

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
автономное образовательное учреждение высшего образования
«Санкт-Петербургский государственный политехнический университет Петра Великого»
(ФГАОУ ВО «СПбПУ»)

Университетский политехнический колледж

КУРСОВАЯ РАБОТА

Программирование в компьютерных системах

специальность

отделение: Информационных Технологий

По дисциплине: Технология разработки программного обеспечения

Тема Выдача кредитов банком

Исполнитель: студент IV курса группы 42928/2

Тимушев Федор Алексеевич

(фамилия, имя, отчество)

Руководитель преподаватель УПК

(ученная степень, звание)

Зернова Елена Николаевна

(фамилия, имя, отчество)

Курсовая работа допущена к защите

Защитил(а) курсовую работу с оценкой

«21» марта 2019 г.

«22» марта 2019 г

г.Санкт-Петербург, 2019

СОДЕРЖАНИЕ

| | |
|---|----|
| Введение..... | 4 |
| 1 Теоретические основы разработки | 6 |
| 1.1 Описание предметной области | 6 |
| 1.2 Анкетирование..... | 8 |
| 1.3 Анализ методов решения..... | 17 |
| 1.4 Обзор средств программирования | 18 |
| 2 Практическая часть | 20 |
| 2.1 Разработка программной части..... | 20 |
| 2.2 Текст программы | 27 |
| 2.3 Руководство оператора | 29 |
| 2.4 Объект испытаний..... | 41 |
| 2.5 Цель испытаний..... | 41 |
| 2.6 Требования к программе..... | 41 |
| 2.7 Требования к программной документации..... | 42 |
| 2.8 Средства и порядок испытаний..... | 42 |
| 2.9 Методы испытаний..... | 43 |
| 2.10 Тест-кейсы..... | 47 |
| 2.11 Юзер-стори..... | 57 |
| 2.12 Юнит тестирование | 57 |
| 2.13 Аудирование | 60 |
| 2.14 Рефакторинг | 63 |
| Заключение..... | 76 |
| Список используемых источников | 77 |
| Приложение А (справочное) Основные вкладки | 78 |
| Приложение Б (справочное) Промежуточные результаты тест-кейсов..... | 80 |
| Приложение В (справочное) Листнинг программы | 83 |

ВВЕДЕНИЕ

Банк (от итал. *banco* — скамья, лавка, стол, на которых менялы раскладывали монеты) — финансово-кредитная организация, производящая разнообразные виды операций с деньгами и ценными бумагами и оказывающая финансовые услуги правительству, юридическим и физическим лицам. Банк — это коммерческое юридическое лицо, которое:

- создано в целях извлечения прибыли,
- имеет право осуществлять банковские операции,
- имеет *исключительное* право на привлечение денежных средств юридических и физических лиц с целью их последующего размещения от своего имени; а также на открытие и ведение банковских счетов юридических и физических лиц,
- действует на основании специального разрешения (лицензии) полномочных государственных органов,
- не имеет права осуществлять производственную, торговую, страховую деятельность.

Основными функциями банка являются:

- Исторически первой функцией банков было безопасное хранение денег клиентов.
- Так как у банка есть много клиентов, которые хранят в нем свои деньги, то банк становится способен переводить деньги от одного из них другому путём изменения записей в банковских счетах (безналичные расчеты). Безналичные расчеты возможны и между клиентами различных банков благодаря системе корреспондентских счетов.
- Банки выдают кредиты. При этом фактически создаётся дополнительная денежная масса (см. Банковский мультипликатор).

Существенные особенности банковской деятельности (отличающие её от производственной, торговой и др.):

- Преобладание в ресурсах банков привлечённых и заёмных средств, влекущее за собой повышенную ответственность перед кредиторами и вкладчиками.
- Чрезвычайная подвижность и изменчивость параметров финансовых рынков, вызываемая не только экономическими, но и политическими, социальными и иными причинами.

- Необходимость постоянной и одновременной работы с клиентами, представляющими различные сферы и отрасли хозяйствования, имеющими противоречивые интересы и цели.
- Нематериальный характер банковских продуктов (услуг) и необходимость участия практически всех подразделений банка в производстве каждого такого продукта.

В последнее время информационные технологии стали неотъемлемой частью нашей жизни. Экономические информационные системы, связанные с предоставлением и обработкой информации для всех уровней управления экономическими объектами, приобретают особую важность в общественной жизни. На данный момент невозможно представить какую-либо организацию, не применяющую компьютерных технологий. Это обусловлено и тем, что государственные структуры требуют обязательных отчетов в электронном виде, следовательно, необходима систематизированная информация.

Целью данной курсовой работы является создание электронной системы, которая позволит улучшить эффективность выполнения процессов, происходящих в банке при процессе кредитования.

Основными задачами данной работы являются:

- изучить теоретические особенности моделирования процессов организации – произвести исследование предметной области – банковской деятельности
- на основании полученных знаний разработать программу для банка.

Объектом исследования является банк.

Предметом исследования являются процессы, происходящие в банке, такие как добавление клиентов, кредитов, кредитных карт.

1 Теоретические основы разработки

1.1 Описание предметной области

Банки являются важным звеном в функционировании экономики страны и мировой экономики. Банковская система обеспечивает бесперебойность и высокую скорость процесса общественного воспроизводства, повышает эффективность всего производства.

Основное призвание банка - посредничество в перемещении денежных средств от кредиторов к заёмщикам и от продавцов к покупателям. Наряду с банками перемещение денежных средств на рынке осуществляют и другие финансовые и кредитно-финансовые учреждения: инвестиционные фонды, страховые компании, брокерские, дилерские фирмы и т.д. Но банки как субъекты финансового рынка имеют два существенных признака, отличающие их от всех других субъектов.

Современные банковские системы ориентированы на использование единой базы данных, являющейся совокупностью структурированных данных, предназначенных для многоцелевого и многократного их применения, и методов доступа к ним.

Отличительной особенностью баз данных информационные банковские технологии (ИБТ) является совместное хранение данных с их описаниями. Эти описания называются метаданными (данные о данных). Они необходимы для контроля и управления данными как ресурсом.

Существует несколько уровней программного обеспечения: операционная система, СУБД, прикладные программы, каждый из которых решает свои специфические задачи. Например, СУБД обеспечивает интерфейс доступа к данным на чтение и запись, блокирование записей в режиме многопользовательской работы, предоставляет средства архивации, восстановления и резервного копирования данных. Наиболее распространенным интерфейсом доступа к данным являются SQL_запросы, которые анализируются на СУБД, и различные программные средства, которые исполняют эти запросы.

Имеется большое количество СУБД, которые используются при построении ИБТ. Все они поддерживают реляционную модель данных, но имеют различные эксплуатационные характеристики. Потенциал программного продукта зависит от применяемой в нем СУБД и от степени использования ключевых свойств СУБД. Важно, что на программный продукт нельзя переносить свойства СУБД, так как

реализация продукта с аналогичным свойством требует специальных усилий со стороны разработчика прикладного программного обеспечения.

Следует различать единую базу данных и единое информационное пространство. Под единым информационным пространством понимается возможность вызывать функции других подсистем, а также общность данных и методов доступа к ним в системе. Сама по себе СУБД не обеспечивает ведение единого информационного пространства, но позволяет его реализовать и использовать с максимально возможной эффективностью.

Под единым информационным пространством банка можно понимать также организацию информации, циркулирующей в банке, включая методы ее обработки, хранения и представления. На уровне автоматизированной банковской системы единое информационное пространство можно интерпретировать, как возможность системы оперировать любыми данными, формирующимися в процессе функционирования системы. При этом должны соблюдаться принципы открытости, защищенности, однократного учета и ввода. Таким образом, реализация единого информационного пространства банковской технологией обеспечивает эффективную организацию работ с информацией, как с точки зрения быстродействия, так и в аспекте удобства работы пользователя с данными.

1.1 Анкетирование

Заинтересованные лица

Кто будет использовать систему:

- 1. руководитель информационно-аналитического центра коммерческого банка*
- 2. клиент (заемщик)*
- 3. рядовой сотрудник банка*

Кто является заказчиком: *владелец банка.*

Обслуживающий персонал: *системный администратор.*

Организация: *коммерческий банк.*

Руководитель информационно-аналитического центра коммерческого банка

| | |
|--------|-------|
| Вопрос | Ответ |
|--------|-------|

| | |
|---|--|
| 1.Имя | <i>Тимушев Федор Алексеевич</i> |
| 2.Наименование структурного подразделения | <i>Коммерческий банк</i> |
| 3.Должность | <i>Руководитель информационно-аналитического центра коммерческого банка</i> |
| 4.Кому подчиняетесь | <i>Коммерческому директору</i> |
| 5.Обязанности | <ul style="list-style-type: none"> • <i>Руководить деятельностью аналитического отдела, в том числе планировать, организовывать и контролировать ее; прогнозировать развитие направлений деятельности аналитического отдела.</i> • <i>Обеспечивать подготовку аналитических материалов и их предоставление руководителю организации.</i> • <i>Разрабатывать и вносить руководителю организации на утверждение положение об аналитическом отделе.</i> • <i>Готовить текущие и перспективные планы работы аналитического отдела, вносить их на утверждение руководителя организации и осуществлять контроль за ходом выполнения утвержденных планов.</i> • <i>Готовить вопросы для рассмотрения на заседаниях аналитического отдела.</i> • <i>Вносить предложения по структуре и штатам, кадровому обеспечению аналитического отдела,ощрению работников аналитического отдела и наложению на них дисциплинарных взысканий.</i> • <i>Разрабатывать проекты должностных регламентов работников аналитического отдела, готовить предложения по распределению обязанностей между ними, давать им обязательные для исполнения задания, поручения, указания и распоряжения, контролировать их исполнение.</i> • <i>Визировать подготовленные работниками аналитического отдела материалы, подписывать служебную документацию в пределах компетенции аналитического отдела.</i> • <i>Соблюдать, а также осуществлять контроль за соблюдением работниками аналитического отдела должностных регламентов, порядка работы со служебной информацией, рабочего распорядка организации, правил делопроизводства.</i> • <i>Изучать и анализировать практику применения действующего законодательства Российской Федерации, мнения ведущих ученых и опыт субъектов Российской Федерации по вопросам, относящимся к компетенции аналитического отдела, готовить соответствующие предложения руководителю организации.</i> • <i>Обобщать, проводить анализ информации, справок и других сообщений работников аналитического отдела, на их основе готовить предложения руководителю организации по вопросам, относящимся к компетенции аналитического отдела.</i> |

| | | |
|--|--|---|
| | | <ul style="list-style-type: none"> Организовывать эффективное взаимодействие аналитического отдела с другими структурными подразделениями организации. |
| 6. Что является результатом работы (документы) | | <ul style="list-style-type: none"> анализ деловой информации в отношении объектов исследования, а также иной информации, предназначенной для удовлетворения информационных потребностей участников рынка, при осуществлении ими предпринимательской деятельности предоставление справок и аналитических записок в электронном виде по запросам Заказчика, на основании информации и совокупности данных медиамониторинга анализ итогов финансово-хозяйственной деятельности юридических лиц. Показатели финансового состояния (ликвидности, платежеспособности, обеспеченности, рентабельности и др.) юридического лица. Аналитические отчеты могут быть представлены в виде таблиц, графиков и диаграмм в электронном виде |
| 7. Необходимые данные или документы | | <p>Для государственной регистрации кредитной организации и получения лицензии на осуществление банковских операций в Банк России в установленном им порядке представляются следующие документы:</p> <ul style="list-style-type: none"> 1) заявление с ходатайством о государственной регистрации кредитной организации и выдаче лицензии на осуществление банковских операций; в заявлении также указываются сведения об адресе (месте нахождения) постоянно действующего исполнительного органа кредитной организации, по которому осуществляется связь с кредитной организацией; 2) учредительный договор (подлинник или нотариально удостоверенная копия), если его подписание предусмотрено федеральным законом; 3) устав (подлинник или нотариально удостоверенная копия); 4) бизнес-план, утвержденный собранием учредителей (участников) кредитной организации, протокол собрания учредителей (участников), содержащий решения об утверждении устава кредитной организации, а также кандидатур для назначения на должности руководителя кредитной организации и главного бухгалтера кредитной организации. Порядок составления бизнес-плана кредитной организации и критерии его оценки устанавливаются нормативными актами Банка России; 5) документы об уплате государственной пошлины за государственную регистрацию кредитной организации и за предоставление лицензии на осуществление банковских операций при создании кредитной организации; 6) документы (согласно перечню, установленному |

| | |
|--|--|
| | <p>нормативным актом Банка России) для оценки финансового положения физических или юридических лиц - учредителей (участников) кредитной организации, физических или юридических лиц, осуществляющих контроль в отношении учредителей (участников) кредитной организации, приобретающих более 10 процентов акций (долей) кредитной организации, физических или юридических лиц, осуществляющих контроль в отношении учредителей (участников) кредитной организации, приобретающих 10 и менее процентов акций (долей) кредитной организации, входящих в состав группы лиц, приобретающей более 10 процентов акций (долей) кредитной организации, в том числе аудиторские заключения о бухгалтерской (финансовой) отчетности учредителей (участников) - юридических лиц; (п. 6 в ред. Федерального закона от 29.07.2017 N 281-ФЗ) (см. текст в предыдущей редакции)</p> <ul style="list-style-type: none"> • 7) документы (согласно перечню, установленному нормативным актом Банка России), подтверждающие источники происхождения средств, вносимых учредителями (участниками) - физическими лицами в уставный капитал кредитной организации; (в ред. Федерального закона от 29.07.2017 N 281-ФЗ) (см. текст в предыдущей редакции) • 8) анкеты кандидатов на должности руководителя кредитной организации, главного бухгалтера, заместителей главного бухгалтера кредитной организации, руководителя, главного бухгалтера филиала кредитной организации (за исключением небанковской кредитной организации, имеющей право на осуществление переводов денежных средств без открытия банковских счетов и связанных с ними иных банковских операций). Указанные анкеты заполняются этими кандидатами собственноручно или с применением технических средств, подписываются ими собственноручно и должны содержать сведения, установленные нормативными актами Банка России, а также сведения: о наличии у кандидата на должность руководителя кредитной организации и руководителя филиала кредитной организации высшего образования (бакалавриат, специалитет, магистратура, подготовка кадров высшей квалификации) (далее - высшее образование) с представлением копии документа об образовании и о квалификации, а также о наличии опыта руководства кредитными организациями, отделом или иным подразделением кредитной организации, связанным с осуществлением банковских операций, опыта руководства лицами, признаваемыми в соответствии со статьей 76.1 Федерального закона "О Центральном банке Российской Федерации (Банке России)" некредитными финансовыми организациями (за |
|--|--|

| | |
|--|--|
| | <p>исключением ломбардов), или опыта работы на руководящих должностях в органах государственной власти Российской Федерации, органах государственной власти субъектов Российской Федерации, Банке России не менее двух лет;</p> <p>о наличии у кандидата на должность главного бухгалтера, заместителя главного бухгалтера кредитной организации и главного бухгалтера филиала кредитной организации высшего образования (с представлением копии документа об образовании и о квалификации) и опыта работы, связанной с ведением бухгалтерского учета, составлением бухгалтерской (финансовой) отчетности либо с аудиторской деятельностью в финансовой организации, не менее одного года;</p> <p>о наличии (об отсутствии) судимости (с представлением оригинала справки о наличии (об отсутствии) судимости, выданной Министерством внутренних дел Российской Федерации);</p> <p>(п. 8 в ред. Федерального закона от 29.07.2017 N 281-ФЗ) (см. текст в предыдущей редакции)</p> <ul style="list-style-type: none"> <p>9) анкеты кандидатов на должности единоличного исполнительного органа и главного бухгалтера небанковской кредитной организации, имеющей право на осуществление переводов денежных средств без открытия банковских счетов и связанных с ними иных банковских операций. Указанные анкеты заполняются этими кандидатами собственноручно или с применением технических средств, подписываются ими собственноручно и должны содержать сведения, установленные нормативными актами Банка России, а также сведения:</p> <p>(в ред. Федерального закона от 29.07.2017 N 281-ФЗ) (см. текст в предыдущей редакции)</p> <p>о наличии у этих лиц высшего образования (с представлением копии документа об образовании и о квалификации);</p> <p>о наличии (об отсутствии) судимости (с представлением оригинала справки о наличии (об отсутствии) судимости, выданной Министерством внутренних дел Российской Федерации);</p> <p>10) документы (согласно перечню, установленному нормативными актами Банка России), необходимые для оценки деловой репутации:</p> <p>лиц, указанных в частях четвертой, седьмой - девятой статьи 11.1 и части второй статьи 11.1-2 настоящего Федерального закона;</p> <p>физических или юридических лиц - учредителей (участников), приобретающих более 10 процентов акций (долей) кредитной организации, и физических или юридических лиц, осуществляющих контроль в отношении указанных юридических лиц;</p> <p>физических или юридических лиц - учредителей</p> |
|--|--|

| | |
|--|---|
| | <p>(участников), приобретающих 10 и менее процентов акций (долей) кредитной организации и входящих в состав группы лиц, приобретающей более 10 процентов акций (долей) кредитной организации, и физических или юридических лиц, осуществляющих контроль в отношении указанных юридических лиц; лиц, осуществляющих функции единоличного исполнительного органа указанных юридических лиц; а также иных лиц, указанных в части четвертой статьи 16 настоящего Федерального закона. (п. 10 в ред. Федерального закона от 29.07.2017 N 281-ФЗ)</p> <p>(см. текст в предыдущей редакции)</p> <p>Помимо документов, указанных в части первой настоящей статьи, Центральный банк Российской Федерации самостоятельно запрашивает в федеральном органе исполнительной власти, осуществляющем государственную регистрацию юридических лиц, физических лиц в качестве индивидуальных предпринимателей и крестьянских (фермерских) хозяйств, сведения о государственной регистрации юридических лиц, являющихся учредителями кредитной организации, а в налоговом органе запрашивает сведения о выполнении учредителями - юридическими лицами обязательств перед федеральным бюджетом, бюджетами субъектов Российской Федерации и местными бюджетами за последние три года. Кредитная организация вправе представить документы, содержащие указанные сведения, по собственной инициативе.</p> |
| 9.Проблемы, влияющие на успешность выполнения работы | <ul style="list-style-type: none"> ● Излишняя документация для заполнения ● Потери в сфере документирования ● Некомпетентность работников ● Финансовые излишки от перезаключения договоров ● Большое кол-во проверок физических и юридических лиц |
| 10.Потребности в разрабатываемом ПО | <ul style="list-style-type: none"> ● Разрабатываемая система должна быть максимально эргономичной, работать стабильно (без сбоев); ● отклик системы не должен вызывать у пользователей раздражения; ● реализуемая функциональность должна полностью удовлетворить потребности пользователя ● ПО должно иметь возможность заполнения данных о клиенте ● система должна точно высчитывать условия кредитования на основе введенных данных ● программа должна иметь расчет по кредитованию |

Клиент (заемщик)

| Вопрос | Ответ |
|---|---|
| 1.Имя | <i>Тимушев Федор Алексеевич 2</i> |
| 2.Удобна ли вам нынешняя система оформления кредитов | <i>Нет</i> |
| 3.Испытываете ли вы трудности с оформлением документов | <i>Их слишком много. Происходят постоянные потери</i> |
| 4.Отвечает ли персонал банка на все вопросы связанные с оформлением | <i>Не всегда</i> |
| 5.Бывали ли случаи оформления на иных условиях? Не на тех, что были оговорены ранее | <i>Да, как то оформили процент выше из-за ошибки в данных о получаемых мною соц. выплат</i> |
| 6.Как долго занимает процесс оформления | <i>Около 2ух недель</i> |

