

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий
Информационные системы

УТВЕРЖДАЮ
Зав. кафедрой ИС
_____ П.П. Дьячук
подпись инициалы, фамилия
« _____ » _____ 2019г.

БАКАЛАВРСКАЯ РАБОТА

09.03.02 Информационные системы и технологии

Разработка приложения «Персональный бухгалтер»

Руководитель _____ доцент, к.н.т. И.А. Легалов
подпись, дата

Выпускник _____ А.В. Кузнецова
подпись, дата

Нормоконтролер _____ ст.преподаватель И.А. Легалов
подпись, дата

Красноярск 2019

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Институт космических и информационных технологий
Информационные системы

УТВЕРЖДАЮ
Заведующий кафедрой ИС
_____ П.П. Дьячук
подпись инициалы, фамилия
«__» _____ 2019г.

**ЗАДАНИЕ
НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ
в форме бакалаврской работы**

Студенту Кузнецовой Анне Витальевне.

Группа КИ15-13Б Направление 09.03.02.

Информационные системы и технологии.

Тема выпускной квалификационной работы: Разработка приложения «Персональная бухгалтерия».

Утверждена приказом по университету №7237/с от 24.05.2019.

Руководитель ВКР: И.А. Легалов, к.т.н., доцент кафедры «Информационные системы и технологии» ИКИТ СФУ.

Исходные данные для ВКР: Требования к разрабатываемому приложению, рекомендации руководителя.

Перечень разделов ВКР: Введение, общие сведения, инструменты реализации, программная реализация, заключение, список сокращений, список использованных источников.

Перечень графического материала: Презентация, выполненная в Microsoft Office PowerPoint.

Руководитель ВКР

И.А. Легалов

подпись

Задание принял к исполнению

А.В. Кузнецова

подпись

«__» _____ 2019г.

РЕФЕРАТ

Выпускная квалификационная работа по теме «Разработка приложения «Персональная бухгалтерия»» содержит 50 страниц текстового документа 43 рисунка, 1 таблицу и 19 использованных источников.

ПРИЛОЖЕНИЕ, РАСЧЁТ, ДЕНЕЖНЫЕ СРЕДСТВА, ПЕРСОНАЛЬНАЯ БУХГАЛТЕРИЯ.

Целью выпускной квалификационной работы является автоматизация процесса ведения персонального бухгалтерского учета денежных средств.

Для достижения поставленной цели были поставлены следующие задачи:

1. Изучение предметной области;
2. Разработка функционала приложения для ведения персонального бухгалтерского учета денежных средств;
3. Реализация приложения.

В результате выпускной квалификационной работы было разработано приложение, которое поможет пользователю в учёте денежных средств.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	5
1 Общие сведения.....	6
1.1 Движения денежных средств	6
1.1.1 Доход	6
1.1.2 Расход	6
1.1.3 Перевод средств	7
1.1.4 Долг	8
1.1.5 Заём.....	8
1.2 Анализ конкурентов	9
1.2.1 MoneyTracker	9
1.2.2 Домашняя экономика	15
2 Инструменты реализации	22
2.1 Среда разработки Embarcadero RAD Studio Delphi XE8	22
2.2 Библиотека VCL	24
2.2.1 Иерархия классов.....	24
2.3 База данных MS Access	27
2.3.1 Основные понятия	28
2.3.2 Обеспечение целостности данных	31
2.4 MS Excel.....	31
2.4.1 Возможности MS Excel	32
2.4.2 Работа с формулами.....	33
3 Программная реализация	35
3.1 Реализация базы данных.	35

3.2 Реализация приложения «Персональный Бухгалтер»	39
3.2.1 Добавление записи.....	39
3.2.2 Фильтрация таблицы	41
3.2.3 Перевод денежных средств	41
3.2.4 Фактические счета денежных средств.....	42
3.2.5 Сокрытие данных.....	43
3.2.6 Статистика.....	43
3.2.7 Регистрация и авторизация пользователя.....	45
ЗАКЛЮЧЕНИЕ	47
СПИСОК СОКРАЩЕНИЙ	48
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	49

ВВЕДЕНИЕ

Каждый день нам приходиться проводить какие-либо манипуляции с денежными средствами: оплата коммунальных услуг, покупка продуктов, оплата проезда на общественном транспорте и так далее. А также иногда приходится давать в долг или занимать у кого-то. И вся информация об денежных манипуляциях находится у нас в голове или записана на бумажном носителе. Но эта информация имеет обыкновение забываться, в случае с запоминанием в голове, или теряться в случае бумажного носителя.

Сейчас в цифровой век у каждого имеется смартфон и/или компьютер, где может содержаться информация, например, в формате таблиц. Однако это не совсем удобно, приходится создавать несколько таблиц для учета транзакции, что тоже неудобно.

Поэтому для выпускной квалификационной работы была выбрана тема «Разработка приложения «Персональная бухгалтерия»», которое призвано помочь в учете денежных транзакций, а также имело бы дружественный интерфейс.

Цель работы – автоматизация процесса ведения персонального бухгалтерского учета денежных средств.

Для достижения поставленной цели были поставлены следующие задачи:

1. Изучение предметной области;
2. Разработка функционала приложения для ведения персонального бухгалтерского учета денежных средств;
3. Реализация приложения.

1 Общие сведения

1.1 Движения денежных средств

Движение денежных средств – это движение, то есть доход или расход денежных средств, в результате какой-либо деятельности. Помимо этих движений, участвующих в повседневной жизни, есть также перевод средств, заём и долг.

1.1.1 Доход

Доход - денежные средства или материальные ценности, полученные государством, физическим или юридическим лицом в результате какой-либо деятельности за определённый период времени. Доход может быть пассивным или активным:

- Пассивный доход – доход, поступающий получателя без каких-либо усилий с его стороны, например, сдача собственности в аренду, может не являться постоянным источником поступления денежных средств;
- Активный доход – доход, получаемый в результате какой-либо деятельности за какой-либо промежуток времени. Является постоянным источником денежных средств. Примером может служить заработка плата или стипендия.

1.1.2 Расход

Расход – затраты (стоимостная оценка ресурсов) в процессе деятельности, приводящие к уменьшению средств на счету. Расходы можно классифицировать по нескольким критериям:

По важности:

- Необходимые/обязательные расходы: жизненно необходимые расходы: расходы на питание, коммунальные услуги, на зимнюю одежду и тому подобное.

По сути это то, что обеспечивает минимальный прожиточный минимум человеку. Обычно такие расходы составляют до 50-60% бюджета;

- Не жизненно необходимые, но очень желательные расходы: это походы в кино, покупка косметики, заказ любимой пиццы и все то, что приносит удовольствие;
- «Статусные» расходы: расходы, соответствующие определенному статусу и доходы, например, приобретение картины в коллекцию;
- Лишние/вредные расходы: расходы, без которых точно стоило бы обойтись, но в силу привычки или иной причины присутствуют в расходах. Например, для курящих людей стоило бы уменьшить, а в будущем исключить привычку курения.

По периодичности:

- Регулярные расходы – расходы с периодичностью, например, ежемесячная оплата коммунальных услуг или ежегодная оплата налогов. Всегда присутствуют в плановых расходах;
- Переменные расходы – непостоянные расходы по необходимости/запланированные расходы: покупка машины или бытовой техники различной габаритности;
- Сезонные расходы – с наступлением разных сезонов требуют соответствующие расходы, например, покупка зимней обуви;
- Непредвиденные расходы – расходы, которые невозможно учесть. Такие расходы присутствуют всегда, иногда они желательны, как покупка мороженого или поломка машины посреди дороги.

1.1.3 Перевод средств

Перевод средств - движение денежных средств от отправителя к получателю с помощью операторов платежных систем через национальные или между-

народные платежные системы с целью зачисления денежных средств на счет получателя или выдачи ему их в наличной форме. В структуре денежного перевода всегда присутствует отправитель, получатель и посредник — оператор платежной системы, взимающий за свои услуги определённую плату. Переводы могут быть:

- Внутренние – переводы внутри одного государства;
- Трансграничными – переводы между странами. Перевод происходит в иностранной валюте, после получения происходит обмен валюты на национальную валюту по курсу на текущий день.

1.1.4 Долг

Долг – денежные средства, передающиеся заёмщику кредитором с условием возврата. Позволяет физическим или юридическим лицам совершать операции, на которые ранее у них не хватало средств.

1.1.5 Заём

Заём – договор, согласно которому одна сторона (заемщик) передаёт/обязуется передать в собственность/управление другой стороне (заёмщику) деньги, ценные бумаги или иные объекты, определенные родовыми признаками (мера, вес), заемщик в свою очередь возмещает равную сумму денег или равное количество, или качество объектов того же рода. Заём может быть беспроцентным и процентным.

1.2 Анализ конкурентов

1.2.1 MoneyTracker

Эта программа является одной из популярных программ для ведения учета денежных средств.

При запуске, программа выглядит так, как на рисунке 1.

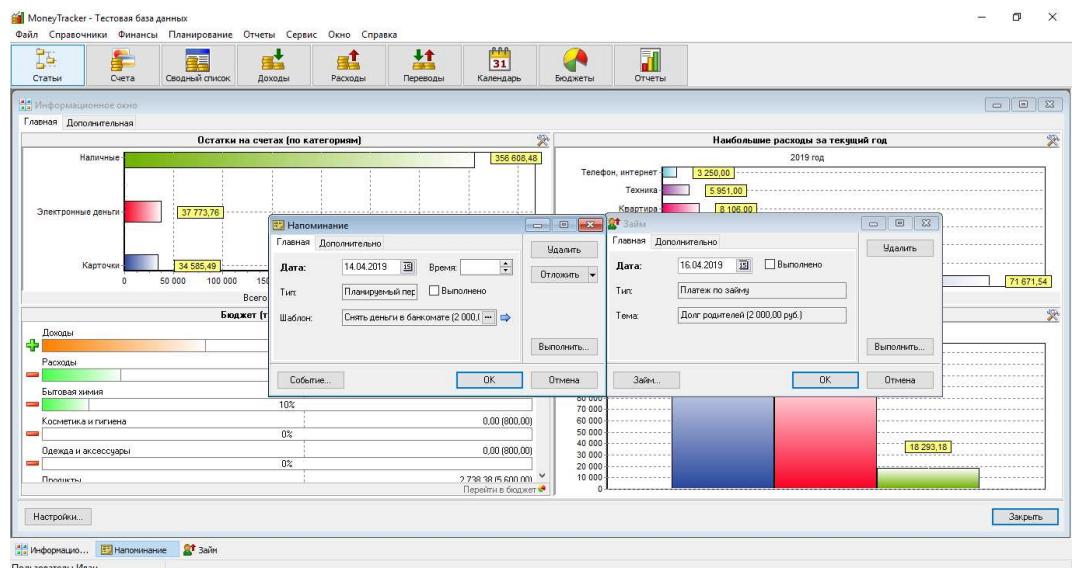


Рисунок 1 – Главный экран приложения

Сразу выводится изобилие окон, в которых есть вероятность запутаться. Пользователь, впервые запустив данную программу, может испугаться такого интерфейса.

Проверим функционал приложения, добавим какую-либо запись. Здесь записи подразделяются на «Доходы», «Расходы» и «Переводы». Допустим, необходимо добавить новую запись в доход, нажимаем «Доходы». Процесс добавления записи показан на рисунке 2. И сразу заметно, что ввод данных слегка не продуман. Пользователю, который впервые использует эту программу, будет не понятно куда, что и зачем вводить. При выборе «Источника», «Статьи» и прочее, открывается новое окно с выбором. Было бы лучше, если бы это все выглядело,

как выпадающий список, потому что постоянное мерцание экрана при открытии/закрытии окна создает дискомфорт глазам пользователя и снижает желание пользоваться этой программой.

