

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ

НАВЧАЛЬНО–НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра інженерії програмного забезпечення

Пояснювальна записка

до бакалаврської роботи
на ступінь вищої освіти бакалавр

на тему: «**РОЗРОБКА МОБІЛЬНОГО ДОДАТКУ "PERSONAL ACCOUNTING" ДЛЯ МОНІТОРИНГУ ФІНАНСОВИХ ВИТРАТ НА МОВІ C#**»

Виконала: студентка 4 курсу, групи ПД-42
спеціальності:

121 Інженерія програмного забезпечення
(шифр і назва спеціальності)

Цілуванська В.В.

(прізвище та ініціали)

Керівник Негоденко О.В.

(прізвище та ініціали)

Рецензент _____

(прізвище та ініціали)

Київ – 2021

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ

НАВЧАЛЬНО–НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра Інженерії програмного забезпечення

Ступінь вищої освіти - «Бакалавр»

Спеціальність – 121 Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ

Завідувач кафедри
інженерії програмного забезпечення

Негоденко О.В.

“ _____ ” _____ 2021 року

ЗАВДАННЯ НА БАКАЛАВРСЬКУ РОБОТУ СТУДЕНТУ

Цілуванська Вікторія Володимирівна

(прізвище, ім'я, по батькові)

1. Тема роботи: «Розробка мобільного додатку "Personal accounting" для моніторингу фінансових витрат на мові С#»

Керівник роботи Негоденко Олена Василівна, завідувач кафедри ІІЗ

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від “12” березня 2021 року №65.

2. Строк подання студентом роботи “01” червня 2021 року

3. Вихідні дані до роботи:

Аналіз існуючих застосунків для моніторингу фінансових витрат;

Встановлено переваги та недоліки існуючих застосунків;

Визначено основні потреби користувачів.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

4.1 Здійснити аналіз і обробку практичної літератури по темі.

4.2 Виконати проектування і реалізацію Personal accounting для моніторингу фінансових витрат.

4.3 Реалізація отриманої БД.

5. Перелік графічного матеріалу

- 5.1. Титульний лист
- 5.2. Актуальність дослідження
- 5.3. Мета, об'єкт дослідження, предмет дослідження
- 5.4. Аналіз існуючих аналогів
- 5.5. Постановка завдань дослідження
- 5.6. Проектні і технічні рішення щодо розробки
- 5.7. Схема даних та потоків
- 5.8. Опис роботи програми та її компонентів
- 5.9. Інструкції по роботі
- 5.10. Відомості про програму
- 5.11. Тестування додатку
- 5.12. Висновки

6. Дата видачі завдання “19” квітня 2021 року**КАЛЕНДАРНИЙ ПЛАН**

№ з/п	Назва етапів бакалаврської роботи	Строк виконання етапів роботи до	Примітка
1	Вибір теми бакалаврської роботи	17.09.2020	Виконано
2	Дослідження актуальності теми	25.09.2020	Виконано
3	Розгляд аналогів	15.10.2020	Виконано
4	Теоретичні аспекти за темою дослідження	05.12.2020	Виконано
5	Проектування додатку	20.01.2021	Виконано
6	Розробка додатку	01.03.2021	Виконано
7	Тестування додатку	07.04.2021	Виконано
8	Розробка демонстраційних матеріалів	21.04.2021	
9	Здача роботи в деканат	01.06.2021	

Студент _____ Цілуванська В.В.
(підпис) (прізвище та ініціали)

Керівник роботи _____ Негоденко О.В.
(підпис) (прізвище та ініціали)

РЕФЕРАТ

Текстова частина бакалаврської роботи 65 с., 23 рис., 40 джерел.

Ключеві слова: ANDROID STUDIO, KOTLIN, GIT, ACTIVITY, LAYOUT, XML

Об'єкт дослідження – підвищення ефективності моніторингу фінансових витрат користувача.

Предмет дослідження – система обробки інформації для моніторингу фінансових витрат на мові С.

Мета роботи – покращення моніторингу фінансових витрат за допомогою додатку Personal accounting мовою С.

Методи дослідження – методи теорії інформації, методи структурного аналізу і проектування, методи розробки програмного забезпечення, методи тестування, валідації та верифікації програмного забезпечення.

Наукова новизна даної роботи полягає в наступному:

1. Розроблено алгоритм для покращення моніторингу фінансових витрат;
2. Встановлено, що використання мови С та середовища розробки Visual Studio це вдале рішення для досягнення поставленої мети.
3. Показано, що мова С зручна у використанні та надає можливість ефективної розробки даного додатку.
4. На основі результатів виконаних досліджень розроблено Personal accounting для моніторингу фінансових витрат на мові С.

Упровадження розробленого застосунку допоможе користувачам здійснювати моніторинг власних фінансових витрат.

В роботі виконано аналіз існуючих застосунків для моніторингу фінансових витрат. Встановлено переваги та недоліки існуючих застосунків. В результаті аналізу було визначено основні потреби користувачів.

Проаналізовано можливості середовища розробки. Розроблено логіку практичних завдань та загальну концепцію представлення інформації для користувачів застосунку.

Галузь використання – завдяки вільному доступу, застосунок може використовувати будь-яка людина, котра бажає здійснювати моніторинг власних витрат. Результат досягнення поставленої мети має можливість застосування в діяльності конкретного закладу, сім'ї, а також може бути використаний і іншими користувачами для вдосконалення аналізу фінансових витрат на мові С, також дана концепція викладення даного дослідження може бути використана в якості методичного допомоги при розробці інформаційних систем.

ЗМІСТ

ПЕРЕЛІК ВИКОРИСТАНИХ СКОРОЧЕНЬ.....	9
ВСТУП.....	10
РОЗДІЛ 1. ДОСЛІДЖЕННЯ АКТУАЛЬНОСТІ МОНІТОРИГУ ФІНАНСОВИХ ВИТРАТ КОРИСТУВАЧІВ.....	13
1.1 Аналіз актуальності підвищення ефективності моніторингу витрат	13
1.2 Огляд існуючих систем та програмних додатків	19
1.3 Постановка завдань дослідження.....	27
РОЗДІЛ 2 РОЕКТНІ І ТЕХНІЧНІ РІШЕННЯ ЩОДО РОЗРОБКИ PERSONAL ACCOUNTING МОВОЮ С#.....	30
2.1 Технічні рішення	30
2.2 Схема даних та інформаційних потоків	35
2.3 Проектне рішення.....	43
РОЗДІЛ 3 ОПИС РОБОТИ ПРОГРАМИ ТА ЇЇ КОМПОНЕНТІВ	50
3.1 Опис роботи програми та її компонентів.....	50
3.2 Інструкції адміністратору та користувачам	52
3.2.1 Загальні відомості про програму	52
3.2.2 Інструкція по роботі.....	52
3.3 Тестування розробленого програмного додатку.....	56
ВИСНОВКИ.....	61
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	62

ПЕРЕЛІК ВИКОРИСТАНИХ СКОРОЧЕНЬ

БД	база даних
ЕОМ	електронно-обчислювальна машина
ІС	інформаційна система
ММД	мова маніпулювання даними
МОД	мова опису даних
ОС	операційна система
ПК	персональний комп'ютер
СУБД	системи управління базою даних
СУБМ	системи управління базою моделей

ВСТУП

Актуальність дослідження. Місія деталізує статус і забезпечує напрям і орієнтири для визначення цілей і стратегій на різних рівнях. Добре сформульована місія прояснює те, чим являється організація і якою вона прагне бути, а також показує відміну організації від інших їй подібних.

Актуальність теми підкреслюється тим, що інститут сім'ї сьогодні переживає кризу. Сім'я відчуває на собі вплив економічних, правових, ідеологічних, моральних відносин. Наслідки переходу до ринкової економіки і усунення держави від комплексної підтримки сім'ї не сповільнили позначитися.

Непрямим підтвердженням тому служить хвиля дитячої безпритульності і бездоглядності. Прямим – низький рівень доходів основної маси українських сімей, житлово-побутова невлаштованість, свідомо орієнтація на малу кіл кіст дітей, заснована на невпевненості у завтрашньому дні, зростання числа цивільних шлюбів і позашлюбних народжень.

Створення нормальних умов для виробництва людського фактора вимагає не тільки ефективного ведення домашнього господарства, але й формування та використання сімейного бюджету.

На сьогоднішній день розробник не пов'язаний рамками якого-небудь конкретного пакету, а залежно від поставленого завдання може використовувати різні додатки.

Темою даної роботи є розробка Personal accounting для моніторингу фінансових витрат на мові С. Ця тема близька кожному. У кожній сім'ї є свої доходи і витрати. Природно виникає необхідність обліку фінансових засобів.

Під дохідними статтями можна розуміти будь-які джерела коштів. Наприклад, зарплата, стипендія, підробітки, фінансова допомога від батьків тощо.

Витрати також можна розділити на безліч підрозділів: витрати на їжу; господарські потреби; витрати на транспорт; комунальні послуги.

Для забезпечення надійності розроблюваної інформаційної системи управління даними слід виконати наступні основні вимоги: цілісність і

несуперечність даних, достовірність даних, простота управління даними, безпека доступу до даних.

Програма повинна дозволяти легко вводити інформацію про доходи та витрати, а також дозволяти здійснювати аналіз інформації. Вона повинна дозволяти відображати інформацію за введений користувачем період з сортуванням за вказаним полем, дозволяючи додавати додаткові елементи, а також надавати зручний користувальницький інтерфейс, який відповідає стандартам ергономіки.

Використання баз даних та інформаційних систем стає невід'ємною складовою ділової діяльності сучасної людини і функціонування процвітаючих організацій.

Важливою категорією є системи обробки інформації, від яких багато в чому залежить ефективність роботи будь-якого підприємства чи установи.

Така система повинна:

1. Забезпечувати отримання загальних і деталізованих звітів за підсумками роботи;
2. Дозволяти легко визначати тенденції зміни найважливіших показників;
3. Забезпечувати отримання інформації, критичної за часом, без істотних затримок;
4. Виконувати точний і повний аналіз даних.

Тому, щоб універсальним способом отримувати з бази даних запису, обробляти, змінювати та видаляти їх, потрібні спеціальні програми, які називаються системами управління базами даних або скорочено СУБД.

Мета та завдання роботи. Метою роботи є покращення моніторингу фінансових витрат за допомогою додатку Personal accounting мовою С.

Для досягнення поставленої мети у роботі варто виконати низку завдань:

- здійснити теоретичний аналіз і обробку практичної і методичної літератури по темі дипломної роботи;
- виконати проектування і реалізацію Personal accounting для моніторингу фінансових витрат на мові С.

- запропонувати реалізацію отриманої БД у програмному середовищі C#.
- навести обґрунтування економічної ефективності та впровадження Personal accounting для моніторингу фінансових витрат на мові C.

Розробка такого роду програм вимагає врахування специфіки роботи. Для цього необхідно виявити *об'єкт дослідження* даного дипломного проекту – підвищення ефективності моніторингу фінансових витрат користувача.

Предметом для розгляду є система обробки інформації для моніторингу фінансових витрат на мові C.

В рамках даного дослідження були використані наступні *методи*:

Методи теорії інформації.

Методи структурного аналізу і проектування.

Методи розробки програмного забезпечення.

Методи тестування, валідації та верифікації програмного забезпечення.

Практична значимість даної роботи полягає в тому, що результат досягнення поставленої мети має можливість застосування в діяльності конкретного закладу, сім'ї, а також може бути використаний і іншими користувачами для вдосконалення аналізу фінансових витрат на мові C, також дана концепція викладення даного дослідження може бути використана в якості методичного допомоги при розробці інформаційних систем.

Структура роботи включає в себе вступ, три розділи, висновок, список літератури з 40 джерел та 1 додаток.

РОЗДІЛ 1.

ДОСЛІДЖЕННЯ АКТУАЛЬНОСТІ МОНІТОРИГУ ФІНАНСОВИХ ВИТРАТ КОРИСТУВАЧІВ

1.1 Аналіз актуальності підвищення ефективності моніторингу витрат

В сучасних умовах підвищених вимог до рівня фінансової грамотності суб'єктів господарювання, укупі з проблемою зниження рівня реальних наявних доходів громадян, на тлі тривалої фінансово-економічної кризи, актуальність ефективного управління особистими фінансами на рівні окремо взятої сім'ї надзвичайно висока.

В умовах нарощення складності фінансових відносин населення, відходу держави від політики патерналізму в сфері обов'язкового пенсійного страхування, стрімкого розвитку в нашій країні ринку кредитування на тлі високої інфляції особлива увага приділяється підвищенню фінансової грамотності українців.

У зазначених умовах високу актуальність набуває розробка зручного і гнучкого інструментарію ведення сім'ями обліку своїх доходів, витрат, майна і зобов'язань, що вимагає розробки і впровадження в українську практику нового підвиду бухгалтерського обліку, адаптованого до особливостей особистих фінансів українців – персонального бухгалтерського обліку. В даний час науковий підхід до персонального бухгалтерського обліку у вітчизняній економічній науці не сформований. Це пояснюється необов'язковістю ведення сім'ями бухгалтерського обліку, що, з одного боку, відповідає принципу не регламентованості державою особистих фінансів в ринковій економіці. А з іншого, істотно ускладнює впровадження в приватну практику методів особистого фінансового планування, оскільки воно базується на даних обліку і аналізу: доходів і витрат сімейного бюджету, а також обсягів і структури активів і пасивів сім'ї.

Історично бухгалтерський облік розвивався як будь-яка наука від простого до складного. Від епохи палеоліту до епохи феодалізму бухгалтерський облік

можна розглядати саме як персональний облік домашніх господарств, і тільки з розвитком індустріалізації суспільства розвиток бухгалтерського обліку, як науки і практичної діяльності, придбав новий вектор розвитку, орієнтований на задоволення інформаційних потреб підприємств і держави в цілому.

Якщо для більшості жителів Європи вести облік як мінімум своїх доходів і витрат – хороша звичка, то для українців швидше виняток з правил, ніж правило. Необхідність враховувати особисті доходи і витрати, на нашу думку, була втрачена українськими громадянами в радянський період історичного розвитку нашої країни. Молода держава на шляху до комунізму постаралася взяти всю соціальну відповідальність за своїх громадян на себе. Безкоштовне навчання, медичне обслуговування, відпочинок, заняття спортом, житло, пільгове харчування для окремих категорій громадян – з одного боку, невеликі доходи, як правило, з одного джерела (заробітна плата), низький рівень розвитку кредитування населення – з іншого боку, призвели до відсутності особистої відповідальності українців за управління своїми фінансами.

На нашу думку, розробка методології та методики нового облікового підвиду - персонального бухгалтерського обліку - повинна ґрунтуватися на наявному теоретичному і практичному потенціалі рахівництва.

Метою персонального фінансового обліку є формування інформаційної бази, релевантної для управління особистими фінансами.

Під управлінням особистими фінансами розуміється процес усвідомленого цілеспрямованого управління фінансовими відносинами індивіда і / або сім'ї з позиції їх оптимізації з метою досягнення поставлених членами сім'ї життєвих цілей [7].

Життєві цілі індивіда або сім'ї визначаються ними суб'єктивно і є по суті усвідомленими мріями і життєвими орієнтирами, вони спрямовані на задоволення особистих потреб як самого індивіда або окремого члена сім'ї, так і всієї родини в цілому. Ці цілі є своєрідними орієнтирами і критеріями сімейного благополуччя. Кожна життєва мета має дві характеристики: вартість і час досягнення.

Підходи до системи обліку особистих фінансів вітчизняних авторів, як правило, ґрунтуються на базових принципах бухгалтерського обліку підприємства (докладніше позиції окремих дослідників в області персонального обліку домашніх господарств були викладені в журналі «Фінансова аналітика» [6]). Ми підтримуємо цю позицію, але вважаємо за доцільне визначити місце персонального обліку в існуючій диференціації сучасного бухгалтерського обліку [3] з метою виявлення специфічного облікового інструментарію, прийняттого для його організації. Для цього проведемо порівняльний аналіз традиційних підвидів бухгалтерського обліку: фінансового та управлінського.

Середню структуру доходів і витрат можна представити таким чином:



Рисунок 1.1 – Структура доходів



Рисунок 1.2 – Структура витрат

У процесі дослідження у нас виникла стійке думка про те, що за сукупністю класифікаційних ознак персональний бухгалтерський облік можна віднести до різновиду управлінського обліку та сформулювати його визначення наступним чином. Персональний бухгалтерський облік - упорядкована система збору, реєстрації та узагальнення інформації про факти господарського життя сім'ї з метою здійснення ефективного управління особистими фінансами. При цьому, за нашу думку, персональний бухгалтерський облік володіє наступними специфічними ознаками:

- здійснюється конфіденційно на добровільній основі, законодавчо не регламентується;
- ведеться по уніграфічній формі за допомогою реєстрації та узагальнення інформації в бухгалтерських регістрах;
- доходи і витрати враховуються касовим методом;
- може здійснюватися як безперервно, так і періодично;

– включає розрахунок як абсолютних, так і відносних показників, що дозволяють акумулювати не тільки поточну облікову інформацію, але прогнозувати її на майбутнє;

– обліковими об'єктами виступають доходи, витрати, майно і зобов'язання сім'ї.