Назначение и цели создания системы:

1. программа должна обеспечивать ввод данных о клиенте
2. программа должна иметь привязку к базе данных банка для экономии времени при оформлении
3. программа должна выдавать печатную документацию о возможных тарифах
4. программа должна выдавать печатную версию реквизитов клиента
5. программа должна осуществлять гашение кредита
6. программа должна осуществлять слежку за просроченными платежами
7. программа должна рассчитывать следующие данные по кредиту:
 - a. полную сумму
 - b. итоговый процент
 - c. платеж в месяц
 - d. время гашения
 - e. стоимость переплаты

1.1.1 Выделение опорных точек зрения на основе анкетирования представлено на рисунке 1

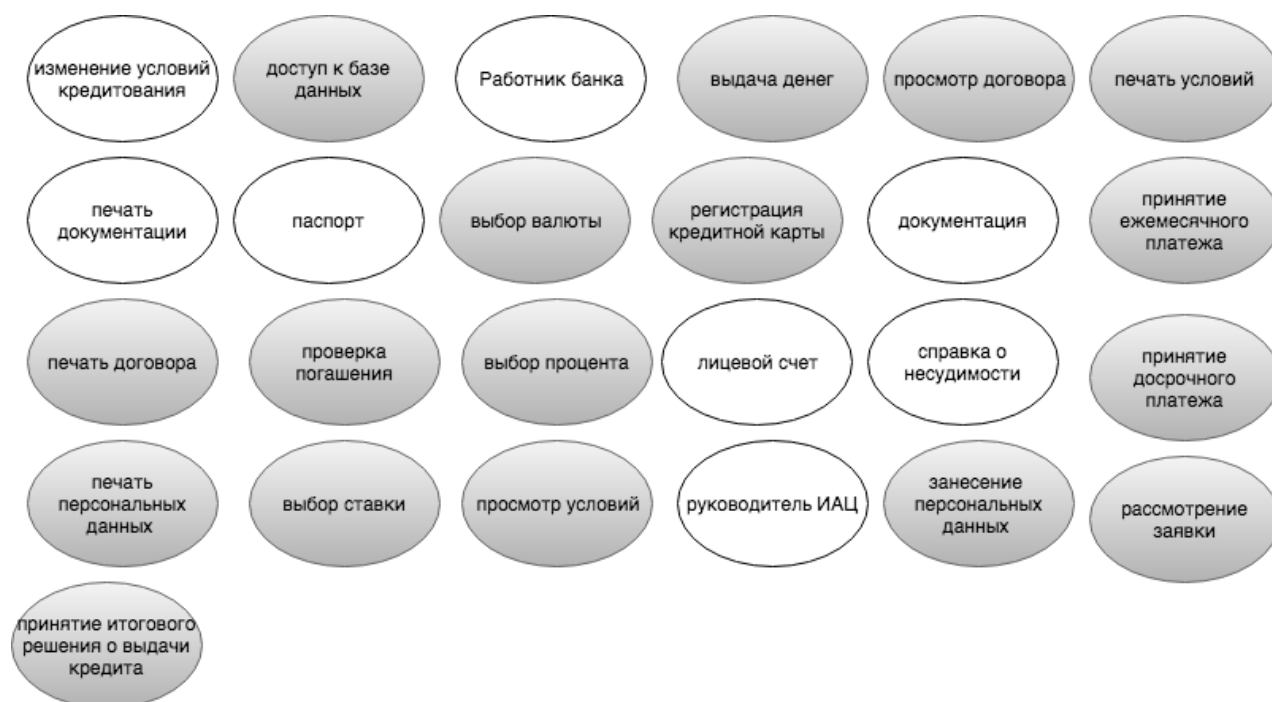


Рисунок 1 – Идентификация точек зрения

Таблица 1 – Сервичы, соотнесенные с точками зрения

| Работник банка | Руководитель |
|-------------------------------|---|
| просмотр договора | |
| рассмотрение заявки | |
| занесение персональных данных | принятие итогового решения о выдаче кредита |
| выдача денег | |
| печать условий | |
| выбор валюты | |
| регистрация кредитной карты | |
| принятие ежемесячного платежа | |
| печать договора | |
| проверка погашения | |
| выбор процента | |
| принятие досрочного платежа | |

| | |
|----------------------------|--|
| печать персональных данных | |
| выбор ставки | |
| просмотр условий | |

1.1.2 Разработка ведётся на основании задания к курсовому проекту по профессиональному модулю ПМ.03 «Участие в интеграции программных модулей» и утверждена Университетским политехническим колледжем.

1.1.3 Программный продукт «Выдача банком кредитов» автоматизирует деятельности банков в сфере кредитования. Программа позволяет рядовому работнику банка полноценно управлять базой данных клиентов, присвоенных им кредитных записей и кредитных карт, отслеживание погашения суммы займа.

1.1.4 Требования к программе

1.1.4.1 К функциональным требованиям относятся

- занесение персональных данных;
- выбор процентной ставки;
- выбор валюты;
- регистрация кредитной карты;
- принятие итогового решения о выдачи кредита;
- способность программы обеспечивать успешное добавление кредитных записей в базу данных;
- проверка погашения;
- принятие ежемесячного платежа;
- принятие досрочного платежа;
- просмотр условий;
- печать условий;
- просмотр договора;
- печать договора.

1.1.4.2 К требованиям надежности относятся:

- использование лицензированного программного обеспечения;
- проверка программы на наличие вирусов;

- организация бесперебойного питания.

1.1.4.3 Программа должна эксплуатироваться в условиях вычислительного центра при температуре от +15 до +35 С и относительной влажности воздуха от 25 до 85%.

1.1.4.4 Для полноценного функционирования данной системы необходимо наличие операционной системы выше Microsoft Windows 7 или совместимой. Язык интерфейса – английский, SQL Server 2012 и выше, SQL Management Studio 2012 и выше.

1.1.4.5 Программа должна поставляться на диске в виде исполняемого (exe) файла, документации и проекта. На диске должна быть наклейка с надписью "ИС Библиотеки". Диск должен быть упакован в пластиковую коробку.

1.1.4.6 Диск с программой должен храниться вдали от электромагнитных полей и не подвергаться механической деформации. Место и условия хранения должны соответствовать санитарным требованиям отрасли. Сроки хранения устанавливаются в соответствии с гарантийными сроками поставщика магнитных носителей

1.2 Анализ методов решения

Жанр программного продукта «Выдача кредитов банком» можно определить как обучающий. Программа может быть реализована следующими способами:

- Автоматизированный способ: исходные данные записываются в базу данных, а далее работник банка взаимодействует с ней (добавляет клиентов, кредитные записи, кредитные карты, изменяет и фильтрует данные).

- Автоматический способ: данные автоматически считываются с внешнего устройства (например, с файла на диске), обрабатываются программой и позволяют работнику банка выполнять прочее взаимодействие с БД.

Автоматический и ручной способы не могут быть использованы, так как программный продукт не будет соответствовать условиям, заданным при создании программы.

В нашем случае мы рассмотрим автоматизированный способ.

1.3 Обзор средств программирования

Средством программирования данного курсового проекта является среда разработки C#. На сегодняшний момент язык программирования C# один из самых мощных, быстро развивающихся и востребованных языков в ИТ-отрасли. В настоящий момент на нем пишутся самые различные приложения: от небольших десктопных программ до крупных веб-порталов и веб-сервисов, обслуживающих ежедневно миллионы пользователей.

По сравнению с другими языками C# достаточно молодой, но в то же время он уже прошел большой путь. Первая версия языка вышла вместе с релизом Microsoft Visual Studio .NET в феврале 2002 года. Текущей версией языка является версия C# 6.0, которая вышла в 20 июля 2015 года вместе с Visual Studio 2015.

C# является языком с Си-подобным синтаксисом и близок в этом отношении к C++ и Java. Поэтому, если вы знакомы с одним из этих языков, то овладеть C# будет легче.

C# является объектно-ориентированным и в этом плане много перенял у Java и C++. Например, C# поддерживает полиморфизм, наследование, перегрузку операторов, статическую типизацию. Объектно-ориентированный подход позволяет решить задачи по построению крупных, но в тоже время гибких, масштабируемых и расширяемых приложений. И C# продолжает активно развиваться, и с каждой новой версией появляется все больше интересных функциональностей, как, например, лямбды, динамическое связывание, асинхронные методы и т.д.

Алфавит языка C#. Все тексты на языке пишутся с помощью его алфавита. В C# используется кодировка символов Unicode. Кодировкой, или кодовой таблицей (character set), называется соответствие между символами и кодирующими их числами. Кодировка Unicode позволяет представить символы всех существующих алфавитов одновременно. Каждому символу соответствует свой уникальный код.

Алфавит C# включает:

- буквы (латинские и национальных алфавитов) и символ подчеркивания (_), который употребляется наряду с буквами.
- цифры.
- специальные символы, например +, *, { и &.
- пробельные символы (пробел и символы табуляции).
- символы перевода строки.

Из символов составляются более крупные строительные блоки: лексемы, директивы препроцессора и комментарии.

Лексема (token) - это минимальная единица языка, имеющая самостоятельный смысл. Существуют следующие виды лексем:

- имена (идентификаторы).
- ключевые слова.
- знаки операций.
- разделители.
- литералы (константы).

Лексемы языка программирования аналогичны словам естественного языка. Например, лексемами являются число 128 (но не его часть 12), имя Vasia, ключевое слово goto и знак операции сложения +. Далее мы рассмотрим лексемы подробнее.

Директивы препроцессора пришли в C# из его предшественника - языка C++. Препроцессором называется предварительная стадия компиляции, на которой формируется окончательный вид исходного текста программы. Например, с помощью директив (инструкций, команд) препроцессора можно включить или выключить из процесса компиляции фрагменты кода. Директивы препроцессора не играют в C# такой важной роли, как в C++.

Комментарии предназначены для записи пояснений к программе и формирования документации. Правила записи комментариев мы рассмотрим чуть позже.

Из лексем составляются выражения и операторы. Выражение задает правило вычисления некоторого значения. Например, выражение $a + b$ задает правило вычисления суммы двух величин.

Операторы языка C#. Любое выражение, завершающееся точкой с запятой, рассматривается как оператор, выполнение которого заключается в вычислении выражения. Частным случаем выражения является пустой оператор; (он используется, когда по синтаксису оператор требуется, а по смыслу - нет).

Блок, или составной оператор, - это последовательность описаний и операторов, заключенная в фигурные скобки. Блок воспринимается компилятором как один оператор и может использоваться всюду, где синтаксис требует одного оператора, а алгоритм - нескольких. Блок может содержать один оператор или быть пустым.

Условный оператор if используется для разветвления процесса вычислений на два направления.

Оператор switch (переключатель) предназначен для разветвления процесса вычислений на несколько направлений.

Операторы цикла используются для вычислений, повторяющихся многократно. Блок, ради выполнения которого и организуется цикл, называется телом цикла. Остальные операторы служат для управления процессом повторения вычислений: это начальные установки, проверка условия продолжения цикла и модификация параметра цикла. Один проход цикла называется итерацией.

Начальные установки служат для того, чтобы до входа в цикл задать значения переменных, которые в нем используются.

Проверка условия продолжения цикла выполняется на каждой итерации либо до тела цикла (тогда говорят о цикле с предусловием), либо после тела цикла (цикл с постусловием).

Параметром цикла называется переменная, которая используется при проверке условия продолжения цикла и принудительно изменяется на каждой итерации, причем, как правило, на одну и ту же величину. Если параметр цикла целочисленный, он называется счетчиком цикла.

Цикл завершается, если условие его продолжения не выполняется. Возможно принудительное завершение как текущей итерации, так и цикла в целом. Для этого служат операторы break, continue, return и goto. Передавать управление извне внутрь цикла запрещается.

2 Практическая часть

2.1 Разработка программной части

2.1.1 Программный продукт «Выдача банком кредитов» автоматизирует деятельности банков в сфере кредитования. Программа позволяет рядовому работнику банка полноценно управлять базой данных клиентов, присвоенных им кредитных записей и кредитных карт, отслеживание погашения суммы займа.

2.1.2 К программной документации относятся:

- техническое задание;
- разрабатываемые программные модули должны быть самодокументированы, т.е. тексты программ должны содержать все необходимые комментарии;
- разрабатываемое программное обеспечение должно включать справочную систему.

2.1.3 Каждый вариант использования определяет последовательность действий, которые должны быть выполнены проектируемой системой при взаимодействии ее с соответствующим актером. USE-CASE Diagram представлена на рисунке 2



Рисунок 2 – Use-case диаграмма

2.1.4 Диаграмма активности представляет по существу обычную блок-схему. На ней показываются деятельности - шаги в выполнении процесса, изображаемые в виде прямоугольников с сопряженными дугами горизонтальными сторонами и

переходы между ними, показываемые стрелками. Предусмотрена возможность ветвления, изображаемая в виде ромба. На этих диаграммах можно показать распараллеливание процесса на подпроцессы и слияние подпроцессов. Для обозначения этих действий используются жирные горизонтальные или вертикальные линии. Все элементы могут быть проименованы. С помощью диаграмм деятельности удобно представлять алгоритмы выполнения работ. В частности, использование ветвления дает возможность легко отобразить основной и альтернативные потоки. Диаграмма активности данной информационной системы представлена на рисунке 3



Text

Рисунок 3 – Диаграмма активности

2.1.5 Диаграмма последовательности отображает взаимодействие объектов - экземпляров классов во времени, выраженное в обмене сообщениями. При этом возможные статические ассоциации между объектами не показываются. Для диаграммы последовательности ключевым моментом является именно динамика взаимодействия объектов во времени (вертикальная временная ось направлена сверху вниз). Линия жизни объекта, изображаемая в виде вертикальной линии, служит для обозначения периода времени, в течение которого объект активен, то есть участвует во взаимодействии. Диаграмма последовательностей представлена на рисунке 4.

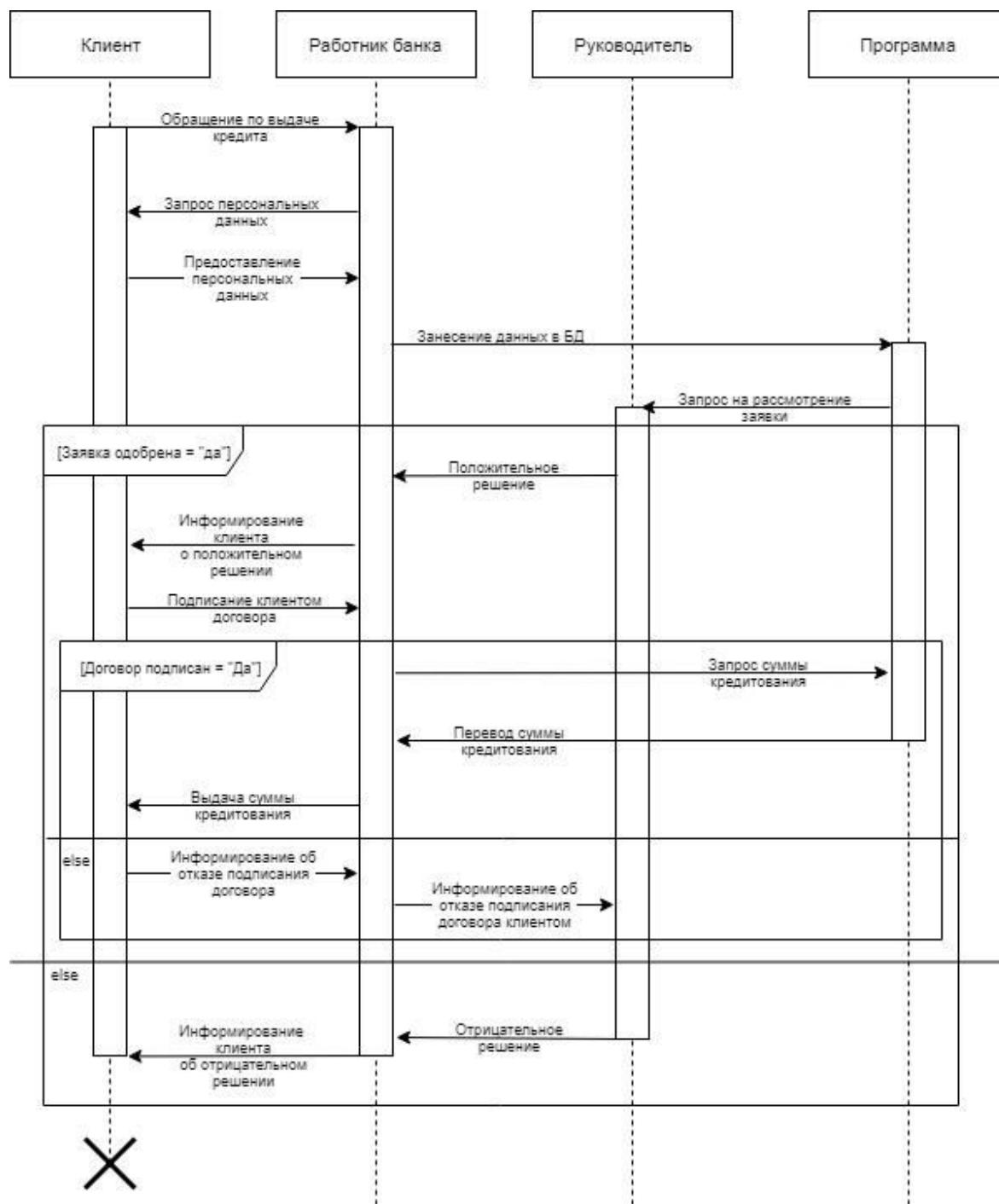


Рисунок 4 – Диаграмма последовательности

2.1.6 База данных была создана в среде SQL Management Studio 2017 (рисунок 5)

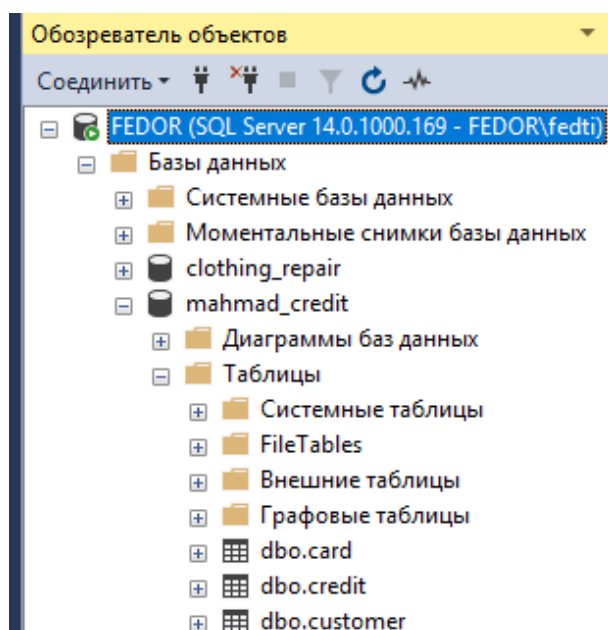


Рисунок 5 – База данных, созданная в SQL Management Studio 17

Сущность «Card» предназначена для хранения данных о кредитной карте для каждого конкретного заемщика.

Сущность «Customer» предназначена для хранения информации о каждом клиенте.

Сущность «Credit» предназначена для хранения информации о каждой кредитной записи.