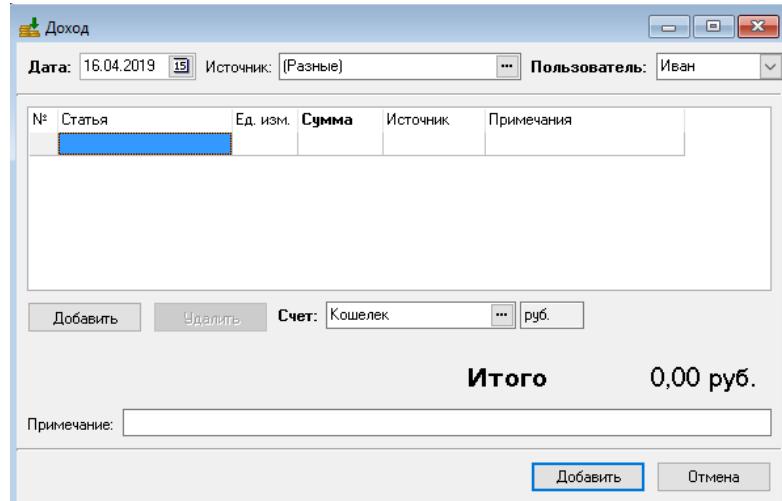


Рисунок 2 – Добавление записи «Доход»

Введя свои данные, сохраняем запись. Сохранённая запись будет выглядеть как на рисунке 3.

Дата	Пользователь	Сумма	Валюта	Счет	Примечания	Источник
16.04.2019	Иван	50,00	руб.	Карт. Сбербанк Иван	Положить в аптечку	Аптека МО
16.04.2019	Иван	1 800,00	руб.	Карт. Сбербанк Иван		
15.04.2019	Иван	10 000,00	руб.	Кошелек		
12.04.2019	Мария	320,00	руб.	Кошелек		
12.04.2019	Мария	400,00	руб.	Кошелек		
12.04.2019	Мария	200,00	руб.	Кошелек		
10.04.2019	Мария	250,00	руб.	Яндекс.Кошелек		
09.04.2019	Мария	541,70	руб.	Яндекс.Кошелек		
09.04.2019	Мария	360,00	руб.	Яндекс.Кошелек		
09.04.2019	Мария	220,00	руб.	Кошелек		
09.04.2019	Мария	550,00	руб.	Кошелек		
09.04.2019	Мария	500,00	руб.	Кошелек		
08.04.2019	Мария	300,00	руб.	Яндекс.Кошелек		
08.04.2019	Мария	550,00	руб.	Яндекс.Кошелек		

Рисунок 3 – Таблица доходов и добавленная запись

Процесс добавление записи в «Расход» и «Переводы» происходит аналогичным образом.

Порадовало, что в «Статьи» есть уже существующие статьи, они показаны на рисунке 4. Пользователь потратит меньше времени на заполнение – можно пользоваться уже существующими статьями.

Рисунок 4 – Статьи доходов/расходов

Было замечено, что таблицы «Доход» и «Расход» являются отдельными таблицами. Следовательно, записи подразделяются на доходы и расходы и являются самостоятельными. Как пользователь, могу сказать, что это не совсем удобно, хотелось бы, чтобы все записи выводились в одну таблицу. Хотя и имеется сводный список, показанный на рисунке 5, но для этого нужно переходить в отдельное окно.

Кроме сводного списка, имеется еще пункт меню «Доходы и расходы детально» в «Финансах». Внешний вид пункта меню выглядит на рисунке 6.

Рисунок 5 – Сводный список

Доходы

494,92 \$
350,00 €

Расходы

19,00 \$
1 046 212,11 руб.
3 742,80 ю.

Баланс

475,92 \$
350,00 €
-259 629,11 руб.
857,20 ю.

Рисунок 6 – Таблица «Доходы и расходы детально»

Не очень понятно, почему сделали две таблицы, если можно оставить более информативную.

В программе есть возможность сделать планирование бюджета для наглядности. Но для этого лучше изучить справку, которую сделали разработчики приложения, иначе придется потратить много времени на то, чтобы разобраться, как в приложении устроено планирование. Справка выглядит как на рисунке 7.

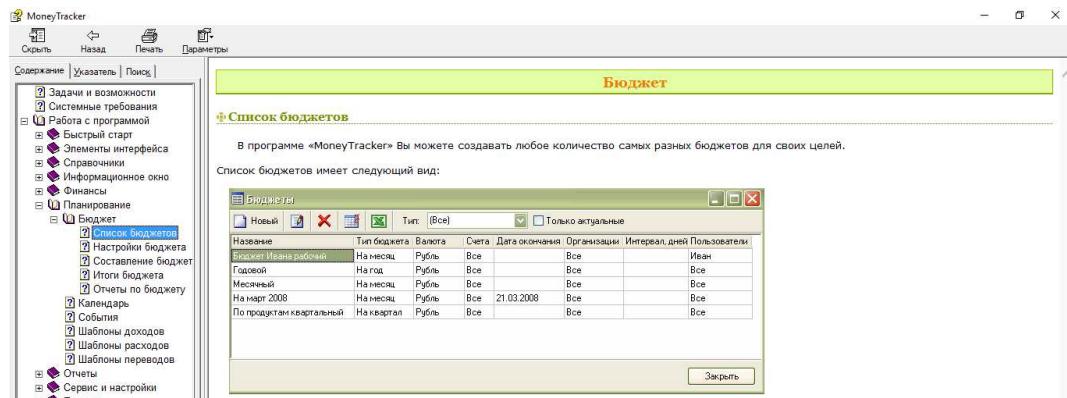


Рисунок 7 – Справка, пункт «Планирование»

Помимо планирования, можно создать напоминание о предстоящих событиях, показано на рисунке 8. Внешний вид календаря событий на рисунке 9. После создания события, можно быть уверенным, что приложение в назначенный день и час напомнит о мероприятии. Есть небольшой недостаток. Приложение формирует события только на текущий месяц. На следующий месяц или на год распланировать не получиться. Это недоработка разработчиков.

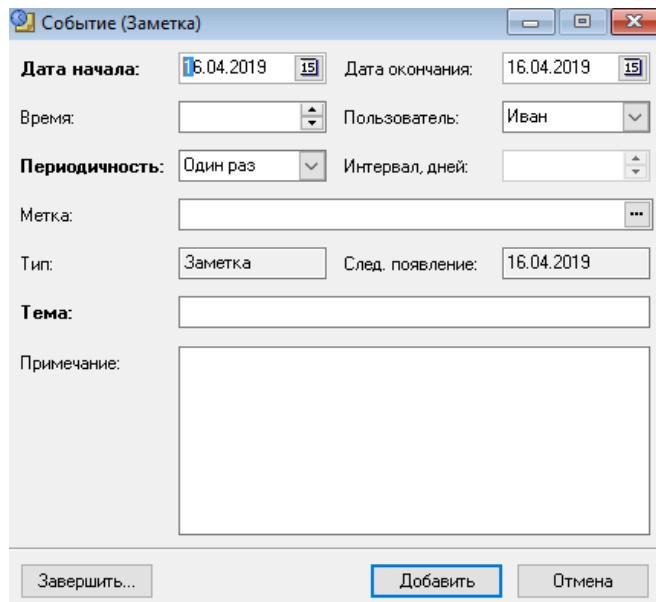


Рисунок 8 – Создание события

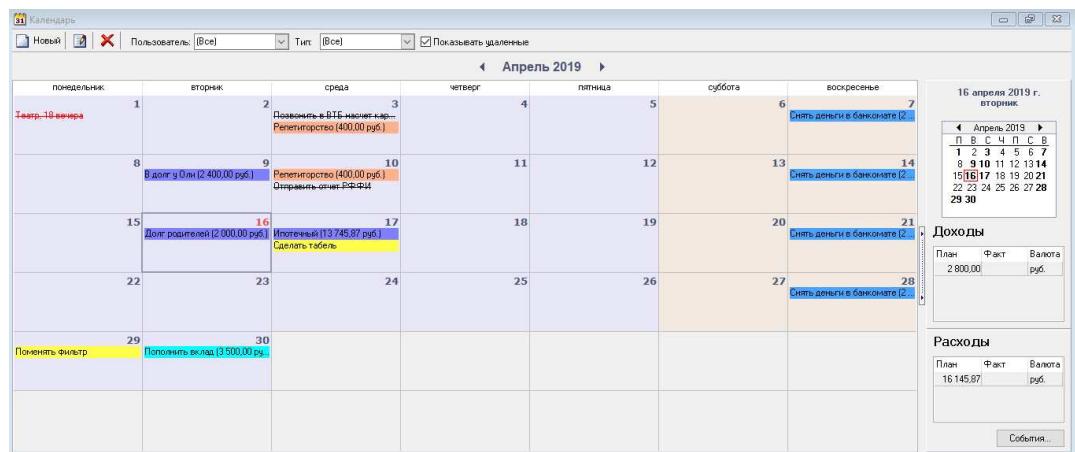


Рисунок 9 – Календарь событий

Программа умеет делать отчетность в любом удобном варианте для пользователя. Для этого необходимо перейти в «Отчеты» и выбрать необходимые параметры. Результат на рисунке 10.

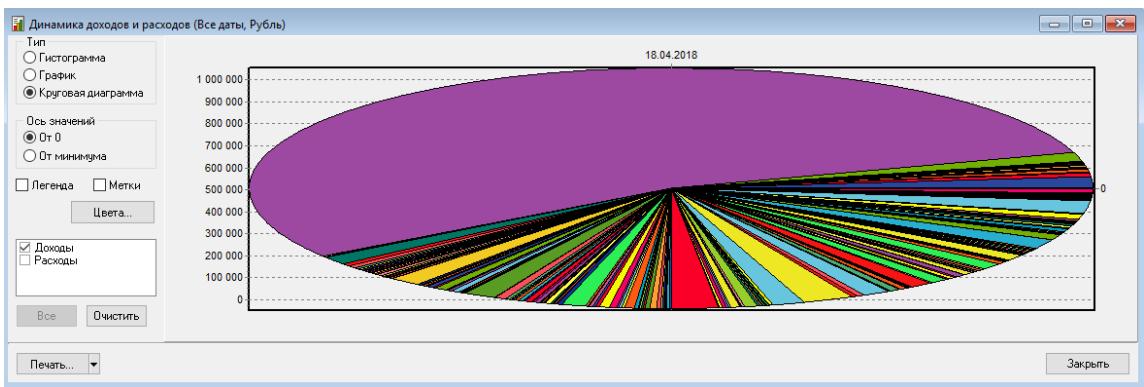


Рисунок 10 – Отчет в виде круговой диаграммы

Программа поддерживает несколько видов отчетов, настраиваемы для каждого индивидуально. Есть возможность экспорта в программу MS Excel и печати отчета.

Если необходимо выполнит резервное копирование, программа может выполнить и это. Пункт меню «Резервное копирование» находится в меню «Файл».

В целом, программа подходит для планирования семейного бюджета, предусмотрен раздельный денежных учет, который защищен паролем.

Однако программа не лишена и недостатков.

- Каждая закладка открывается в новом окне. С одной стороны, удобно работать с несколькими окнами одновременно. С другой, обилие окон может создавать дискомфорт пользователю;
- Как уже говорилось выше, не продуман процесс ввода данных. Этот процесс должен быть максимально понятен и упрощен;
- Отсутствуют калькуляторы. Приходится открывать другое приложение;
- Трудна в освоении, новичок может просто испугаться приложения и не оценит в полной мере все прелести программы
- Присутствуют некоторые ненужные параметры для обычного пользователя, которые необходимо заполнять.

Если пользоваться этим приложением, необходимо затратить некоторое время на ее освоение и привыкнуть к многооконному интерфейсу.

1.2.2 Домашняя экономика

Программа «Домашняя экономика» давно зарекомендовала себя как отличную программу для ведения персонального бухгалтерского учёта.

Внешний вид главного экрана программы демонстрирует рисунок 11. Здесь показан график текущего баланса счёта, остатки по счетам на текущую дату и ближайшие события. Справа показаны расходы по категориям, гистограммы доходов и расходов и новости из мира финансов. Слева присутствуют вкладки с действиями, основные счета и сводный отчёт. В левом нижнем углу есть вкладка со справками. В целом внешний вид программы привлекательный, есть сходство с программой Microsoft Word, что импонирует.