Утвердившись в думці про те, що до персонального бухгалтерського обліку можливе застосування практичного інструментарію управлінського обліку, вважаємо, що в основу методики його ведення може бути покладена система бюджетування, що включає в себе як мінімум три види річних і місячних базових бюджетів:

- 1) бюджет доходів;
- 2) бюджет споживчих витрат (споживчий бюджет);
- 3) бюджет заощаджень та інвестицій (інвестиційно-ощадний бюджет).

Система базових бюджетів може бути доповнена бюджетом зобов'язань сім'ї та стратегічними бюджетами (наприклад, бюджетами пенсійної стратегії, страховий стратегії та ін.).

Майно і зобов'язання індивіда і / або сім'ї формуються в результаті їх участі в створенні і використанні особистих і загальносімейних коштів – бюджетів. Індивід, якщо він проживає в сім'ї, самостійно визначає ступінь своєї участі у формуванні загального бюджету [2]. Розглядаючи бюджет сім'ї, більш коректно говорити про систему бюджетів в рамках єдиного сімейного бюджету, так як кожен з її членів може вести індивідуальний облік своїх доходів і витрат [1]. Відповідно, представлена система бюджетів може дублюватися на загальносімейним і індивідуальних (персональних) рівнях і вестися (складатися) одноосібно обраним членом сім'ї та / або спільно, і / або окремо кожним членом сім'ї.

Теоретично часовий період охоплення системи бюджетування може бути різним - денним, місячним, річним або навіть на все життя особи або сім'ї, на практиці оптимально складання місячного і річного бюджетів. Безперервність персонального обліку визначається частотністю подій: отримання / надходження

доходу, вчиненням витрат, здійсненням заощаджень і придбанням ощадно-інвестиційних інструментів. Інвентаризація активів і зобов'язань повинна проводитися зі встановленою регулярністю, але не рідше ніж раз на місяць і в кінці - початку кожного року.

Завдання персонального обліку впливають з необхідності і потреби членів сім'ї [7]:

- в фіксації і співвіднесенні джерела надходження і напрямки використання грошових коштів сім'ї і окремих її членів як за сумами, так і за термінами;
- оптимізації співвідношення доходів сім'ї з її витратами на задоволення потреб її членів окремо і в сукупності (особистих і загальносімейних потреб), з огляду на правило випереджаючого розвитку потреб людини над можливостями їх задоволення (тобто витрати сім'ї завжди ростуть швидше її доходів);
- виділення із сімейного бюджету коштів на неспоживацькі витрати – на заощадження та інвестиції, а також на ведення бізнесу (приватне підприємництво);
- можливості варіативного управління кредитними зобов'язаннями сім'ї та окремих її членів, їх активами і пасивами (страховим, інвестиційним, пенсійним та іншими видами портфелів);
- створення фінансової подушки безпеки сім'ї та окремих її членів як певної структури високоліквідних активів, основне призначення яких - забезпечити сім'ї та окремих її членів засобами для існування в разі виникнення непередбачених поточних витрат, і компенсувати тимчасову втрату основного джерела доходу терміном від 3 до 6 місяців (наприклад, в разі непрацездатності основного добувача сім'ї).

Виходячи із зазначених цілей, бюджети доходів і витрат повинні вестися паралельно, безперервно в зіставленні і взаємозалежності, фіксуючи всі поточні доходи і витрати. З огляду на можливий розрив у часі і за обсягами доходів і

витрат, необхідно робити грошові запаси (резерви) на погашення цих розривів. Необхідність формування таких резервів визначається негативним сальдо між отриманими доходами і планованих витрат на певний момент часу. Бюджет заощаджень та інвестицій досить складати раз на місяць. Результати місячних бюджетів доходів, витрат, заощаджень та інвестицій необхідно щорічно зводити до загального річного бюджету. Дані річних бюджетів необхідні для побудови особистого фінансового плану сім'ї [4].

Бюджет доходів акумулює дані обліку джерел, обсягу, структури за видами і динаміки доходів сім'ї, виміряні за певний період часу (день, місяць, рік, все життя). При цьому під доходом з позиції приток ресурсів розуміється збільшення обсягу грошових ресурсів сім'ї за рахунок припливу грошових і не грошових (матеріальних і нематеріальних) ресурсів в сім'ю, в тому числі у формі економії, отриманні вигод, повинні розраховуватися за певний період часу [7].

З позиції відтоку ресурсів дохід сім'ї є максимально можливі грошові витрати (відтоки грошових коштів) сім'ї протягом певного періоду часу, які сім'я здатна здійснити без зміни свого майнового статусу (положення), тобто нічого не продаючи зі свого майна [7]. Необхідно регулярно і безперервно планувати, враховувати і оцінювати (моніторити) всі доходи сім'ї в абсолютному вираженні (в тій валюті, в якій передбачається їх витратити) і відносному вираженні (у відсотках від базового показника) за обсягом, за структурою (джерелами) і за динамікою.

1.2 Огляд існуючих систем та програмних додатків

Планування власних витрат і доходів вважається головною основою благополуччя в фінансовому плані.

За особистими доходами і витратами стежити досить просто. Сімейний бюджет - це вже повноцінна бухгалтерія, в якій потрібно враховувати всі джерела надходження коштів і інтереси кожного.

Для ефективного використання додаток повинен бути розрахований на багато користувачів і підходити під різні операційні системи, щоб кожен член сім'ї міг вносити зміни в режимі реального часу. За можливість доступу кількох людей розробники, як правило, пропонують заплатити.

Розглянемо деякі популярні програми для економії бюджету сім'ї, його розумного і кваліфікованого складання.

1. Домашня бухгалтерія

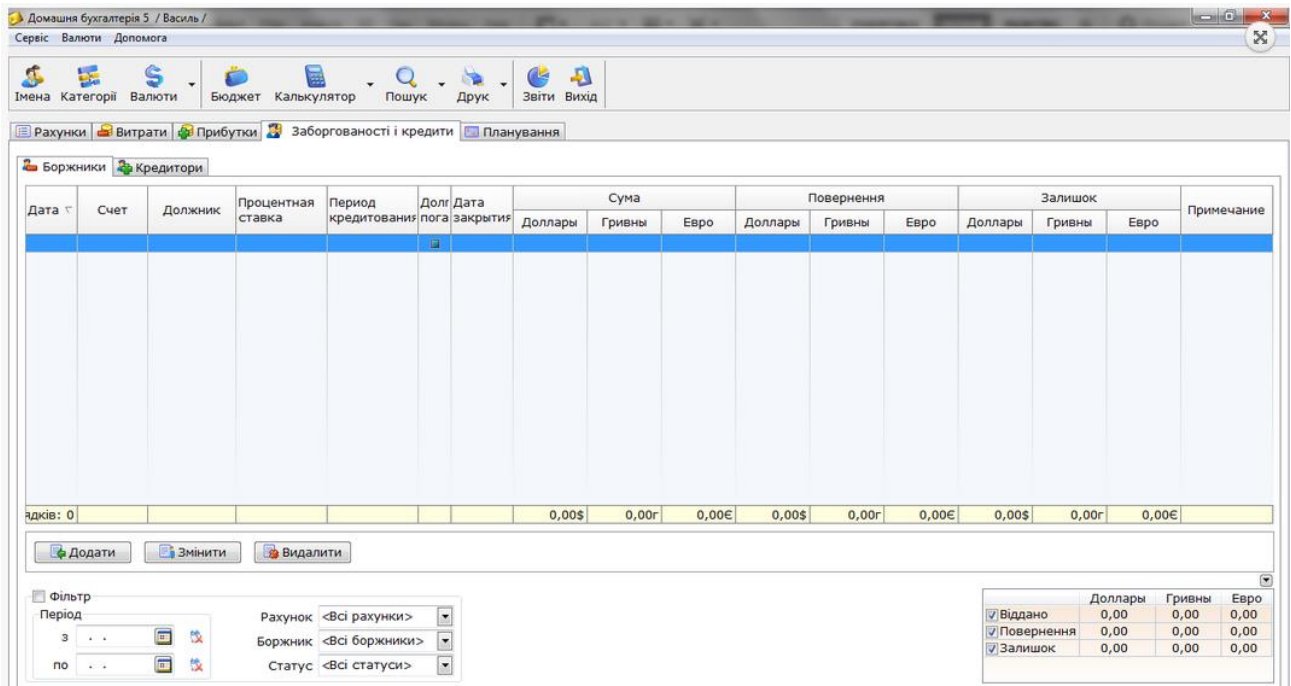


Рисунок 1.3 – Домашня бухгалтерія

Додаток становить докладний звіт щодо загального бюджету і показує індивідуальні витрати. Кожен член сім'ї може коментувати свої витрати. «Домашня бухгалтерія» дозволяють налаштувати розпізнавання СМС від банку і їх автоматичний облік в русі коштів. Бухгалтерію можна вести в різних валютах і на різних рахунках. Від сторонніх очей додаток захищено паролем і ПІН-кодом.

2. Сімейний бюджет

Сімейний бюджет на місяць

Всього заплановані витрати	Загальні	Загальні
1 203 грн.	1 317 грн.	-114 грн.

Щитло	Запланована	Фактична	Різниця
Іпотека або орендні	1 000 грн.	1 000 грн.	0 грн.
Друга іпотека або	0 грн.	0 грн.	0 грн.
Телефон	62 грн.	100 грн.	-38 грн.
Електрика	44 грн.	125 грн.	-81 грн.
Газ	22 грн.	35 грн.	-13 грн.
Вода та каналізація	8 грн.	8 грн.	0 грн.
Кабельне	34 грн.	39 грн.	-5 грн.
Вивезена сміття	10 грн.	10 грн.	0 грн.
Обслуговування та	23 грн.	0 грн.	23 грн.
Начеена	0 грн.	0 грн.	0 грн.
Інше	0 грн.	0 грн.	0 грн.
Разом	1 203 грн.	1 317 грн.	-114 грн.

Транспорт	Запланована	Фактична витр.	Різниця
Виплата за			0 грн.
Виплата за			0 грн.
Автобус, таксі			0 грн.
Страхованя			0 грн.
Права			0 грн.
Паливо			0 грн.
Обслуговуваня			0 грн.
Інше			0 грн.
Разом	0 грн.	0 грн.	0 грн.

Страхованя	Запланована	Фактична	Різниця
Головна			0 грн.

Запланований дохід на місяць	
Income 1	4 000 грн.
Income 2	1 200 грн.
Extra income	300 грн.
Total monthly income	5 500 грн.

Фактичний дохід за місяць	
Дохід 1	4 000 грн.
Дохід 2	1 200 грн.
Додатковий дохід	300 грн.
Загальний дохід за	5 500 грн.

Запланований баланс	4 297 грн.
Фактичний баланс	4 183 грн.
Різниця	-114 грн.

Позики	Різниця
Особисті	0 грн.
Студентська	0 грн.
Кредитна картка	0 грн.
Кредитна картка	0 грн.
Кредитна картка	0 грн.
Інше	0 грн.
Разом	0 грн.

Дозвілля	Різниця
Відео/DVD	0 грн.
Компакт-диски	0 грн.
Кіно	0 грн.

Рисунок 1.4 – Сімейний бюджет

Багатокористувацький додаток, в якому можна одночасно планувати особистий і сімейний бюджет. Він дозволяє імпортувати операції з банківських служб, систем електронних грошей. «Сімейний бюджет» надає детальний аналіз витрат і враховує рух коштів в різних валютах, включаючи криптовалюту. Статистичні дані доступні з програми і на сайті.

3. CoinKeeper

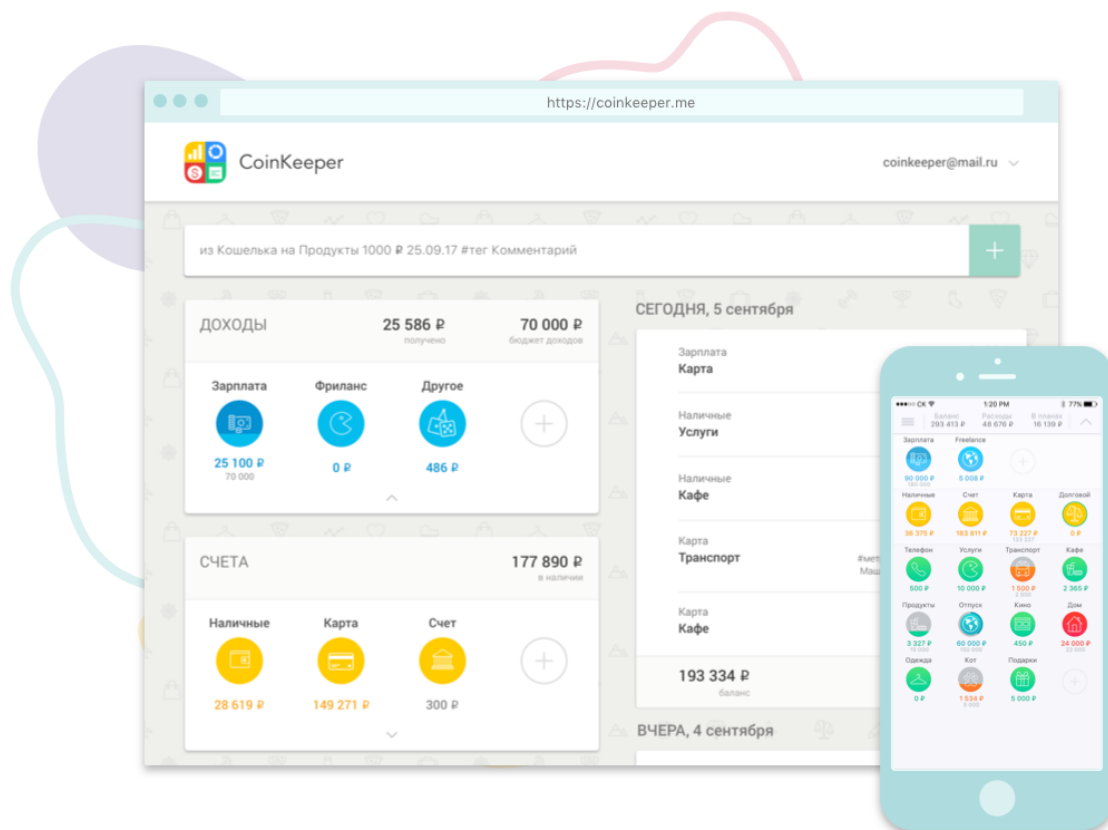


Рисунок 1.5 – CoinKeeper

Управляти фінансами можна за допомогою мобільного додатку або веб-версії сервісу. Розробники запевняють, що CoinKeeper дозволяє розібратися і з грошовими потоками сім'ї, і з бюджетом невеликої фірми. Додаток підтримує імпорт операцій з більш ніж 150 банків, розпізнає витрати з СМС. Можна встановити нагадування про повернення боргів і ліміт на витрати в звітний період.

4. Toshl

Додаток підтримує майже 200 валют, включаючи 30 криптовалют, і вносить відповідні зміни курсу. Імпортує дані з файлів онлайн-банку, нагадує про необхідність внести витрати у відповідні графи, автоматично синхронізується з сайтом. Стежити за бюджетом допоможуть веселі монстри. Для консерваторів додаток генерує звіти в PDF, Excel і Google Docs.