Схема данной БД представлена на рисунке 6.

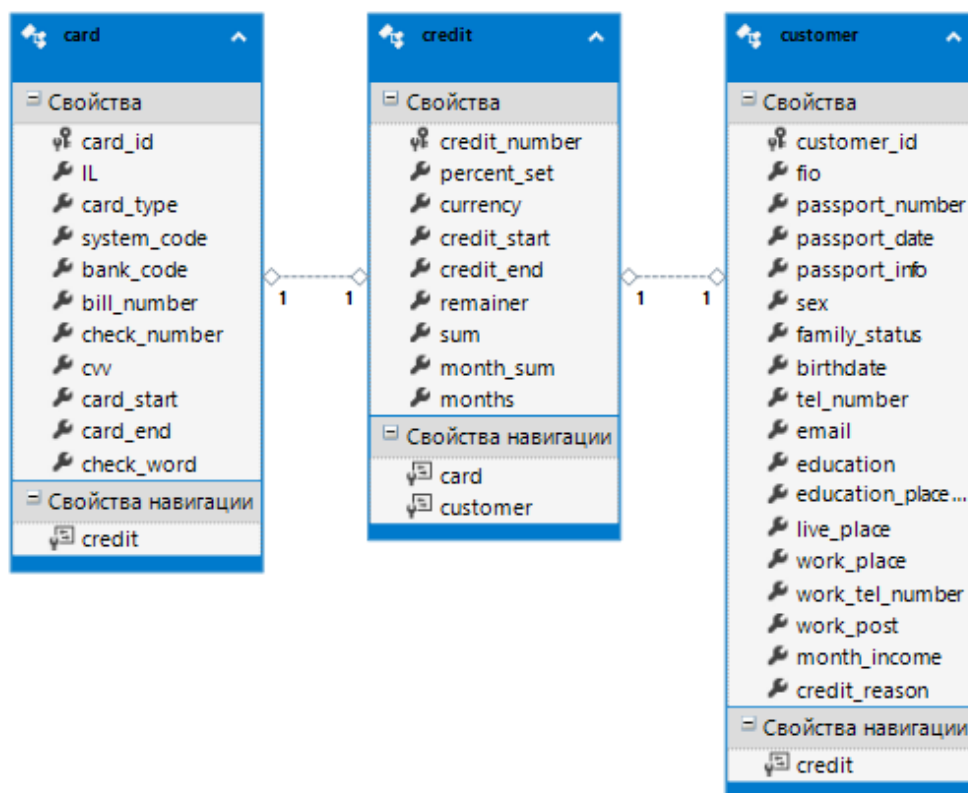


Рисунок 6 – Схема базы данных

В каждой таблице БД должно быть определено ключевое поле. Этот, так называемый первичный ключ, обеспечивает уникальность записей в таблице, препятствуя вводу повторяющихся данных. С помощью ключа СУБД обеспечивает быстрый поиск требуемых данных, их хранение и обработка.

Таблица Card (карта) связана с таблицей Credit (кредит) связью «один-к-одному», поскольку одной кредитной записи можно соотнести одну кредитную карту, выданную по этому кредиту.

Таблица Customer (клиент) связана с таблицей Credit (кредит) также связью «один-к-одному», поскольку в банке один конкретный кредит выдается на одно конкретное физическое лицо.

2.2 Текст программы

Текст программы в соответствии с ГОСТ 19.101-77 (СТ СЭВ 1626-79) и ГОСТ 19.401-79 (СТ СЭВ 3746-82) представляет собой запись программы на исходном языке программирования с необходимыми комментариями. Текст программы представляет собой документ, выполненный машинным способом, и приведен в приложении «В».

2.2.1 Описание программы

2.2.1.1 С помощью данной программы сотрудник банка вносит в базу данных информацию об клиентах (заемщиках), кредитных записях, кредитных картах - изменяет, удаляет и фильтрует их. А также выполняет процессы выдачи и списания кредитов.

2.2.1.2 Программа содержит 6 модулей: MainWindow – главное окно программы, Clients – форма выбора клиентов, класс date, рассчитывающий количество месяцев кредитования, класс Observe, необходимый для автоматического обновления таблиц и списков после редактирования, добавления, удаления из них данных, класс Itemsourse, предназначенный для синхронизации таблиц с базой данных, класс Rem, рассчитывающий итоговую сумму кредитования.

При запуске работы приложения загружается главная форма, на которой появляются:

- поля для заполнения данными о заемщике и кредите
- кнопка «Выбрать клиента»
- кнопка «Перейти к регистрации кредитной карты -> »

Программа использует функции следующих библиотек среды C#:

System, System.Collections.Generic, System.Data, System.Data.SqlClient, System.Linq, System.Text, System.Threading.Tasks, System.Windows, System.Windows.Controls, System.Windows.Data, System.Windows.Documents, System.Windows.Input, System.Windows.Media, System.Windows.Media.Imaging, System.Windows.Navigation, System.Windows.Shapes, System.Data.Entity, Interface.Models, Interface.Classes, System.Collections.ObjectModel, System.IO;

2.2.1.3 Программа может быть загружена как с диска, так и с жесткого диска. В последнем случае требуется предварительно переписать программу с диска на жесткий диск. Исполняемым файлом программы является файл interYOURMOMface.exe. Для его запуска необходимо дважды щелкнуть по исполняемому файлу левой кнопкой мышки.

2.3 Руководство оператора

2.3.1 Назначение программы

2.3.1.1 Основной функцией программы «Выдача банком кредитов» является автоматизирование деятельности банков в сфере кредитования.

Программа позволяет рядовому работнику банка полноценно управлять базой данных клиентов, присвоенных им кредитных запись и кредитных карт, отслеживание погашения суммы займа. Также, к функциональным характеристикам относится:

- добавление клиентов, кредитных записей, кредитных карт;
- создание списка кредитных записей;
- поиск клиентов;
- просмотр всех кредитных записей;
- просмотр и печать договора и условий.

2.3.1.2 Основное назначение программы «Выдача банком кредитов» - ускорить выдачу кредита клиенту, заполнение базы данных, регистрацию кредитной карты, поиск клиентов.

2.3.2 Условия выполнения программы

2.3.2.1 Минимальный состав используемых технических (аппаратных) средств:

- PC, совместимый с процессором 80386 и выше;
- ОЗУ более 32 Мбайт;
- 16 МБ видеопамяти и выше;
- наличие свободного места на жестком диске более 500 Мбайт.

2.3.2.2 Системные программные средства, используемые программой «Библиотека», должны быть представлены локализованной версией операционной системы Windows XP или выше. Также для функционирования программы «Библиотека» на компьютере должны быть установлены следующие программы:

- MS SQL Server Management Studio версии 12 и выше
- SQL Server версии 12 и выше

2.3.2.3 Сотрудник банка должен обладать практическими навыками работы с графическим пользовательским интерфейсом операционной системы, обладать минимальными знания в работе с SQL Server'ом и MS SQL Management Studio.

2.3.3 Выполнение программы

2.3.3.1 Загрузка и запуск программы осуществляется двойным кликом по ярлыку программы с названием «interYOURMOMface».

В случае успешного запуска программы на рабочем столе будет отображено главное окно программы (показано на рисунке 7).

The screenshot shows the 'Make credit 1.0' application window. The title bar includes the application name and standard window controls. The menu bar contains 'Регистрация клиента в кредитной системе', 'Регистрация кредитной карты', and 'Погаше'. The main area is divided into two panels: 'Данные о клиенте:' and 'Данные о кредите:'. The 'Данные о клиенте:' panel contains fields for client information, including FIO, passport details, date of birth, gender, family status, education, department code, workplace, position, company phone, monthly income, place of residence, contact phone, and email. The 'Данные о кредите:' panel contains fields for credit details, including interest rate, currency, credit period, purpose of credit, loan amount, credit term, monthly payment, and total payment. A blue button at the bottom right of the credit panel reads 'Перейти к регистрации кредитной карты ➔'.

| Данные о клиенте: | Данные о кредите: |
|---|--|
| ФИО клиента: _____ | Процентная ставка: _____ |
| Паспортные данные: 0000 000000 _____ кем выдан _____ | Валюта: _____ |
| Дата рождения: _____ | Период кредитования: с _____ по: _____ |
| Пол: <input type="radio"/> М <input type="radio"/> Ж | Цель получения кредита: адрес _____ |
| Семейное положение: _____ | Желаемая сумма займа: 150000 _____ |
| Образование: _____ | Кол-во месяцев кредитования: _____ |
| Код подразделения: _____ | Ежемесячное погашение: _____ |
| Место работы: _____ | Итого к погашению: _____ |
| Должность: должность _____ | |
| Телефон предприятия: 88121234567 _____ | |
| Ежемесячные доходы на месте работы: доходы _____ | |
| Место проживания (регистрации): адрес _____ | |
| Контактный телефон: 89111234567 _____ | |
| Эл. почта: email _____ | |

Перейти к регистрации кредитной карты ➔

Рисунок 7 – Главное окно программы

2.3.3.2 После открытия главной страницы будут доступны для ввода поля, которые необходимо заполнить соответствующей информацией о клиенте (см. рис. 8), а также о б оформляющемся на него кредите (см. рис. 9). В окне «Данные о кредите», поля «Кол-во месяцев кредитования», «Ежемесячное погашение», «Итого к погашению» высчитываются автоматически по щелчку мыши по ним соответственно.

Make credit 1.0

Регистрация клиента в кредитной системе | Регистрация кредитной карты | Погаше

Данные о клиенте:

ФИО клиента: *Тимушев Федор Алексеевич*

Паспортные данные: *4013 757424* *27.06.2013*

ТП №141

Дата рождения: *17.06.1999*

Пол: ☒ М ☐ Ж

Семейное положение: *не в браке*

Образование: *среднее специальное*

Код подразделения: *123*

Место работы: *УПК*

Должность: *студент*

Телефон предприятия: *88121234567*

Ежемесячные доходы на месте работы: *5000*

Место проживания (регистрации): *1-ая Утиная, д. 32, кв. 104*

Контактный телефон: *89112195183*

Эл. почта: *fed.timushev@gmail.com*

Данные о кредите:

Процентная ставка: _____

Валюта: _____

Период кредитования: с _____ по: _____

Цель получения кредита: *адрес*

Желаемая сумма займа: *150000*

Кол-во месяцев кредитования: _____

Ежемесячное погашение: _____

Итого к погашению: _____

Перейти к регистрации кредитной карты ►

Рисунок 8 – Заполнение информации о клиенте

Make credit 1.0

Регистрация клиента в кредитной системе | Регистрация кредитной карты | Погаше

Данные о клиенте:

ФИО клиента: *Тимушев Федор Алексеевич*

Паспортные данные: *4013 757424* *27.06.2013*

ТП №141

Дата рождения: *17.06.1999*

Пол: ☒ М ☐ Ж

Семейное положение: *не в браке*

Образование: *среднее специальное*

Код подразделения: *123*

Место работы: *УПК*

Должность: *студент*

Телефон предприятия: *88121234567*

Ежемесячные доходы на месте работы: *5000*

Место проживания (регистрации): *1-ая Утиная, д. 32, кв. 104*

Контактный телефон: *89112195183*

Эл. почта: *fed.timushev@gmail.com*

Данные о кредите:

Процентная ставка: *10*

Валюта: *EURO*

Период кредитования: с *01.12.2018* по: *01.05.2019*

Цель получения кредита: *ремонт*

Желаемая сумма займа: *100000*

Кол-во месяцев кредитования: *5*

Ежемесячное погашение: *22000*

Итого к погашению: *110000*

Перейти к регистрации кредитной карты ►

Рис. 9 – Заполнение информации о кредите

После того, как все данные заполнены, необходимо нажать на кнопку «Перейти к регистрации кредитной карты». Откроется вкладка с полями для заполнения соответствующей информации по регистрации кредитной карты. Данные заполняются в соответствии с параметрами и установленными стандартами банка (см. рис. 10).

The screenshot displays the 'Make credit 1.0' application window. The top navigation bar includes tabs for 'Регистрация клиента в кредитной системе', 'Регистрация кредитной карты' (which is active), and 'Погашение'. The main interface is split into two panels. The left panel, titled 'Данные для регистрации карты:', contains a form for entering card details. The right panel, titled 'Вид карты:', shows a visual representation of the credit card.

Данные для регистрации карты:

- Тип карты: **MASTERCARD**
- Код системы: 1
- Код банка: 22222
- Номер счета: 333333333
- Проверочное число: 4
- CVV: 5
- Начало действия: 01.12.2018
- Окончание действия: 01.05.2019
- Индекс: 197375
- Проверочное слово: колледж

Buttons at the bottom of the left panel:

- Внести данные на карту →
- Создать кредитную запись ☒
- Полная отмена регистрации ✕

Вид карты:

The right panel shows two visualizations of the credit card. The top one is a purple card with the text 'MAHMADCREDIT' and a Mastercard logo. The bottom one is a similar purple card with the text 'MAHMADCREDIT' and a Mastercard logo.

Рис. 10 – Заполнение информации по карте

Заполнив все необходимые поля, следует нажать на кнопку «Внести данные на карту». Действие, в свою очередь, заполнит макет карты справа (см. рис. 11).

Make credit 1.0

Регистрация клиента в кредитной системе | Регистрация кредитной карты | Погашение

Данные для регистрации карты:

Тип карты: MASTERCARD

Код системы: 1 Код банка: 22222 Номер счета: 333333333 Проверочное число: 4 CVV: 5

Начало действия: 01.12.2018 Окончание действия: 01.05.2019

Индекс: 197375

Проверочное слово: колледж

Внести данные на карту

Создать кредитную запись

Полная отмена регистрации

Вид карты:

MAHMADCREDIT

1 22222 333333333 4

12/2018 5/2019

Тимушев Федор Алексеевич

MAHMADCREDIT

11111111111111111111 5

Рис. 11 – Внесение данных на макет карты

Далее необходимо нажать на кнопку «Создать кредитную запись», после чего выведется сообщение «Кредитная запись добавлена!» и откроется главная страница, что свидетельствует об удачном добавлении данных о кредите (см. рис. 12).

Make credit 1.0

Регистрация клиента в кредитной системе | Регистрация кредитной карты | Погашение

Данные о клиенте:

ФИО клиента: Тимушев Федор Алексеевич

Паспортные данные: 4013 757424 27.06.2013

ТП №141

Дата рождения: 17.06.1999

Пол: М Ж

Семейное положение: не в браке

Образование: среднее специальное

Код подразделения: 123

Место работы: УПК

Должность: студент

Телефон предприятия: 88121234567

Ежемесячные доходы на месте работы: 5000

Место проживания (регистрации): 1-ая Утиная, д. 32, кв. 104

Контактный телефон: 89112195183

Эл. почта: fed.timushev@gmail.com

Данные о кредите:

Процентная ставка: 10

Валюта: EURO

Период кредитования: с 01.12.2018 по 01.05.2019

Цель получения кредита: ремонт

Желаемая сумма займа: 100000

Кол-во месяцев кредитования: 5

Ежемесячное погашение: 22000

Полное погашение: 110000

Кредитная запись добавлена!

OK

Перейти к регистрации кредитной карты

Рис. 12 – Добавление новой кредитной записи

2.3.3.3 После загрузки программы производится формирование списка кредитных записей из базы данных SQL Server и показ их на экране во вкладке «Погашение», в разделе «Список заемщиков» (см. рис. 13).

Make credit 1.0

Регистрация клиента в кредитной системе | Регистрация кредитной карты | **Погашение**

Список заемщиков

| Клиент | Валюта | Начало | Конец | Сумма | Остаток |
|-----------|--------|-----------|----------|--------|---------|
| Тимушев Ф | EURO | 12/1/2018 | 5/1/2019 | 100000 | 110000 |

Поиск по ФИО *фuo*

ФИО клиента: *фuo*

Ежемесячное погашение: *сумма*

Досрочное погашение: *месяцы* *кол-во*

Договор

Условия

Рис. 13 – Вывод списка кредитных записей

2.3.3.4 На вкладке «Погашение», если выбрать в списке одну из кредитных записей, «кликнув» по ней, то ниже под списком в областях «ФИО клиента», «Месячное погашение» - выведется, следовательно, информация о клиенте и его месячном погашении, в соответствии с договором (см. рис. 14).

Make credit 1.0

Регистрация клиента в кредитной системе | Регистрация кредитной карты | **Погашение**

Список заемщиков

| Клиент | Валюта | Начало | Конец | Сумма | Остаток |
|-----------|--------|-----------|----------|--------|---------|
| Тимушев Ф | EURO | 12/1/2018 | 5/1/2019 | 100000 | 110000 |

Поиск по ФИО фюо **Поиск**

ФИО клиента: Тимушев Федор Алексеевич

Ежемесячное погашение: 22000 **Внести ➤**

Досрочное погашение: месяцы кол-во **Внести ➤**

Договор

Тимушев Федор Алексеевич
 6.17.1999
 1-ая Утиная, д. 32, кв. 104
 ТП №141
 4013 757424
 ремонт

Условия

10
 EURO
 12.1.2018
 5.1.2019
 100000
 MASTERCARD

Печать договора

Печать условий

Рис. 14 – Выбор кредитной записи

Далее, если клиент готов внести погашение, следует нажать на кнопку справа от поля – «Внести», тогда автоматически внесение вычтется из поля остатка (см. рис. 15).

Make credit 1.0

Регистрация клиента в кредитной системе | Регистрация кредитной карты | **Погашение**

Список заемщиков

| Клиент | Валюта | Начало | Конец | Сумма | Остаток |
|-----------|--------|-----------|----------|--------|---------|
| Тимушев Ф | EURO | 12/1/2018 | 5/1/2019 | 100000 | 88000 |

Поиск по ФИО фюо **Поиск**

ФИО клиента: Тимушев Федор Алексеевич

Ежемесячное погашение: 22000 **Внести ➤**

Досрочное погашение: месяцы кол-во **Внести ➤**

Договор

Тимушев Федор Алексеевич
 6.17.1999
 1-ая Утиная, д. 32, кв. 104
 ТП №141
 4013 757424
 ремонт

Условия

10
 EURO
 12.1.2018
 5.1.2019
 100000
 MASTERCARD

Печать договора

Печать условий

Рис. 15 – Месячное погашение

Также, если клиент готов оплатить досрочно несколько месяцев вперед из своего кредита, то стоит воспользоваться полем досрочного погашения «Кол-во месяцев». Необходимо занести количество месяцев, которые готов оплатить заемщик, далее в следующее поле справа – данные высчитаются в зависимости от месячного погашения и количества месяцев для досрочного погашения. Затем необходимо нажать на кнопку «Внести», тогда автоматически внесение вычтется из остатка (см. рис. 16).

Make credit 1.0

Регистрация клиента в кредитной системе | Регистрация кредитной карты | **Погашение**

Список заемщиков

| Клиент | Валюта | Начало | Конец | Сумма | Остаток |
|-----------|--------|-----------|----------|--------|---------|
| Тимушев Ф | EURO | 12/1/2018 | 5/1/2019 | 100000 | 44000 |

Поиск по ФИО *фuo* **Поиск**

ФИО клиента: *Тимушев Федор Алексеевич*

Ежемесячное погашение: *22000* **Внести ➤**

Досрочное погашение: *месяцы 2* *44000* **Внести ➤**

Договор

Тимушев Федор Алексеевич
6.17.1999
1-ая Утиная, д. 32, кв. 104
ТП №141
4013 757424
ремонт

Печать договора

Условия

10
EURO
12.1.2018
5.1.2019
100000
MASTERCARD

Печать условий

Рис. 16 – Досрочное погашение

2.3.3.5 После нажатия на определенную кредитную запись в списке, производится вывод в соответствующее поле справа – договор заемщика (полная информация о клиенте), условия (полная информация по кредиту) (см. рис. 17).