Начнём работу с программой и добавим счёт. Для этого нужно открыть пункт меню «Счета» и нажать во вкладке действий «Добавить счёт». Открывается форма заполнения как на рисунке 12.

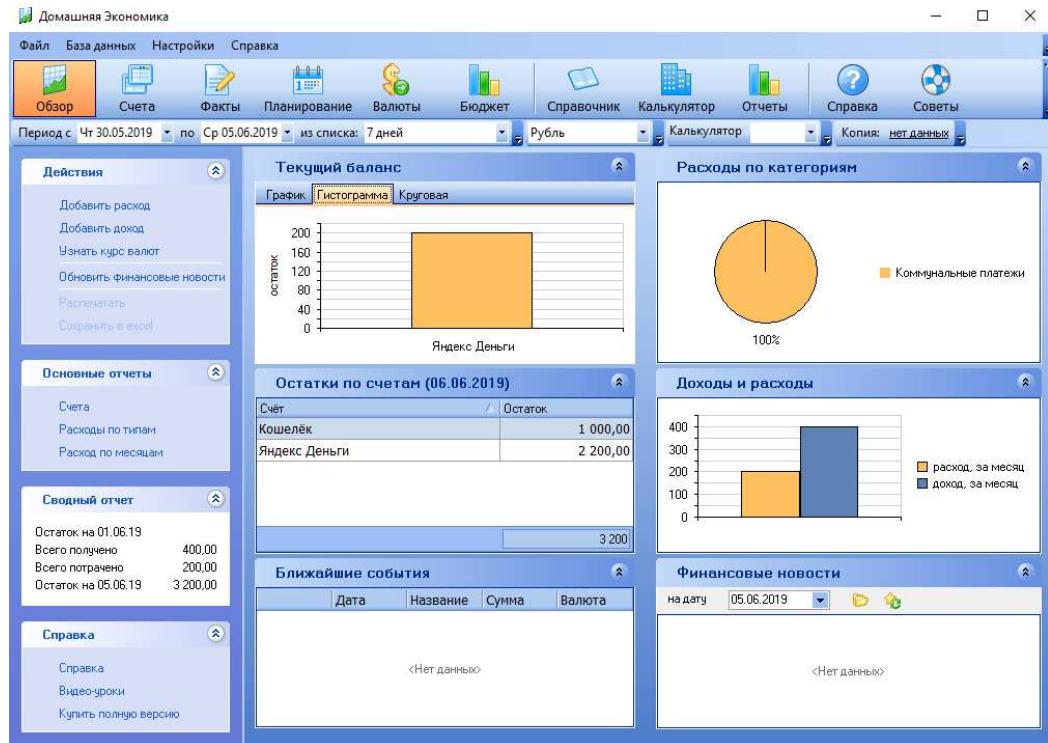


Рисунок 11 – Главный экран программы

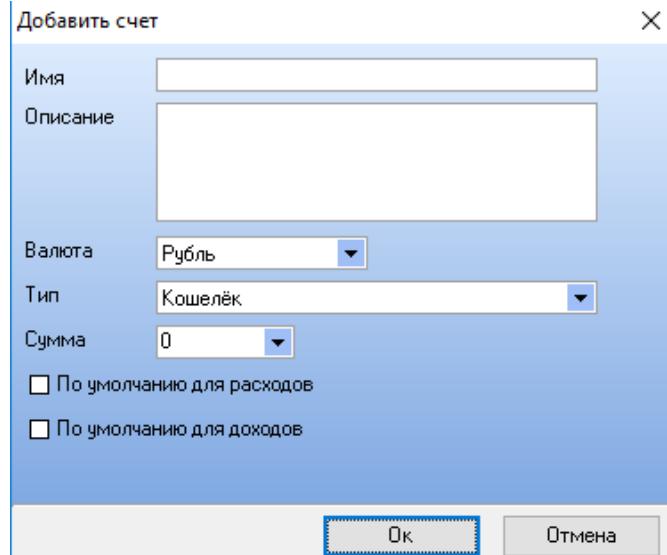


Рисунок 12 – Форма добавления счёта

Здесь обязательными полями являются имя счета, валюта (если не в рублях), тип и сумма. После добавления, счёт появляется в списке счетов как на рисунке 13.

Имя	Тип	Остат...	Ва...	-	+
Кошелёк	Кошелёк	1 000 руб		<input type="checkbox"/>	<input type="checkbox"/>
Яндекс Деньги	Кошелёк	2 200 руб		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Рисунок 13 – Список счетов

Для добавления расхода или дохода нужно перейти в пункт меню «Факты» и выбрать соответствующая вкладку. Допустим, нужно добавить расход, во вкладке «Действия» выбираем пункт «Добавить расход». Открывается окно заполнения как на рисунке 14.

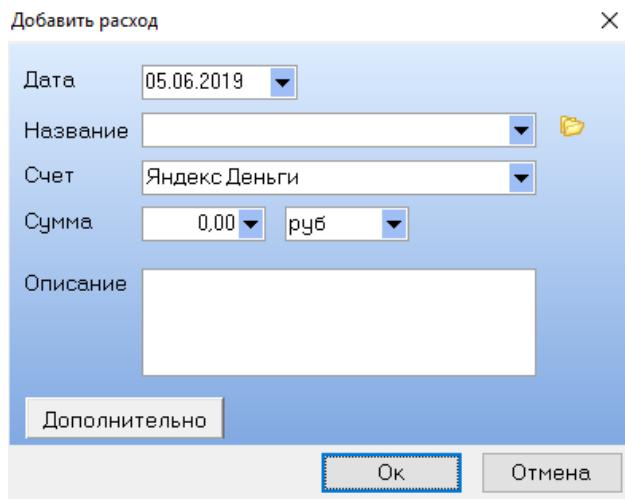


Рисунок 14 – Добавление расхода

Обязательными полями являются все поля, кроме описания. После добавления, запись появляется в таблице расходов, она показана на рисунке 15.

Дата	Название	Сумма, руб	Счет	Кол-во	Единица	Проект
■ Дата : 05.06.2019		Итого: 200,00				
05.06.2019	Internet	200,00	Яндекс Деньги	1	- ? -	

Рисунок 15 – Таблица расходов

Такие же действия нужно сделать для добавления дохода.

Вообще, такое разделение доходов и расходов не очень удобно. Для просмотра за определенную дату всей истории движения денежных средств, нужно будет переключаться между вкладками.

В программе реализован перевод между счетами. Для этого надо перейти в пункт меню «Счета» и выбрать вкладку «Переводы между счетами», выбрать действие «Добавить перевод». Процесс перевода денежных средств выглядит как на рисунке 16.

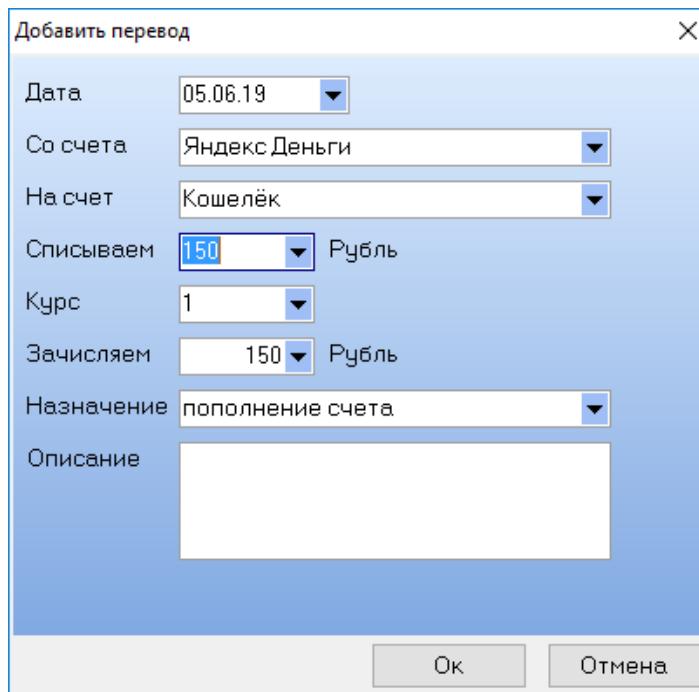


Рисунок 16 – Перевод между счетами

После завершения запись появится таблице и будет выглядеть как на рисунке 17.

Дата	Со счета	Списываем	На счет	Зачисляем	Назначение
05.06.19			Кошелёк	1 000,00	пополнение счета
05.06.19			Яндекс Деньги	2 000,00	пополнение счета
► 05.06.19	Яндекс Деньги	150,00	Кошелёк	150,00	пополнение счета

Рисунок 17 – Таблица переводов

Также в программе можно вести учёт личного имущества. Например, у пользователя есть в собственности квартира, которую он сдаёт в аренду. Значит в таблицу собственность можно внести квартиру. Для этого в этом же пункте меню нужно перейти во вкладку «Собственность» и совершить действие «Добавить собственность». Открывается окно добавления собственности как на рисунке 18.

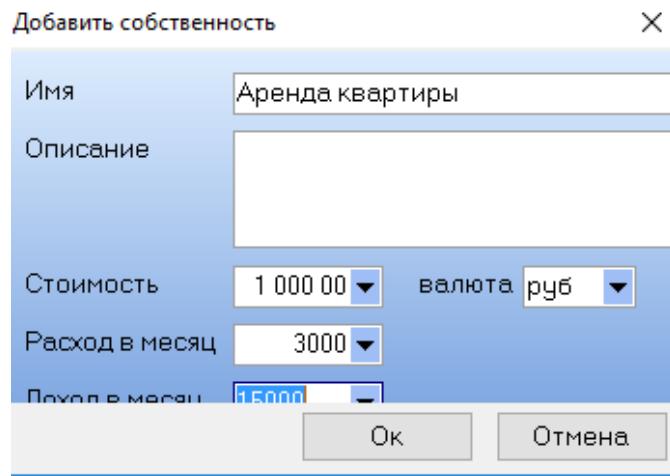


Рисунок 18 – Добавление собственности

Сразу видна недоработка разработчиков, поле «Доход в месяц» наполовину не виден. Расширить окно также невозможно.

После добавления, собственность появляется в таблице собственности, представленной на рисунке 19.

Имя	Стоимость	валюта	Описание	Доход в мес	Расход в ме	Актив
Аренда квартиры	1 000 000	руб		15000	3000	<input checked="" type="checkbox"/>

Рисунок 19 – Таблица собственности

Отчеты в “Домашней экономике” выполнены достаточно информативно, вариантов просмотра множество. Можно выбрать различные временные рамки, можно смотреть отчет по категориям, по расходам/доходам, по различным счетам, либо совокупный. Аналитическая часть выполнена хорошо.

Пример отчётности приведен на рисунке 20.

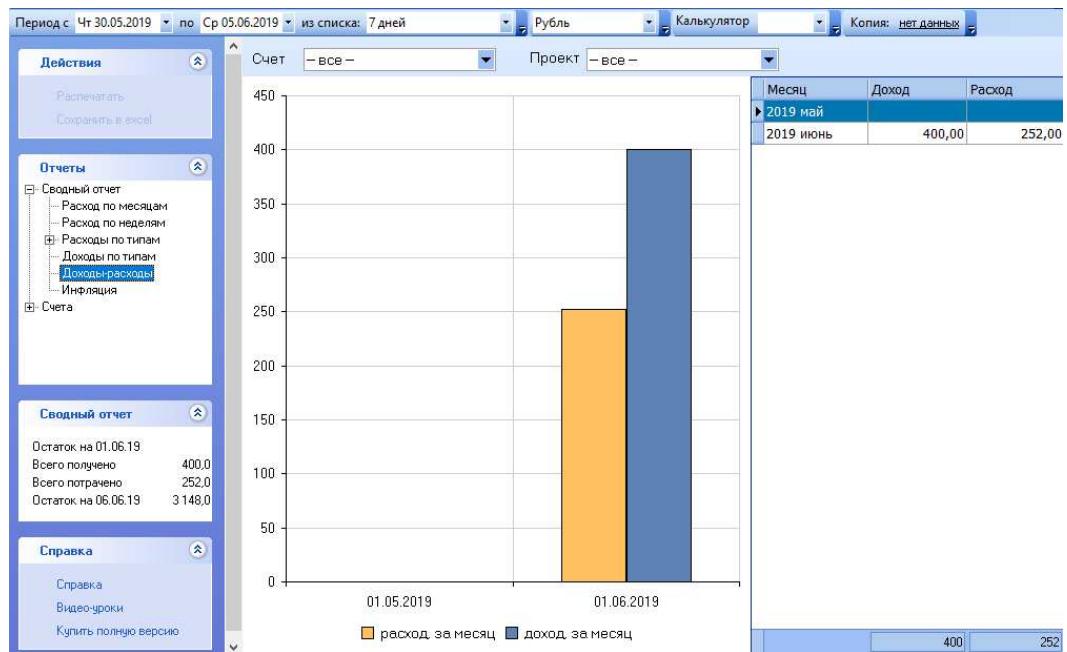


Рисунок 20 – Отчёт в программе

Здесь можно выбрать отчёты по расходам/доходы (по месяцам, по неделям, по типам), а также совместный отчёт доходов и расходов. Также есть отчёт по счетам. Отчёт представлен на рисунке 21.