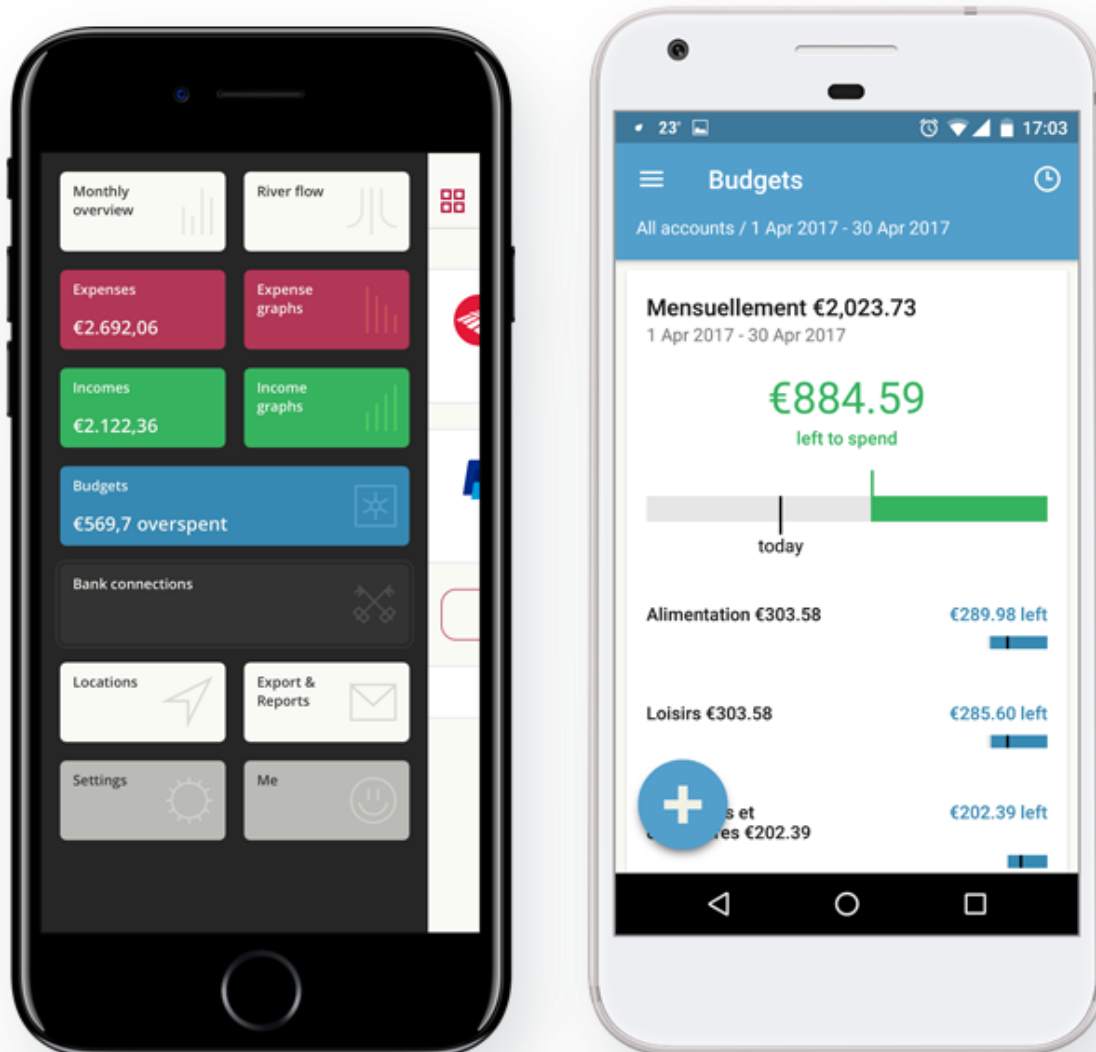


Рисунок 1.6 – Toshl

5. EasyFinance

Розробники називають свій сервіс фінансовим навігатором в браузері і смартфоні і обіцяють облік особистих фінансів, імпорт і синхронізацію операцій з більш ніж 200 банків десяти країн. Додаток дозволяє прогнозувати витрати і планувати бюджет на короткострокову та довгострокову перспективу.

The screenshot displays the EasyFinance web application interface. At the top, there is a navigation bar with tabs for 'Инфо', 'Учет', 'План', 'Календарь', 'Отчеты', and 'Магазины'. Below this, there are buttons for 'Добавить' and 'Запланировать'. The main content area is titled 'ЖУРНАЛ ОПЕРАЦИЙ' and shows a list of transactions for the year 2014. The transactions are summarized as follows:

Дата	Тип	Сумма	Все счета	Все категории	Метки и комментарии
28.12	-	-5 000	Debit card	Питание дома	[ашан]
25.12	-	-30 000	Debit card	Аренда жилья	[аренда]
21.12	-	-5 000	Debit card	Питание дома	[ашан]
16.12	+	70 000	Debit card	Зарботная плата	[зарплата]
15.12	↔	-10 000	Debit card	Debit card => Loan	
14.12	-	-5 000	Debit card	Питание дома	[ашан]
07.12	-	-5 000	Debit card	Питание дома	[ашан]
30.11	-	-5 000	Debit card	Питание дома	[ашан]
22.11	-	-5 000	Debit card	Питание дома	[ашан]
20.11	+	70 000	Debit card	Зарботная плата	[зарплата]

Summary statistics: Было: 708 758.23 P, Изменение: 86 130.14 P, Стало: 794 888.37 P. The interface also includes a sidebar with account balances and a footer with contact information and social media links.

Рисунок 1.7 – EasyFinance

6. Alzex Finance

Програма дозволяє створити кожному користувачеві свій обліковий запис. Завдяки цьому кожен член сім'ї зможе вибирати, які рухи коштів зробити доступними всім, а які приховати. Деревоподібна система тегів дозволяє враховувати витрати по великим і дрібним категоріям. У додатку можна стежити за боргами і ставити фінансові цілі.

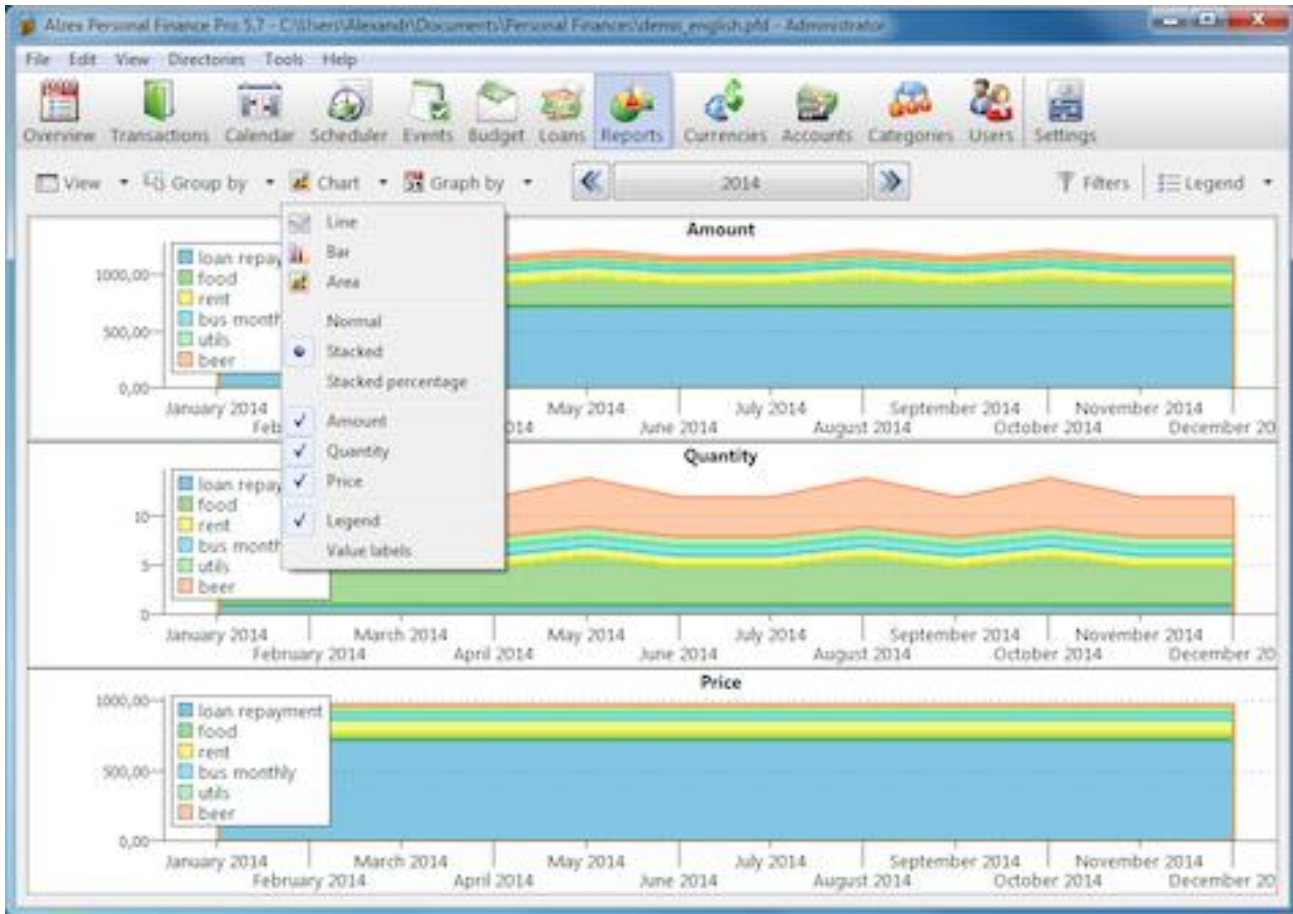


Рисунок 1.8 – Alzex Finance

7. YNAB

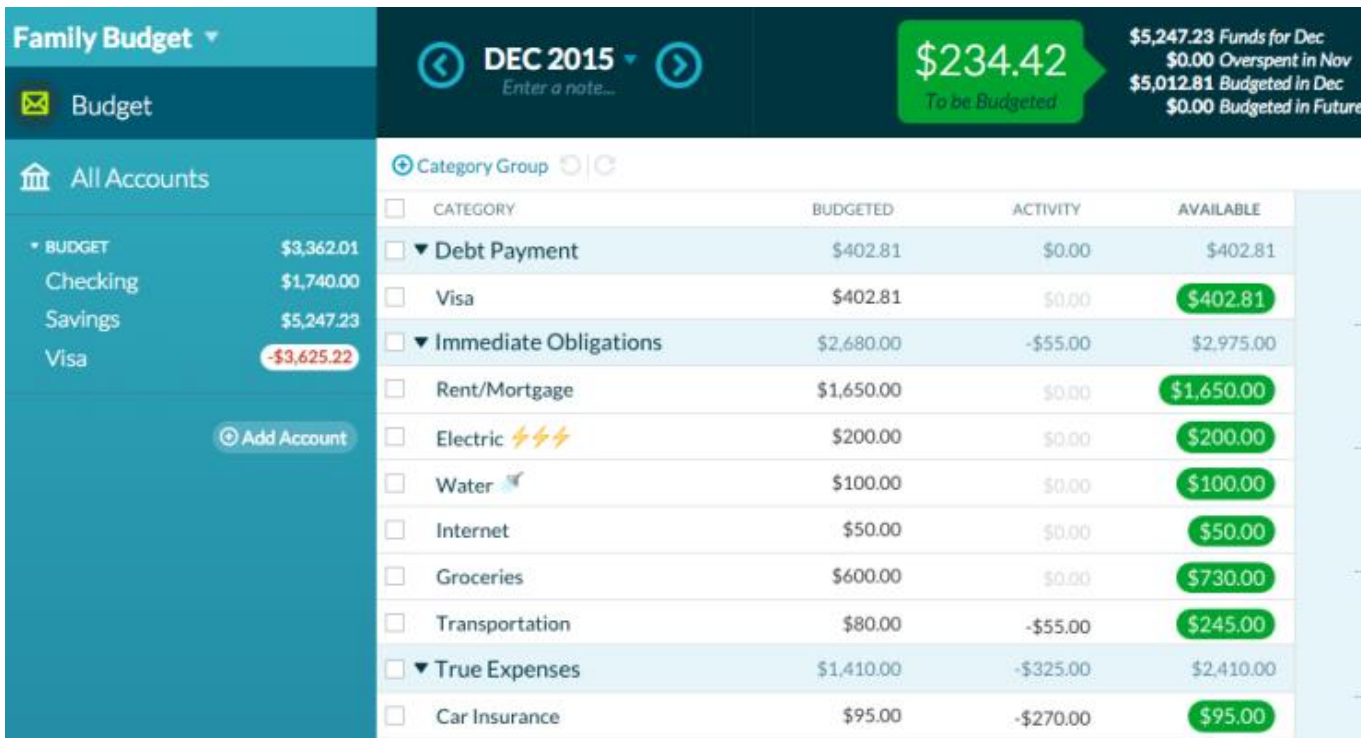


Рисунок 1.9 – YNAB

Програма працює тільки з однією валютою і не русифікована, але ці недоліки перебиваються її перевагами. YNAV не просто враховує доходи і витрати. Це повноцінна система, яка дозволяє планувати бюджет на короткострокову та довгострокову перспективу. Вона визначає недовитрати і перевитрати і коректно працює з боргами.

8. Cash Organizer

Розрахована на багато користувачів програма, в якій кожен користувач може приховати частину переміщень коштів за рахунками від інших членів сім'ї. Дані можна вносити офлайн, вони згодом синхронізуються з усіма пристроями через хмару. Розробники обіцяють захистити дані 256-бітовим шифруванням, так що фінансові відомості про користувачів не впізнають навіть вони.

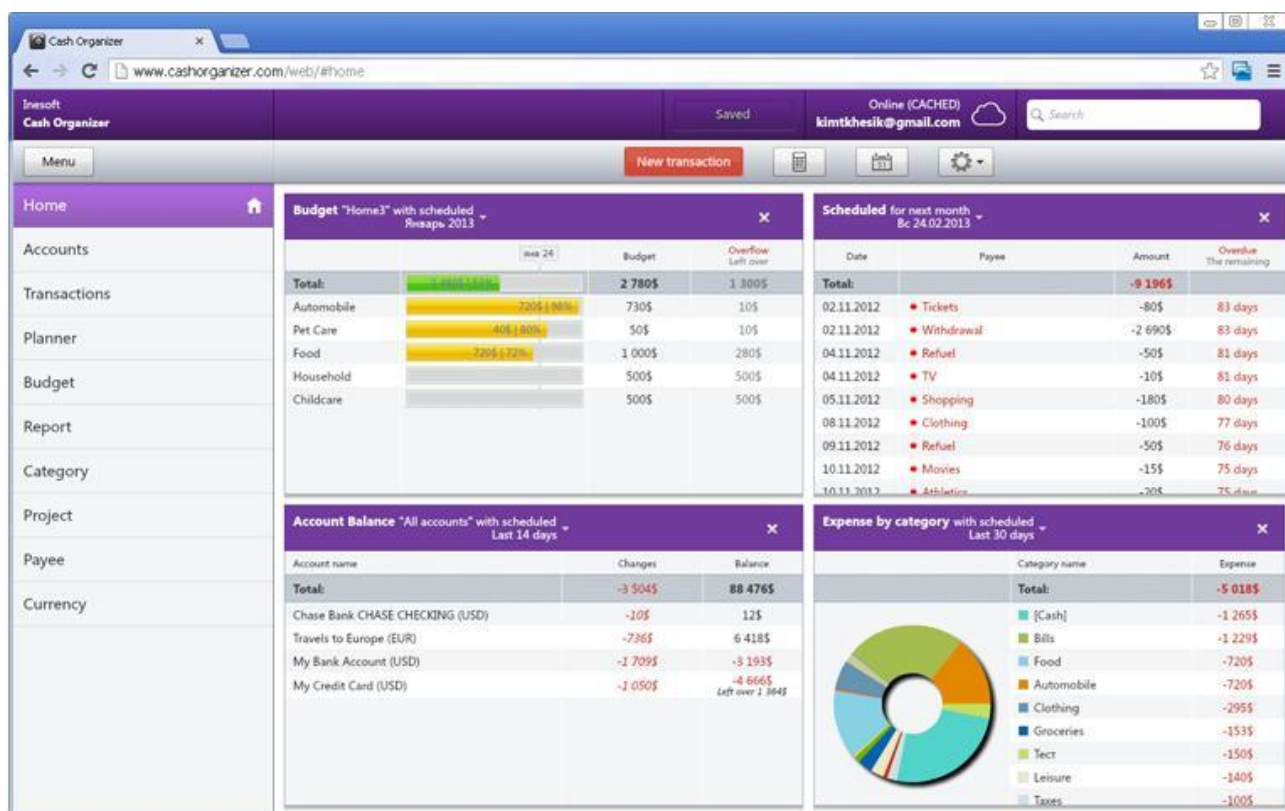


Рисунок 1.10 – Cash Organizer

9. Домашня бухгалтерія

The screenshot displays the Alzex Personal Finance Free 5.9 application window. The title bar shows the file path: C:\Users\Alex\Documents\Personal Finances\Demo.pfd - Администратор. The menu bar includes: Файл, Правка, Справочники, Инструменты, Помощь, and Перейти на версию Pro!. The toolbar contains icons for Валюты, Счета, Категории, Семья, and Настройки. The main interface is divided into several sections:

- Left Panel (Accounts):**
 - Общий баланс: 55 725,30р.
 - Кошелек Саша: 3 600,00р.
 - Кошелек Аня: 35 800,00р.
 - Заначка Саша (скрытый): 9 900,00р.
 - Visa Саша: \$-179,19
 - MasterCard Аня: €0,00
 - Счет Сбербанк: 16 948,50р.
- Top Center (Navigation):**
 - Вид: [icon]
 - Группировка: [icon]
 - Апрель 2015
- Main Table (Transactions):**

Описание	Дата	Сумма	Категория	Член семьи
+ Доходы		24 410,00		
Зарплата		14 900,00		
Халтура		2 500,00		
Скрытая категория		10,00		
Проценты		7 000,00		
- Расходы		-12 568,14		
Еда для питомца		-900,00		
Продукты		-4 700,00		
Транспорт		-500,00		
Отдых		-1 958,14		
Услуги		-2 500,00		
Автомобиль		-2 000,00		
Обслуживание		-1 000,00		
Бензин	20.04.2015	-500,00	Автомобиль	Аня
Бензин	06.04.2015	-500,00	Автомобиль	Аня
Скрытая категория		-10,00		
Итого		11 841,86р.		
- Right Panel (Actions):**
 - + Добавить
 - Копировать
 - Изменить
 - Удалить

Рисунок 1.11 – Домашня бухгалтерія

Додаток дозволяє вести облік витрат і доходів на всіх валютах світу, одночасно можна використовувати будь-які дві. Всі дані синхронізуються з додатком на ПК. Доступна портативна версія програми, яка встановлюється на флешку. Записи кожного члена сім'ї захищаються паролем.