The screenshot shows the 'Make credit 1.0' application interface. The top navigation bar includes 'Регистрация клиента в кредитной системе', 'Регистрация кредитной карты', and 'Погаше'. The left sidebar contains 'Список заемщиков'. The main content area is divided into two panels: 'Договор' (Contract) and 'Условия' (Conditions).

Список заемщиков (Left Panel):

| Клиент | Валюта | Начало | Конец | Сумма | Остаток |
|-----------|--------|-----------|----------|--------|---------|
| Тимушев Ф | EURO | 12/1/2018 | 5/1/2019 | 100000 | 44000 |

Below the table, there are search and input fields:

- Поиск по ФИО:
- ФИО клиента:
- Ежемесячное погашение:
- Досрочное погашение:

Договор (Right Panel):

Тимушев Федор Алексеевич
 6.17.1999
 1-ая Утиная, д. 32, кв. 104
 ТП №141
 4013 757424
 ремонт

Условия (Right Panel):

10
 EURO
 12.1.2018
 5.1.2019
 100000
 MASTERCARD

Рис. 17 – Вывод договора и условий заемщика

2.3.3.6 После регистрации многочисленных клиентов, поиск их в базе усложняется, а графа в вою очередь заполняется (см. рис. 18).

The screenshot shows the 'Make credit 1.0' application interface. The top navigation bar includes 'Регистрация клиента в кредитной системе', 'Регистрация кредитной карты', and 'Погаше'. The left sidebar contains 'Список заемщиков'. The main content area is divided into two panels: 'Договор' (Contract) and 'Условия' (Conditions).

Список заемщиков (Left Panel):

| Клиент | Валюта | Начало | Конец | Сумма | Остаток |
|-----------|--------|-----------|----------|--------|---------|
| Тимушев Ф | EURO | 12/1/2018 | 5/1/2019 | 100000 | 44000 |
| Рустам Ма | DOLLAR | 12/1/2018 | 5/1/2019 | 150000 | 162000 |

Below the table, there are search and input fields:

- Поиск по ФИО:
- ФИО клиента:
- Ежемесячное погашение:
- Досрочное погашение:

Договор (Right Panel):

Тимушев Федор Алексеевич
 6.17.1999
 1-ая Утиная, д. 32, кв. 104
 ТП №141
 4013 757424
 ремонт

Условия (Right Panel):

10
 EURO
 12.1.2018
 5.1.2019
 100000
 MASTERCARD

Рис. 18 – Заполнение базы клиентов

Чтобы воспользоваться поиском кредитной записи – необходимо на вкладке «Погашение», в графе «Поиск по ФИО» вписать полные данные ФИО клиента в соответствии с его ранней регистрацией, и нажать на кнопку «Поиск» справа, после чего в графе базы будет доступна соответствующая поиску кредитная запись (см. рис. 19).

The screenshot displays the 'Make credit 1.0' application window. The top navigation bar includes tabs for 'Регистрация клиента в кредитной системе', 'Регистрация кредитной карты', and 'Погашение'. The 'Погашение' tab is active. The interface is divided into two main sections: 'Список заемщиков' (Borrowers List) on the left and 'Договор' (Contract) on the right.

Список заемщиков (Borrowers List):

| Клиент | Валюта | Начало | Конец | Сумма | Остаток |
|-----------|--------|-----------|----------|--------|---------|
| Рустам Ма | DOLLAR | 12/1/2018 | 5/1/2019 | 150000 | 162000 |

Below the table, there are search and input fields:

- Поиск по ФИО:** Рустам Махмадкадамов (with a 'Поиск' button)
- ФИО клиента:** Тимушев Федор Алексеевич
- Ежемесячное погашение:** 22000 (with a 'Внести' button)
- Досрочное погашение:** месяцы 2, 44000 (with a 'Внести' button)

Договор (Contract):

Тимушев Федор Алексеевич
 6.17.1999
 1-ая Утиная, д. 32, кв. 104
 ТП №141
 4013 757424
 ремонт

Печать договора

Условия (Conditions):

10
 EURO
 12.1.2018
 5.1.2019
 100000
 MASTERCARD

Печать условий

Рис. 19 – Результат выполнения поиска

2.3.3.7 Печать документа (договора или условий) выполняется путем нажатия соответствующей клавиши «Печать» в правом нижнем углу областей «Договор» и «Условия». После нажатия на клавишу откроется диалоговое окно с выбором устройства печати и прямой отправкой документа в очередь на печать (см. рис.20).

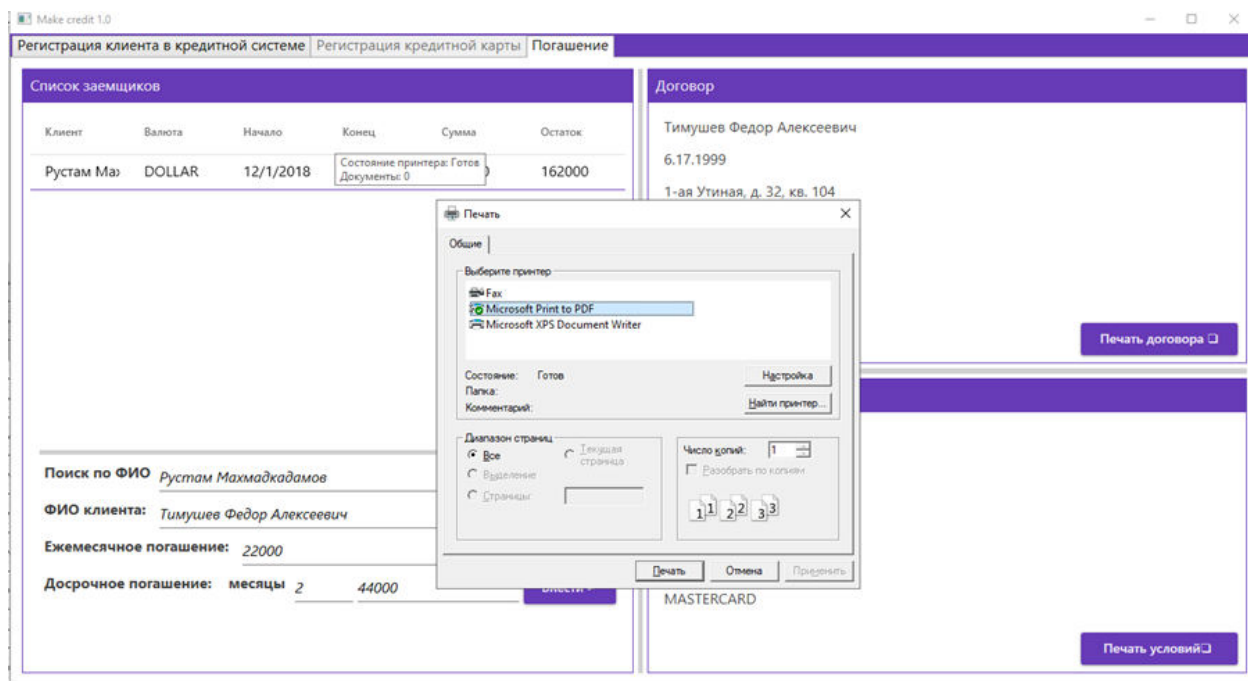


Рис. 20 – Отправка документа в очередь на печать

После, следует нажать на кнопку «Печать», после чего документ будет распечатан выбранным ранее устройством.

2.3.3.8 Завершение программы производится стандартным путём, т.е. нажатием на «крестик» в правом верхнем углу.

В ходе работы с системой оператору могут выдаваться сообщения, приведенные в таблице ниже.

Таблица 2 - Сообщения оператору

| Текст сообщения | Описание сообщения | Возможные действия |
|---|--|---|
| «Конечная дата должна быть позже начала!» | Сообщение появляется в тот момент, когда оператор выбирает на вкладке «Регистрация клиента в кредитной системе» поле «Кол-во месяцев кредитования» | Нажав на кнопку «ОК» - оператор меняет конечную дату на валидную. Затем жмет на поле «Кол-во месяцев кредитования» для правильного подсчета месяцев |

| Текст сообщения | Описание сообщения | Возможные действия |
|--|---|--|
| «Принять заявку и перейти к регистрации К/К?» | Сообщение появляется в тот момент, когда оператор прожимает на вкладке «Регистрация клиента в кредитной системе» кнопку «Перейти к регистрации кредитной карты» | Нажав на кнопку «Да» - оператор перейдет к вкладке «Регистрация кредитной карты». Нажав «Нет» - получит сообщение «Заявка отклонена!» |
| «Заявка отклонена!» | Сообщение появляется в тот момент, когда оператор жмет «Нет» в сообщении «Принять заявку и перейти к регистрации К/К?» | Нажав на кнопку «ОК» поля для заполнения на вкладке «Регистрация кредитной карты» очистятся и регистрация дальше не продолжится |
| «Отменить регистрацию?» | Сообщение появляется в тот момент, когда оператор нажимет на кнопку «Полная отмена регистрации» на вкладке «Регистрация кредитной карты» | Нажав на кнопку «Да» - оператор получит сообщение «Заявка отклонена!» и перейдет к вкладке «Регистрация клиента в кредитной системе» Нажав «Нет» - оператор остается на текущей вкладке |
| «Кредитная запись добавлена!» | Сообщение появляется в тот момент, когда оператор нажимет на кнопку «Создать кредитную запись» на вкладке «Регистрация кредитной карты» | Нажав на кнопку «ОК» окно с сообщением закроется, а оператор будет переведен на вкладку «Регистрация клиента в кредитной системе» для дальнейшей работы |
| «Кол-во месяцев не может быть больше оставшегося срока!» | Сообщение появляется в тот момент, когда была нажата кнопка «Внести» напротив поля «месяцы» с числом месяцев больше чем указано в поле «Срок» на вкладке «Погашение». | Нажав на кнопку «ОК» окно с сообщением закроется, а оператору седует указать валидное количество месяцев в поле «месяцы». |
| «У клиента имеется задолженность!» | Сообщение появляется в тот момент, когда оператор выбирает кредитную запись, подсвеченную красным цветом, что означает долг по кредиту на вкладке «Погашение». | Нажав на кнопку «ОК» окно с сообщением закроется, а оператору седует сообщить заемщику об его задолженности. |

| Текст сообщения | Описание сообщения | Возможные действия |
|-----------------------------------|--|--|
| «Кредит минимум от 2-ух месяцев!» | Сообщение появляется в тот момент, когда оператор выбирает дату окончания кредита раньше чем 2 месяца от даты начала кредита на вкладке «Регистрация клиента в кредитной системе». | Нажав на кнопку «ОК» окно с сообщением закроется, а оператору сдует выставить валидную дату окончания кредита отличную (большую) от двух.. |

2.4 Объект испытаний

2.4.1 Объектом испытаний является программный продукт «Выдача кредитов банком».

2.4.2 Программа предназначена к применению в банком.

2.4.3 Обозначения испытываемой программы не используются.

2.5 Цель испытаний

Целью испытаний является проверка соответствия программы требованиям Технического Задания.

2.6 Требования к программе

В процессе испытаний подлежат проверке следующие требования к программе:

2.6.1 К функциональным требованиям относятся:

- программа должна формировать списки заемщиков и кредитных записей и выводить их в таблице в окне «Клиенты» и на вкладке «Погашение» соответственно;
- программа должна обеспечивать добавление новых клиентов в базу данных;
- программа должна обеспечивать добавление новых кредитных записей в базу данных;
- программа должна обеспечивать добавление новых кредитных карт в базу данных;
- программа должна иметь функцию поиска кредитной записи с помощью ФИО по маске;

- программа должна иметь функцию поиска клиента с помощью ФИО по маске;
- программа должна выводить договор о кредитовании по каждому клиенту;
- программа должна выводить условия кредитования по каждому клиенту;
- программа должна обеспечивать ежемесячное погашение кредита для по каждому клиенту;
- программа должна обеспечивать досрочное погашение кредита для по каждому клиенту;
- программа должна сообщать о задолженности по кредиту для по каждому клиенту если таковая имеется;
- программа должна закрывать если он выплачен для каждого клиента;

2.6.2 Исходные коды программы должны быть реализованы на языке C#.

Системные программные средства, используемые программой, должны быть представлены локализованной версией операционной системы Windows 7 и выше.

2.7 Требования к программной документации

Состав программной документации должен включать в себя:

- спецификация;
- текст программы;
- описание программы;
- программу и методики испытаний;
- ведомость эксплуатационных документов;
- описание применения;
- руководство программиста;
- руководство оператора.

2.8 Средства и порядок испытаний

2.8.1 В состав технических средств должен входить персональный компьютер, включающий в себя:

- процессор Pentium - 4 с тактовой частотой, 1.2 ГГц, не менее;
- оперативную память объемом, 128 Мб, не менее;

- жесткий диск объемом 40 Гб, и выше;
- мышь;
- клавиатура;

2.8.2 Для проведения испытаний необходимы следующие программные средства:

- Операционная система Windows 7 или выше;
- SQL Management Studio 2012 или выше;
- SQL Server 2012 или выше;

2.8.3 Испытания проводятся в следующем порядке:

- проверяется наличие и комплектность программной документации (п.2.6.4);
- проверяется соответствие требованиям к маркировке и упаковке (п.2.6.3.3);
- проверяется соответствие требованиям к функциональным характеристикам (п.2.6.3.1);
- проверяется соответствие требованиям к информационной и программной совместимости (п.2.6.3.2);

2.9 Методы испытаний

Для тестирования данного программного продукта применяется – системное тестирование. Это тестирование программного обеспечения, выполняемое на полной, интегрированной системе, с целью проверки соответствия системы исходным требованиям, как функциональным, так и не функциональным.

Системное тестирование выполняет основную задачу - определяет разницу между функционалом готовой системы и функционалом, заявленным исходными требованиями.

Преимущества данного типа тестирования:

- Сокращение количества дефектов в опытно-промышленной эксплуатации;
- Возможность использования тестовых сценариев в качестве обучающих материалов для будущих пользователей системы;
- Выявление ошибок настройки стенда, что облегчает работу администраторов при установке системы в опытно-промышленную эксплуатацию.

Выполняя системное тестирование, можно обнаружить следующие типы дефектов:

- Неправильное использование системных ресурсов.
- Непредусмотренные комбинации пользовательских данных.
- Проблемы с совместимостью окружения.
- Непредусмотренные сценарии использования.
- Плохое удобство использования.

Системное тестирование выполняется методом «Черного ящика», т.к. проверяемое множество является «внешними» сущностями, которые не требуют взаимодействия с внутренним устройством программы.

Для проверки способности программы обеспечивать необходимый функционал тестирующему, необходимо пройти тест-кейс (см. приложение).

Тест-кейс тестирования программы представлен в таблице 1.

Таблица 3 – Результаты испытаний программы

| Проверяемые требования | Действия тестирующего | Полученные результаты |
|--|--|---|
| Способности программы заносить персональные данные | <ul style="list-style-type: none"> - запустить программу - открыть вкладку «Регистрация клиента в кредитной системе» - заносить в соответствующие поля информацию о клиенте | Во всех полях для заполнения будет информация о клиенте, необходимая для занесения в базу данных и для последующей итоговой регистрации в системе |
| Способность программы к выбору процентной ставки | <ul style="list-style-type: none"> - запустить программу - открыть вкладку «Регистрация клиента в кредитной системе» - после занесения персональных данных, в разделе «Данные о кредите» нажать по выпадающему списку с графой «Процентная ставка» - выбрать необходимый процент | После выбора нужного процента, он будет отображаться в соответствующей графе до занесения информации в базу данных. |
| Способность программы к выбору валюты | <ul style="list-style-type: none"> - запустить программу - открыть вкладку «Регистрация клиента в кредитной системе» - после занесения персональных данных, в разделе «Данные о кредите» нажать по выпадающему списку с графой «Валюта» - выбрать необходимую валюту | После выбора нужной валюты, он будет отображаться в соответствующей графе до занесения информации в базу данных. |

| Проверяемые требования | Действия тестировщика | Полученные результаты |
|--|---|---|
| Способность программы к регистрации кредитной карты | <ul style="list-style-type: none"> - запустить программу - открыть вкладку «Регистрация клиента в кредитной системе» - занести в соответствующие поля информацию о клиенте - занести в соответствующие поля информацию о кредите - нажать на кнопку «Перейти к регистрации кредитной карты» - в полученном сообщении «Перейти к регистрации К/К?» нажать «Да» - в открывшейся вкладке «Регистрация кредитной карты» внести все необходимые данные - нажать на кнопку «Внести данные на карту» | После нажатия последней кнопки, макет карты справа будет автоматически заполнен данными с левой формы, после чего данные будут ожидать занесения в базу. |
| Способность программы обеспечивать успешное добавление кредитных записей в базу данных | <ul style="list-style-type: none"> - запустить программу - открыть вкладку «Регистрация клиента в кредитной системе» - занести в соответствующие поля информацию о клиенте - занести в соответствующие поля информацию о кредите - нажать на кнопку «Перейти к регистрации кредитной карты» - в полученном сообщении «Перейти к регистрации К/К?» нажать «Да» - в открывшейся вкладке «Регистрация кредитной карты» внести все необходимые данные - нажать на кнопку «Внести данные на карту» - нажать кнопку «Создать кредитную запись» | При нажатии на кнопку «Создать кредитную запись» форма откроет вкладку с пустыми полями для заполнения «Регистрация клиента в кредитной системе», а также выведет сообщение «Кредитная запись добавлена!», что будет свидетельствовать об успешном добавлении записи в базу данных. |
| Способность программы производить проверку погашения | <ul style="list-style-type: none"> - запустить программу - открыть вкладку «Погашение» | В открывшейся вкладке будет раздел «Список заемщиков» будет выведена таблица с кредитными записями, узнать об остатке погашения можно в графе таблицы «Остаток». |

| Проверяемые требования | Действия тестировщика | Полученные результаты |
|---|---|---|
| Способность программы производить принятие ежемесячного платежа | <ul style="list-style-type: none"> - запустить программу - открыть вкладку «Погашение» - в разделе «Список заемщиков», в таблице с кредитными записями, выбрать необходимую и нажать на нее - после этого в графу «Ежемесячное погашение» будет передана сумма ежемесячной оплаты на основе регистрации договора - нажать кнопку «Внести» напротив поля | В в разделе «Список заемщиков», в таблице с кредитными записями, в выбранной записи будет изменен остаток в соответствии с внесенной оплатой. |
| Способность программы производить принятие досрочного платежа | <ul style="list-style-type: none"> - запустить программу - открыть вкладку «Погашение» - в разделе «Список заемщиков», в таблице с кредитными записями, выбрать необходимую и нажать на нее - после этого в графе «Досрочное погашение» необходимо выбрать нужное кол-во месяцев погашения - далее нажать на графу справа для автоматического подсчета суммы - после этого в графу «Досрочное погашение» будет передана сумма досрочной оплаты на основе регистрации договора - нажать кнопку «Внести» напротив поля | В в разделе «Список заемщиков», в таблице с кредитными записями, в выбранной записи будет изменен остаток в соответствии с внесенной оплатой. |
| Способность программы выполнять просмотр условий | <ul style="list-style-type: none"> - запустить программу - открыть вкладку «Погашение» - в разделе «Список заемщиков», в таблице с кредитными записями, выбрать необходимую и нажать на нее | В разделе «Условия» справа будут выведены все необходимые данные. |

| Проверяемые требования | Действия тестирующего | Полученные результаты |
|---|--|---|
| Способность программы выполнять печать условий | <ul style="list-style-type: none"> - запустить программу - открыть вкладку «Погашение» - в разделе «Список заемщиков», в таблице с кредитными записями, выбрать необходимую и нажать на нее - в разделе «Условия» нажать на кнопку «Печать условий» - в открывшемся диалоговом окне выбрать нужное устройство печати и нажать на кнопку «Печать» | Условия будут распечатаны выбранным способом. |
| Способность программы выполнять просмотр договора | <ul style="list-style-type: none"> - запустить программу - открыть вкладку «Погашение» - в разделе «Список заемщиков», в таблице с кредитными записями, выбрать необходимую и нажать на нее | В разделе «Договор» справа будут выведены все необходимые данные. |
| Способность программы выполнять печать договора | <ul style="list-style-type: none"> - запустить программу - открыть вкладку «Погашение» - в разделе «Список заемщиков», в таблице с кредитными записями, выбрать необходимую и нажать на нее - в разделе «Договор» нажать на кнопку «Печать договора» - в открывшемся диалоговом окне выбрать нужное устройство печати и нажать на кнопку «Печать» | Договор будет распечатан выбранным способом. |

2.10 Тест-кейсы

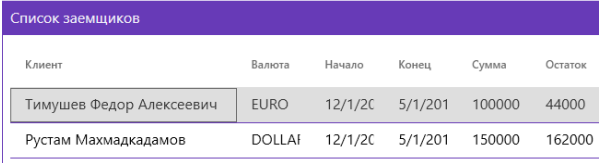
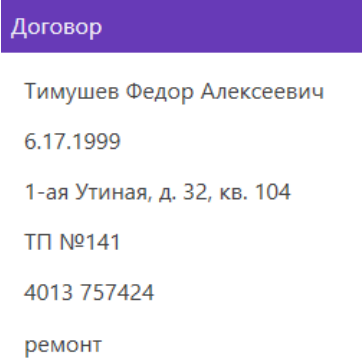
Программный продукт «Выдача банком кредита» автоматизирует деятельность банка, ускоряя оформление и выдачу кредита, регистрацию кредитной карты, просмотр списка заемщиков, просмотр условий и договора. Позволяет ускоренно работать с базой данных (добавлять и удалять кредитные записи, заемщиков), изменять и фильтровать данные.