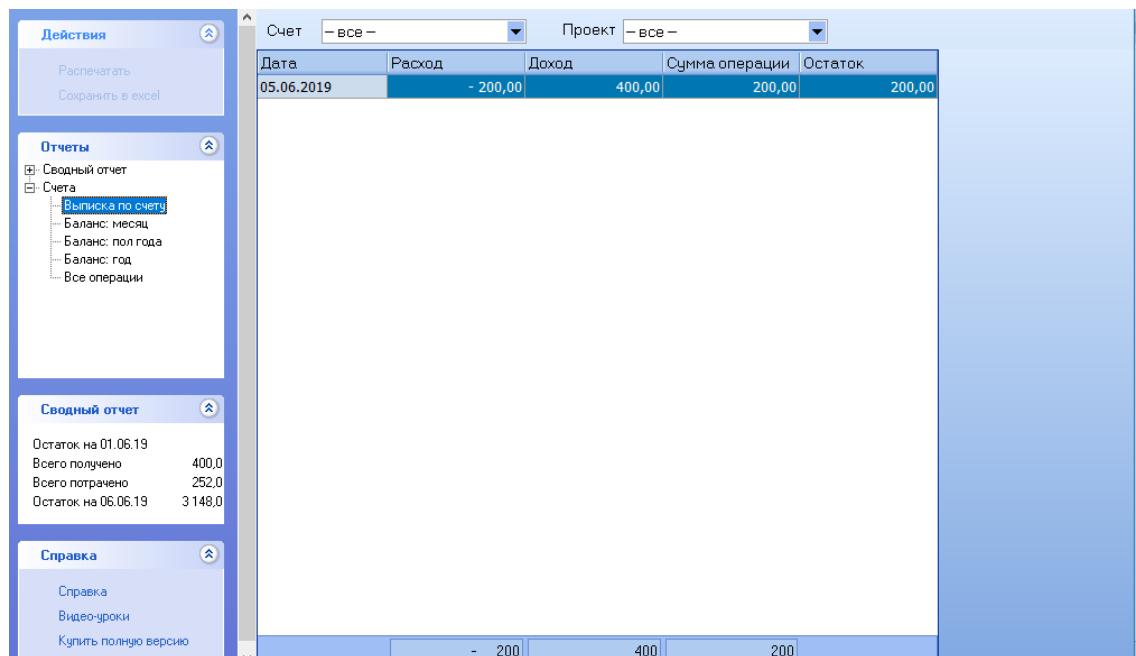


Рисунок 21 – Отчёт по счетам

Все отчёты возможно выгрузить в программу MS Excel, однако функция недоступна в демоверсии.

Также, как и в предыдущей программы, возможно резервное копирования базы данных.

Подведем итоги. Программа имеет достаточно неплохой и дружественный интерфейс. Помимо справки в самой программе, есть также видеоуроки, описывающие работу с программой.

Недостатки данной программы:

- Неудобное отображение доходов и расходов, они находятся на разных вкладках;
- Нет возможности защиты паролем данных, пользоваться программой может только один пользователь;
- В программе присутствуют слишком много функций, пользователь может растеряться.

В целом программа приятна в использовании, справка сделана информативно. Однако, полный функционал программы доступен лишь платно.

2 Инструменты реализации

2.1 Среда разработки Embarcadero RAD Studio Delphi XE8

Embarcadero Delphi (ранее Borland Delphi) — интегрированная среда разработки ПО для операционных систем Microsoft Windows, Mac OS, iOS и Android на языке Delphi, созданная первоначально фирмой Borland и на данный момент принадлежащая и разрабатываемая Embarcadero Technologies. Также является частью пакета Embarcadero RAD Studio.

Среда предназначена для быстрой разработки прикладного ПО. Созданные программы не требуют наличия стороннего ПО для запуска. Благодаря уникальной совокупности простоты языка и генерации машинного кода позволяет непосредственно, и, при желании, достаточно низкоуровнево взаимодействовать с операционной системой, а также с библиотеками, написанными на C/C++. Выделение и освобождение памяти управляется в основном пользовательским кодом, что, с одной стороны, ужесточает требования к качеству кода, а с другой — делает возможным создание сложных приложений с высокими требованиями к отзывчивости (работа в реальном времени). Среда разработки производит небольшие исполняемые модули .exe и .dll. В Delphi входит компилятор кода, локальный SQL-сервер, генераторы отчетов и библиотеки визуальных компонентов. Также среда поддерживает несколько форматов БД. Остановимся на dBase и Paradox.

dBase является первым появившимся универсальным форматом, поэтому достаточно долго он являлся популярным, так как многие СУБД взаимодействовали. Формат использует для хранения каждой таблицы до 3-х файлов:

- Dbf – основная таблица;
- Dbt – blob-данные, необязательный файл. Появляется файл только при наличии таких данных. (Blob-данные - специальный тип данных для хранения изображений или программного кода);
- Mdx - мультииндексный файл.

В таблицах присутствует ограничения на имена полей в виде максимальной длины имени – 10 символов и допустимые значения имен – только латинские буквы и цифры. Также отсутствует контроль целостности данных и не поддерживаются некоторые типы ключей, индексов и т.д. Поэтому этот формат хранения данных является не конкурентоспособным.

Paradox является наиболее развитым вариантом хранения данных. В нем присутствует поддержка любых символов в имени таблицы и ее полей, а также увеличен лимит на длину имени до 25 символов. Также появился автоинкрементный тип данных – счетчик, который автоматически присваивал данным код записи. Появилась возможность защиты таблиц посредством пароля и разграничения доступа к ним. Расширился перечень поддерживаемых типов данных, они указаны в таблице 1.

Таблица 1 – Поддерживаемые файлы в Paradox

Тип	Обозначение	Описание
Alpha	A	Строка, длиной не более 255 символов
Number	N	Число с плавающей точкой
Money	\$	Число с обозначением денежного знака
Short	S	Малое целое
LongInteger	I	Целое
BCD	#	Число в двоично-десятичном коде
Date	D	Дата
Time	T	Время
Timestamp	@	Дата и время
Memo	M	Строка произвольной длины. Первые 240 символов хранятся в таблице, остальные – в файле с blob-данными
Formatted Memo	F	Мемо для RTF-файлов
Graphic	G	Графическое изображение формата BMP, TIFF, GIF, хранятся в файле с blob-данными
OLE	O	Поддерживаемые OLE произвольные данные, хранятся в файле с blob-данными
Logical	L	Логическое значение: true(1) или false(0)
Autoincrement	+	Автоинкрементное поле(при каждом следующем поле, значение увеличивается на единицу)
Binary	B	Двоичная информация, первые 240 символов – в таблице, остальные в blob-файле
Bytes	Y	Двоичная информация, длиной не более 255 символов

Файлы хранения данных в Paradox следующие:

- db - файл с основными данными таблицы;
- mb - файл с BLOB-данными;
- px - файл с первичным индексом (ключом);
- xg* и ug* - файлы со вторичными индексами (вместо * будет порядковый номер индекса);
- val - файл с информацией для проверки данных и целостности ссылок.

2.2 Библиотека VCL

Библиотека визуальных компонентов (VCL) — объектно-ориентированная библиотека для разработки программного обеспечения, разработанная компанией Borland (на данный момент поддерживается Embarcadero) для поддержки принципов визуального программирования. Предоставляет огромное количество готовых к использованию компонентов для работы в самых разных областях программирования, таких, например, как интерфейс пользователя (экранные формы и элементы управления — т. н. «контролы», «контроли»), работа с базами данных, взаимодействие с операционной системой, программирование сетевых приложений и прочее. Это значит, что программистам не придется вручную програмировать всех компоненты, нужно будет всего лишь изменять некоторые параметры.

2.2.1 Иерархия классов

Класс библиотеки VCL имеют простой механизм наследования: один класс имеет только одного предка. Иерархия классов представлена на рисунке 22.

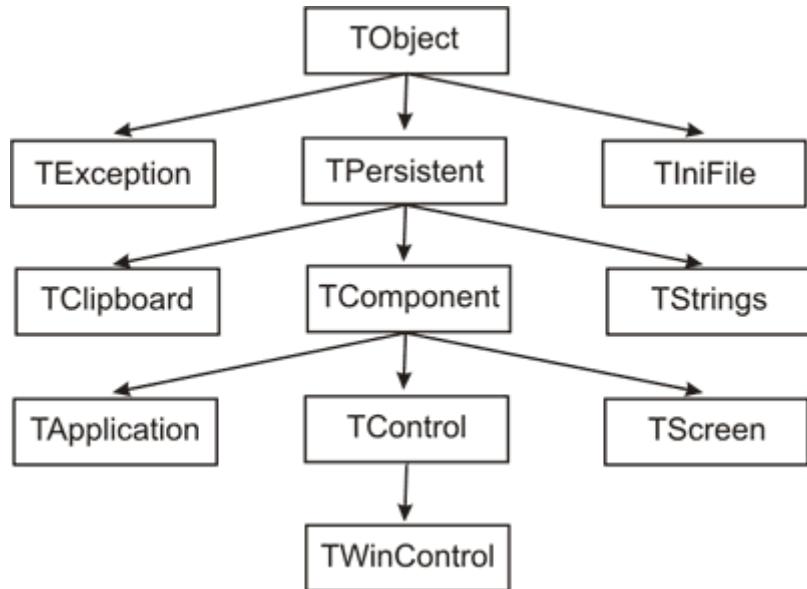


Рисунок 22 – Иерархия классов в VCL

Класс – определённый пользователем тип данных, обладающий процедурами или функциями со своими внутренними данными. Обычно описывается родовые признаки объекта. Объект – экземпляр класса.

Класс TObject является базовым классом для всех созданных объектов, если не указан другой тип. Представляет собой абстрактный класс с общими механизмами для всех объектов. Все дальнейшие классы будут образованы именно от него. Производные классы:

- Класс TPersistent предназначен для организации взаимодействия своих потомков с другими объектами и потомками, через метод Assign. Метод выполняет копирование содержимого одно объекта в другой, не изменяя самих объектов;
- Класс TException работает с исключительными ситуациями, которые возникают при неправильной работе с программой: попытка открытий несуществующий файл, попытка деления на ноль и так далее;
- Класс TIniFile инкапсулирует методы для работы с INI-файлами, которые содержат информацию о прикладных программах, их параметров (высота и ширина окна, логотип программы и так далее), положение окна и тому подобное. Такие файлы обычно хранятся в каталоге Windows;

В свою очередь класс TPersistent также имеет потомков:

- Класс TClipboard работает с буфером обмена Windows;
- Класс TString предназначен для хранения наборов строк. В компонентах используется как свойство компонента: например, свойство Lines типа TString в компоненте Memo;
- Класс TComponent - предок всех компонентов библиотеки. Компоненты обычно расположены в палитре компонентов. Также класс содержит свойство Name, которое уникально для каждого компонента, производного от него. Класс позволяет определять:

Класс TComponent позволяет определять:

- Родительский элемент управления – элемент, в который помещен данный компонент;
- Владелец компонента – форма, на которой расположен компонент.