1.3 Постановка завдань дослідження

Місія деталізує статус і забезпечує напрям і орієнтири для визначення цілей і стратегій на різних рівнях. Добре сформульована місія прояснює те, чим є

організація і якою вона прагне бути, а також показує відміну організації від інших їй подібних.

Темою даної роботи є Personal accounting для моніторингу фінансових витрат. Дана тема близька кожному. У будь-якої людини є свої доходи і витрати. Природно виникає необхідність врахування фінансових коштів.

Під прибутковими статтями можна розуміти будь-які джерела коштів. Наприклад, зарплата, стипендія, підробіток, фінансова допомога від батьків тощо.

Витрати також можна розділити на безліч підрозділів: витрати на їжу; господарські потреби; витрати на транспорт; комунальні послуги.

Для забезпечення надійності розроблювальної інформаційної системи управління даними слід виконати наступні основні вимоги:

- цілісність і несуперечність даних,
- достовірність даних,
- простота управління даними,
- безпека доступу до даних.

Програма повинна дозволяти легко вводити інформацію про доходи та витрати, а також дозволяти здійснювати аналіз інформація. Вона повинна дозволяти відображати інформацію за якою користувач вів період з сортуванням за вказаним полем, вести класифікатор статей доходу і витрати, дозволяючи додавати додаткові елементи, графічно відображати частки доходів і витрат з угрупованням за статтями, а також надавати зручний призначений для користувача інтерфейс, який відповідає стандартам ергономіки.

Пріоритетним завданням є створення бази даних для маленької власної бухгалтерії, щоб швидко і точно враховувати всі доходи і витрати.

Отже, до переліку обов'язкових повинні увійти такі вимоги:

- облік всіх надходжень за сумами;
- облік джерел надходжень;

- облік поточного курсу, наприклад до долара, так як високі темпи інфляції в країні не дозволяють коректно порівнювати навіть ціни на масло тримісячної давності;
- облік найменувань статей витрат по групах товарів;
- облік витрат по найменуванню товарів;
- облік залишків коштів;
- облік різних варіантів вкладень коштів у різні фінансові інструменти, наприклад в валюту, на депозити і так далі.

Чим докладніше і ретельніше буде виконана вся первісна підготовча робота, тим менше потім піде часу на конкретне програмування. Проблема полягає в тому, що помилка, яка "спливла" на будь-якому етапі або логічна невідповідність при програмуванні обертається часом навіть повною зміною всього проекту в цілому, що нерідко викликає переробку таблиць, переписування запитів і написання нових форм.

РОЗДІЛ 2

ПРОЕКТНІ І ТЕХНІЧНІ РІШЕННЯ ЩОДО РОЗРОБКИ PERSONAL ACCOUNTING МОВОЮ С#

2.1 Технічні рішення

Технічне забезпечення – комплекс технічних засобів, призначених для роботи інформаційної системи, а також відповідна документація на ці засоби і технологічні процеси.

Комплекс технічних засобів становить:

- Комп'ютери будь-яких моделей;
- Пристрої збору, накопичення, обробки, передачі та виведення інформації;
- Пристрої передачі даних і ліній зв'язку;
- Оргтехніка й пристрої автоматичного знімання інформації;
- Експлуатаційні матеріали та ін.

Документацією оформлюється попередній вибір технічних засобів, організація їх експлуатації, технологічний процес обробки даних, технологічне оснащення. Документацію можна умовно розділити на три групи:

- Загальносистемну, що включає державні та галузеві стандарти по технічному забезпеченню;
- Спеціалізовану, яка містить комплекс методик по всіх етапах розробки технічного забезпечення;
- Нормативно-довідкову, використовувану при виконанні розрахунків по технічному забезпеченню.

До теперішнього часу склалися дві основні форми організації технічного забезпечення (форми використання технічних засобів): централізована і частково або повністю децентралізоване.

Централізоване технічне забезпечення базується на використанні в інформаційній системі великих ЕОМ і обчислювальних центрів.

Децентралізація технічних засобів припускає реалізацію функціональних підсистем на персональних комп'ютерах безпосередньо на робочих місцях.

Перспективним підходом варто вважати, мабуть, частково децентралізований підхід – організацію технічного забезпечення на базі розподілених мереж, що складаються з персональних комп'ютерів і великій ЕОМ для зберігання баз даних, загальних для будь-яких функціональних підсистем.

Еволюція технічного забезпечення, яке включає в себе апаратні засоби, засоби комунікації, програмне забезпечення, проходить нерівномірно, стрибкоподібно. Розвиток комп'ютерної техніки поки відбувається в геометричній прогресії. Кожні чотири роки відбувається подвоєння продуктивності комп'ютерів.

Класифікація комп'ютерних технологій за типом користувацького інтерфейсу (як взаємодіє користувач технології з комп'ютером) - пакетні, діалогові, мережеві. У першому випадку користувач отримує тільки результати роботи технології, в інших він взаємодіє з нею на індивідуальному комп'ютері чи комп'ютері, підключеному до мережі ЕОМ.

Сучасні технічні засоби забезпечення управління інформаційними ресурсами за своїм складом і функціональним можливостям вельми різноманітні. Засоби обчислювальної техніки, засоби комунікаційної техніки, засоби організаційної техніки [6].

Комп'ютерна техніка призначена, в основному, для реалізації комплексних технологій обробки і зберігання інформації і є базою інтеграцією всіх сучасних технічних засобів забезпечення управління інформаційними ресурсами.

Комунікаційна техніка призначена, в основному, для реалізації технологій передачі інформації і припускає як автономне функціонування, так і функціонування в комплексі із засобами комп'ютерної техніки.

Організаційна техніка призначена для реалізації технологій зберігання, подання та використання інформації, а також для виконання різних допоміжних операцій в рамках тих чи інших технологій інформаційної підтримки управлінської діяльності.

Програмні засоби автоматизації офісу діляться на системні та прикладні програми.

До системних програм відносяться: тестові та діагностичні програми, антивірусні програми, операційні системи, командно-файлові процесори.

Тестові і діагностичні програми призначені для перевірки працездатності окремих вузлів комп'ютера і компонентів програмно-файлових систем і, можливо, для усунення виявлених несправностей.

Антивірусні програми призначені для виявлення і, можливо, усунення вірусних програм, що порушують нормальну роботу обчислювальної системи.

Операційні системи є основними системними програмними комплексами, які виконують такі основні функції: тестування працездатності обчислювальної системи і її настройку при первісному включенні; забезпечення синхронної та ефективної взаємодії всіх апаратних і програмних компонентів обчислювальної системи в процесі її функціонування; забезпечення ефективної взаємодії користувача з обчислювальною системою.

Операційні системи класифікуються наступним чином: одно користувачькі одно задачні системи (MS-DOS, DR-DOS); одно користувачькі багато задачні системи (OS/2, Windows 2007); багатокористувачькі системи (системи сімейства UNIX).

Командно-файлові процесори (оболонки) призначені для організації системи взаємодії користувача з обчислювальною системою на принципах, відмінних від реалізованих операційною системою, з метою полегшення роботи або надання додаткових можливостей (наприклад, Norton Commander, Windows 3.11) [2].

Прикладні програмні засоби забезпечення управлінської діяльності класифікуються наступним чином: системи підготовки текстових документів; системи обробки фінансово-економічної інформації; системи управління базами даних; особисті інформаційні системи; системи підготовки презентацій; системи управління проектами; експертні системи та системи підтримки прийняття

рішень; системи інтелектуального проектування і вдосконалення систем управління; інші системи.

Таким чином, можна виділити основні компоненти для організації реалізації програного забезпечення:

- 32- розрядний (x86) або 64 - розрядний (x64) процесор з тактовою частотою не менш 1 гігагерц (ГГц);
- Оперативна пам'ять повинна бути обсягом не менше 1 ГБ;
- Жорсткий диск об'ємом не менше 20 ГБ;
- Матричний або лазерний принтер;
- Пристрої периферії (миша, клавіатура);
- Монітор;

Інструментальне забезпечення.

У якості інструментального засобу для створення Personal accounting для моніторингу фінансових витрат на мові C, було обрано мову програмування C#. Мова C# розроблялася як основна мова розробки додатків для платформи Microsoft .NET. Компілятор C# входить в стандартну установку .NET, тому програми на ньому можна створювати і компілювати навіть без інструментальних засобів кшталт Visual Studio.

Технологія .NET являє собою нову платформу, новий API для програмування Windows, а C# - нова мова, створена з нуля для роботи з цією платформою, а також для вилучення всіх вигод з прогресу середовищ розробки і нашого розуміння принципів об'єктно-орієнтованого програмування протягом останніх 20 років.

C# і .NET дозволяють розробникам легко створювати сучасні програмні продукти.

Особливості C#:

Повний і певний набір основних типів.

Вбудована підтримка автоматичної генерації XML документації.
Автоматичне звільнення динамічно розподіленої пам'яті.

Можливість позначки класів і методів атрибутами, що визначаються користувачем. Це може бути корисно при документуванні і здатне впливати на процес компіляції (наприклад, можна позначити методи, які повинні компілюватися тільки в налагоджувальному режимі).

Повний доступ до бібліотеки базових класів .NET, а також легкий доступ до Windows API (якщо це дійсно необхідно).

Показчики і прямий доступ до пам'яті, якщо вони необхідні. Однак мова розроблена таким чином, що практично у всіх випадках можна обійтися і без цього.

Підтримка властивостей і подій в стилі VB.

Проста зміна ключів компіляції. Дозволяє отримувати виконувані файли або бібліотеки компонентів .NET, які можуть бути викликані іншим кодом так само, як елементи керування ActiveX компоненти COM).

У якості бази даних обрано MS Access.

СУБД Access має досить високі швидкісні характеристики і входить до складу надзвичайно популярного в нашій країні і за кордоном пакету Microsoft Office.

MS Access - це сама популярна сьогодні настільна система керування базами даних. Але корінь успіху, швидше за все, полягає в прекрасній реалізації продукту, розрахованого як на починаючого, так і кваліфікованого користувача.

Набір команд і функцій по потужності і гнучкості відповідають більшості сучасних вимог до представлення і обробки даних. В Access підтримується різноманітні спливаючі і багаторівневі меню, робота з вікнами та мишею, реалізовані функції низькорівневого доступу до файлів, управління кольорами, настройки принтера, представлення даних у вигляді електронних таблиць і. т.п. Система також має засоби швидкої генерації екранів, звітів і меню, підтримує мову управління запитамі SQL, має вбудовану мову Visual Basic for Application (VBA), що дозволяє створювати масиви, свої типи даних, викликати DLL-функції, за допомогою OLE Automation контролювати роботу додатків, які можуть

функціонувати як OLE-сервери. Можливо навіть цілком створювати бази даних, за допомогою кодування, добре працює в мережі [20].

Головна якість Access, що залучає до нього багатьох користувачів, - тісна інтеграція з Microsoft Office. Наприклад, скопіювавши в буфер графічний образ таблиці, відкривши Microsoft Word і застосувавши вставку з буфера, ми відразу одержимо в документі готову таблицю з даними з БД.

Вся робота з базою даних здійснюється через вікно контейнера бази даних. Звідси здійснюється доступ до всіх об'єктів, а саме: таблицям, запитам, формам, звітам, макросам, модулям. Вони є основними компонентами БД.

2.2 Схема даних та інформаційних потоків

Принципи формування Personal accounting для моніторингу фінансових витрат на мові С базуються на реалізації класів та методів. На початку роботи програми відбувається створення екземпляра класу tIniFile, що представляє собою реалізацію windows ini файлу. І інкапсулюючого в собі методи роботи з ним. У конструкторі класу вказується ім'я файлу для відкриття. Потім за допомогою методу ReadString читається ім'я останньої робочої БД.

Якщо файл існує, то запускається процедура відкриття БД. Якщо ні, то його створення.

Після цього встановлюється властивість "Date" компонента tMonthCalendar в поточну дату. Далі перемикаються закладки на першу сторінку і встановлюється джерело даних для навігатора БД (TDBNavigator).

Далі викликається процедура відкриття запиту до БД.

В тілі будь-якої програми windows знаходиться глобальний цикл обробки повідомлень. При програмуванні на С# даний цикл ховається від програміста, тим самим, дозволяючи абстрагуватися від рутинних внутрішніх описів і фокусуватися власне на рішенні прикладної задачі. Такий принцип називається подієво орієнтованим програмуванням. Тут програміст тільки описує реакцію програми на різні події, що відбуваються.

У програмі обробляються наступні користувальницькі події:

- зміна поля для сортування. Обробник цієї події викликає процедуру перевідкриття запиту до БД;
- додавання запису. Додавання даних відбувається автоматично. В програмі після цього відбувається запуск процедури оновлення запиту;
- редагування. Аналогічно попередньому;
- видалення. Аналогічно додаванню;
- створення нової БД. Обробник викликає однойменну процедуру;
- зміна періоду перегляду. Дія аналогічна зміні поля сортування;
- перемикання закладок. При перемиканні закладок переустановлюється властивість DataSource для компонентів tDBNavigator і tDBMemo;
- класифікатор статей. Тут створюється однойменна форма, в якій користувач може редагувати список статей. Після чого форма показується модально після виклику методу ShowModal. Після цього пам'ять виділена для форми звільняється і оновлюються всі запити.
- вихід. Зариває головну форму, при цьому автоматично закриваються всі запити і з'єднання з БД.

Процедура відкриття БД полягає у наступному.

За допомогою методу Execute невізуального компонента tOpenDialog відкривається діалог відкриття файлу. Якщо користувач відмовився від вибору, то відбувається вихід з процедури.

Якщо файл вибрати його ім'я підставляється в рядок з'єднання з БД. Далі поточне активне з'єднання закривається, властивість ConnectionString компонента доступу до даних tADODConnection записується отриманий рядок з'єднання. В кінці відбувається відкриття з'єднання за допомогою виклику методу Open.

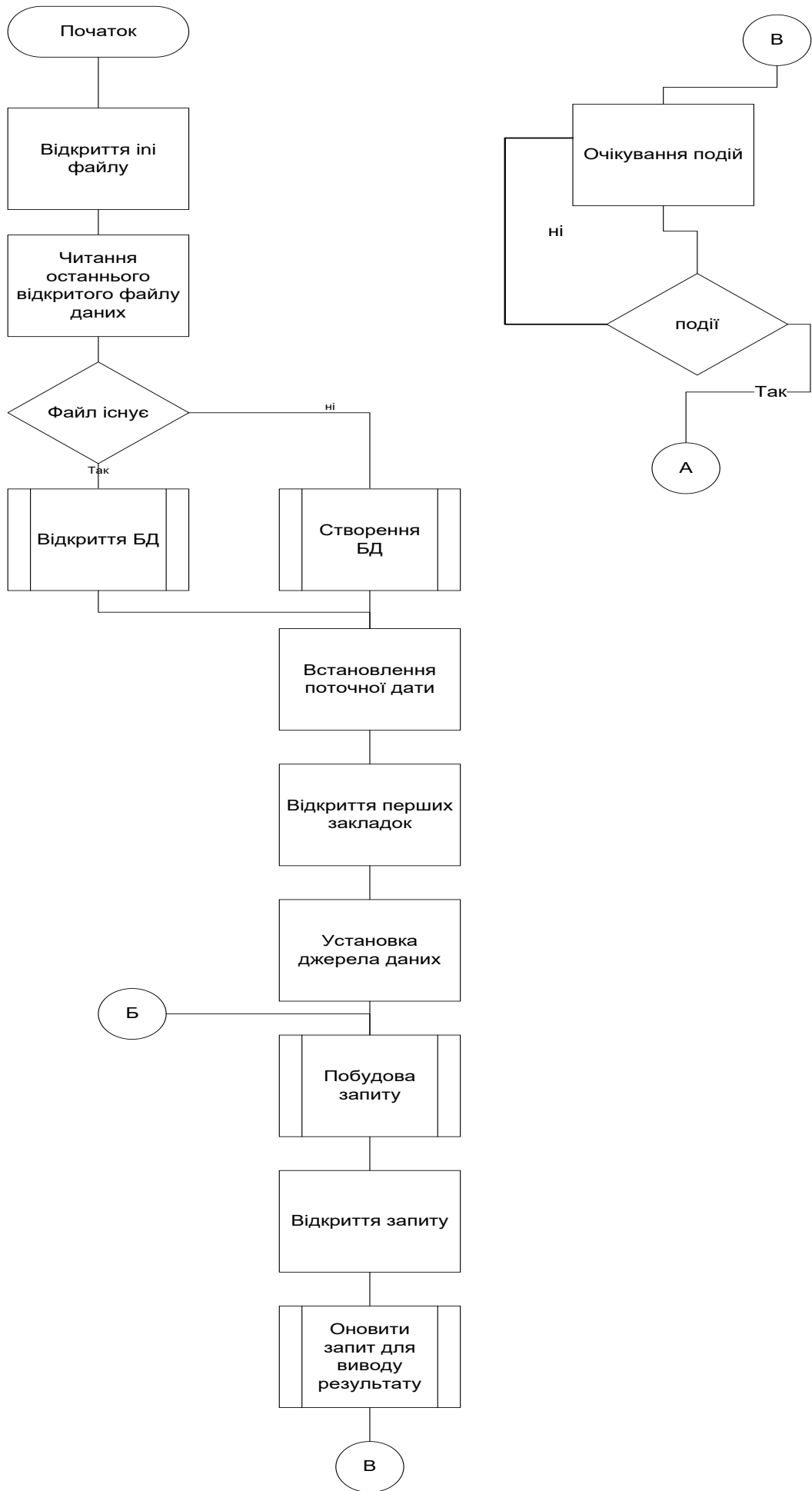
Процедура створення нового файлу БД полягає у запиті користувача нового імені файлу. Якщо користувач відмовився від вибору відбувається вихід з процедури.