Тест-кейс № 1. Вывод списка заемщиков

| ШАГИ | ОЖИДАЕМЫЙ РЕЗУЛЬТАТ |
|--|--|
| Открыть программу «Выдача банком кредита» (двойным кликом по файлу InterYOURMOMface.exe) | Открывается главное окно программы (показано на рисунке А.1) |

| | |
|---|---|
| Перейти на вкладку «Погашение» | Отображается вкладка «Погашение» (показано на рисунке А.2) |
| <p>Убедиться, что на вкладке «Погашение», в окне «Список заемщиков» отобразился список заемщиков, соответствующий следующим данным:</p> <p>Тимушев Федор Алексеевич</p> <p>Рустам Махмадкадамов</p> <p>Коваленко Андрей</p> <p>Скарговский Федор Борисович</p> <p>Кобжицкая Ирина Андреевна</p> | <p>Отобразился список заемщиков</p> <p>Клиент</p> <hr/> <p>Тимушев Федор Алексеевич</p> <hr/> <p>Рустам Махмадкадамов</p> <hr/> <p>Коваленко Андрей</p> <hr/> <p>Скарговский Федор Борисович</p> <hr/> <p>Кобжицкая Ирина Андреевна</p> <hr/> |

Тест-кейс № 2. Вывод договора

| ШАГИ | ОЖИДАЕМЫЙ РЕЗУЛЬТАТ |
|--|--|
| Открыть программу «Выдача банком кредита» (двойным кликом по файлу InterYOURMOMface.exe) | Открывается главное окно программы (показано на рисунке А.1) |
| Перейти на вкладку «Погашение» | Отображается вкладка «Погашение» (показано на рисунке А.3) |
| Нажать на кредитную запись «Тимушев Федор Алексеевич» в окне «Список заемщиков» | Выбранная запись выделиться серым цветом  |
| Убедиться, что на вкладке «Погашение», в окне «Договор» отобразились данные по кредитной записи, соответствующий следующим: <i>Тимушев Федор Алексеевич</i> <i>17.06.1999</i> <i>1-ая Утиная, д. 32, кв. 104</i> <i>ТП №141</i> <i>4013 757424</i> <i>ремонт</i> | Отобразились следующие данные:  |

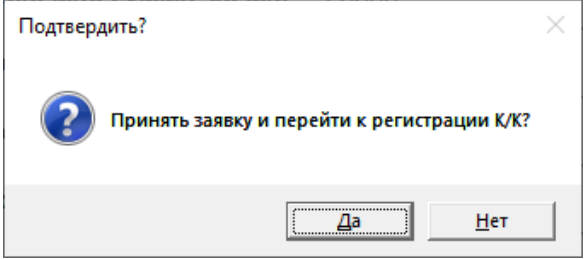
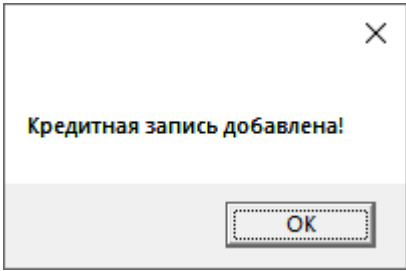
Тест-кейс № 3. Вывод условий

| ШАГИ | ОЖИДАЕМЫЙ РЕЗУЛЬТАТ | | | | | | | | | | | | | | | | | | |
|---|---|---------|---------|--------|---------|-------|---------|--------------------------|------|---------|---------|--------|-------|---------------------|--------|---------|---------|--------|--------|
| Открыть программу «Выдача банком кредита» (двойным кликом по файлу InterYOURMOMface.exe) | Открывается главное окно программы (показано на рисунке А.1) | | | | | | | | | | | | | | | | | | |
| Перейти на вкладку «Погашение» | Отображается вкладка «Погашение» (показано на рисунке А.3) | | | | | | | | | | | | | | | | | | |
| Нажать на кредитную запись «Тимушев Федор Алексеевич» в окне «Список заемщиков» | Выбранная запись выделиться серым цветом <div><div>Список заемщиков</div><table><tr><td>Клиент</td><td>Валюта</td><td>Начало</td><td>Конец</td><td>Сумма</td><td>Остаток</td></tr><tr><td>Тимушев Федор Алексеевич</td><td>EURO</td><td>12/1/20</td><td>5/1/201</td><td>100000</td><td>44000</td></tr><tr><td>Рустам Махмададамов</td><td>DOLLAR</td><td>12/1/20</td><td>5/1/201</td><td>150000</td><td>162000</td></tr></table></div> | Клиент | Валюта | Начало | Конец | Сумма | Остаток | Тимушев Федор Алексеевич | EURO | 12/1/20 | 5/1/201 | 100000 | 44000 | Рустам Махмададамов | DOLLAR | 12/1/20 | 5/1/201 | 150000 | 162000 |
| Клиент | Валюта | Начало | Конец | Сумма | Остаток | | | | | | | | | | | | | | |
| Тимушев Федор Алексеевич | EURO | 12/1/20 | 5/1/201 | 100000 | 44000 | | | | | | | | | | | | | | |
| Рустам Махмададамов | DOLLAR | 12/1/20 | 5/1/201 | 150000 | 162000 | | | | | | | | | | | | | | |
| Убедиться, что на вкладке «Погашение», в окне «Условия» отобразились данные по кредитной записи, соответствующий следующим: 10 EURO 12.1.2018 5.1.2019 100000 MSTERCARD | Отобразились следующие данные: <div><div>Условия</div><div>10 EURO 12.1.2018 5.1.2019 100000 MASTERCARD</div></div> | | | | | | | | | | | | | | | | | | |

Тест-кейс № 4. Добавление кредитной записи

| ШАГИ | ОЖИДАЕМЫЙ РЕЗУЛЬТАТ |
|---|---|
| Открыть программу «Выдача банком кредита» (двойным кликом по файлу InterYOURMOMface.exe) | Открывается главное окно программы (показано на рисунке А.1) |
| <p>Заполнить все поля на вкладке «Регистрация клиента в кредитной системе» в окне «Данные о клиенте» следующими данными:</p> <ul style="list-style-type: none">– ФИО клиента: «Тимушева Елизавета Алексеевна»– Паспортные данные: «1234 567890 / 31.07.2018 / Отдел милиции №1»– Дата рождения: «15.07.2004»– Пол: «Ж»– Семейное положение: «не в браке»– Образование: «полное среднее»– Код подразделения: «4321»– Место работы: «Гимназия №17»– Должность: «ученик»– Телефон предприятия: «88121234567»– Ежемесячные доходы на месте работы: «15000»– Место проживания (регистрации): «ул. 1-я Утиная, д. 32, кв. 104»– Контактный телефон: «89111234567»– Эл. Почта: «liza@gmail.com» | Заполненная форма вкладки «Регистрация клиента в кредитной системе» в окне «Данные о клиенте» (показано на рисунке Б.1) |

| | |
|--|--|
| <p>Заполнить все поля на вкладке «Регистрация клиента в кредитной системе» в окне «Данные о кредите» следующими данными:</p> <ul style="list-style-type: none"> – Процентная ставка: «10» – Валюта: «RUB» – Период кредитования: с «10.03.2019» по «10.07.2019» – Цель получения кредита: «обучение в вузе» – Желаемая сумма займа: «50000» – Кол-во месяцев кредитования: <i>высчитывается автоматически при щелчке на поле</i> – Ежемесячное погашение: <i>высчитывается автоматически при щелчке на поле</i> – Итого к погашению: <i>высчитывается автоматически при щелчке на поле</i> | <p>Заполненная форма вкладки «Регистрация клиента в кредитной системе» в окне «Данные о кредите» (показано на рисунке Б.2)</p> |
|--|--|

| ШАГИ | ОЖИДАЕМЫЙ РЕЗУЛЬТАТ |
|--|--|
| <p>Нажать на кнопку «Перейти к регистрации кредитной карты»</p> <p>Во всплывшем окне «Принять заявку и перейти к регистрации КК?» нажать кнопку «Да»</p> | <p>Вкладка «Регистрация клиента в кредитной системе» закрывается, и открывается вкладка «Регистрация кредитной карты» (показано на рисунке А.2)</p>  |
| <p>Заполнить все поля на вкладке «Регистрация кредитной карты» в окне «Данные для регистрации карты» следующими данными:</p> <ul style="list-style-type: none"> – Тип карты: «VISA» – Код системы: «1» – Код банка: «22222» – Номер счета: «333333333» – Проверочное число: «4» – CVV: «5» – Начало действия: <i>заполняется автоматически</i> – Окончание действия: <i>заполняется автоматически</i> – Индекс: «197375» – Проверочное слово: «брат» | <p>Заполненная форма вкладки «Регистрация кредитной карты» в окне «Данные для регистрации карты» (показано на рисунке Б.3)</p> |
| <p>Нажать на кнопку «Внести данные на карту» в окне «Данные для регистрации карты»</p> | <p>Заполненный макет карты на вкладке «Регистрация кредитной карты» в окне «Вид карты» (показано на рисунке Б.4)</p> |
| <p>Нажать на кнопку «Создать кредитную запись» в окне «Данные для регистрации карты»</p> <p>Во всплывшем окне «Кредитная запись добавлена!» нажать кнопку «Ок»</p> |  <p>Открывается главное окно программы (показано на рисунке А.1)</p> |

Тест-кейс № 5. Ежемесячное погашение

| ШАГИ | ОЖИДАЕМЫЙ РЕЗУЛЬТАТ | | | | | | | | | | | | | | | | | | |
|---|---|---------|---------|--------|---------|-------|---------|--------------------------|------|---------|---------|--------|-------|----------------------|--------|---------|---------|--------|--------|
| Открыть программу «Выдача банком кредита» (двойным кликом по файлу InterYOURMOMface.exe) | Открывается главное окно программы (показано на рисунке А.1) | | | | | | | | | | | | | | | | | | |
| Перейти на вкладку «Погашение» | Отображается вкладка «Погашение» (показано на рисунке А.3) | | | | | | | | | | | | | | | | | | |
| Нажать на кредитную запись «Тимушев Федор Алексеевич» в окне «Список заемщиков» | <p>Выбранная запись выделиться серым цветом:</p> <div><div>Список заемщиков</div><table><tr><th>Клиент</th><th>Валюта</th><th>Начало</th><th>Конец</th><th>Сумма</th><th>Остаток</th></tr><tr><td>Тимушев Федор Алексеевич</td><td>EURO</td><td>12/1/20</td><td>5/1/201</td><td>100000</td><td>44000</td></tr><tr><td>Рустам Махмадкадамов</td><td>DOLLAR</td><td>12/1/20</td><td>5/1/201</td><td>150000</td><td>162000</td></tr></table></div> <p>ФИО записи отобразится в поле «ФИО клиента» ниже таблицы:</p> <p>ФИО клиента: <u>Тимушев Федор Алексеевич</u></p> | Клиент | Валюта | Начало | Конец | Сумма | Остаток | Тимушев Федор Алексеевич | EURO | 12/1/20 | 5/1/201 | 100000 | 44000 | Рустам Махмадкадамов | DOLLAR | 12/1/20 | 5/1/201 | 150000 | 162000 |
| Клиент | Валюта | Начало | Конец | Сумма | Остаток | | | | | | | | | | | | | | |
| Тимушев Федор Алексеевич | EURO | 12/1/20 | 5/1/201 | 100000 | 44000 | | | | | | | | | | | | | | |
| Рустам Махмадкадамов | DOLLAR | 12/1/20 | 5/1/201 | 150000 | 162000 | | | | | | | | | | | | | | |
| Заполнить поле «Ежемесячное погашение» на вкладке «Погашение » в окне «Список заемщиков» следующими данными: — Ежемесячное погашение: заполняется автоматически | <p>Заполненное поле «Ежемесячное погашение»:</p> <p>Ежемесячное погашение: <u>22000</u></p> | | | | | | | | | | | | | | | | | | |
| Нажать на кнопку «Внести» левее поля «Ежемесячное погашение» на вкладке «Погашение» в окне «Список заемщиков» | <p>Данные в столбце «Остаток» в таблице кредитных записей на вкладке «Погашение» в окне «Список заемщиков» соответственно записи «Тимушев Федор Алексеевич» изменились:</p> <div><table><tr><th>Клиент</th><th>Валюта</th><th>Начало</th><th>Конец</th><th>Сумма</th><th>Остаток</th></tr><tr><td>Тимушев Федор Алексеевич</td><td>EURO</td><td>12/1/20</td><td>5/1/201</td><td>100000</td><td>22000</td></tr></table></div> | Клиент | Валюта | Начало | Конец | Сумма | Остаток | Тимушев Федор Алексеевич | EURO | 12/1/20 | 5/1/201 | 100000 | 22000 | | | | | | |
| Клиент | Валюта | Начало | Конец | Сумма | Остаток | | | | | | | | | | | | | | |
| Тимушев Федор Алексеевич | EURO | 12/1/20 | 5/1/201 | 100000 | 22000 | | | | | | | | | | | | | | |

Тест-кейс № 6. Досрочное погашение

| ШАГИ | ОЖИДАЕМЫЙ РЕЗУЛЬТАТ | | | | | | |
|--|---|----------------------------|-----------|-----------|-----------|-------|-------|
| Открыть программу «Выдача банком кредита» (двойным кликом по файлу InterYOURMOMface.exe) | Открывается главное окно программы (показано на рисунке А.1) | | | | | | |
| Перейти на вкладку «Погашение» | Отображается вкладка «Погашение» (показано на рисунке А.3) | | | | | | |
| Нажать на кредитную запись «Тимушева Елизавета Алексеевна» в окне «Список заемщиков» | Выбранная запись выделяться серым цветом: <table><tr><td>Тимушева Елизавета Алексее</td><td>RUB</td><td>3/10/20</td><td>7/10/20</td><td>50000</td><td>55000</td></tr></table> ФИО записи отобразится в поле «ФИО клиента» ниже таблицы: ФИО клиента: <u>Тимушева Елизавета Алексеевна</u> | Тимушева Елизавета Алексее | RUB | 3/10/20 | 7/10/20 | 50000 | 55000 |
| Тимушева Елизавета Алексее | RUB | 3/10/20 | 7/10/20 | 50000 | 55000 | | |
| Заполнить поле «Досрочное погашение» на вкладке «Погашение» в окне «Список заемщиков» следующими данными: — месяцы: «2» / <i>заполняется автоматически</i> | Заполненное поле «месяцы»: Досрочное погашение: <u>месяцы 2</u> Заполненное поле «Досрочное погашение»: Досрочное погашение: <u>месяцы 2</u> <u>27500</u> | | | | | | |
| Нажать на кнопку «Внести» левее поля «Досрочное погашение» на вкладке «Погашение» в окне «Список заемщиков» | Данные в столбце «Остаток» в таблице кредитных записей на вкладке «Погашение» в окне «Список заемщиков» соответственно записи «Тимушева Елизавета Алексеевна» изменились: <table><tr><td>Тимушева </td><td>RUB</td><td>3/10/2019</td><td>7/10/2019</td><td>50000</td><td>27500</td></tr></table> | Тимушева | RUB | 3/10/2019 | 7/10/2019 | 50000 | 27500 |
| Тимушева | RUB | 3/10/2019 | 7/10/2019 | 50000 | 27500 | | |

Тест-кейс № 7. Поиск по ФИО

| ШАГИ | ОЖИДАЕМЫЙ РЕЗУЛЬТАТ | | | | | | | | |
|--|---|----------|--------|--------|-------|----------------------|--------|----------|--------|
| Открыть программу «Выдача банком кредита» (двойным кликом по файлу InterYOURMOMface.exe) | Открывается главное окно программы (показано на рисунке А.1) | | | | | | | | |
| Перейти на вкладку «Погашение» | Отображается вкладка «Погашение» (показано на рисунке А.3) | | | | | | | | |
| Заполнить поле «Поиск по ФИО» на вкладке «Погашение» в окне «Список заемщиков» следующими данными: – Поиск по ФИО: «Рустам Махмадкадамов» | Заполненное поле «Поиск по ФИО»: Поиск по ФИО <u>Рустам Махмадкадамов</u> | | | | | | | | |
| Нажать на кнопку «Поиск» левее поля «Поиск по ФИО» на вкладке «Погашение» в окне «Список заемщиков» | В таблице кредитных записей на вкладке «Погашение» в окне «Список заемщиков» отображена запись согласно поиску: <div><div>Список заемщиков</div><table><tr><th>Клиент</th><th>Валюта</th><th>Начало</th><th>Конец</th></tr><tr><td>Рустам Махмадкадамов</td><td>DOLLAR</td><td>12/1/201</td><td>5/1/20</td></tr></table></div> | Клиент | Валюта | Начало | Конец | Рустам Махмадкадамов | DOLLAR | 12/1/201 | 5/1/20 |
| Клиент | Валюта | Начало | Конец | | | | | | |
| Рустам Махмадкадамов | DOLLAR | 12/1/201 | 5/1/20 | | | | | | |

2.11 Юзер-стори

User Story—это короткая формулировка намерения, описывающая что-то, что система должна делать для пользователя.