Наследниками класса TComponent являются:

- TApplication – основа для самого графического приложения, обеспечивая взаимодействие между приложением и операционной системой. Также выполняет ряд вспомогательных функций: перехват и обработка нажатий на клавиатуру и тому подобное;
- TScreen инкапсулирует свойства и методы, необходимые для работы с экраном;
- TControl отвечает за визуальные компоненты. Для экземпляров класса определены свойства, такие как позиция, курсор, события для манипуляций с помощью мыши и так далее.

TControl имеет единственного потомка – TWinControl, который является базовый класс для оконных элементов приложения.

Также компоненты разделяются на четыре базисных типа:

- Стандартные компоненты: некоторые компоненты скрывают поведение типовых элементов управления операционной системы Windows от прикладного

программиста. VCL делает их видимыми для пользователя, добавляя обрамление для модификации вида компонента и его поведения через свойства и методы. Могут служить контейнерами для других компонентов, а также они могут быть активизированы, то есть принимать фокус. Примеры: компоненты TButton, TEdit, TListbox;

- Оригинальные компоненты: представляют собой элементы управления без заранее определенного поведения. Для этого разработчик должен предусмотреть код для определения поведения компонента, когда пользователь будет с ней взаимодействовать. Могут служить контейнерами для других компонентов, а также они могут быть активизированы, то есть принимать фокус. Примеры: компоненты TPanel, TStringGrid;

- Графические компоненты: представляет собой видимые компоненты управления без способности принятия фокуса. Обеспечивают отображение объектов без использования системных ресурсов. Не могут быть контейнерами для других объектов. Примеры: компоненты TImage, TShape;

- Невидимые компоненты: их поведение определяется программным кодом, никогда не появляются на форме ни в каком виде. Примеры: компоненты TTable, TTimer.

2.3 База данных MS Access

Microsoft Access – реляционная система управления базами данных, разработанная компанией Microsoft. Слово «реляционная» означает отношения, то есть база данных основана на отношениях между сущностями.

Характерной особенностью баз данных MS Access является файл .mdb, в которой хранятся таблицы и средства их обработки.

Основной элемент БД является таблица. Столбцы таблицы являются полями, строки – записями.

Этапы создания таблицы БД:

1. Задание структуры таблицы: определение количества и типов полей;

2. Ввод записей в БД;

3. Редактирование записей в БД (если необходимо);

Основные свойства полей в БД:

- Имя поля – используются как заголовки таблиц, имя поля позволяет обращаться к данным поля;
- Тип поля – тип содержащихся данных, например, число, дата, текст и так далее;
- Размер поля – определяет максимальную длину размещённых в поле данных;
- Формат поля – способ форматирования данных в ячейках.

2.3.1 Основные понятия

Таблица – список атрибутов, характеризующие сущность.

Сущность - объект, о котором необходимо хранить информацию.

Атрибут – определённая характеристика сущности.

Базовая таблица - таблица, имеющая первичный ключ, однозначно определяющая объект.

Ключи:

- *Первичный ключ* - атрибут или группа атрибутов, уникально идентифицирующая каждый экземпляр сущности;
- *Внешний ключ* – поле, значения которого соответствуют значениям первичного ключа таблицы.
- *Составной ключ* – ключ, состоящий из нескольких полей, уникально идентифицирующий объект.

Связь – логическая ассоциация между двумя сущностями.

Связь «один-к-одному» формируется, когда ключевой столбец одной таблицы присутствует в другой таблице как ключ, или свойствами столбца задана

уникальность – одно и тоже значение не повторяется в других строках. Визуально связь выглядит как на рисунке 23.

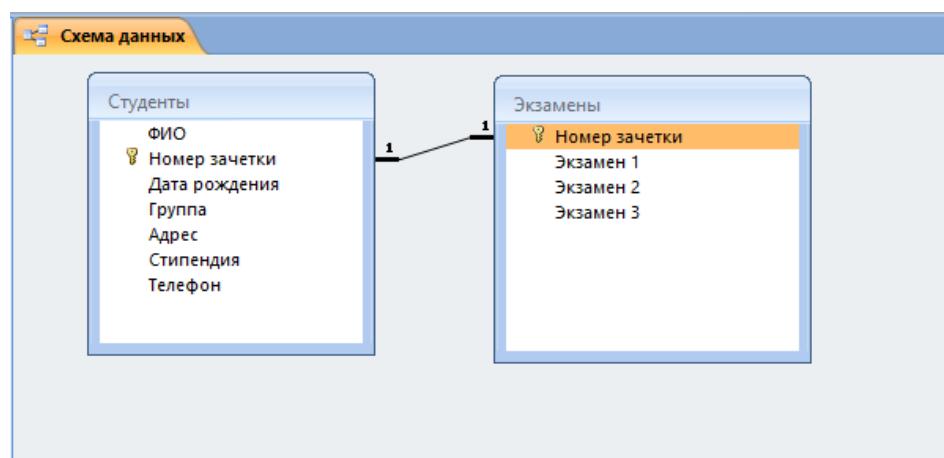


Рисунок 23 – Связь «один-к-одному»

Связь означает, что каждому значению атрибута первой таблицы соответствует одно и только одно значение связанного с ним атрибута второй таблицы. На рисунке показано, что атрибуту «Номер зачётки» таблицы «Студенты» обязательно соответствует атрибут «Номер зачётки» таблицы «Экзамены».

Связь «один-ко-многим» формируется таким образом, что одной записи первой таблицы соответствует несколько значений другой таблицы. Визуально связь выглядит на рисунке 24.

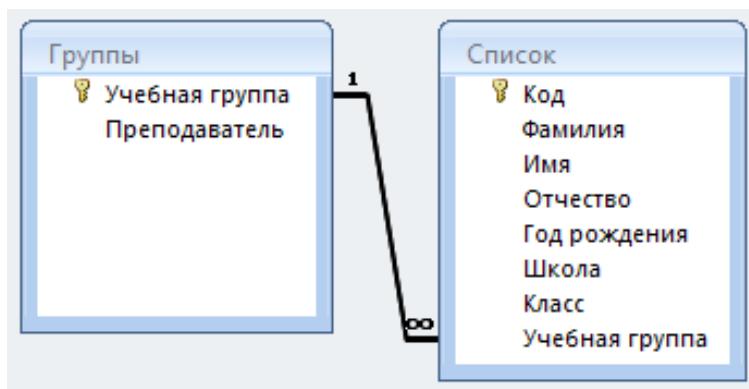


Рисунок 24 – Связь «один-ко-многим»

Связь означает, что каждому значению ключевого атрибута первой таблицы может соответствовать ноль, одно или несколько значений связанного с ним неключевого атрибута второй таблицы. В данном случае, одна и та же учебная группа присваивается нескольким студентам, если они являются одногруппниками. Например, Студент-1 и Студент-2 обучаются в группе 1, а Студент-3 обучаются в группе 3. Значит в таблице «Список», у Студент-1 и Студент-2 в атрибуте «Учебная группа» значение будет равно 1.

Связь «многие-ко-многим» формируется, если нескольким записям первой таблицы соответствует несколько записей второй таблицы. В этом случае создаётся промежуточная таблица и связывается с таблицами, которым необходимо сформировать такую связь, с помощью связи «один-ко-многим». Визуально это выглядит как на рисунке 25.

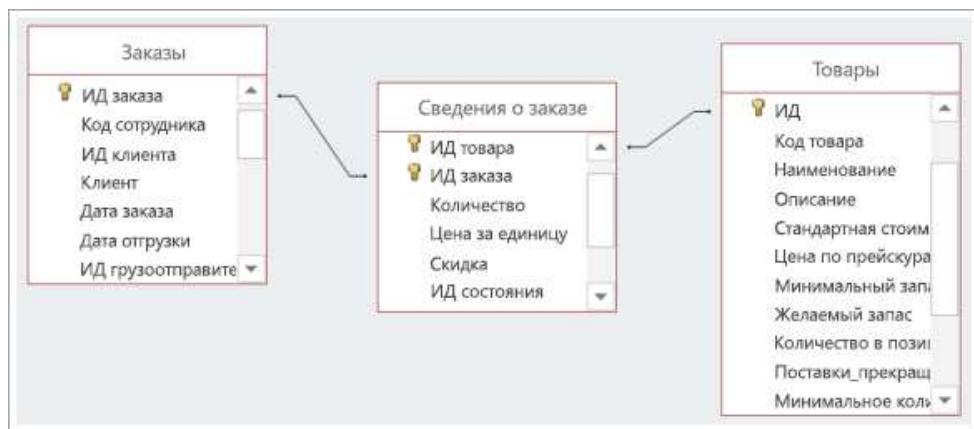


Рисунок 25 – Связь «многие-ко-многим»

Связь означает, что каждому значению атрибута первой таблицы соответствует несколько значений атрибута второй таблицы и в обратную сторону. На рисунке заказ содержит один или несколько товаров, также товар может быть фигурировать в нескольких заказах. Связи «многие-ко-многим» в MS Access нет, поэтому формируется промежуточная таблица «Сведения о заказах», которая связывает таблицы «Заказы» и «Товары». Таким образом имеется две связи «один-ко-многим», которые формируют связь «многие-ко многим».

2.3.2 Обеспечение целостности данных

Для первых типов связи можно обеспечить целостность данных. Это значит осуществить связность данных, их автоматическое каскадное обновление и удаление связанных записей. MS Access обеспечивает контроль целостности данных, соблюдая условия:

- В подчинённую таблицу не может быть добавлена запись, если записи не существует в главной таблице со значением соответствующего ключа;
- В главной таблице нельзя удалить запись, если она присутствует в подчинённых таблицах. Для этого сначала необходимо удалить все упоминания записи в связанных таблицах;
- Изменения значений ключа в главной таблице невозможны, так как этот ключ присутствует в подчинённых таблицах и уникально идентифицирует запись.

При нарушении этих условий, система MS Access генерирует сообщение о невозможности выполнения операции.

Также MS Access может не позволить установить параметр целостности для связи таблиц, если ранее введённые данные не отвечают требованиям целостности.

2.4 MS Excel

MS Excel – программа для работы с электронными таблицами компании Microsoft для операционных систем Windows, Mac OS, а также для мобильных устройств под управлением Android, iOS и Windows Phone. На данный момент MS Excel входит в пакет программ Microsoft Office, предназначенный для работы с различными видами документов: текст, электронные таблицы и прочее.

Электронная таблица – специальная модель структурирования, представления и обработки произвольной информации, связанная с базами данных и текстовыми документами.

2.4.1 Возможности MS Excel

Весомым аргументом применения MS Excel является универсальность. Приложение разработано для большинства популярных операционных систем. Ниже перечислены возможности приложения:

- *Работа с численными данными*: например, можно применить для формирования семейного бюджета или бюджета для малого бизнеса;
- *Работа с текстом*: в приложении предложен широкий спектр инструментов для работы с текстовыми данными;
- *Создание графиков и диаграмм*: большое количество инструментов позволяет формировать отчета и диаграммы различной сложности, что позволяет наиболее наглядно представить данные;
- *Создание схем и рисунков*: приложение позволяет вставлять в рабочий лист геометрические фигуры и рисунки SmartArt (картинки с готовыми блоками, которые можно изменять);
- *Организация списков и баз данных*: в приложении заложена структура строк и столбцов, что позволяет создавать списки и базу данных;
- *Импорт и экспорт данных*: есть возможность обмениваться данными с другими источниками;
- *Автоматизация однотипных задач*: в приложении предусмотрено написание макросов (программный алгоритм действий, созданный пользователем, предназначенный для автоматизации однотипных работ), что позволяет свести работу до единственного нажатия компьютерной мышью;
- *Создание панелей управления*: на рабочем листе можно размещать интерактивные элементы управления;
- *Встроенный язык программирования*: язык программирования VBA позволяет расширить возможности приложения в несколько раз, разрабатывая необходимые надстройки.