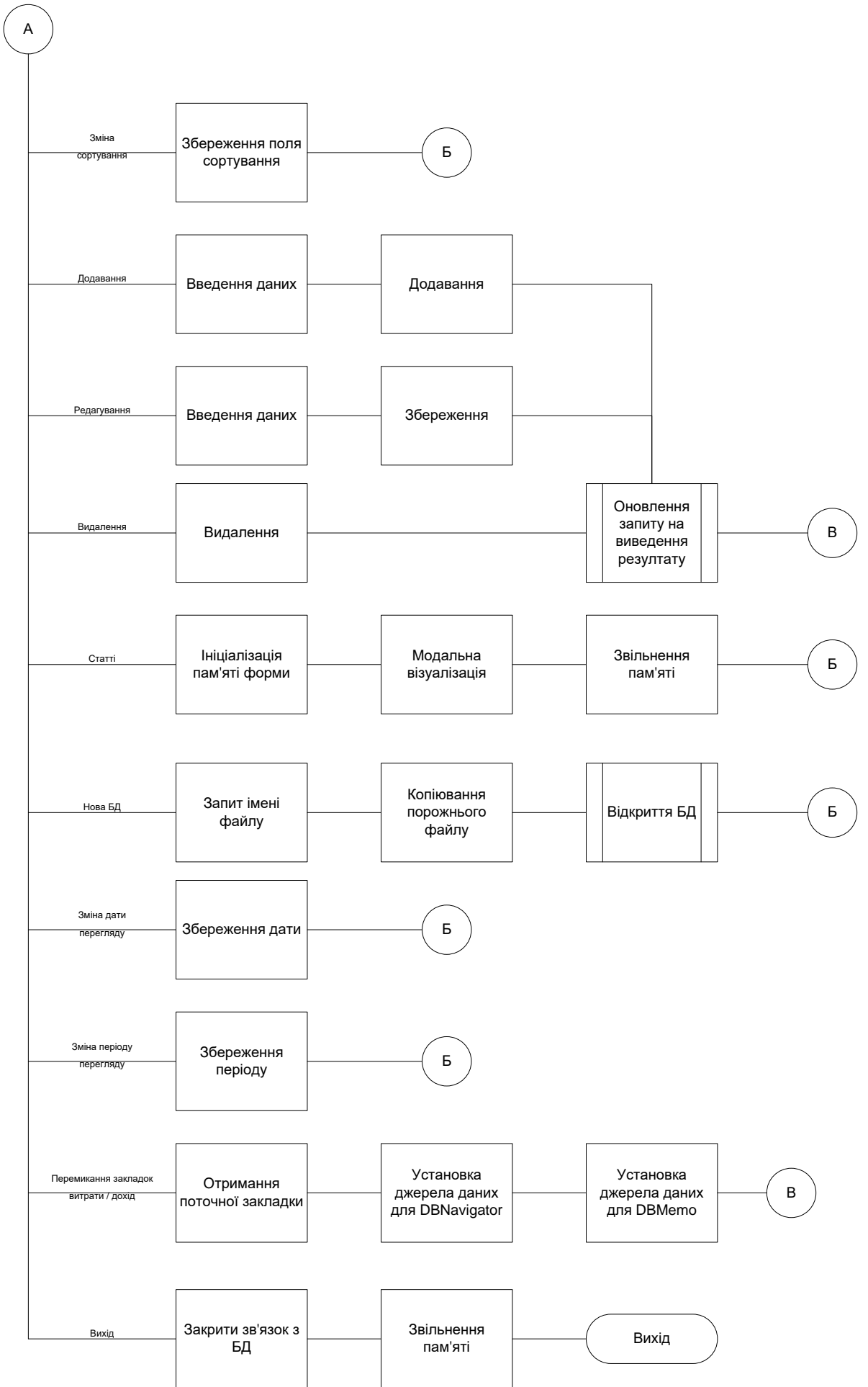
Якщо введене ім'я вже існуючого файлу, ім'я файлу, то відбувається запит на підтвердження перезапису файлу. Якщо підтвердження не отримано, то виходимо з процедури.

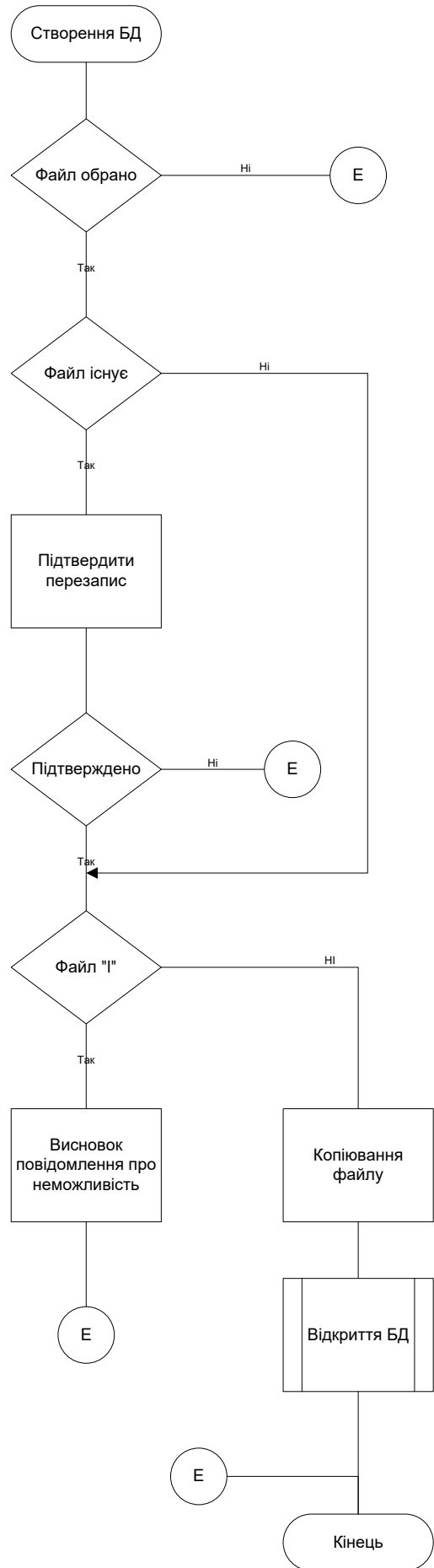
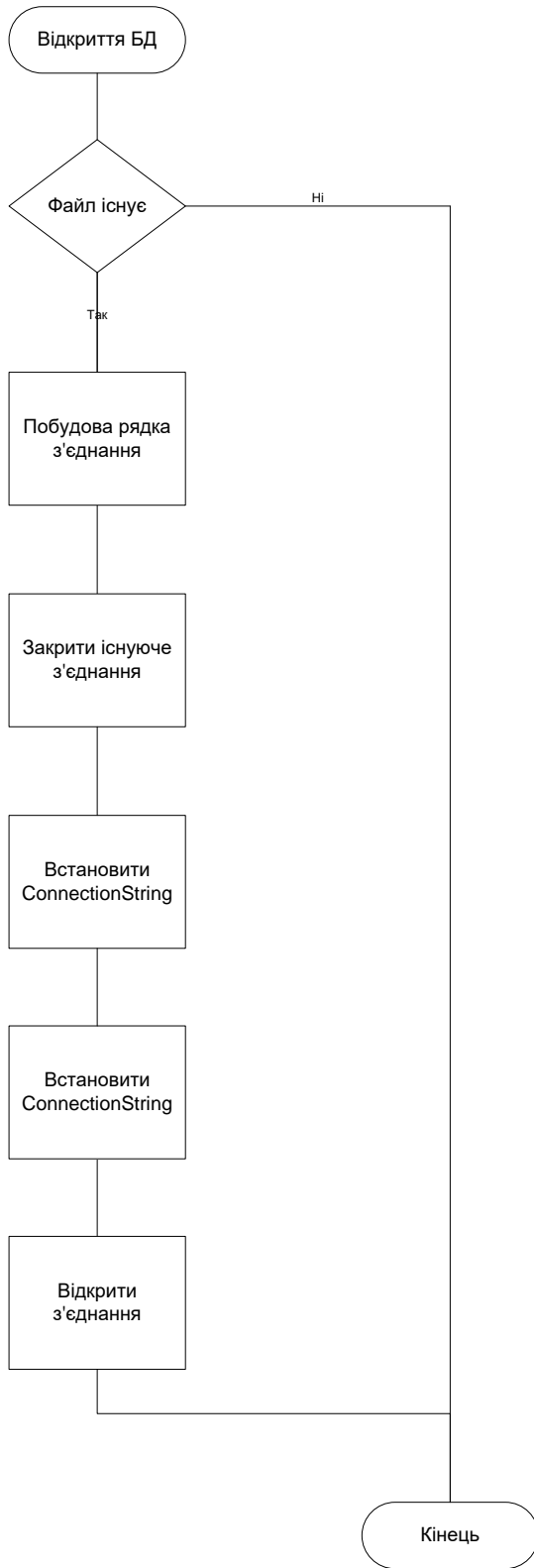
Далі введене ім'я файлу порівнюється з "l.mdb". Даний файл не може бути зіпсований.

Якщо користувач вибрав його, то видається відповідне повідомлення про помилку і процедура завершує свою роботу.

Якщо все нормально, то відбувається копіювання файлу "l.mdb" під введеним ім'ям і запускається процедура відкриття БД. Розглянемо блок-схему алгоритму (Рис. 2.1.):







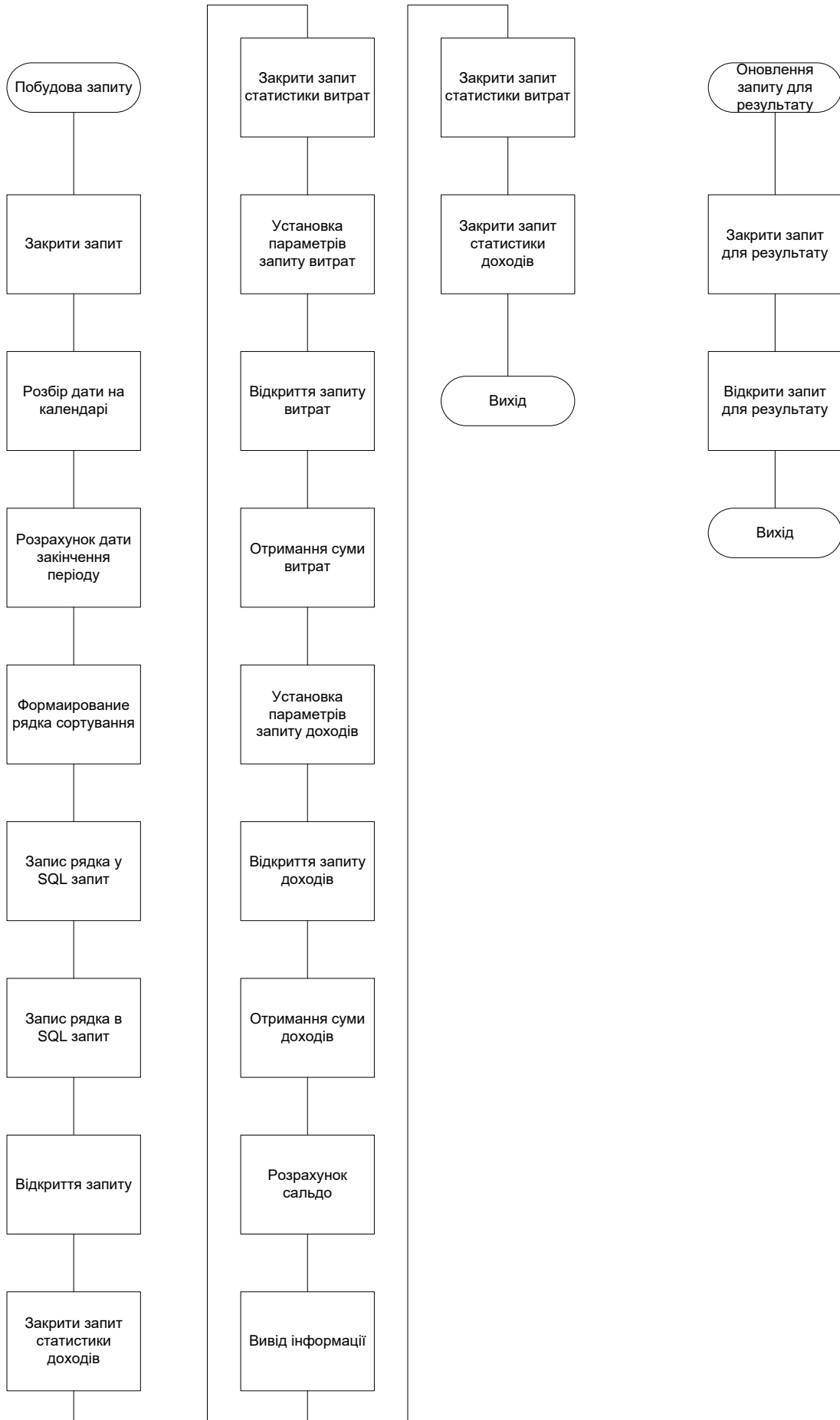


Рисунок 2.1 – Блок схема алгоритму

Модуль повідомлень. Містить у собі опис трьох основних експортованих процедур

```
procedure Notify(Msg: string);
```

Показує повідомлення з кнопкою 'Ок'

```
function Confirm(Msg: string): Boolean;
```

Функція виводить діалог підтвердження з рядком переданої параметрами і кнопками "Так", "Ні". Повертає істину, якщо користувач вибрав "Так"

```
procedure Error(Msg: string);
```

Процедура видає повідомлення про помилку.

Модуль класифікатора типів статей. Форма містить візуальні компоненти для відображення таблиці "I", а також компонент tDataSource для зв'язування їх з джерелом даних.

Тут на формі знаходяться набір компонент, які забезпечують роботу програми. Візуальні компоненти забезпечують візуалізацію даних і забезпечують користувацьке введення даних. Не візуальні компоненти поділяються по призначенню:

- компоненти доступу до даних (З'єднання з БД, таблиці, запити)
- компоненти сховища даних (зберігання зображень і подій)
- Діалоги (діалог відкриття файлу, діалог збереження)

Код програми даного модуля містить обробники подій:

- вибір пунктів меню
- зміна порядку сортування
- зміна глибини запитів
- зміна поточної дати в календарі
- перемикання закладок
- зміна даних у таблицях

На рисунку представлена структурна схема програми (Рис 2.2.).

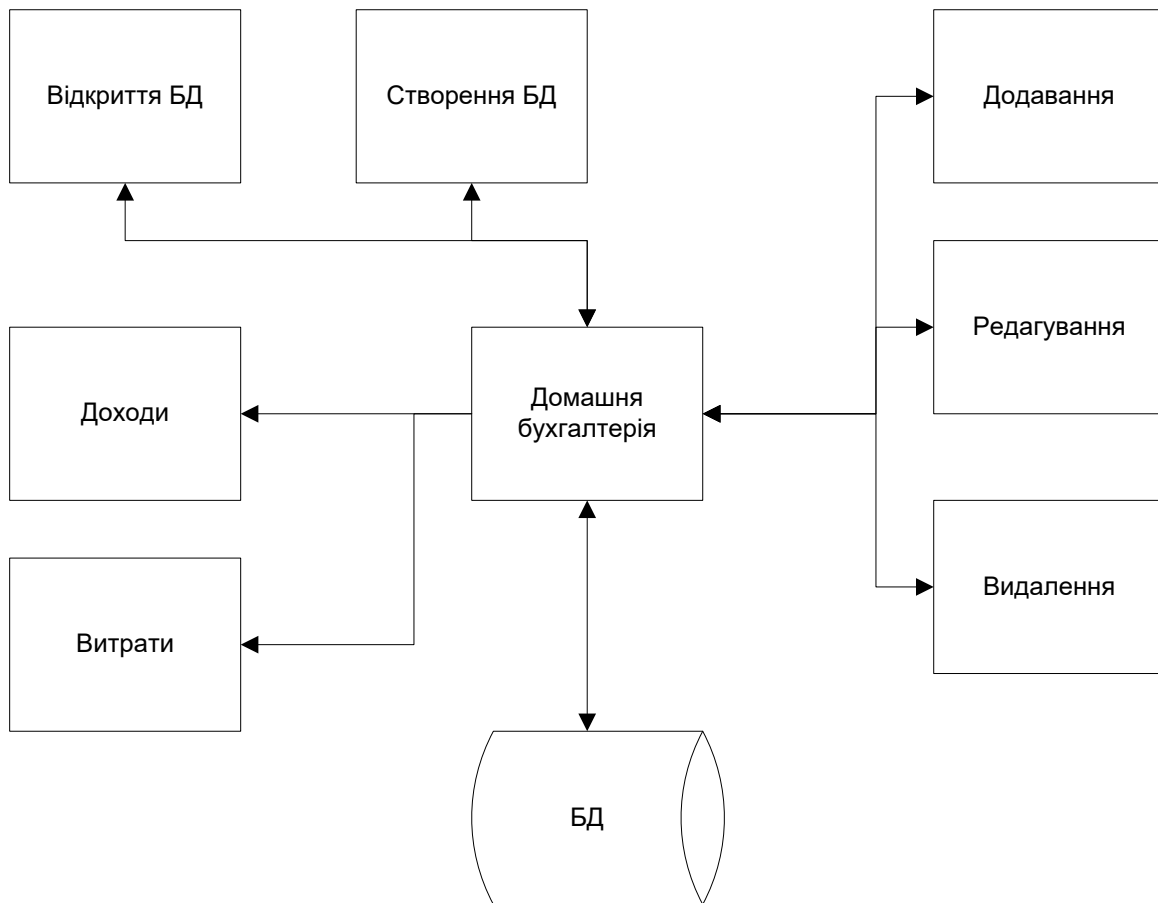


Рисунок 2.2 – Структурна схема програми

Програма має дуже простий і зрозумілий інтерфейс, тому користувачі без зусиль можуть виконувати розрахунки. Програма розроблена українською мовою, і немає складних бухгалтерських операцій за статтями витрат і доходів.