- Как рабочий банка я имею возможность просматривать всех заемщиков в одном списке;
- Как рабочий банка я имею возможность просматривать договор и условия кредитования о каждом заемщике в отдельном окне;
- Как рабочий банка я имею возможность отправлять на печать документы (договор и условия) отдельно по каждому заемщику.
- Как рабочий банка, я хочу иметь возможность списания кредитной карты;
- Как рабочий банка, я хочу иметь возможность поиска заемщика в списке по ФИО;

2.12 Юнит тестирование

Создание проекта тестирования

Модульное тестирование используется разработчиками для проверки правильности работы методов классов. Модульные тесты сохраняются в системе контроля версий и выполняются при каждом построении приложения. Кроме того, модульные тесты являются основой регрессионного тестирования, которое выполняется при добавлении новых возможностей или модификации приложения.

Для реализации модульного тестирования разрабатываемого приложения создадим тестовый проект модульного тестирования. Для этого добавим в решение Interface проект модульного теста - UnitTestProjectLibrary (рисунок 21)

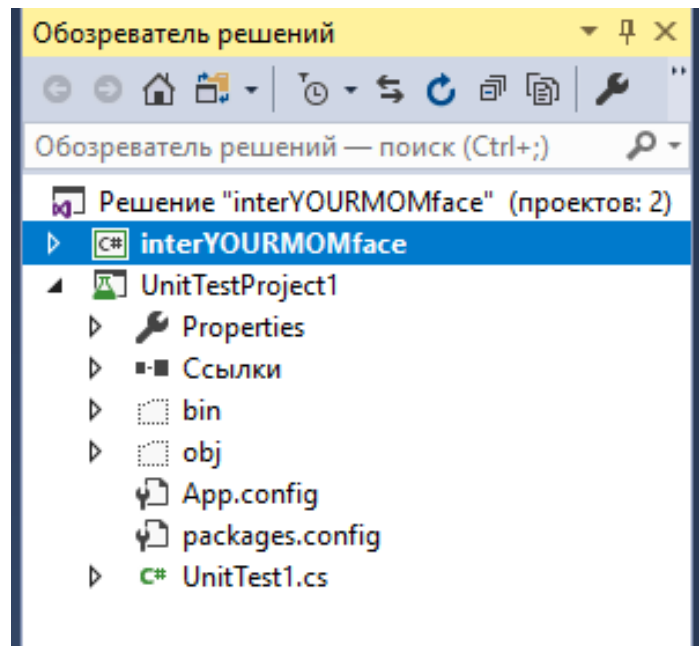


Рисунок 21 – Проект модульного тестирования

Создание теста:

При создании проекта модульного теста генерируется класс для тестирования UnitTest (рисунок 22)

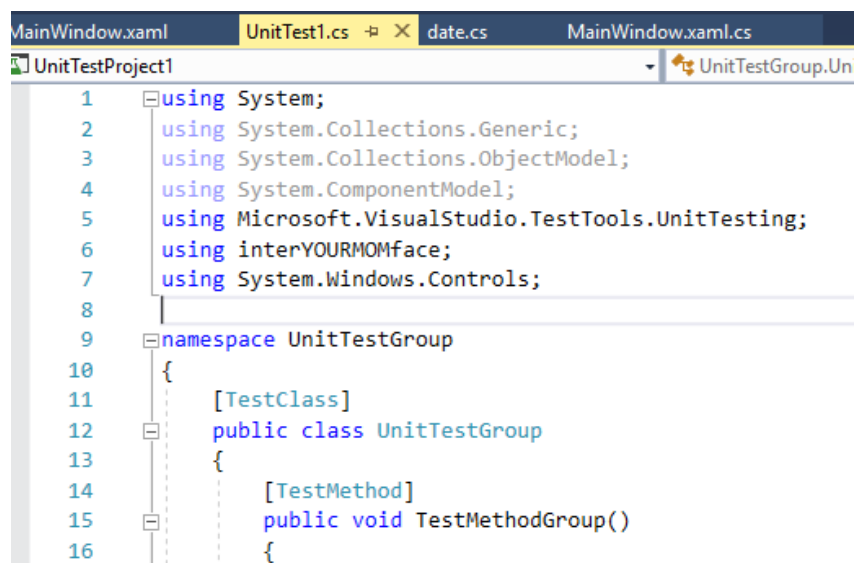


Рисунок 22 – Код созданного класса модульного тестирования

Исходя из содержания классов основной программы, были разработаны следующие юнит-тесты:

1. Тестирование метода, использующегося для вывода данных о всех заемщиках в элемент DataGridView (рисунок 23)

```

[TestMethod]
public void dgv()
{
    Model1Container db = new Model1Container();
    DataGrid dgv = new DataGrid();

    interYOURMOMface.itemsource.dgv(db, dgv);
    Assert.IsNotNull(dgv.Items); //проверка на вывод данных в dgv
}

```

Рисунок 23 – Тестирование вывода данных о всех заемщиках

2. Тестирование метода, использующегося для расчета итоговой суммы кредитования с учетом процентной ставки (рисунок 24)

```

public void sum()
{
    TextBox text = new TextBox();
    TextBox text2 = new TextBox();
    ComboBox combo = new ComboBox();

    text.Text = "15000";
    combo.Items.Add(12);
    combo.SelectedItem = 12;

    interYOURMOMface.rem.remainer(text, combo, text2);
    Assert.IsNotNull(text2.Text); //проверка на подсчет итоговой суммы кредитования
}

```

Рисунок 24 – Тестирование расчета итоговой суммы кредитования

3. Тестирование метода, использующегося для подсчета количества месяцев кредитования (рисунок 25)

```

[TestMethod]
public void period()
{
    DatePicker start = new DatePicker();
    DatePicker end = new DatePicker();
    DatePicker start_card = new DatePicker();
    DatePicker end_card = new DatePicker();
    TextBox sum_text = new TextBox();

    start.SelectedDate = DateTime.Now.Date;
    end.SelectedDate = DateTime.Now.Date;
    start_card.SelectedDate = DateTime.Now.Date;
    end_card.SelectedDate = DateTime.Now.Date;

    interYOURMOMface.date.date_rem(end, start, sum_text, start_card, end_card);
    Assert.IsNotNull(sum_text.Text); //подсчет кол-ва месяцев кредитования
}

```

Рисунок 25 – Тестирование подсчета количества месяцев кредитования

Результаты тестирования формируются классом Assert, который производит сравнение ожидаемых результатов с фактическими.

После построения тестового проекта созданные тесты отображаются в обозревателе тестов (рисунок 26).

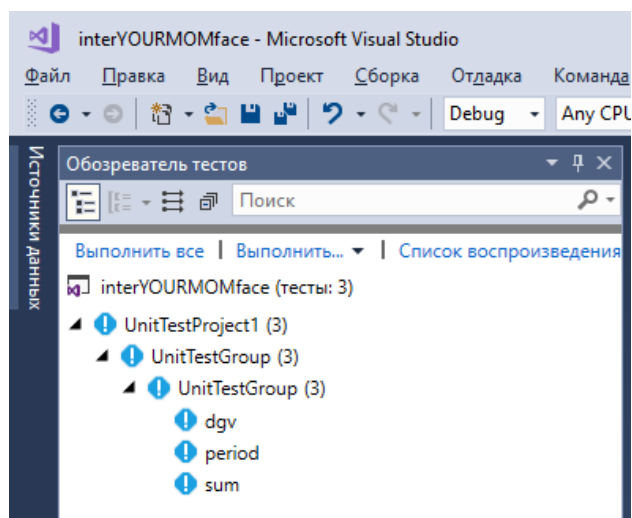


Рисунок 26 – Созданные тесты в обозревателе

При выполнении теста в обозревателе тестов отображаются параметры его выполнения (рисунок 27).

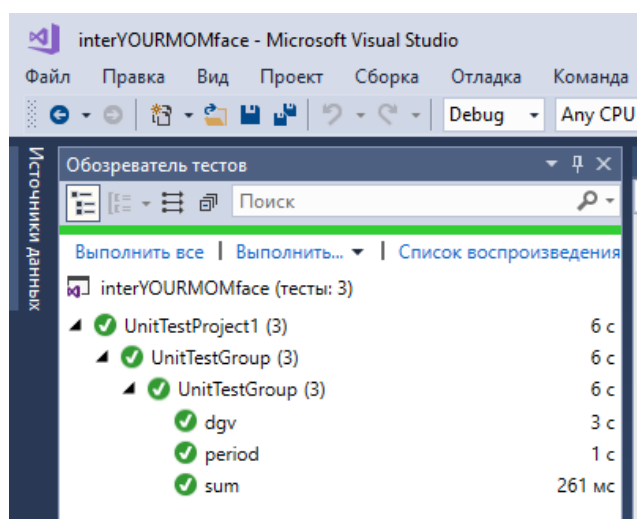


Рисунок 27 – Выполненные тесты в обозревателе тестов

2.13 Аудирование

Аудирование было проведено – Махмадкадамовым Рустамом Шодиевичем

Для данного этапа тестирования были выданы:

- Программа (главное окно программы представлено на рисунке А.1)
- Техническое задание
- Руководство оператора
- Тест-кейсы

Проверка программы на соответствия требованиям ТЗ:

В результате анализа технического задания мною было сформировано следующее замечание: в списке кредитных записей, просроченных или завершенных заемщиком, желательно выделять другим цветом. Данный аспект значительно упростит поиск во время не закрытых заемщиками кредитов или уже завершенных.

Полнота покрытия тестами:

Предоставленные тест-кейсы были выполнены успешно. Замечаний нет

Юзабилити:

Мною были произведены исследования программного продукта, и были выявлены следующие замечания:

1. Для экономии места и ускоренного заполнения поля "пол" и "дата рождения" целесообразнее поместить в одну строку.
2. Поле "код подразделения" необходимо ограничить по вводу символов, так как в РФ установлен стандарт шести чисел.
3. Для экономии места и ускоренного заполнения поля "индекс" и "проверочное слово" целесообразнее поместить в одну строку.
4. Поле "CVV" необходимо ограничить по вводу символов, так для банковских карт установлен стандарт трех чисел.
5. На макете карты необходимо убрать поле "начало действия", так как по стандарту оно не указывается.
6. Для экономии места и ускоренного заполнения поля "код подразделения" и "образование" целесообразнее поместить в одну строку.
7. Функцию поиска по ФИО стоит поместить выше таблицы, так как это позволяет проще сравнивать запрос с выдаваемым результатом, находясь на одном зрительном уровне.
8. В ситуации, когда клиент изъявляет желание получить повторно кредит, вносить его данные заново не будет целесообразным. Вместо этого стоит разместить на форме дополнительную кнопку "выбрать клиента", которая открывала бы форму с перечнем клиентов, уже зарегистрированных клиентов.

Дизайн:

При работе с программным продуктом, прежде всего, обращаешь внимание на внешний вид и на удобство использования. В ходе работы с программой, мною были обнаружены следующие недочёты в дизайне продукта:

1. Шрифт окон подобран слишком тусклый. Он слабо выделяется на фоне структурных элементов (рисунок 28):

The screenshot shows a web application window titled 'Make credit 1.0'. It has three tabs: 'Регистрация клиента в кредитной системе', 'Регистрация кредитной карты', and 'Погашение'. The 'Регистрация клиента в кредитной системе' tab is active. It contains two main sections: 'Данные о клиенте' and 'Данные о кредите'. The 'Данные о клиенте' section includes fields for: ФИО клиента, Паспортные данные (0000 000000), кем выдан, Дата рождения, Пол (radio buttons for M and F), Семейное положение, Образование, Код подразделения, Место работы, Должность (должность), Телефон предприятия (88121234567), Ежемесячные доходы на месте работы (доходы), Место проживания (регистрации) (адрес), Контактный телефон (89111234567), and Эл. почта (email). The 'Данные о кредите' section includes: Процентная ставка, Валюта, Период кредитования (с and по), Цель получения кредита (адрес), Желаемая сумма займа (150000), Кол-во месяцев кредитования, Ежемесячное погашение, and Итого к погашению. At the bottom right, there is a button labeled 'Перейти к регистрации кредитной карты'.

Рисунок 28 – Тусклый шрифт окон

Понятность и полнота Руководства оператора:

Изучив руководство оператора, я пришел к выводу, что данный документ полностью сопоставим в своем содержании с функционалом программного обеспечения и не нуждается в каком-либо редактировании.

Оценка уровня защищенности данных:

При тестировании и внедрении программного продукта в рабочую среду, были учтены такие факторы как:

- Программа была просканирована антивирусным ПО с новейшими базами;
- К программному продукту был организован уровень доступа на должности сотрудника библиотеки;

– Каждый из участников тестирования авторизовался под своей учетной записью на устройстве тестирования (персональном компьютере), с последующим обязательным выходом из учетной записи по окончании тестирования;

– Доступ к программному обеспечению для сторонних лиц (читателей) предоставлен не был.

Оценка качества написания кода:

Осмотр исходников программного продукта показал, что:

- 1) Исходный код программы не структурирован.
- 2) Присутствуют комментарии, упрощающие понимание кода.
- 3) Все переменные соответствуют хранящемуся в них значению.
- 4) Файлы исходных кодов слишком большие, код долго пролистывать и сложно читать.
- 5) Хотелось бы видеть классы, описывающие однотипные действия, так как код повторяется во многих местах.

2.14 Рефакторинг

На основе проведенного аудирования:

1. Шрифт окон теперь ярче и выделен четче:

До рефакторинга (рисунок 29):

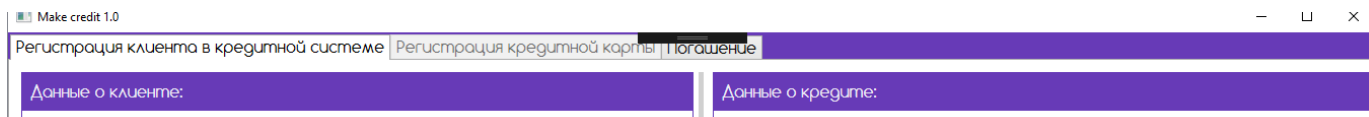


Рисунок 29 – Шрифт окон до рефакторинга

После рефакторинга (рисунок 30):

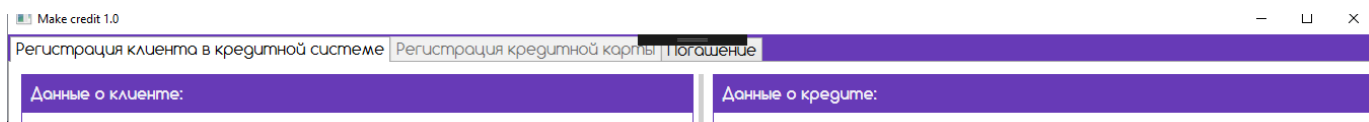


Рисунок 30 – Шрифт окон после рефакторинга

2. Поля "Пол" и "Дата рождения" теперь помещены в одну строку:
До рефакторинга (рисунок 31):

Make credit 1.0

Регистрация клиента в кредитной системе | Регистрация кредитной карты | Пог

Данные о клиенте:

ФИО клиента: _____

Паспортные данные: 0000 000000 _____
кем выдан

Дата рождения: _____

Пол: ☐ М ☐ Ж

Семейное положение: _____

Образование: _____

Код подразделения: _____

Место работы: _____

Рисунок 31 – Положение полей "Пол" и "Дата рождения" до рефакторинга

После рефакторинга (рисунок 32):

Make credit 1.0

Регистрация клиента в кредитной системе | Регистрация кредитной карты | Пог

Данные о клиенте:

ФИО клиента: _____

Паспортные данные: 0000 000000 _____
кем выдан

Дата рождения: _____ Пол: ☐ М ☐ Ж

Семейное положение: _____

Образование: _____ Код подразделения: _____

Место работы: _____

Рисунок 32 – Положение полей "Пол" и "Дата рождения" после рефакторинга

3. Поля "Код подразделения" и "Образование" теперь помещены в одну строку:

До рефакторинга (рисунок 33):

Make credit 1.0

Регистрация клиента в кредитной системе | Регистрация кредитной карты | Лог

Данные о клиенте:

ФИО клиента: _____

Паспортные данные: 0000 000000 _____
 кем выдан _____

Дата рождения: _____

Пол: ☐ М ☐ Ж

Семейное положение: _____

Образование: _____

Код подразделения: 123456789

Место работы: _____

Рисунок 33 – Положение полей "Код подразделения" и "Образование" до рефакторинга

После рефакторинга (рисунок 34):

Make credit 1.0

Регистрация клиента в кредитной системе | Регистрация кредитной карты | Лог

Данные о клиенте:

ФИО клиента: _____

Паспортные данные: 0000 000000 _____
 кем выдан _____

Дата рождения: _____ Пол: ☐ М ☐ Ж

Семейное положение: _____

Образование: _____ Код подразделения: 123456

Место работы: _____

Рисунок 34 – Положение полей "Код подразделения" и "Образование" после рефакторинга

4. Поле "Код подразделения" теперь ограничено по вводу на 6 символов:

До рефакторинга (рисунок 35):

Make credit 1.0

Регистрация клиента в кредитной системе | Регистрация кредитной карты | Логин

Данные о клиенте:

ФИО клиента: _____

Паспортные данные: 0000 000000 _____

_____ кем выдан

Дата рождения: _____

Пол: ☐ М ☐ Ж

Семейное положение: _____ ▾

Образование: _____ ▾

Код подразделения: 123456789

Место работы: _____

Рисунок 35 – Поле "Код подразделения" до рефакторинга

После рефакторинга (рисунок 36):

Make credit 1.0

Регистрация клиента в кредитной системе | Регистрация кредитной карты | Логин

Данные о клиенте:

ФИО клиента: _____

Паспортные данные: 0000 000000 _____

_____ кем выдан

Дата рождения: _____ Пол: ☐ М ☐ Ж

Семейное положение: _____ ▾

Образование: _____ ▾ Код подразделения: 123456

Место работы: _____

Рисунок 36 – Поле "Код подразделения" после рефакторинга

5. Поля "Индекс" и "Проверочное слово" теперь находятся на одной строке:

До рефакторинга (рисунок 37):

Рисунок 37 – Поля "Индекс" и "Проверочное слово" до рефакторинга

После рефакторинга (рисунок 38):

Рисунок 38 – Поля "Индекс" и "Проверочное слово" после рефакторинга

6. Поле "CVV" на теперь ограничено по вводу на 3 символа:

До рефакторинга (рисунок 39):

Make credit 1.0

Регистрация клиента в кредитной системе Регистрация кредитной карты Логин

Данные для регистрации карты:

Тип карты: _____

Код системы Код банка Номер счета Проверочное число CVV

* *** ** ** ***** *

Начало действия Окончание действия

Рисунок 39 – Поле "CVV" до рефакторинга

После рефакторинга (рисунок 40)

Make credit 1.0

Регистрация клиента в кредитной системе Регистрация кредитной карты Логин

Данные для регистрации карты:

Тип карты: _____

Код системы Код банка Номер счета Проверочное число CVV

* *** ** ** ***** *

Начало действия Окончание действия

Индекс 000000 Проверочное слово *СЛОВО*

Рисунок 40 – Поле "CVV" после рефакторинга

7. На макете карты теперь отсутствует поле "Начало действия":

До рефакторинга (рисунок 41):