2.4.2 Работа с формулами

Для осуществления отчёта в ячейки вводятся различные формулы. Как правило, они начинаются с символа «=». Без символа «=», приложение воспринимает информацию как данные: текст или число. В записи формул могут быть присутствовать ссылки на необходимые ячейки:

- *Относительная ссылка*: на ячейку состоит из буквы столбца и номера строки, например, A5. Автоматически изменяется при удалении, копировании и вставке;
- *Абсолютная ссылка* – жёстко фиксированная ячейка, не изменяющаяся не при каком условии. Перед буквой столбца и перед номером строки ставится символ «\$», например, \$B\$5;
- *Смешанная ссылка* – фиксируется строго только буква столбца или номер строки, при этом, не фиксируемая часть изменяется при копировании или перемещении. Например, при перемещении данных из ячейки B\$5 в ячейку A5, адрес ячейки изменяется на A\$5.

Одно иногда при записи формулы допускаются ошибки, которые начинаются с символа «#»:

- #ДЕЛ/0 – означает, что в формуле предпринимается попытка деления на ноль, это запрещено;
- #Н/Д – «Неопределённые данные», появляется, когда ячейка должна содержать данные, но данные отсутствуют;
- #ИМЯ? – означает, когда невозможно распознать имя;
- #ПУСТО! – показывает, что в пересечении заданных областей нет общих ячеек;
- #ЧИСЛО! – показывает, что при использовании чисел возникли проблемы;
- #ССЫЛКА! – используется недопустимая ссылка на ячейку;

- #ЗНАЧ! – использован недопустимый тип аргумента, значение или операнд (аргумент операции);
- ##### - показывает, что число невозможно вместить в ячейку – слишком большое число.

3 Программная реализация

После изучения предметной области и определения инструментов разработки, необходимо приступить к реализации базы данных и собственно приложения.

3.1 Реализация базы данных.

Необходимо определить, какие таблицы будут присутствовать в будущем приложении.

Во-первых, нужна таблица для авторизации пользователей, так как приложение является многопользовательским.

Во-вторых, необходима таблица, содержащая все денежные движения, совершенные пользователем.

Таблицы будут содержаться в базе данных MS Access.

Были созданы 2 таблицы: таблица «Пользователи» и таблица «Список». Схема данных представлена на рисунке 26.

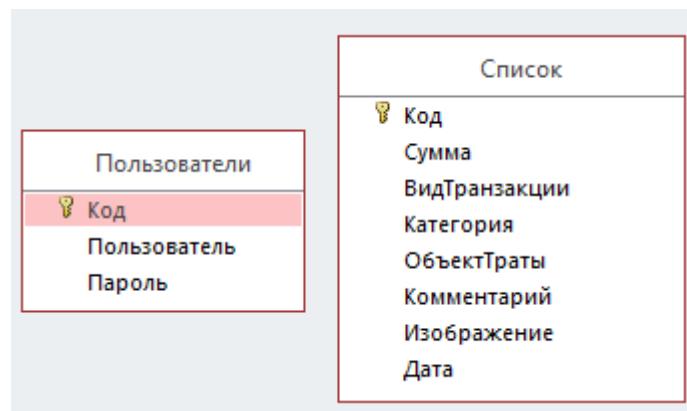


Рисунок 26 – Схема базы данных

В таблице «Пользователи» имеются поля:

- Поле «Код» с типом данных «Счётчик», является ключевым. Хранит в себе уникальный идентификатор пользователя;

- Поле «Пользователь» с типом данных «Короткий текст», ограничение по символам – 255 символов. Хранит в себе имя пользователя;
- Поле «Пароль» с типом данных «Короткий текст», ограничение по символам – 30 символов. Хранит в себе пароль пользователя.

Внешний вид таблицы выглядит на рисунке 27.

Имя поля	Тип данных
Код	Счетчик
Пользователь	Короткий текст
Пароль	Короткий текст

Рисунок 27 – Таблица «Пользователи»

Таблица «Список» имеет следующие поля:

- Поле «Код» с типом данных «Счётчик», является ключевым. Хранит в себе уникальный идентификатор записи;
- Поле «Сумма» с типом данных «Числовой». Содержит сумму, введённую пользователем при добавлении записи;
- Поле «ВидТранзакции» с типом данных «Числовой». Хранит вид транзакции, введенный пользователем, например, «Доход»;
- Поле «Категория» с типом данных «Числовой». Хранит категорию траты, с которым было взаимодействие;
- Поле «ОбъектТраты» с типом данных «Числовой». Хранит объект траты, на который была направлена транзакция;
- Поле «Комментарий» с типом данных «Короткий текст», ограничение по символам – 255 символов. Хранит дополнительную информацию о транзакции;
- Поле «Изображение» с типом данных «Короткий текст», ограничение по символам – 255 символов. Хранит путь до изображения, которое сохранил пользователь;
- Поле «Дата» с типом данных «Дата и время». Хранит дату, когда была совершена транзакция.

Таблица «Список» внешне выглядит как на рисунке 28.

	Имя поля	Тип данных
Код	Счетчик	
Сумма	Числовой	
ВидТранзакции	Числовой	
Категория	Числовой	
ОбъектТраты	Числовой	
Комментарий	Короткий текст	
Изображение	Короткий текст	
Дата	Дата и время	

Рисунок 28 – Таблица «Список»

Для таблицы «Список» также нужно создать три таблицы для хранения категорий, объектов траты и вида транзакций, так как они могут повторяться в таблице.

Дополнительно были созданы таблицы «ВидТранзакций», «Категории» и «ОбъектыТраты».

Таблица «ВидТранзакций» имеет поля:

- «Код» с типом данных «Счётчик», ключевое поле. Хранит уникальный код транзакции;
- «Транзакция» с типом данных «Короткий текст». Хранит название транзакций: доход, расход, перевод, заём и долг.

Таблица для хранения видов транзакций представлена на рисунке 29.

	Имя поля	Тип данных
Код	Счетчик	
Транзакция	Короткий текст	

Рисунок 29 – Таблица «ВидТранзакций»

Таблица «Категории» имеет следующие поля:

- «Код» с типом данных «Счётчик», ключевое поле. Хранит уникальный код категории;

- «Категория» с типом данных «Короткий текст». Хранит название категории, которое внесёт пользователь.

Так на рисунке 30 выглядит таблица «Категории».

	Имя поля	Тип данных
Код		Счетчик
Категория		Короткий текст

Рисунок 30 – Таблица «Категории»

Таблица «ОбъектыТраты» содержит поля:

- «Код» с типом данных «Счётчик», ключевое поле. Хранит уникальный код объекта траты;
- «Категория» с типом данных «Короткий текст». Хранит название объекта траты, которое внесёт пользователь.

Таблица «ОбъектыТраты» представлена на рисунке 31.

	Имя поля	Тип данных
Код		Счетчик
ОбъектТраты		Короткий текст

Рисунок 31 – Таблица «ОбъектыТраты»

Таким образом схема данных будет содержать 5 таблиц и выглядит следующим образом на рисунке 31.

У таблицы «Список» будет связь с таблицами «ВидТранзакции», «Категории» и «ОбъектыТраты», тип связи «один ко многим», так как дополнительные таблицы содержат данные с уникальными идентификаторами, в то время, как в таблице «Список», данные могут повторяться.

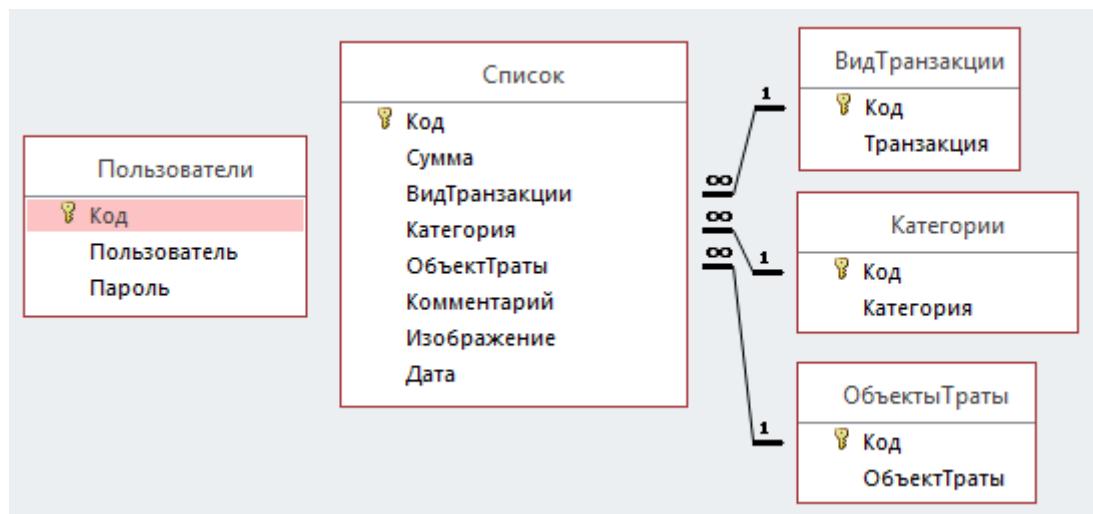


Рисунок 32 – Схема данных с добавленными таблицами

После создания базы данных можно приступить к реализации приложения.

3.2 Реализация приложения «Персональный Бухгалтер»

Реализация приложения декомпозируется на составляющие для проработки соответствующего функционала.

3.2.1 Добавление записи

Добавление записи происходит следующим образом:

1. Необходимо обозначить сумму;
2. Выбрать тип операции: доход, расход, заём или долг;
3. Выбрать категорию, то есть счёт, с которым будет совершена операция;
4. Выбрать объект траты, на что направлена операция;
5. Написать комментарий (необязательное действие);
6. Добавить изображение (необязательное действие);
7. Выбрать дату, когда была совершена операция. По умолчанию, запись будет добавлена по системному времени.

Если все данные были введены неверно, можно очистить поля.

После заполнения, необходимо нажать на кнопку «Добавить запись».

Форма добавления записи представлена на рисунке 33.

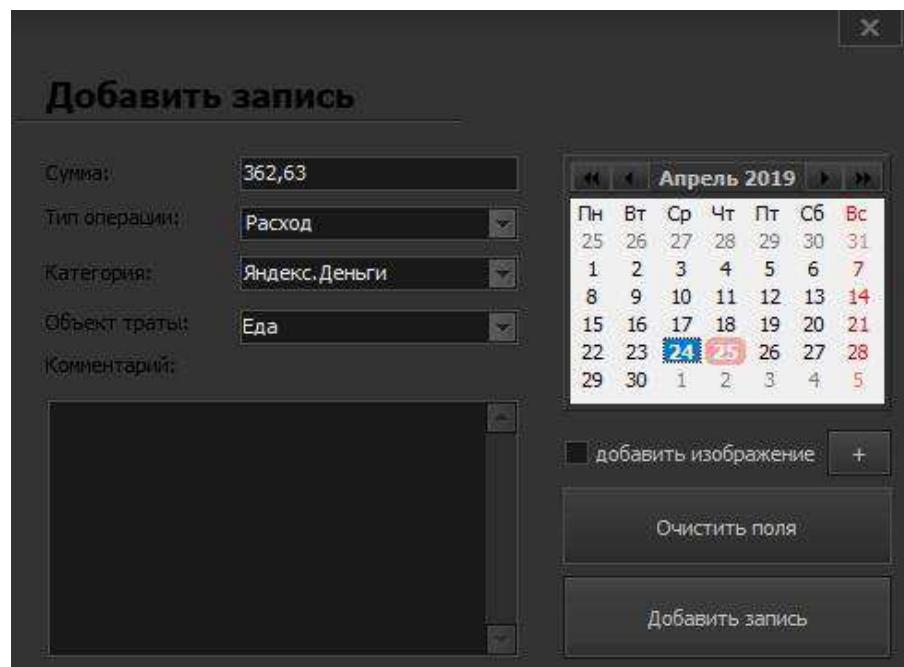


Рисунок 33 – Добавление записи

После добавления запись появляется в таблице, изображенной на рисунке 34. Последняя добавленная запись будет самой последней в списке. Таблица отсортирована по умолчанию по дате.