2.3 Проектне рішення

Математичне моделювання можна розглядати як засіб вивчення реальної системи шляхом її заміни зручнішою для експериментального дослідження системою (моделлю), що зберігає істотні риси оригінала. При моделюванні здійснюється апроксимація функції опису більш простою і зручною для практичного аналізу функцією – моделлю.

Математичні моделі, особливо ті, що використовують чисельні методи, потребують для свого створення значних інтелектуальних, фінансових та часових затрат. Тому рішення про створення нової моделі приймається лише в разі відсутності більш простих шляхів вирішення поставленої проблеми (наприклад, модифікації однієї з існуючих моделей).

Дослідження об'єкту моделювання і складання його математичного опису полягають у встановленні зв'язків між характеристиками процесу, виявлених його граничних і початкових умов та формалізації процесу у вигляді системи математичних співвідношень.

Процес побудови математичної моделі представимо послідовністю етапів, зображених на рис. 2.3.

На етапі дослідження об'єкта моделювання потрібно виконати наступні дії:

- аналіз взаємодії об'єкта з зовнішнім середовищем, виділення характеристик вхідних впливів та реакції об'єкту, класифікація їх на вимірні та невимірні, керуючі та перешкоди;
- проведення декомпозиції та дослідження внутрішньої структури об'єкту;
- дослідження порядку функціонування об'єкту, виявлення зв'язку між входом та виходом, формування множини станів об'єкту;
- збір та перевірка існуючих експериментальних даних про об'єкти-аналоги, проведення, при необхідності, додаткових експериментів;
- класифікація об'єкта моделювання на стаціонарний чи нестаціонарний, визначення міри впливу випадкових факторів на об'єкт та порядку нелінійності зв'язків між характеристиками об'єкту;
- аналітичний огляд літературних джерел, аналіз та порівняння побудованих раніше моделей подібних об'єктів;
- аналіз та узагальнення всього накопиченого матеріалу, розроблення загального плану створення математичної моделі.

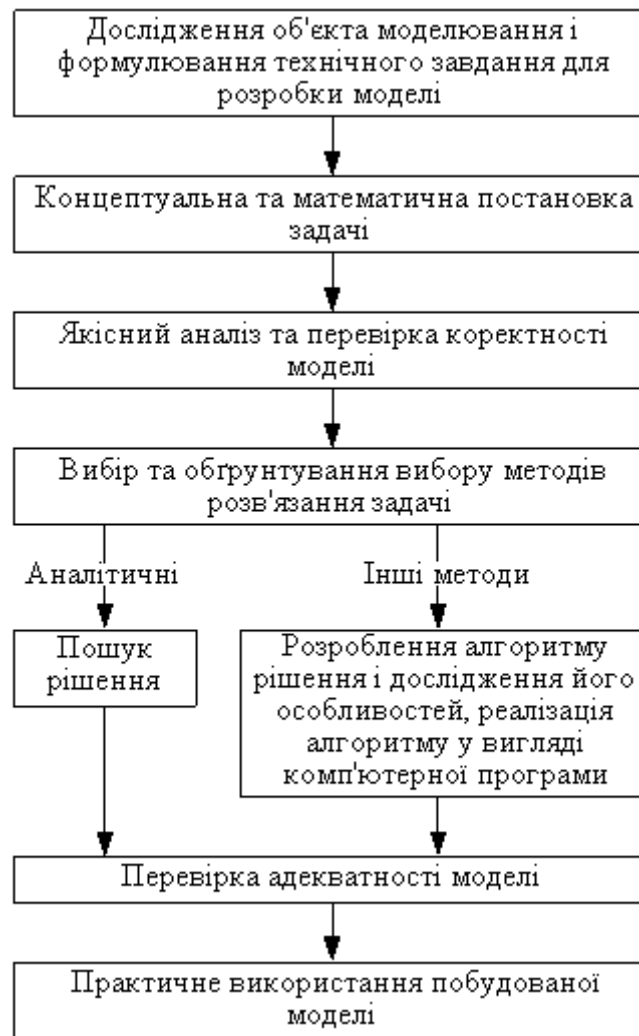


Рисунок 2.3 – Послідовність етапів побудови математичної моделі [7]

В деяких випадках дослідження внутрішньої будови та порядку функціонування об'єкта моделювання неможливе або економічно недоцільне. Тому можливо розглядати об'єкт як „чорний ящик”, стосовно якого нам відомі лише його входи та виходи.

На підставі аналізу об'єкту моделювання формується змістовна постановка моделювання, в якій мають бути зазначені:

- мета моделювання;
- тип моделі;
- вимоги до адекватності моделі та якості розв'язку.

Тип моделі має відповідати результатам класифікації об'єкта моделювання, інакше модель навряд чи буде адекватною.

Весь накопичений в результаті дослідження матеріал, змістовна постановка задачі моделювання, додаткові вимоги до реалізації моделі, оформлюються у вигляді технічного завдання на проектування та розробку моделі.

Концептуальна постановка задачі моделювання – це сформульований в термінах конкретних дисциплін (фізики, хімії, біології тощо) список основних питань, а також сукупність гіпотез відносно особливостей та поведінки об'єкта моделювання. Розробник моделі на підставі результатів аналізу об'єкта моделювання формує своє бачення стосовно процесів на об'єкті і формулює його на природній мові в термінах предметної області. При цьому з метою спрощення моделі він приймає низку припущень та обмежень. Припущення можуть містити нехтування певними процесами або зміну характеру їх протікання. Концептуальна модель має пройти погодження з експертами по даній предметній області з метою перевірки на адекватність. Адекватність концептуальної моделі визначає адекватність математичної моделі, яка формується на її основі [11].

Математична постановка задачі моделювання – це сукупність математичних співвідношень, які описують поведінку та характеристики об'єкта моделювання. Необхідно формалізувати змінні моделі та зв'язки між ними. Математичний опис моделі складається на основі законів фізики, хімії тощо, які характеризують динаміку і статику процесів в досліджуваному об'єкті, і виражається на мові будь-яких розділів математики. Найбільше поширення при побудові детермінованих моделей набули алгебраїчні рівняння та системи, звичайні диференціальні рівняння і диференціальні рівняння в частинних похідних, матрична алгебра, а при стохастичному моделюванні і методи теорії імовірності, математичної статистики та теорії випадкових процесів. Якщо апріорні відомості про об'єкт недостатні, вигляд математичних моделей уточнюється за допомогою методів багатовимірної статистики: регресійного, кореляційного, багатофакторного і інших аналізів, а також проведення пасивного або планування активного експериментів. Для більшості складних об'єктів структура моделі містить параметри, які відображають особливості конкретних об'єктів. Пошук значень

цих параметрів відбувається за допомогою методів параметричної ідентифікації на основі проведення пасивного або активного експериментів.

Під доходами населення розуміється сума грошових коштів і матеріальних благ, отриманих або вироблених домашніми господарствами за певний проміжок часу. Роль доходів визначається тим, що рівень споживання населення прямо залежить від рівня доходів.

Грошові доходи населення включають усі надходження грошей у вигляді оплати праці працюючих осіб, доходів від підприємницької діяльності, пенсій, стипендій, різних допомог, доходів від власності у вигляді відсотків, дивідендів, ренти, сум від продажу цінних паперів, нерухомості, продукції сільського господарства, різних виробів, доходів від наданих на сторону різних послуг та ін.

Рівень доходів членів суспільства є найважливішим показником їхнього добробуту, тому що визначає можливості матеріального і духовного життя індивідуума: відпочинку, отримання освіти, підтримки здоров'я, задоволення насущних потреб. Серед факторів, що роблять безпосередній вплив на величину доходів населення, крім розмірів самої заробітної плати, виступає динаміка роздрібних цін, ступінь насиченості споживчого ринку товарами та ін.

Для оцінки рівня і динаміки доходів населення використовуються показники номінального, наявного і реального доходу.

Номінальний дохід - кількість грошей, отримане окремими особами протягом певного періоду також він характеризує рівень грошових доходів незалежно від оподаткування.

Наявний дохід - дохід, який може бути використаний на особисте споживання і особисті заощадження. Наявний дохід менше номінального доходу на суму податків та обов'язкових платежів, тобто це кошти, які використовуються на споживання та заощадження. Для вимірювання динаміки наявних доходів застосовується показник «реальні наявні доходи», що розраховується з урахуванням індексу цін.

Реальний дохід - являє собою кількість товарів і послуг, яку можна купити на дохід протягом певного періоду, тобто з поправкою на зміну рівня цін.

Прагнення до максимізації свого доходу диктує економічну логіку поведінки будь-якого ринкового суб'єкту. Прибуток є кінцевою метою дій кожного активного учасника ринкової економіки, об'єктивним і потужним стимулом його повсякденній діяльності.

Всього можуть існувати три можливих варіанти використання інформації: інформація, що вводиться, виводиться або обчислюється. Припустимо, заробітна плата, інші надходження і діючий курс долара (неважливо який) зазвичай однозначно вводяться користувачем, у той час як величина доходів у валютному виразі однозначно обчислюється самою системою виходячи із суми одержаних гривень і курсу на момент їх отримання. Проставляючи у кожному рядку спосіб використання цієї інформації, слід передбачати всі можливі варіанти. Наприклад, однозначно видно, що ціну на молоко можна не тільки вводити в якості вхідного параметра, але і час від часу виводити на екран для перегляду (наприклад, при демонстрації загальної статистики за той або інший період).

У СУБД всі дані зберігаються в таблицях. Цих таблиць зазвичай декілька, що дозволяє уникнути зайвої витрати системних ресурсів на багаторазове заповнення ключових полів уже наявною інформацією. У створюваній СУБД облік проводиться в гривнях, причому програма не буде самостійно визначати поточну вартість товарів і послуг в USD виходячи з введеного курсу гривні до долара. Однак таблиця є таблиця, при її веденні неминуче доведеться або проставляти курс кожен раз, як тільки будуть вводитися які б то не було суми, або відразу їх перераховувати валюту і знову ж вводити результат самостійно у відповідну графу. І те й інше однаково незручно. По-перше, будь-яка з перерахованих величин може легко бути отримана шляхом простої математичної операції ділення ціни на курс. По-друге, одночасно і гривневі і валютні величини навряд чи коли-небудь будуть потрібні одночасно, та ще й щодня. По-третє, раз дані можуть бути отримані розрахунковим шляхом, то їх дублювання оператором є не що інше, як порожня витрата робочого часу. Звідси вимальовується наступне очевидне рішення: дані розділити на дві таблиці - в одній містити найменування товарів і гривневі величини платежів, а в іншій - поточний курс і дату його

введення. Крім таблиць, знадобиться ще запит, який буде брати гривні, ділити на курс і показувати, скільки це у валюті. В результаті можемо досягти, як мінімум, двох важливих ефектів: СУБД не буде зберігати надлишкової інформації і в той же час вона стане виробляти перерахунок тільки для тих даних, які запросив користувач в даний конкретний момент, чим значно заощадить обчислювальну потужність персонального комп'ютера.

Зазвичай до єдиного об'єкту прагнуть відносити однорідні речі. Наприклад, цілком очевидно, що всі надходження, звідки б вони не виходили, належать до однієї групи, так як для даної СУБД це однозначно вхідна інформація.

Цілком детально конкретизується той тип даних, який слід використовувати в тому чи іншому випадку, - якщо це найменування, значить передбачається текстовий тип, якщо це дата, значить знадобиться тип дата/час і так далі. Причому тут же рекомендується прикинути і бажану величину даних. Якщо дати це не має особливого значення, то в разі тексту або чисел від початково заданого розміру в чому залежить і точність отриманого результату і загальний обсяг всієї СУБД в цілому.

За для вирішення задачі визначення ресурсів об'єкту необхідно створити СУБД, яка буде включати основні набори даних таких як сума доходів сім'ї та сума витрат. Також одним з головних завдань є складання реєстру витрат та доходів у діапазоні дат [13].

РОЗДІЛ 3

ОПИС РОБОТИ ПРОГРАМИ ТА ЇЇ КОМПОНЕНТІВ

3.1 Опис роботи програми та її компонентів

При розробці програмного додатку «Personal accounting для моніторингу фінансових витрат на мові C» багато уваги приділялось розробці інтерфейсу користувача, його розроблено максимально зручним та зрозумілим для користувача (рис. 3.1).

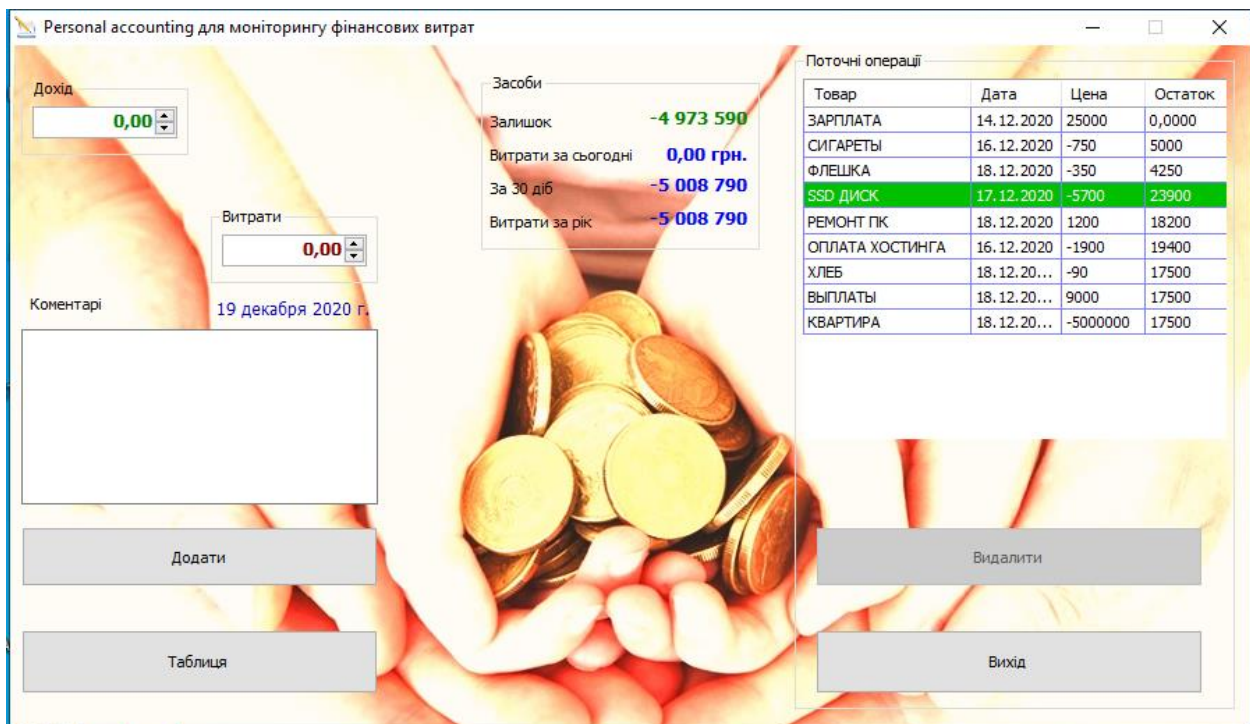
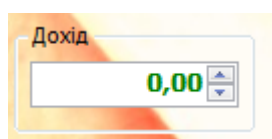
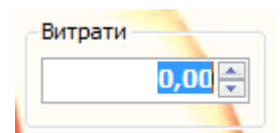


Рисунок 3.1 – Інтерфейс користувача

Головне вікно програми складається з двох форм витрати



та дохід . У даних формах користувач має можливість вводити суму витрат та доходу відповідно.