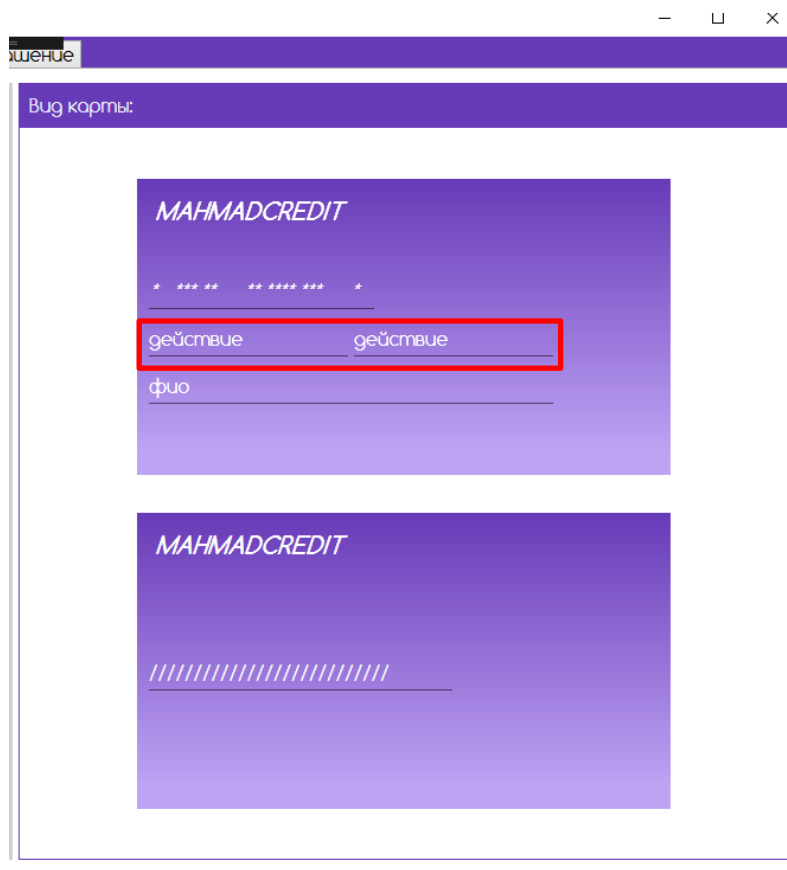


Рисунок 41 – Макет карты до рефакторинга

После рефакторинга (рисунок 42):

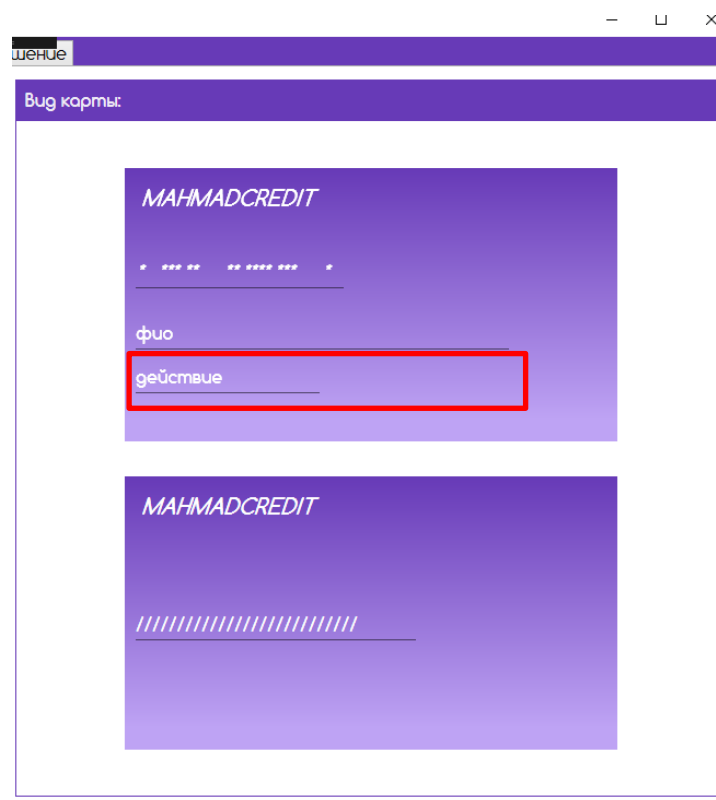


Рисунок 42 – Макет карты после рефакторинга

8. Поле поиска по ФИО теперь находится выше таблицы кредитных записей:

До рефакторинга (рисунок 43):

Make credit 1.0

Регистрация клиента в кредитной системе | Регистрация кредитной карты | Лого

Список заемщиков

| Клиент | Валюта | Начало | Конец | Сумма | Остаток |
|-----------|--------|-----------|-----------|--------|-------------|
| Тимушев Д | EURO | 12/1/2018 | 5/1/2019 | 100000 | 0 |
| Тимушева | RUB | 3/10/2019 | 7/10/2019 | 50000 | 0 |
| Рустам М | RUB | 3/11/2019 | 1/11/2020 | 150000 | 0 |
| Тимушев Д | RUB | 3/12/2019 | 3/15/2019 | 150000 | 153137,9666 |

Поиск по ФИО:

Поиск

ФИО клиента:

Ежемесячное погашение:

Внести ➤

Досрочное погашение:

месяцы:

Внести ➤

Рисунок 43 – Расположение поля поиска до рефакторинга

После рефакторинга (рисунок 44):

Make credit 1.0

Регистрация клиента в кредитной системе | Регистрация кредитной карты | Пог

Список заемщиков

Поиск по ФИО: фuo Поиск

| Клиент | Валюта | Начало | Конец | Сумма | Остаток |
|-----------|--------|-----------|-----------|--------|-------------|
| Тимушев Д | EURO | 12/1/2018 | 5/1/2019 | 100000 | 0 |
| Тимушева | RUB | 3/10/2019 | 7/10/2019 | 50000 | 0 |
| Рустам М | RUB | 3/11/2019 | 1/11/2020 | 150000 | 0 |
| Тимушев Д | RUB | 3/12/2019 | 3/15/2019 | 150000 | 153137,9666 |

ФИО клиента: фuo

Ежемесячное погашение: сумма Внести ➤

Досрочное погашение: месяцы: кол-во Внести ➤

срок: месяцы

Рисунок 44 – Расположение поля поиска после рефакторинга

9. Была добавлена новая кнопка для открытия формы с уже добавленными клиентами:

До рефакторинга (рисунок 45):

Make credit 1.0

Регистрация клиента в кредитной системе | Регистрация кредитной карты | Погашение

Данные о клиенте:

ФИО клиента: _____

Паспортные данные: 0000 000000

кем выдан

Дата рождения: _____

Пол: ☐ М ☐ Ж

Семейное положение: _____

Образование: _____

Код подразделения: _____

Место работы: _____

Должность: должность

Телефон предприятия: 88121234567

Ежемесячные доходы на месте работы: доходы

Место проживания (регистрации): адрес

Контактный телефон: 89111234567

Эл. почта: email

Данные о кредите:

Процентная ставка: _____

Валюта: _____

Период кредитования: с _____ по: _____

Цель получения кредита: адрес

Желаемая сумма займа: 150000

Кол-во месяцев кредитования: _____

Ежемесячное погашение: _____

Итого к погашению: _____

Перейти к регистрации кредитной карты ➡

Рисунок 45 – Главная страница до рефакторинга

После рефакторинга (рисунок 46):

Make credit 1.0

Регистрация клиента в кредитной системе | Регистрация кредитной карты | Погашение

Данные о клиенте:

ФИО клиента: _____

Паспортные данные: 0000 000000

кем выдан

Дата рождения: _____ Пол: ☐ М ☐ Ж

Семейное положение: _____

Образование: _____ Код подразделения: _____

Место работы: _____

Должность: должность

Телефон предприятия: 88121234567

Ежемесячные доходы на месте работы: доходы

Место проживания (регистрации): адрес

Контактный телефон: 89111234567

Эл. почта: email

Данные о кредите:

Процентная ставка: _____

Валюта: _____

Период кредитования: с _____ по: _____

Цель получения кредита: _____

Желаемая сумма займа: 150000

Кол-во месяцев кредитования: _____

Ежемесячное погашение: _____

Итого к погашению: _____

Выбрать клиента

Перейти к регистрации кредитной карты ➡

Рисунок 46 – Главная страница после рефакторинга

10. Была добавлена возможность выбрать уже существующего клиента – открытие новой формы с уже зарегистрированными клиентами (рисунок 47):

| Клиент | Паспортные данные |
|-----------------------------|-------------------|
| Тимушев Федор Алексеевич | ТП №141 |
| Тимушева Елизавета Алексеев | Отдел милиции №1 |
| Рустам Махмадкадамов | Отдел милиции №1 |
| Тимушев Федор Алексеевич | ТП №141 |

Рисунок 47 – Добавленная форма с клиентами после рефакторинга

11. Также была исправлена формула кредитования с простой математической на годовую по специальной формуле:

До рефакторинга (рисунок 48):

```
int res = (((sum/100)*per)+sum)/ month;
```

Рисунок 48 – Формула кредитования до рефакторинга

После рефакторинга (рисунок 49):

```
double rem = sum * (i * (Math.Pow((1+i), n)))/((Math.Pow((1 + i), n))-1);
```

Рисунок 49 – Формула кредитования после рефакторинга

12. Стала доступна возможность закрытия кредитных записей (запись выделяется зеленым цветом и становится не доступна):

До рефакторинга (рисунок 50):

Make credit 1.0

Регистрация клиента в кредитной системе | Регистрация кредитной карты | Поиск

Список заемщиков

| Клиент | Валюта | Начало | Конец | Сумма | Остаток |
|-----------|--------|-----------|-----------|--------|-------------|
| Тимушев Д | EURO | 12/1/2018 | 5/1/2019 | 100000 | 0 |
| Тимушева | RUB | 3/10/2019 | 7/10/2019 | 50000 | 0 |
| Рустам М | RUB | 3/11/2019 | 1/11/2020 | 150000 | 0 |
| Тимушев Д | RUB | 3/12/2019 | 3/15/2019 | 150000 | 153137,9666 |

Рисунок 50 – Кредитные записи до рефакторинга

После рефакторинга (рисунок 51):

Make credit 1.0

Регистрация клиента в кредитной системе | Регистрация кредитной карты | Поиск

Список заемщиков

Поиск по ФИО: фuo Поиск

| Клиент | Валюта | Начало | Конец | Сумма | Остаток |
|-----------|--------|-----------|-----------|--------|-------------|
| Тимушев Д | EURO | 12/1/2018 | 5/1/2019 | 100000 | 0 |
| Тимушева | RUB | 3/10/2019 | 7/10/2019 | 50000 | 0 |
| Рустам М | RUB | 3/11/2019 | 1/11/2020 | 150000 | 0 |
| Тимушев Д | RUB | 3/12/2019 | 3/15/2019 | 150000 | 153137,9666 |

Рисунок 51 – Кредитные записи после рефакторинга

13. Добавлен транскрипт на имя в карте при создании макета:

До рефакторинга (рисунок 52):

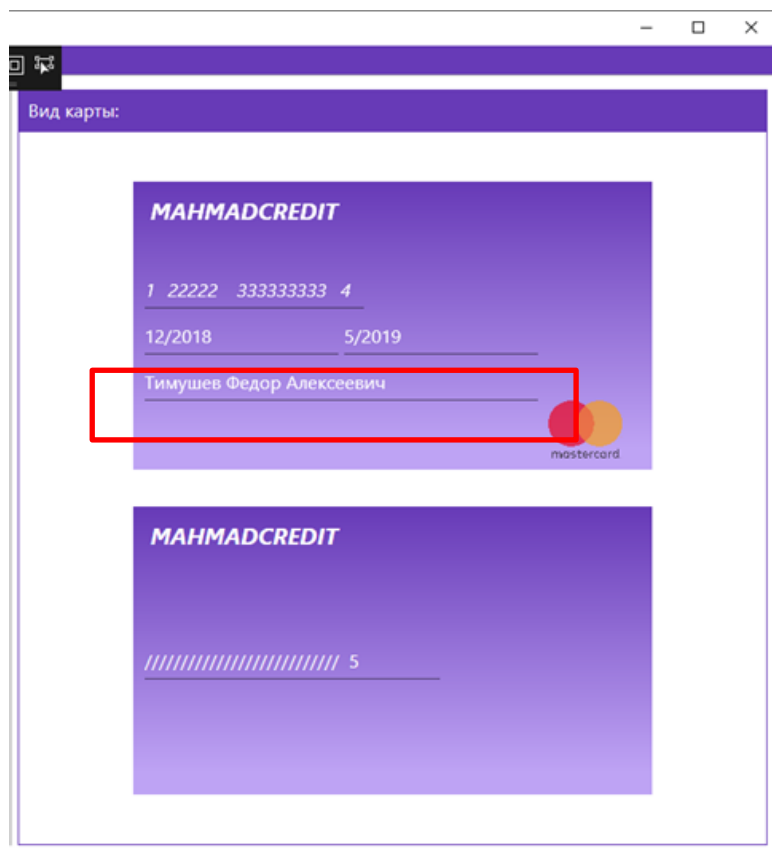


Рисунок 52 – Имя на макете карты до рефакторинга

После рефакторинга (рисунок 53):

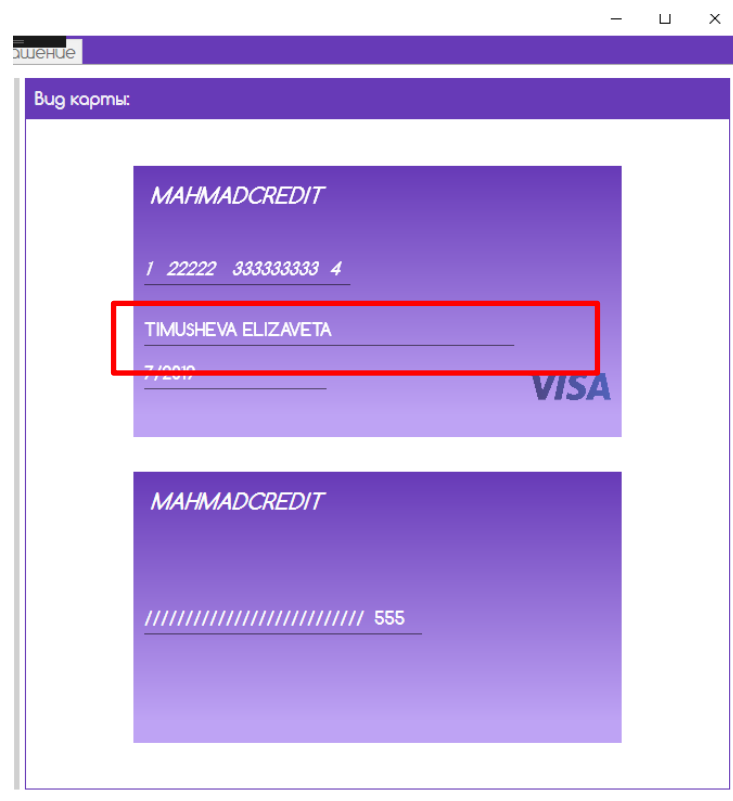


Рисунок 53 – Имя на макете карты после рефакторинга

ЗАКЛЮЧЕНИЕ

Разработанная в ходе выполнения курсового проекта программа удовлетворяет всем требованиям технического задания, что подтверждается протоколом испытаний.

Разработанная программа может быть использована в практических целях.

СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

- 1 А. Хейлсберг, М. Торгерсен, С. Вилтамут, П. Голд ., Язык программирования С#, 4-е издание, СПб:-ПИТЕР, 2011
- 2 Фленов М., Библия С#, 3-е издание, СПб:-БХВ, 2016
- 3 Культин Н., Microsoft Visual С# в задачах и примерах, 2-е издание, СПб:-БХВ, 2015
- 4 Язык С# [Электронный ресурс]: ru.wikipedia.org/wiki/C_Sharp
- 5 Форум программистов и сисадминов Киберфорум [Электронный ресурс]: www.cyberforum.ru

ПРИЛОЖЕНИЕ А

(справочное)

Основные вкладки

The screenshot shows the 'Make credit 1.0' application window with the 'Регистрация клиента в кредитной системе' tab selected. The window is divided into two main sections: 'Данные о клиенте' (Client Data) on the left and 'Данные о кредите' (Credit Data) on the right.

Данные о клиенте:

- ФИО клиента: _____
- Паспортные данные: 0000 000000, кем выдан: _____
- Дата рождения: _____
- Пол: ☐ М ☐ Ж
- Семейное положение: _____
- Образование: _____
- Код подразделения: _____
- Место работы: _____
- Должность: должность
- Телефон предприятия: 88121234567
- Ежемесячные доходы на месте работы: доходы
- Место проживания (регистрации): адрес
- Контактный телефон: 89111234567
- Эл. почта: email

Данные о кредите:

- Процентная ставка: _____
- Валюта: _____
- Период кредитования: с _____ по: _____
- Цель получения кредита: адрес
- Желаемая сумма займа: 150000
- Кол-во месяцев кредитования: _____
- Ежемесячное погашение: _____
- Итого к погашению: _____

At the bottom right, there is a button: **Перейти к регистрации кредитной карты ►**

Рисунок А.1 – Загрузка программы

The screenshot shows the 'Make credit 1.0' application window with the 'Регистрация кредитной карты' tab selected. The window is divided into two main sections: 'Данные для регистрации карты' (Card Registration Data) on the left and 'Вид карты' (Card View) on the right.

Данные для регистрации карты:

- Тип карты: _____
- Код системы: _____ Код банка: _____ Номер счета: _____ Проверочное число: _____ CVV: _____
- Начало действия: _____ Окончание действия: _____
- Индекс: 000000
- Проверочное слово: *слово*

At the bottom left, there are three buttons:

- Внести данные на карту ►**
- Создать кредитную запись ✓**
- Полная отмена регистрации ✕**

Вид карты:

The card view shows two identical representations of a credit card. Each card has the logo 'MAHMADCREDIT' at the top, followed by a card number (represented by asterisks), an expiration date (represented by two 'действие' fields), and a name (represented by a 'фио' field).