Учет бюджета						
Тип	Сумма	Дата	Категория	Объект	Комментарий	
-	663,03	31.03.2019	Яндекс.Деньги	AliExpress	Фитнес-трекер x2	
-	775,77	01.04.2019	Яндекс.Деньги	Еда		
-	625,00	01.04.2019	Наличка	Пища		
-	550,16	02.04.2019	Яндекс.Деньги	Еда		
-	559,80	03.04.2019	Яндекс.Деньги	Еда		
-	320,00	04.04.2019	Яндекс.Деньги	KFC		
-	1350,00	05.04.2019	Наличка	Пивка		
-	585,00	05.04.2019	Наличка	Пища		
-	400,00	06.04.2019	Наличка	Транспорт	Такси 200 x2	
-	1000,00	06.04.2019	Наличка	Другое	Денису на ДР	
-	637,00	07.04.2019	Яндекс.Деньги	Пища		
-	530,69	08.04.2019	Яндекс.Деньги	Еда		
-	116,00	13.04.2019	Наличка	Еда	Хотдоги	
-	3385,00	14.04.2019	Наличка	Уравнивание		
-	686,00	14.04.2019	Наличка	Еда		
-	92,00	14.04.2019	Наличка	Еда	Хотдоги	
+	8559,00	16.04.2019	Яндекс.Деньги	FreelanceHunt	Оплата за 03.2019	
-	52,00	16.04.2019	Наличка	Транспорт	Автобус 26x2	
-	104,00	17.04.2019	Наличка	Транспорт	Автобус 26x2x2	
-	610,00	17.04.2019	Яндекс.Деньги	Пища		
-	552,15	17.04.2019	Яндекс.Деньги	Еда		
-	184,00	18.04.2019	Наличка	Еда	Хотдоги	
-	154,46	19.04.2019	Яндекс.Деньги	Еда		
-	340,72	19.04.2019	Яндекс.Деньги	Еда		
-	104,00	20.04.2019	Наличка	Еда	Хотдоги	
+	2000,00	20.04.2019	Яндекс.Деньги	Зарплата	Проектирование БД Менеджера автомагазина	
-	610,00	21.04.2019	Яндекс.Деньги	Пища		
-	631,80	21.04.2019	Яндекс.Деньги	Еда		
-	545,24	22.04.2019	Яндекс.Деньги	Еда		
-	576,85	22.04.2019	Яндекс.Деньги	Еда		
-	308,73	23.04.2019	Яндекс.Деньги	Еда		
-	362,63	24.04.2019	Яндекс.Деньги	Еда		

Рисунок 34 – Таблица учета бюджета

3.2.2 Фильтрация таблицы

Таблица может быть отфильтрована по различным полям. Для этого необходимо совершить двойное нажатие по любой записи в интересующем поле. На рисунке 35 таблица отфильтрована по полю «Яндекс.Деньги».

Тип	Сумма	Дата	Категория	Объект	Комментарий
-	327,12	21.03.2019	Яндекс.Деньги	Еда	
-	402,51	22.03.2019	Яндекс.Деньги	Еда	
-	510,00	22.03.2019	Яндекс.Деньги	Аптека	
-	659,00	23.03.2019	Яндекс.Деньги	Еда	
-	637,00	24.03.2019	Яндекс.Деньги	Лица	
+	1100,00	26.03.2019	Яндекс.Деньги	AliExpress	Возврат за повербанки
-	585,00	26.03.2019	Яндекс.Деньги	Лица	
-	684,36	28.03.2019	Яндекс.Деньги	Еда	
-	1605,47	28.03.2019	Яндекс.Деньги	Еда	
-	490,57	29.03.2019	Яндекс.Деньги	Еда	
-	945,00	29.03.2019	Яндекс.Деньги	Лица	
-	2850,00	31.03.2019	Яндекс.Деньги	DNS	Весы
-	633,59	31.03.2019	Яндекс.Деньги	Еда	
-	663,03	31.03.2019	Яндекс.Деньги	AliExpress	Фитнес-трекер x2
-	775,77	01.04.2019	Яндекс.Деньги	Еда	
-	550,16	02.04.2019	Яндекс.Деньги	Еда	
-	559,80	03.04.2019	Яндекс.Деньги	Еда	
-	320,00	04.04.2019	Яндекс.Деньги	KFC	
-	637,00	07.04.2019	Яндекс.Деньги	Лица	
-	530,69	08.04.2019	Яндекс.Деньги	Еда	
+	8555,00	16.04.2019	Яндекс.Деньги	FreelanceHunt	Оплата за 03.2019
-	610,00	17.04.2019	Яндекс.Деньги	Лица	
-	552,15	17.04.2019	Яндекс.Деньги	Еда	
-	154,46	19.04.2019	Яндекс.Деньги	Еда	
-	240,72	19.04.2019	Яндекс.Деньги	Еда	
+	2000,00	20.04.2019	Яндекс.Деньги	Забота	Проектирование БД Менеджера автомагазина
-	610,00	21.04.2019	Яндекс.Деньги	Лица	
-	631,80	21.04.2019	Яндекс.Деньги	Еда	
-	545,24	22.04.2019	Яндекс.Деньги	Еда	
-	576,85	22.04.2019	Яндекс.Деньги	Еда	
-	308,73	23.04.2019	Яндекс.Деньги	Еда	
-	362,63	24.04.2019	Яндекс.Деньги	Еда	

Рисунок 35 – Отфильтрованная таблица

3.2.3 Перевод денежных средств

Также помимо обычных видов транзакций, есть особый вид – перевод со счета на счет. Форма перевода изображена на рисунке 36. Здесь также можно учесть комиссию, если необходимо.

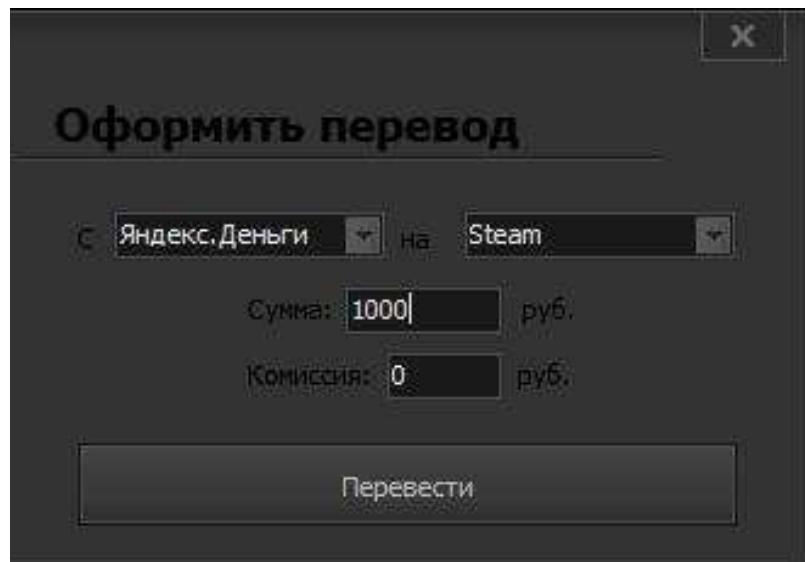


Рисунок 36 – Форма перевода со счёта

В таблице перевод выглядит как на рисунке 37.

-	1000,00	21.05.2019	Наличка	Перевод	Наличка --> Яндекс.Деньги
+	1000,00	21.05.2019	Яндекс.Деньги	Перевод	Начисление с Яндекс.Деньги

Рисунок 37 – Выполненный перевод

3.2.4 Фактические счета денежных средств

Счета выглядят как на рисунке 38. По счетам формируется общий доход за всю историю программы, общий расход и итоговая сумма, который показывает фактическую сумму, которая должна быть на счету физически. Результат показывает итоговую сумму всех указанных счетов.

	Доход	Расход	Итог
Наличка	2000,00	52,00	1948,00
Сбербанк	0,00	0,00	0,00
Результат	0,00	0,00	5648,00

Рисунок 38 – Внешний вид счетов

3.2.5 Сокрытие данных

Счета возможно скрыть при необходимости. Выглядит это следующим образом, на рисунке 39.

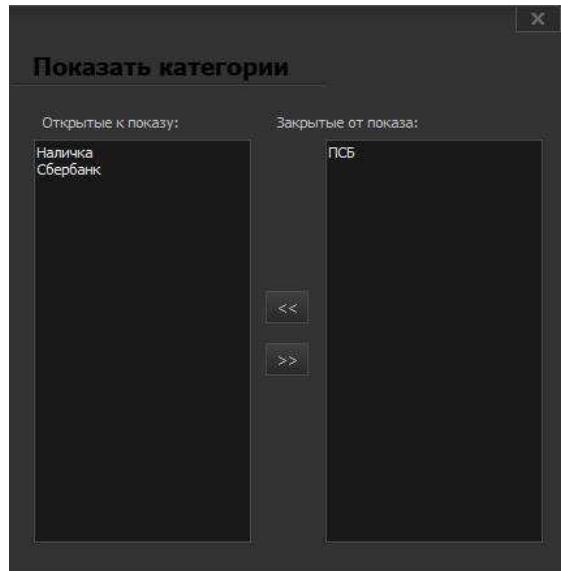


Рисунок 39 – Параметры сокрытия счетов

В результате сокрытия таблица поменяла свой вид и выглядит как на рисунке 40. На итоговую сумму сокрытие данных не влияет.

	Доход	Расход	Итог
Наличка	2000,00	52,00	1948,00
Сбербанк	0,00	0,00	0,00
Результат	0,00	0,00	5648,00

Рисунок 40 – Изменённый внешний вид таблицы

3.2.6 Статистика

Статистика показывает различную информацию, интересующую пользователя по датам и суммам. Статистика составляется с учетом заданного проме-

жутка, указанной категории (счета) и объекта траты. Результатом является гистограмма и круговая диаграмма, они показаны на рисунке 41. На рисунке показана статистика за неделю по всем категориям и всем объектах траты. Доход помечается красным цветом, чтобы показать пользователю, что расходы преобладают над доходами.

Данная статистика показывает, сколько было потрачено денежных средств за неделю.

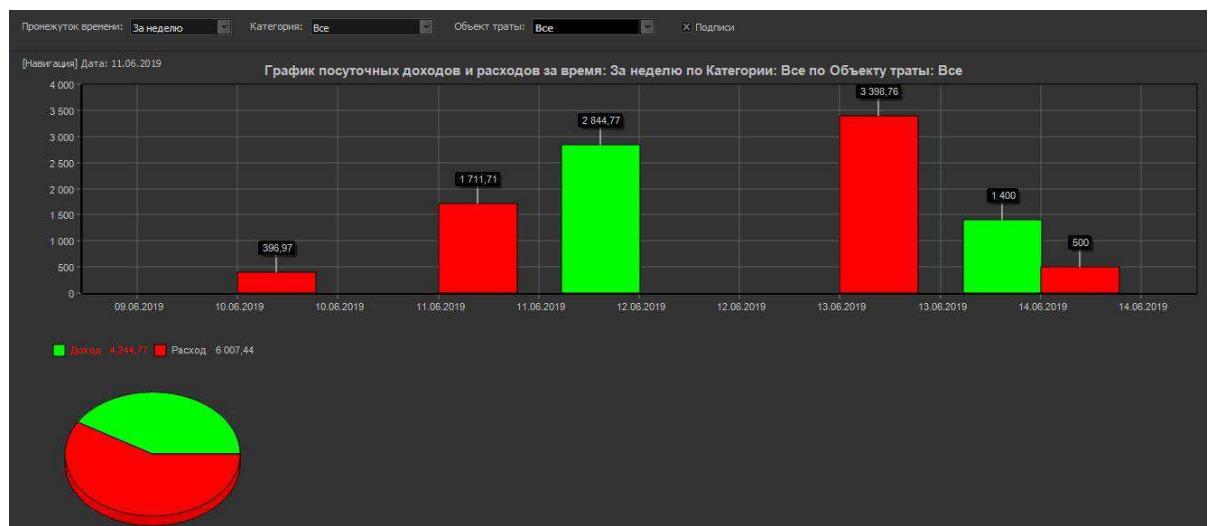


Рисунок 41 – Гистограмма и круговая диаграмма

На рисунке 42 представлен график за месяц по всем категориям и всем объектам трат.

