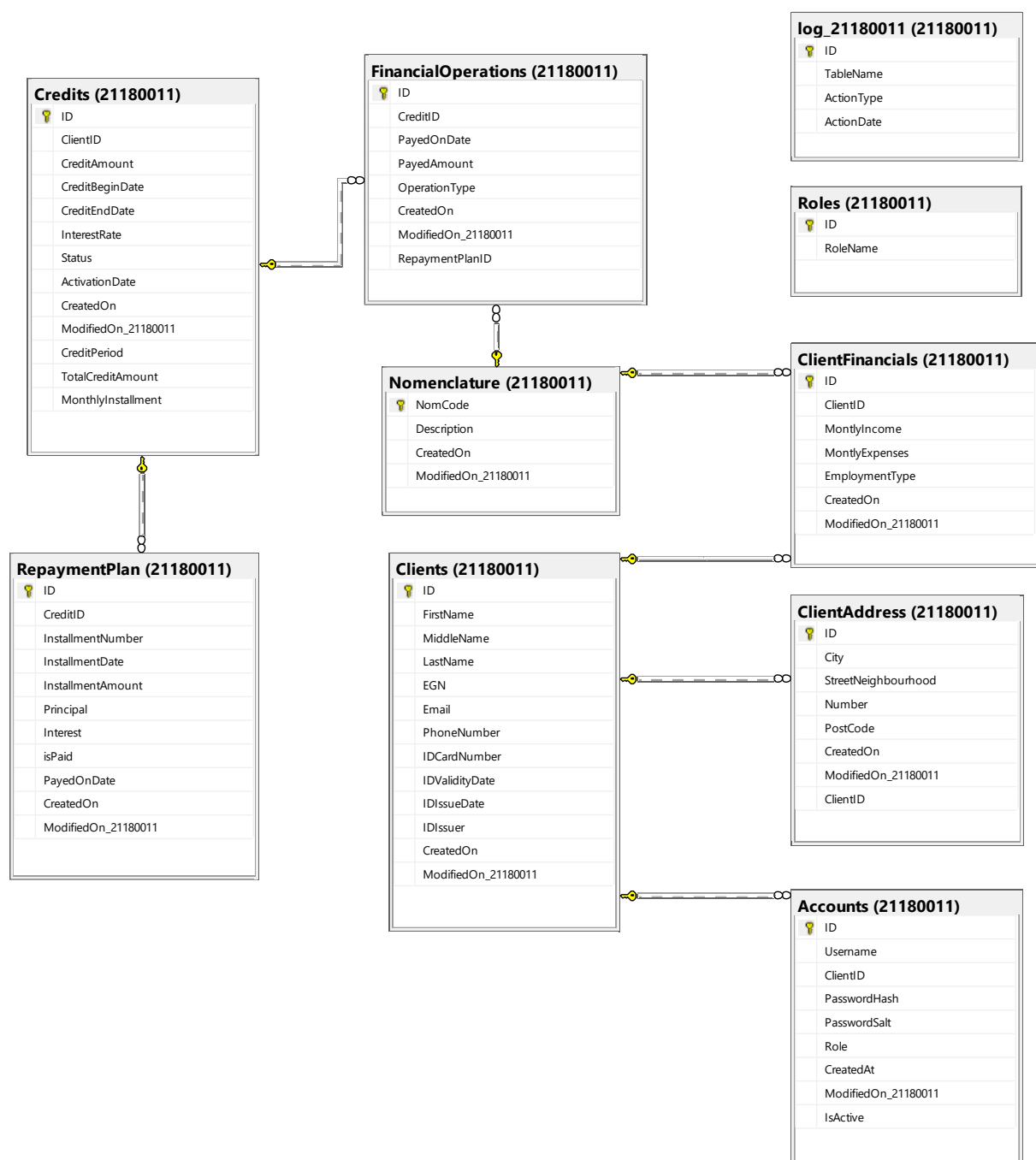
Фигура 5. Процес по сторниране на вноски

Приложение 2



Фигура 6. Процесна диаграма

Приложение 3



Фигура 7. Схема на База от данни