Форма коментарі , дозволяє описати сутність надходжень та витрат.

Активна кнопка додає до бази даних суми та найменування витрат да доходів.

Товар	Дата	Цена	Остаток

У вкладенні виводиться таблиці з бази даних з описом усіх надходжень та витрат у сімейному бюджеті.

Активна кнопка здійснює вихід з додатку и закриття всіх з'єднань з базою даних.

Кнопка дозволяє перейти на другу сторінку додатку «Personal accounting для моніторингу фінансових витрат на мові С» та переглянути надходження та витрати за періодом дат як разом так і окремо.

3.2 Інструкції адміністратору та користувачам

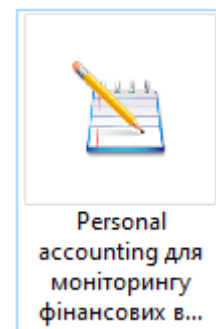
3.2.1 Загальні відомості про програму

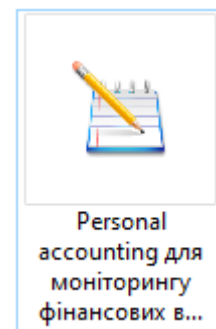
Програма «Personal accounting для моніторингу фінансових витрат на мові С» розроблена так, що будь-який користувач зміг розібратися в роботі з додатком. У даній програмі можна зберігати такі дані: назва покупки, вартість, дата покупки, дату отримання доходів і в якій кількості. Додаток може використовуватися для контролю фінансів сім'ї, і планування покупок.

3.2.2 Інструкція по роботі

Дана програма не вимагає установки на жорсткий диск комп'ютера. Для того, щоб її відкрити, досить запустити файл програми. Додаток має два робочих вікна (рис. 3.2)

Взаємодія користувача з програмним середовищем здійснюється за допомогою 4 кнопок для додавання доходів, витрат, перегляд таблиці, перегляд за датою (діапазон дат), вихід.



Запуск програми здійснюється натисканням на знак . На екрані з'являється головне вікно програми (рис. 3.3), і користувач може вносити свої витрати та доходи.

Для того щоб внести суму доходів варто внести суму у відповідне вікно и натиснути на кнопці додати, внесена сума відобразиться у таблиці праворуч (рис. 3.4).

Для того щоб внести суму витрат варто внести суму у відповідне вікно и натиснути на кнопці додати, внесена сума відобразиться у таблиці праворуч (рис. 3.5).

Personal accounting для моніторингу фінансових витрат

Дохід: **0,00**

Витрати: **0,00**

Коментарі: 19 декабря 2020 г.

Засоби:

- Залишок: **-4 973 590**
- Витрати за сьогодні: **0,00 грн.**
- За 30 дб: **-5 008 790**
- Витрати за рік: **-5 008 790**

Поточні операції

Товар	Дата	Цена	Остаток
ЗАРПЛАТА	14.12.2020	25000	0,0000
СИГАРЕТЫ	16.12.2020	-750	5000
ФЛЕШКА	18.12.2020	-350	4250
SSD ДИСК	17.12.2020	-5700	23900
РЕМОНТ ПК	18.12.2020	1200	18200
ОПЛАТА ХОСТИНГА	16.12.2020	-1900	19400
ХЛЕБ	18.12.20...	-90	17500
ВЫПЛАТЫ	18.12.20...	9000	17500
КВАРТИРА	18.12.20...	-5000000	17500

Додати

Видалити

Таблиця

Вихід

Таблиця

Товар	Дата	Цена	Остаток
ЗАРПЛАТА	14.12.2020	25000	0,0000
СИГАРЕТЫ	16.12.2020	-750	5000
ФЛЕШКА	18.12.2020	-350	4250
SSD ДИСК	17.12.2020	-5700	23900
РЕМОНТ ПК	18.12.2020	1200	18200
ОПЛАТА ХОСТИНГА	16.12.2020	-1900	19400
ХЛЕБ	18.12.20...	-90	17500
ВЫПЛАТЫ	18.12.20...	9000	17500
КВАРТИРА	18.12.20...	-5000000	17500

Всього: 0 грн.

За період

Вибірка у діапазоні дат

19.12.2020 - 19.12.2020

Тип операції

Доходи Витрати

Містить текст (або залиште поле порожнім)

Показати

Вихід

Рисунок 3.2 – Робочі вікна програми

При натисканні на кнопці таблиця, додаток переходить на другу сторінку де є можливість переглянути витрати та доходи у діапазоні меж (рис. 3.5).

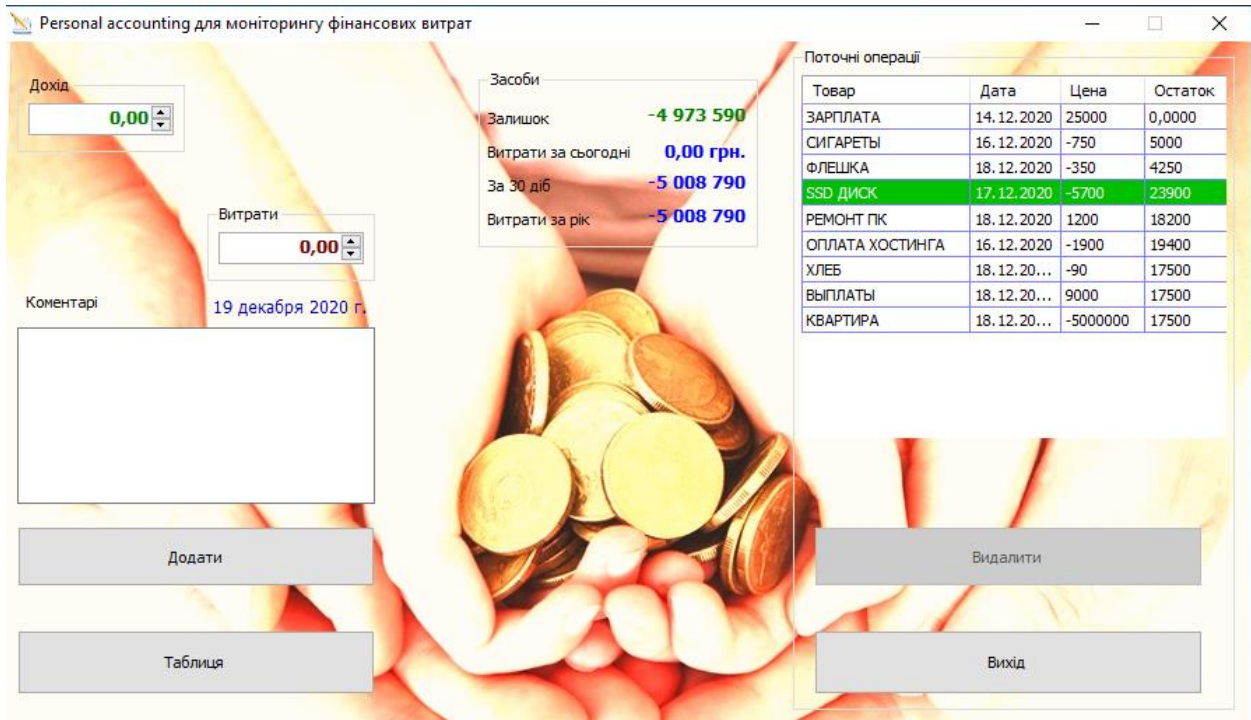


Рисунок 3.3 – Головне вікно програми

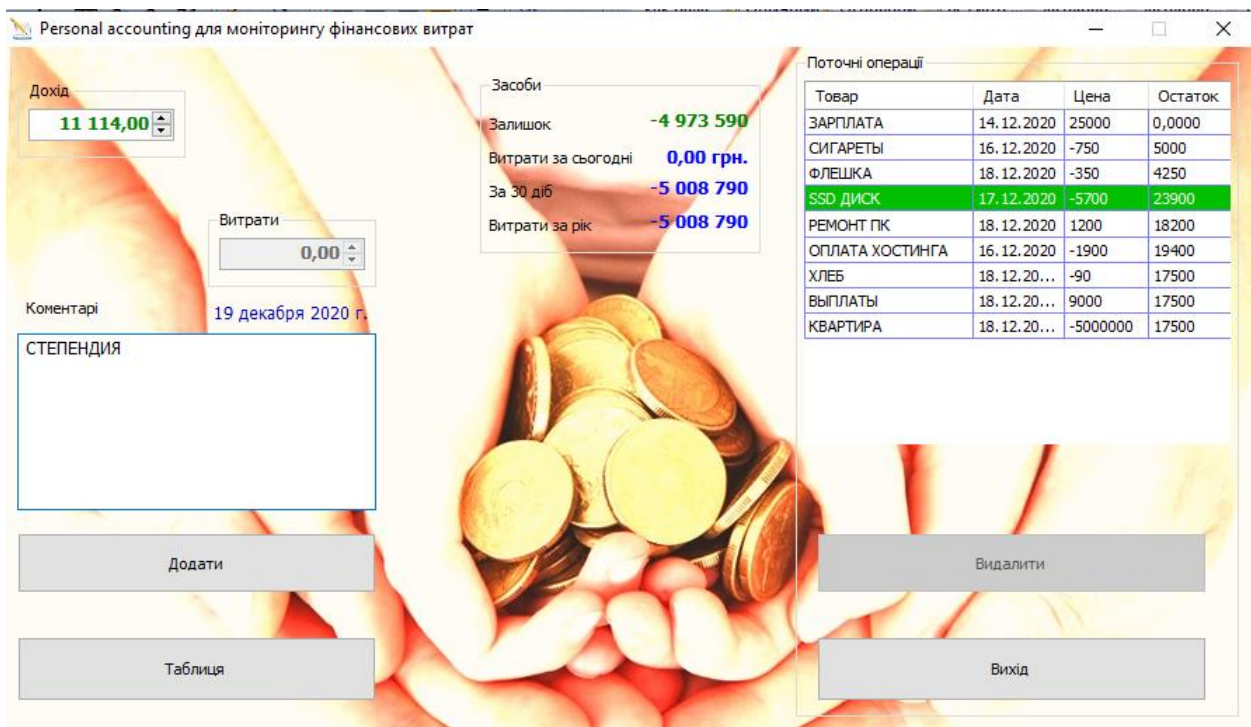


Рисунок 3.4 – Головне вікно програми. Додавання суми доходів

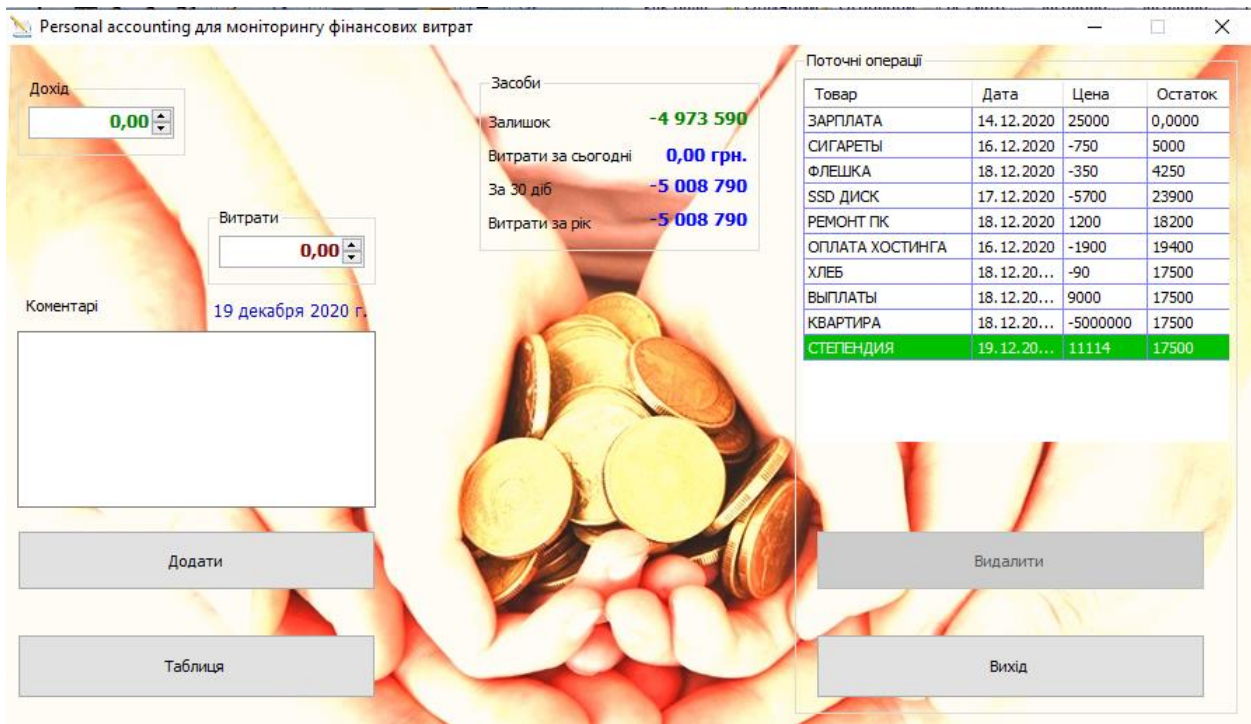


Рисунок 3.5 – Головне вікно програми. Додавання суми витрат

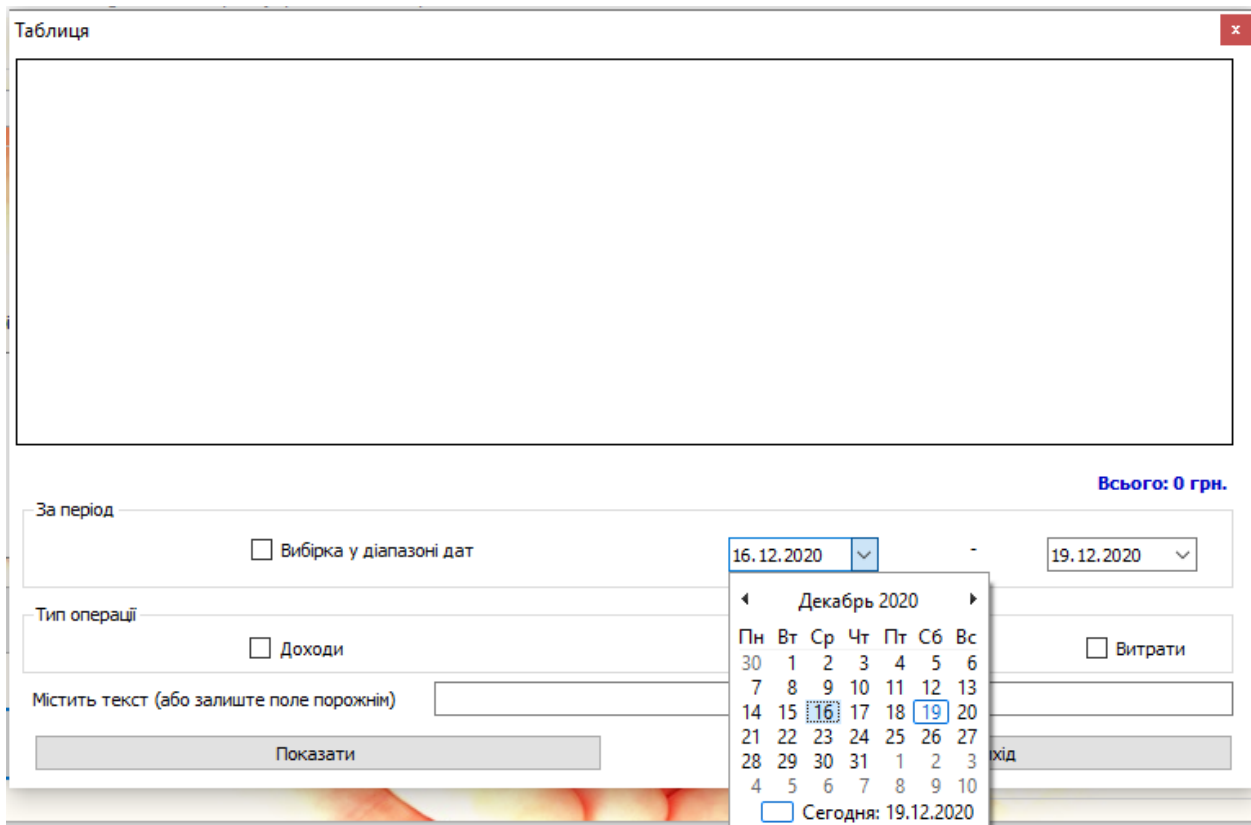


Рисунок 3.6 – Головне вікно програми. Перегляд історії за періодом дат

Таблиця

Товар	Дата	Цена	Остаток
ЗАРПЛАТА	14.12.2020	25000	0,0000
ОПЛАТА ХОСТИНГА	16.12.2020	-1900	19400
СИГАРЕТЫ	16.12.2020	-750	5000
SSD ДИСК	17.12.2020	-5700	23900
РЕМОНТ ПК	18.12.2020	1200	18200
ФЛЕШКА	18.12.2020	-350	4250
ХЛЕБ	18.12.2020 13:36	-90	17500
ВЫПЛАТЫ	18.12.2020 14:14	9000	17500
КВАРТИРА	18.12.2020 14:16	-5000000	17500
СТЕПЕНДИЯ	19.12.2020 7:46	11114	17500

-4 962 476 грн.