Рисунок А.2 – Вкладка «Регистрация кредитной карты»

Make credit 1.0

Регистрация клиента в кредитной системе

Регистрация кредитной карты

Погашение

Список заемщиков

| Клиент | Валюта | Начало | Конец | Сумма | Остаток |
|------------|--------|-----------|-----------|--------|---------|
| Тимушев Ф | EURO | 12/1/2018 | 5/1/2019 | 100000 | 44000 |
| Рустам Ма | DOLLAR | 12/1/2018 | 5/1/2019 | 150000 | 162000 |
| Коваленко | DOLLAR | 12/1/2018 | 5/1/2019 | 150000 | 162000 |
| Скагровски | DOLLAR | 12/1/2018 | 5/1/2019 | 150000 | 162000 |
| Кобжицкая | DOLLAR | 12/1/2018 | 5/1/2019 | 150000 | 162000 |
| Тимушева | RUB | 3/10/2019 | 7/10/2019 | 50000 | 55000 |

Поиск по ФИО

фио

Поиск

ФИО клиента:

фио

Ежемесячное погашение:

сумма

Внести ➤

Досрочное погашение:

месяцы

кол-во

Внести ➤

Договор

Печать договора

Условия

Печать условий

Рисунок А.3 – Вкладка «Погашение»

ПРИЛОЖЕНИЕ Б
(справочное)
Промежуточные результаты тест-кейсов

Регистрация клиента в кредитной системе Регистрация кредитной карты Погашение

Данные о клиенте:

ФИО клиента:

Тимушева Елизавета Алексеевна

Паспортные данные:

1234 567890

31.07.2018

Отдел милиции №1

Дата рождения:

15.07.2004

Пол:

☐ М ☒ Ж

Семейное положение:

не в браке

Образование:

полное среднее

Код подразделения:

4321

Место работы:

Гимназия №17

Должность:

ученик

Телефон предприятия:

88121234567

Ежемесячные доходы на месте работы:

15000

Место проживания (регистрации):

ул. 1-я Утиная, д. 32, кв. 104

Контактный телефон:

89111234567

Эл. почта:

liza@gmail.com

Рисунок Б.1 – Заполненная форма на вкладке «Регистрация кредитной карты» в окнах «Данные о клиенте»

Данные о кредите:

Процентная ставка:

10

▼

Валюта:

RUB

▼

Период кредитования: с

10.03.2019

📅

по:

10.07.2019

📅

Цель получения кредита:

обучение в вузе

Желаемая сумма займа:

50000

Кол-во месяцев кредитования:

4

Ежемесячное погашение:

13750

Итого к погашению:

55000

Перейти к регистрации кредитной карты ➤

Рисунок Б.2 – Заполненная форма на вкладке «Регистрация кредитной карты» в окнах «Данные о кредите»

Make credit 1.0

Регистрация клиента в кредитной системе | Регистрация кредитной карты | Погашение

Данные для регистрации карты:

Тип карты: **VISA**

| Код системы | Код банка | Номер счета | Проверочное число | CVV |
|-------------|-----------|-------------|-------------------|-----|
| 1 | 22222 | 333333333 | 4 | 5 |

Начало действия 10.03.2019 Окончание действия 10.07.2019

Индекс 197375

Проверочное слово брат

Внести данные на карту ➔

Создать кредитную запись ☒

Полная отмена регистрации ✕

Рисунок Б.3 – Заполненная форма на вкладке «Регистрация кредитной карты» в окне «Данные для регистрации карты»

Вид карты:

MAHMADCREDIT

1 22222 333333333 4

3/2019 7/2019

Тимушева Елизавета Алексеевна

VISA

MAHMADCREDIT

//////////////////// 5

Рисунок Б.4 – Заполненная форма на вкладке «Регистрация кредитной карты» в окне «Вид карты»

ПРИЛОЖЕНИЕ В

(информационное)
Листинг программы

MainWindow (главное окно программы)

```
using NickBuhro.Translit;

using System;

using System.Data.Entity;

using System.Data.Entity.Core.Objects;

using System.Linq;

using System.Windows;

using System.Windows.Controls;

using System.Windows.Input;

using System.Windows.Media;

using System.Windows.Media.Imaging;

namespace interYOURMOMface
{
    /// <summary>
    /// Логика взаимодействия для MainWindow.xaml
    /// </summary>

    public partial class MainWindow : Window
    {

        String sex;

        ModelContainer db;

        DateTime passDate, birthDate, creditStart, creditEnd, cardStart, cardEnd;

        public MainWindow()
        {
```


InitializeComponent();

db = new Model1Container();

tel_number.MaxLength = 11;

education_place_code.MaxLength = 6;

work_place_number.MaxLength = 11;

sum_month.MaxLength = 2;

some_month.MaxLength = 2;

system_code.MaxLength = 1;

bank_code.MaxLength = 5;

bill_number.MaxLength = 9;

check_number.MaxLength = 1;

cvv.MaxLength = 3;

//добавление процентов в комбобокс

percent.Items.Add("8");

percent.Items.Add("10");

percent.Items.Add("12");

//добавление валют в комбобокс

currency.Items.Add("RUB");

currency.Items.Add("DOLLAR");

currency.Items.Add("EURO");

//добавление вариантов семейного положения в комбобокс

family_status.Items.Add("в браке");

family_status.Items.Add("не в браке");

```

//добавление вариантов образования в комбобокс
education.Items.Add("полное среднее");
education.Items.Add("среднее специальное");
education.Items.Add("полное специальное");

//добавление типов карт в комбобокс
card_type.Items.Add("VISA");
card_type.Items.Add("MASTERCARD");
card_type.Items.Add("MAESTRO");

dgv.ItemsSource = new observe<credit>(db.credit.ToList()).List.ToBindingList();

}

private void incert_Click(object sender, RoutedEventArgs e)
{
    var r = ((credit)dgv.SelectedItem);
    decimal s = decimal.Parse( r.remainer);
    r.remainer = (s - decimal.Parse(month_check.Text)).ToString();

    db.SaveChanges();
    itemsourse.dgv(db, dgv);
}

private void incert_month_Click(object sender, RoutedEventArgs e)
{
    int mon = Convert.ToInt32(month_check.Text);
    int some = Convert.ToInt32(some_month.Text);
    int mon2 = Convert.ToInt32(some_month_2.Text);
    var r = ((credit)dgv.SelectedItem);

```

```

decimal s = decimal.Parse(r.remainer);

if ( mon2 < some)
{
    MessageBox.Show("Кол-во месяцев не может быть больше оставшегося срока!");
}

r.remainer = (s -(mon*some)).ToString();

db.SaveChanges();
}

private void dgv_SelectionChanged(object sender, SelectionChangedEventArgs e)
{
    var r = ((credit)dgv.SelectedItem);
    if(r!=null)
    {
        dogovor.Items.Clear();
        usloviya.Items.Clear();

        month_check.Text = r.month_sum.ToString();
        client_fio.Text = r.customer.fio.ToString();

        var f = ((r.credit_end.Year - DateTime.Now.Year) * 12) + r.credit_end.Month -
        DateTime.Now.Month;

        // var f2 = r.credit_end.Subtract(DateTime.Now).Days / (365.25 / 12);

        if (f <= 0)
        {
            some_month_2.Text = "ДОЛГ!";
            MessageBox.Show("У клиента имеется задолженность!");
        }
        else

```

```

{
    some_month_2.Text = Convert.ToString(f);
}

dogovor.Items.Add(r.customer.fio);
dogovor.Items.Add(r.customer.birthdate);
dogovor.Items.Add(r.customer.live_place);
dogovor.Items.Add(r.customer.passport_info);
dogovor.Items.Add(r.customer.passport_number);
dogovor.Items.Add(r.customer.credit_reason);

usloviya.Items.Add(r.percent_set);
usloviya.Items.Add(r.currency);
usloviya.Items.Add(r.credit_start);
usloviya.Items.Add(r.credit_end);
usloviya.Items.Add(r.sum);
usloviya.Items.Add(r.card.card_type);
}
}

private void res_month_incert_SelectionChanged(object sender, RoutedEventArgs e)
{
    if (sum_credit.Text != "")
    {
        rem.remainer(sum_credit, percent, sum_month, res_month_incert);
    }
}

private void random_check_SelectionChanged(object sender, RoutedEventArgs e)
{
    int mon = Convert.ToInt32(month_check.Text);
    int some = Convert.ToInt32(some_month.Text);

```

```
random_check.Text = (mon * some).ToString();  
}
```

```
private void fio_PreviewTextInput(object sender, TextCompositionEventArgs e)  
{  
    if (Char.IsLetter(e.Text, 0)) e.Handled = false;  
    else e.Handled = true;  
}
```

```
private void passport_number_PreviewTextInput(object sender, TextCompositionEventArgs e)  
{  
    if (Char.IsDigit(e.Text, 0)) e.Handled = false;  
    else e.Handled = true;  
}
```

```
private void to_card_Click(object sender, RoutedEventArgs e)  
{  
    MessageBoxResult res_to_card = MessageBox.Show("Принять заявку и перейти к  
    регистрации К/К?", "Подтвердить?", MessageBoxButton.YesNo,  
    MessageBoxImage.Question);  
    if (res_to_card == MessageBoxResult.Yes)  
    {  
        main_tabcon.SelectedItem = card_reg;  
        main_reg.IsEnabled = false;  
        rem_reg.IsEnabled = false;  
    }  
    else  
    {  
        MessageBox.Show("Заявка отклонена!");  
    }  
}
```

```

private void full_reset_Click(object sender, RoutedEventArgs e)
{
    MessageBoxResult reset = MessageBox.Show("Отменить регистрацию клиента и
    вернуться на главную страницу?", "Отмена", MessageBoxButton.YesNo,
    MessageBoxImage.Question);

    if (reset == MessageBoxResult.Yes)
    {
        main_reg.IsEnabled = true;

        rem_reg.IsEnabled = true;

        main_tabcon.SelectedItem = main_reg;

        card_reg.IsEnabled = false;

    }
}

```

```

private void Button_Click(object sender, RoutedEventArgs e)
{
    //проверка на пол

    if (male.IsChecked == true) { sex = "men"; }
    else if (female.IsChecked == true) { sex = "female"; }

    //переменная выбранной даты для паспорта и даты рождения, начала и конца
    кредитования, действия карты

    passDate = passport_date.SelectedDate.Value.Date;

    birthDate = birthdate.SelectedDate.Value.Date;

    creditStart = start.SelectedDate.Value.Date;

    creditEnd = end.SelectedDate.Value.Date;

    cardStart = card_start.SelectedDate.Value.Date;

    cardEnd = card_end.SelectedDate.Value.Date;

    //информация о деталях кредита

```

```

db.Database.ExecuteSqlCommand("INSERT INTO customer VALUES({0}, {1}, {2}, {3}, {4},
{5}, {6}, {7}, {8}, {9}, {10}, {11}, {12}, {13}, {14}, {15}, {16}) ",

    fio.Text, passport_number.Text, passDate, passport_info.Text, sex, family_status.Text,
    birthDate, tel_number.Text, email.Text, education.Text, education_place_code.Text,
    live_place.Text, work_place.Text, work_place_number.Text, work_post.Text,
    month_income.Text, credit_reason.Text);

db.Database.ExecuteSqlCommand("INSERT INTO card VALUES({0}, {1}, {2}, {3}, {4}, {5},
{6}, {7}, {8}, {9}) ",

    IL.Text, card_type.Text, system_code.Text, bank_code.Text, bill_number.Text,
    check_number.Text, cvv.Text, cardStart, cardEnd, check_word.Text);

db.Database.ExecuteSqlCommand("INSERT INTO credit VALUES({0}, {1}, {2}, {3}, {4}, {5},
{6}, {7}, {8}, {9}) ",

    percent.Text, currency.Text, creditStart, creditEnd, remainder.Text, sum_credit.Text,
    res_month_incert.Text, sum_month.Text, db.card.ToList().Last().card_id,
    db.customer.ToList().Last().customer_id);

main_reg.IsEnabled = true;

rem_reg.IsEnabled = true;

dgv.ItemsSource = new observe<credit>(db.credit.ToList()).List.ToBindingList();

main_tabcon.SelectedItem = main_reg;

MessageBox.Show("Кредитная запись добавлена!");

}

private void sum_month_SelectionChanged(object sender, RoutedEventArgs e)
{
    if (DateTime.Today < end.SelectedDate)
    {
        date.date_rem(end, start, sum_month, card_start, card_end);
    }
    else
    {
        MessageBox.Show("Конечная дата должна быть позже начала!");
    }
}

```

}

}

private void exit_MouseDoubleClick(object sender, MouseButtonEventArgs e)

{

System.Windows.Application.Current.Shutdown();

}

private void card_type_SelectionChanged(object sender, SelectionChangedEventArgs e)

{

if(card_type.SelectedItem == "VISA")

{

img.Source = new BitmapImage(new Uri("pack://application:,,,/res/visa.png"));

img_Copy.Source = new BitmapImage(new Uri("pack://application:,,,/res/visa.png"));

}

else

if (card_type.SelectedItem == "MASTERCARD")

{

img.Source = new BitmapImage(new Uri("pack://application:,,,/res/mastercard.png"));

img_Copy.Source = new BitmapImage(new Uri("pack://application:,,,/res/mastercard.png"));

}

else

if (card_type.SelectedItem == "MAESTRO")

{

img.Source = new BitmapImage(new Uri("pack://application:,,,/res/maestro.png"));

img_Copy.Source = new BitmapImage(new Uri("pack://application:,,,/res/maestro.png"));

}

}


```

private void Already_reg_client_Click(object sender, RoutedEventArgs e)
{

    clients client_window = new clients();
    if(client_window.ShowDialog() == true)
    {
        var cus = client_window.GetCustomer;
        fio.Text = cus.fio;
        passport_number.Text = cus.passport_number;
        passport_date.Text = Convert.ToString(cus.passport_date);
        passport_info.Text = cus.passport_info;
        birthdate.Text = Convert.ToString(cus.birthdate);
        if (cus.sex == "male")
        {
            male.IsChecked = true;
        }
        else
        {
            female.IsChecked = true;
        }
        family_status.Text = cus.family_status;
        education.Text = cus.education;
        education_place_code.Text = cus.education_place_code;
        work_place.Text = cus.work_place;
        work_post.Text = cus.work_post;
        work_place_number.Text = cus.work_tel_number;
        month_income.Text = cus.month_income;
        live_place.Text = cus.live_place;
        tel_number.Text = cus.tel_number;
        email.Text = cus.email;
    }
}

```

}

}

private void Dgv_LoadingRow(object sender, DataGridRowEventArgs e)

{

if (((credit)(e.Row.DataContext)).remainder == "0")

{

e.Row.Background = new SolidColorBrush(Color.FromArgb(85, 145, 255, 133));

e.Row.IsEnabled = false;

}

if (((credit)(e.Row.DataContext)).credit_end < DateTime.Today)

e.Row.Background = new SolidColorBrush(Color.FromArgb(85, 207, 91, 91));

}

private void to_card1_Click(object sender, RoutedEventArgs e)

{

res_card.Text = "//////////////////// " + Convert.ToString(cvv.Text);

system_code_Copy.Text = Convert.ToString(system_code.Text);

bank_code_Copy.Text = Convert.ToString(bank_code.Text);

bill_number_Copy.Text = Convert.ToString(bill_number.Text);

check_number_Copy.Text = Convert.ToString(check_number.Text);

*card_period_end.Text = Convert.ToString(card_end.SelectedDate.Value.Month) + "/" +
Convert.ToString(card_end.SelectedDate.Value.Year);*

var fi = fio.Text.Split(' ');

*var latin = Transliteration.CyrillicToLatin(string.Concat(fi[0], " ", fi[1]),
NickBuhro.Translit.Language.Russian);*

card_fio.Text = latin.ToUpper();

}

```

private void dogovor1_Click(object sender, RoutedEventArgs e)
{
    PrintDialog pd = new PrintDialog();
    if ((pd.ShowDialog() == true))
    {
        //use either one of the below
        pd.PrintVisual(dogovor as Visual, "printing as visual");
    }
}

```

```

private void usloviya1_Click(object sender, RoutedEventArgs e)
{
    PrintDialog pd = new PrintDialog();
    if ((pd.ShowDialog() == true))
    {
        //use either one of the below
        pd.PrintVisual(usloviya as Visual, "printing as visual");
    }
}

```

```

private void btn_search_Click(object sender, RoutedEventArgs e)
{
    if(search.Text == "")
    {
        dgv.ItemsSource = new observe<credit>(db.credit.ToList()).List.ToBindingList();
    }
    else
    {
        dgv.ItemsSource = null;
    }
}

```

```

        if (db.credit.Any(x => x.customer.fio == search.Text))
        {
            dgv.ItemsSource = db.credit.Where(x => x.customer.fio == search.Text).ToList();
        }

        private void remainder_SelectionChanged(object sender, RoutedEventArgs e)
        {
            double months = Convert.ToDouble(sum_month.Text);
            double res_month = Convert.ToDouble(res_month_incert.Text);
            remainder.Text = Convert.ToString(months * res_month);
        }
    }
}

```

Client (окно с клиентами)

```
using System;
using System.Collections.Generic;
using System.Data.Entity;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows;
using System.Windows.Controls;
using System.Windows.Data;
using System.Windows.Documents;
using System.Windows.Input;
using System.Windows.Media;
using System.Windows.Media.Imaging;
using System.Windows.Shapes;

namespace interYOURMOMface
{
    /// <summary>
    /// Логика взаимодействия для clients.xaml
    /// </summary>
    public partial class clients : Window
    {
        Model1Container db = new Model1Container();
        public clients()
        {
            InitializeComponent();
            client_dgv.ItemsSource = new observe<customer>(db.customer.ToList()).List.ToBindingList();
        }
    }
}
```

```
public customer GetCustomer { get; set; }
```

```
private void Select_client_btn_Click(object sender, RoutedEventArgs e)
```

```
{
```

```
var r = ((customer)client_dgv.SelectedItem);
```

```
GetCustomer = r;
```

```
this.DialogResult = true;
```

```
this.Close();
```

```
}
```

```
}
```

```
}
```

Date.cs (Класс для подсчета кол-ва месяцев кредитования)

```
using System;

using System.Collections.Generic;

using System.Data.Entity;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows;

using System.Windows.Controls;

namespace interYOURMOMface

{

    public class date

    {

        public static void date_rem(DatePicker end_pick, DatePicker start_pick, TextBox sum,
        DatePicker card_st, DatePicker card_ed)

        {

            int res_month = (((end_pick.SelectedDate.Value).Year - (start_pick.SelectedDate.Value).Year) *
            12) + (end_pick.SelectedDate.Value).Month - (start_pick.SelectedDate.Value).Month;

            if (res_month == 1 || res_month == 0)

            { MessageBox.Show("Кредит минимум от 2-ух месяцев!"); }

            else

            {

                sum.Text = Convert.ToString(res_month);

                card_st.SelectedDate = start_pick.SelectedDate.Value.Date;

                card_ed.SelectedDate = end_pick.SelectedDate.Value.Date;

            }

        }

    }

}
```

ItemSource.cs (Класс для синхронизации таблиц с базой данных)

```
using System;

using System.Collections.Generic;

using System.Data.Entity;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Controls;


namespace interYOURMOMface
{
    public class itemsource
    {
        public static void dgv (ModelIContainer db, DataGrid dgv)
        {
            dgv.ItemsSource = new observe<credit>(db.credit.ToList()).List.ToBindingList();
        }
    }
}
```


Observe.cs (Класс, предназначенный для автоматической синхронизации таблиц и списков с содержимым БД)

using System;

using System.Collections.Generic;

using System.Collections.ObjectModel;

using System.ComponentModel;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace interYOURMOMface

{

class observe<T> : INotifyPropertyChanged

{

public observe()

{

List = new ObservableCollection<T>();

}

public observe(List<T> list)

: this()

{

list.ForEach(x => List.Add(x));

}

public ObservableCollection<T> List { get; set; }

public event PropertyChangedEventHandler PropertyChanged

{

add

{

((INotifyPropertyChanged)List).PropertyChanged += value;

}

remove

```
{  
  ((INotifyPropertyChanged)List).PropertyChanged -= value;  
}  
}  
}  
}
```

Rem.cs (Класс для подсчета итоговой суммы кредитования)

```
using System;

using System.Collections.Generic;

using System.Data.Entity;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Controls;

namespace interYOURMOMface
{
    public class rem
    {
        public static void remainder (TextBox tb1, ComboBox cmb, TextBox tb2, TextBox tb3)
        {
            double sum = Convert.ToDouble(tb1.Text);

            double i = Convert.ToDouble(cmb.Text)/100/12;

            double n = Convert.ToDouble(tb2.Text);

            double rem = sum * (i * (Math.Pow((1+i), n)))/((Math.Pow((1 + i), n))-1);

            tb3.Text = Convert.ToString(rem);
        }
    }
}
```