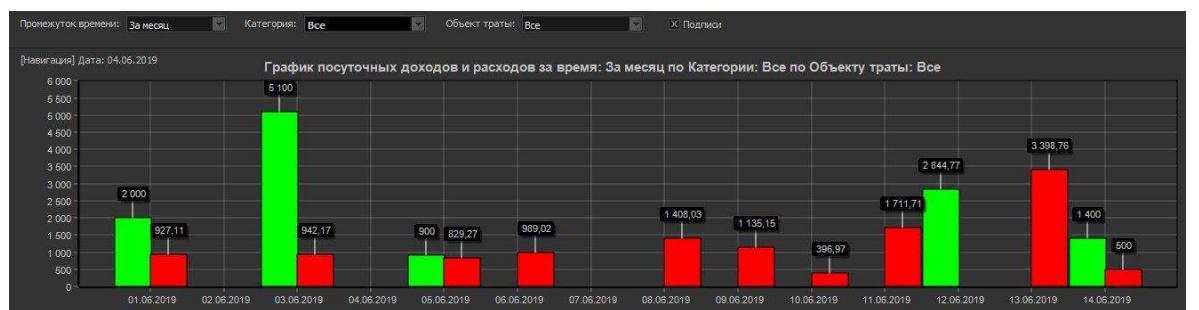


Рисунок 42 – Статистика за месяц

На рисунке 43 представлена статистика за 3 месяца по всем категориям и объекту траты «Транспорт». Статистика показывает, сколько было потрачено денежных средств на использование транспорта.



Рисунок 43 – Статистика за 3 месяца по объекту траты «Транспорт»

3.2.7 Регистрация и авторизация пользователя

При первом запуске программа выведет форму регистрации пользователя. Выглядит как на рисунке 13. После регистрации и в дальнейшем форма регистрации заменится на форму авторизации, показанной на рисунке 14. Если необходимо сделать второго пользователя, то нужно нажать на кнопку «у меня нет аккаунта». Таким образом реализовано многопользовательство приложения.

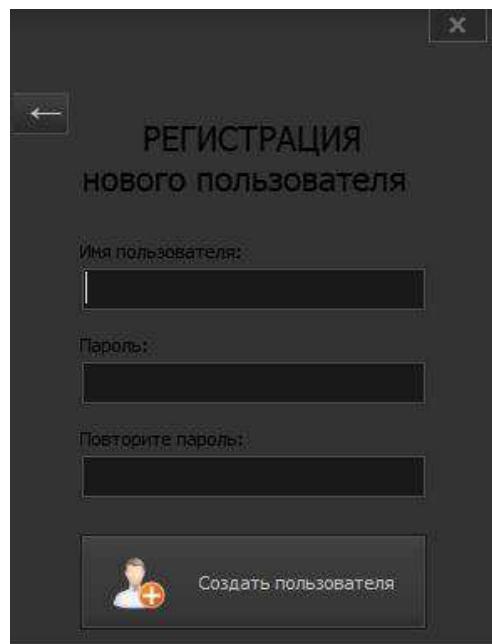


Рисунок 42 – Регистрация нового пользователя

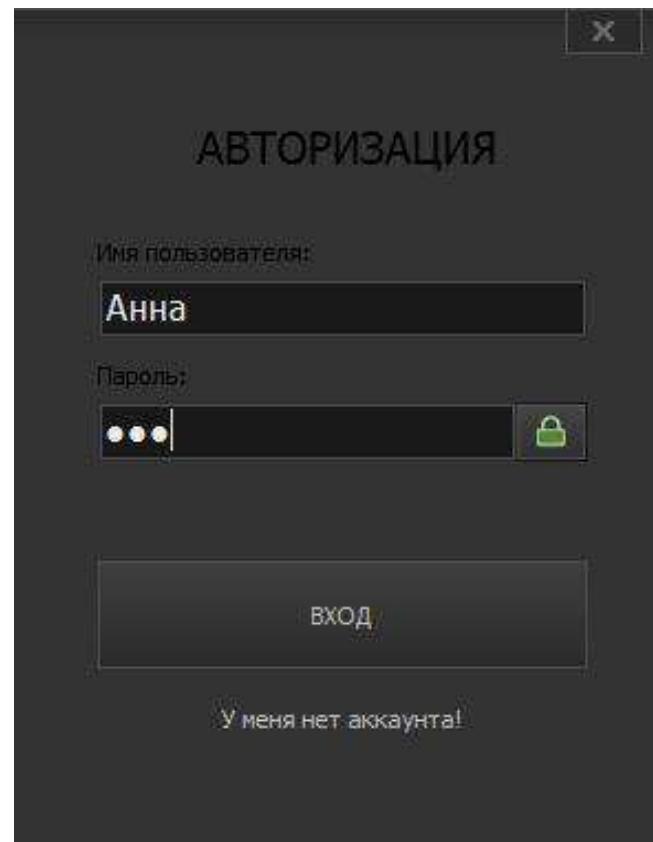


Рисунок 43 – Авторизация пользователя

ЗАКЛЮЧЕНИЕ

Подведем итоги. В ходе выполнения выпускной квалификационной работы была выявлена необходимость разработки приложения «Персональная бухгалтерия», а выбраны средства разработки приложения.

Необходимость разработки приложения состоит в автоматизации процесса учёта движения денежных средств, а также в создании дружественного интерфейса.

В качестве среды разработки приложения было выбрано Embarcadero RAD Studio Delphi XE8. Среда разработки позволяет в короткие сроки создать функциональное приложение, не требующее стороннего ПО для его работы.

Библиотека VCL в составе Embarcadero RAD Studio Delphi содержит в себе огромное количество готовых компонентов, позволяющих работать в различных областях программирования.

В качестве СУБД был выбран продукт компании Microsoft – MS Access. Данная СУБД является распространенной среди других СУБД так как она может работать практически со всеми программами, созданными различными студиями разработки приложений.

MS Excel используется для удобного представления пользователю различных видов отчетности.

Результатом работы стало создание приложения «Персональный бухгалтер». В приложении реализовано:

- авторизация и регистрация пользователя;
- добавление записей по всем движениям денежных средств;
- удобная таблица, содержащая все записи за всю историю программы;
- фильтрация таблицы по интересующему полю;
- список доступных счетов;
- статистика по различным параметрам;

Приложение протестировано и полностью готово к работе.

СПИСОК СОКРАЩЕНИЙ

ПО – программное обеспечение.

СУБД – система управления базами данных.

БД – база данных.

RAD – Rapid application development – быстрая разработка приложений.

OS – Operating system – операционная система.

SQL – Structured query language – язык структурированных запросов.

BLOB – Binary large object – большой двоичный объект.

DBF – Data base file – файл базы данных.

BCD – Binary-coded decimal – десятично-двоичный код.

OLE – Object Linking and Embedding – связывание и внедрение объектов.

VCL – Visual Component Library – библиотека визуальных компонентов.

MS - Microsoft.

VBA – Visual Basic for Applications - Visual Basic для приложений.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. СТАНДАРТ ОРГАНИЗАЦИИ Система менеджмента качества. Общие требования к построению, изложению и оформлению, документов учебной деятельности СТО 4.2-07-2014 – Красноярск: ИПК СФУ, 2014. – 60 с.
2. Microsoft Office Excel – что это? [Электронный ресурс] / Microsoft Excel для начинающих – Бесплатные уроки MS Excel и MS Word. – Режим доступа: <http://office-guru.ru/excel/microsoft-office-excel-chto-eto-59.html>
3. Типы связей в реляционных базах данных [Электронный ресурс] / Office-menu – Статьи и уроки Excel и SQL. – Режим доступа: <http://officemenu.ru/uroki-sql/41-tipy-svyazej-v-relyatsionnykh-bazakh-dannykh>
4. Типы таблиц и ключей в реляционных базах данных. Индексы. Взаимосвязи таблиц. Обеспечение целостности данных. [Электронный ресурс] / Единый образовательный портал Алтайского государственного университета. – Режим доступа:
<https://portal.edu.asu.ru/mod/book/view.php?id=22428&chapterid=501>
5. Microsoft Access [Электронный ресурс] / Википедия – свободная энциклопедия. – Режим доступа: https://ru.wikipedia.org/wiki/Microsoft_Access
6. Visual Component Library [Электронный ресурс] / Википедия – свободная энциклопедия. – Режим доступа:
https://ru.wikipedia.org/wiki/Visual_Component_Library
7. DFB [Электронный ресурс] / Википедия – свободная энциклопедия. – Режим доступа: <https://ru.wikipedia.org/wiki/DBF>
8. BLOB [Электронный ресурс] / Википедия – свободная энциклопедия. – Режим доступа: <https://ru.wikipedia.org/wiki/BLOB>
9. Класс TComponent [Электронный ресурс] / KANSoftWare – команда разработчиков. – Режим доступа: <http://www.kansoftware.ru/?tid=5102>
10. Класс TString [Электронный ресурс] / Borland Russian Community. – Режим доступа: <https://bdrc.ru/articles/delphi-7/urok-37-klass-tstrings.html>

11. Класс Exception – обработка исключений [Электронный ресурс] / Delphi Plus – ежедневные новости IT-технологий. – Режим доступа: <http://www.delphiplus.org/iskustvo-sozdania-komponentov-delphi/41-klass-exception--obrabortka-iskluchenii.html>

12. Описание класса TObject [Электронный ресурс] / Codenet – все для программиста! – Режим доступа: <http://www.codenet.ru/progr/delphi/stat/TObject.php>

13. Назначение классов TPersistent и TComponent. Примеры. [Электронный ресурс] / Лекции.Ком. – Режим доступа: <https://lektsii.com/1-41460.html>

14. Использование и создание визуальных компонентов [Электронный ресурс] / HELLOWORLD.RU - документация и книги по программированию. – Режим доступа: <http://www.helloworld.ru/texts/comp/lang/builder/builder/6.htm>

15. Класс TControl [Электронный ресурс] / Интуит – национальный открытый университет. – Режим доступа:

<https://www.intuit.ru/studies/courses/27/27/lecture/871?page=2>

16. VCL (Visual Component Library) [Электронный ресурс] / Национальная библиотека им. Н.Э. Баумана. – Режим доступа:

[https://ru.bmstu.wiki/VCL_\(Visual_Component_Library\)](https://ru.bmstu.wiki/VCL_(Visual_Component_Library))

17. Обзор программы Домашняя Экономика [Электронный ресурс] / МОЙРУБЛЬ – финансовый блог. – Режим доступа: <https://myrouble.ru/obzor-programmy-domashnyaya-ekonomika/>

18. Обзор программы MoneyTracker – Домашняя бухгалтерия [Электронный ресурс] / МОЙРУБЛЬ – финансовый блог. – Режим доступа:

<https://myrouble.ru/obzor-programmy-moneytracker-domashnyaya-buxgalteriya/>

19. Delphi и база данных [Электронный ресурс] / SNK SoftWare. – Режим доступа: <http://www.snkey.net/books/delphi/ch4-2.html>

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт космических и информационных технологий
Информационные системы

УТВЕРЖДАЮ
Зав. кафедрой ИС

П.П. Дьячук
подпись инициалы, фамилия
«21» 06 2019г.

БАКАЛАВРСКАЯ РАБОТА

09.03.02 Информационные системы и технологии

Разработка приложения «Персональная бухгалтерия»

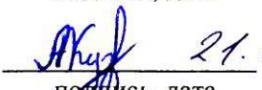
Руководитель

 21.06.19
подпись, дата

доцент, к.т.н.

И.А. Легалов

Выпускник

 21.06.19
подпись, дата

А.В. Кузнецова

Нормоконтролер

 21.06.19
подпись, дата

ст. преподаватель

Ю. В. Шмагрис

Красноярск 2019