За період

Вибірка у діапазоні дат

16.12.2020 - 19.12.2020

Тип операції

Доходи Витрати

Містить текст (або залиште поле порожнім)

Рисунок 3.7 – Головне вікно програми. Перегляд історії за періодом дат

3.3 Тестування розробленого програмного додатку

Тестування програми – це етап, на якому перевіряється, як веде себе програма на як можна більшій кількості вхідних наборів даних, у тому числі і на завідомо невірних.

Основні принципи організації тестування:

– необхідною частиною кожного тесту повинен бути опис очікуваних результатів роботи програми, щоб можна було швидко з'ясувати наявність або відсутність помилки в ній;

– слід по можливості уникати тестування програми її автором, так як крім вже вказаної об'єктивної складності тестування для програмістів тут присутній і той чинник, що виявлення недоліків в своїй діяльності суперечить людській психології (однак відладка програми ефективніше усього виконується саме автором програми);

– за тими же міркуваннями організації – розробник програмного забезпечення не повинен "одноосібно" його тестувати (повинні існувати організації, що спеціалізуються на тестуванні програмних засобів);

– повинне бути правилом доскональне вивчення результатів кожного тесту, щоб не пропустити малопомітну на поверхневий погляд помилку в програмі;

– необхідно ретельно підбирати тест не тільки для правильних (передбачених) вхідних даних, але і для неправильних (непередбачених);

– слід зберігати використані тести (для підвищення ефективності повторного тестування програми після її модифікації або установки у замовника);

– тестування не повинно плануватися виходячи з припущення, що в програмі не будуть виявлені помилки (зокрема, потрібно виділяти для тестування достатні тимчасові і матеріальні ресурси);

– слід враховувати так званій "принцип скупчення помилок": імовірність наявності не виявлених помилок в деякій частині програми прямо пропорційно числу помилок, вже виявлених в цій частині.

Методика проведення та результати тестування

Тестування є важливим етапом у розробці програми. Потрібно домогтися взаємозв'язку таких складових, як надійність, практичність, ефективність, мобільність, функціональність. Тим не менш, при розробці програмного продукту допускалися різні помилки:

– застосування змінних без перетворень типу даних;

– при роботі з індексами рядків були допущені помилки, які приводили до неправильної роботи програми.

– некоректні умови в циклах;

– використовувалися кілька циклів поспіль, замість одного, що вимагало оптимізації програмного коду.

Для знаходження такого роду помилок застосовувалося модульне тестування, особливістю якого є розбиття всього програмного коду на мінімально можливе для тестування компонент, наприклад, окремий клас або метод. Мета

модульного тестування – ізолювати окремі частини програми і показати, що окремо ці частини працездатні. Таке тестування полегшує виявлення і усунення помилок.

Методи і способи усунення помилок

Налагодження програми – найбільш трудомісткий етап. Його мета – перевірка синтаксичної та логічної правильності програми, а також визначення того, що програма функціонує на всьому діапазоні допустимих даних.

В процесі відладки програми виділяються наступні етапи.

1) Трансляція. При трансляції виконується переклад програми, зрозумілою людині, на мову, зрозумілу комп'ютеру. Якщо мета трансляції – перетворення всього вихідного тексту на внутрішню мову комп'ютера (тобто отримання деякого нового коду) і тільки, то така трансляція називається компіляцією. Вихідний текст називається також вихідною програмою або вихідним модулем, а результат компіляції – об'єктним кодом або об'єктним модулем. Якщо ж трансляції піддаються окремі оператори вихідних текстів і при цьому отримані коди виконуються, така трансляція називається інтерпретацією. Оскільки трансляція виконується спеціальними програмними засобами – компіляторами або інтерпретаторами відповідно.

2) Тестування. Метою є визначення працездатності програми на всьому необхідному діапазоні вихідних даних. Програмістом складається представницька вибірка вихідної інформації, яка дозволить визначити коректність програми при будь-яких вхідних параметрах.

3) Виконання програми з метою знаходження логічних помилок. Здійснюється після успішного компонування або по ходу інтерпретації кожного оператора. При цьому виявляються такі помилки, як, наприклад, ділення на нуль або обчислення логарифма від'ємного числа. Тут же перевіряється коректність опису використовуваних змінних, а також закінченість циклів.

Налагодження виявлених помилок, виявлених при тестуванні

Під час виконання налагодження, були використані наступні методи:

– unit тестування;

- аналіз коду без виконання програми;
- запуск програми з під відладчика.

Всі синтаксичні помилки були виправлені при компіляції проекту, враховуючи синтаксичні особливості середовища програмування .NET.

Налагодження ряду помилок пройшло успішно, були внесені зміни в інтерфейс програми, а також були передбачені деякі виняткові ситуації. В результаті тестування та налагодження програми були виявлені і виправлені наступні помилки:

- змінні, що вимагають зміни типу даних, були перетворені;
- була проведена оптимізація програмного коду для кожного з методів окремо.

Таблиця

Товар	Дата	Цена	Остаток
ЗАРПЛАТА	14.12.2020	25000	0,0000
РЕМОНТ ПК	18.12.2020	1200	18200
ВЫПЛАТЫ	18.12.2020 14:14	9000	17500
СТЕПЕНДИЯ	19.12.2020 7:46	11114	17500

46 314 грн.

За період Вибірка у діапазоні дат -

Тип операції Доходи Витрати

Містить текст (або залиште поле порожнім)

Рисунок 3.8 – Головне вікно програми. Перегляд історії за періодом дат. Тільки доходи

Таблиця

Товар	Дата	Цена	Остаток
ОПЛАТА ХОСТИНГА	16.12.2020	-1900	19400
СИГАРЕТЫ	16.12.2020	-750	5000
SSD ДИСК	17.12.2020	-5700	23900
ФЛЕШКА	18.12.2020	-350	4250
ХЛЕБ	18.12.2020 13:36	-90	17500
КВАРТИРА	18.12.2020 14:16	-5000000	17500

-5 008 790 грн.

За період

 Вибірка у діапазоні дат

16.12.2020

-

19.12.2020

Тип операції

 Доходи Витрати

Містить текст (або залиште поле порожнім)

Показати

Вихід

Рисунок 3.9 – Головне вікно програми. Перегляд історії за періодом дат.
Тільки витрати

Таким чином, спираючись на результати тестування є можливість стверджувати про належну якість розробки та можливість використання додатку за потребою.

ВИСНОВКИ

У межах даної дипломної роботи здійснено розробку Personal accounting для моніторингу фінансових витрат на мові С.

У роботі варто виконано низку завдань:

- здійснено теоретичний аналіз і обробку практичної і методичної літератури по темі дипломної роботи;
- виконано проектування і реалізацію Personal accounting для моніторингу фінансових витрат на мові С.
- запропоновано реалізацію отриманої БД у програмному середовищі С#.
- наведено обґрунтування економічної ефективності та впровадження Personal accounting для моніторингу фінансових витрат на мові С.

В роботі уперше розроблено Personal accounting для моніторингу фінансових витрат на мові С, на основі модифікованого алгоритму. Запропоновано структуроване подання всіх етапів розробки з описом відповідних функцій.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Антощук С.Г. Прогнозирование количества ошибок на этапе эксплуатации адаптируемых учетных информационных систем. / С.Г. Антощук, Д.А. Маевский, С. А. Яремчук // Радиоэлектронные и компьютерные системы, НТЖ, Харьков: ХАИ. – 2010. – № 6(47). – С. 204-209.
2. Атре Ш. Структурный подход к организации баз данных. – М.: Финансы и статистика, 1983. – 320 с.
3. Банк В.С., Зверев В.С. Информационные технологии в экономике, - 2003
4. Бойко В.В., Савинков В.М. Проектирование баз данных информационных систем. – М.: Финансы и статистика, 1989. – 351 с.
5. В. В. Липаев. Обеспечение качества программных средств. Методы и стандарты. Синтег, Москва, 2001.
6. Голицина О.Л., Максимов Н.В., Попов И.И. Базы данных: Учебное пособие. – М.: ФОРУМ: ИНФРА-М, 2003. – 352 с.
7. Гуда А.Н., Калинин Т.С., Чернов А.В. Реализация надежного программного обеспечения задач технической диагностики информационно-управляющих систем // Известия высших учебных заведений. Северо-Кавказский регион. Технические науки, 2011. – №4. – С.26-31.
8. Джексон Г. Проектирование реляционных баз данных для использования с микроЭВМ. -М.: Мир, 1991. – 252 с.
9. Докучаев А.А., Мошенский С.А., Назаров О.В. Средства информатики в офисе торговой фирмы. Средства компьютерных коммуникаций. – СП б, ТЭИ, 1996. – 32с.
10. ДСТУ 3918–1999 (ISO/IEC 12207:1995). Інформаційні технології. Процеси життєвого циклу програмного забезпечення. – Введ. 01.07.2000. – К.: Держстандарт України, 1995. – 55 с.
11. Канер С., Фолк Д., Нгуен Е.К. Тестирование программного обеспечения: Пер с англ. - К.: DiaSoft. – 2000. – 544 с.

12. Карпова Т.С. Базы данных: модели, разработка, реализация. – СПб.: Питер, 2002. – 304 с.
13. Кент Бек. Экстремальное программирование: разработка через тестирование. – «Питер», 2003.
14. Кириллов В.В. Структуризованный язык запросов (SQL). – СПб.: ИТМО, 1994. – 80 с.
15. Клименко С.В., Уразметов В.Ф. Internet: среда обитания информационного общества. Протвино.: РЦ ФТИ, 1995.
16. Климова Р.Н., Сорокина М.В., Хахаев И.А., Мошенский С.А. Информатика торговой фирмы / Учебное пособие. Для студентов всех специальностей всех форм обучения. – СПб.: СПбТЭИ, 1998. – 32с.
17. Компьютерные технологии обработки информации./Под ред. Назарова С.И. – М.: Финансы и статистика, 1996.
18. Корнеев И.К., Машурцов В.А. Информационные технологии в управлении. – М.: ИНФРА-М, 2001. – 158 с.
19. Лайза Криспин, Джанет Грегори Гибкое тестирование: практическое руководство для тестировщиков ПО и гибких команд = Agile Testing: A Practical Guide for Testers and Agile Teams. – М.: «Вильямс», 2010. – 464 с.
20. Майерс Г. Искусство тестирования программ. – М: Финансы и статистика. – 1982. – 176 с.
21. Нельсона Э. Обеспечение качества программных средств. Методы и стандарты. – М.: СИНТЕГ, 2001. – 380 с.
22. Харченко В.С., Тарасюк О.М., Гордеев А.А. Мамутов С.С. Инструментальные средства оценки качества программного обеспечения // Материалы конф. «Информационные технологии – в науку и образование». – Харьков: Харьк. нац. ун-т радиоэлектроники, 2005. – С. 98-99.
23. Agilian - All-in-one Modeling Tool supporting Agile Modeling. Support latest UML, BPMN, ERD, DFD and Textual Analysis, webpage: <http://www.visual-paradigm.com/product/ag/>

24. B. W. Boehm, J. R. Brown, H. Kaspar, M. Lipow, G. MacLeod, and M. J. Merritt. *Characteristics of Software Quality*. North Holland, 1978.
25. Beck, K. *Test-Driven Development by Example*, Addison Wesley, 2003
26. Beck, K., *Extreme Programming Explained: Embrace Change*, Addison-Wesley, 1999.
27. Boehm B.. *Software Risk Management*. IEEE Computer Society Press, CA, 1989.
28. International Standard ISO/IEC 9126. *Information technology – Software product evaluation – Quality characteristics and guidelines for their use*. International Organization for Standardization, International Electrotechnical Commission, Geneva, 1991.
29. ISO/IEC 9126-1. 2001. *Software engineering – Software product quality – Part 1: Quality model*. Geneva, Switzerland: International Organization for Standardization.
30. ISO/IEC DTR 9126-2. 2001. *Software engineering – Software product quality – Part 2: External metrics*. Geneva, Switzerland: International Organization for Standardization.
31. ISO/IEC DTR 9126-3. 2000. *Software engineering – Software product quality – Part 3: Internal metric*. Geneva, Switzerland: International Organization for Standardization.
32. J. McCall, P. Richards, G. Walters. *Factors in Software Quality*. three volumes, NTIS AD-A049-014, AD-A049-015, AD-A049-055, November 1977.
33. James Martin: *Rapid Application Development*, Macmillan Coll Div.
34. Jones E.L. *Integrating testing into the curriculum – arsenic in small doses.* / Jones E.L. // In Proc. 32nd SIGCSE Technical Symp. Computer Science Education, ACM, 2001, p. 337-341.
35. Juran, Joseph M., Frank M. Gryna, *Juran's Quality Control Handbook*. McGraw-Hill.

36. Kuo S. Y., Hung C. Y. and Lyu M. R., “Framework for modeling software reliability, using various testing efforts and fault detection rates”, IEEE Transactions on Reliability, 2007. – Vol. 54. – №.2. – pp 198-211.
37. Manifesto for Agile Software Development, Agile Alliance, 2001, webpage: [Manifesto-for-Agile-Software-Dev](#).
38. Walter A. Shewhart, Economic Control of Quality of Manufactured Product // 50th Anniversary Commemorative Issue. American Society for Quality December 1980, 501 p.
39. Yamada S., Ohba M., Osaki S.. S-shaped Reliability Growth Modeling for Software Fault Detection. IEEE Transactions on Reliability, 1983. – vol. 32. – no. 5. – pp. 475–484.
40. Yamada S., Osaki S.. Software Reliability Growth Modeling: Models and Applications. IEEE Transactions on Software Engineering, 1985. – vol. 11. – no. 12. – pp. 1431–1437.

ДОДАТОК

РОЗРОБКА МОБІЛЬНОГО ДОДАТКУ "PERSONAL ACCOUNTING" ДЛЯ МОНІТОРИНГУ ФІНАНСОВИХ ВИТРАТ НА МОБІ С#

Бакалаврська робота

Актуальність дослідження

- Актуальність теми підкреслюється тим, що інститут сім'ї сьогодні переживає кризу. Сім'я відчуває на собі вплив економічних, правових, ідеологічних, моральних відносин. Наслідки переходу до ринкової економіки і усунення держави від комплексної підтримки сім'ї не сповільнили позначитися.
- Непрямим підтвердженням тому служить хвиля дитячої безпритульності і бездоглядності. Прямим – низький рівень доходів основної маси українських сімей, житлово-побутова невлаштованість, свідомо орієнтація на малу кількість дітей, заснована на невпевненості у завтрашньому дні, зростання числа цивільних шлюбів і позашлюбних народжень.
- Створення нормальних умов для виробництва людського фактора вимагає не тільки ефективного ведення домашнього господарства, але й формування та використання сімейного бюджету.

Мета, об'єкт дослідження, предмет дослідження

- Метою роботи є покращення моніторингу фінансових витрат за допомогою додатку Personal accounting мовою С#.
- **Об'єкт дослідження** – підвищення ефективності моніторингу фінансових витрат користувача.
- **Предметом** для розгляду є система обробки інформації для моніторингу фінансових витрат на мові С#.

Аналіз існуючих аналогів

The screenshot displays a software interface for personal accounting. The main window shows a table for 'Борженики / Кредитори' (Debtors / Creditors) with columns for date, account, amount, and currency. Below the table are filters for account, period, and status.

Overlaid on the interface is a 'Сімейний бюджет на місяць' (Monthly Family Budget) summary. It shows a total planned expense of 1,283 UAH and a total actual expense of 1,217 UAH, resulting in a surplus of 114 UAH.

Вид витрат	Заплановано	Фактично	Різниця
Всього за період	1 000 грн.	1 000 грн.	0 грн.
Друга частина або	0 грн.	0 грн.	0 грн.
Телефон	62 грн.	100 грн.	-38 грн.
Електроенергія	44 грн.	125 грн.	-81 грн.
Газ	22 грн.	35 грн.	-13 грн.
Вода та комунальні послуги	0 грн.	0 грн.	0 грн.
Кабельне	34 грн.	29 грн.	5 грн.
Використання смт	10 грн.	10 грн.	0 грн.
Обслуговування та	23 грн.	0 грн.	23 грн.
Навесина	0 грн.	0 грн.	0 грн.
Інше	0 грн.	0 грн.	0 грн.
Всього	1 066 грн.	1 217 грн.	-114 грн.

Additional summary data from the interface:

Запланований дохід на місяць	Різниця
Income 1	4 000 грн.
Income 2	1 200 грн.
Extra income	300 грн.
Total monthly income	5 500 грн.

Запланований баланс	Різниця
Income 1	4 000 грн.
Income 2	1 200 грн.
Додатковий дохід	300 грн.
Загальний дохід за	5 500 грн.

Запланований баланс	Різниця
Income 1	4 000 грн.
Income 2	1 200 грн.
Додатковий дохід	300 грн.
Загальний дохід за	5 500 грн.

Запланований баланс	Різниця
Income 1	4 000 грн.
Income 2	1 200 грн.
Додатковий дохід	300 грн.
Загальний дохід за	5 500 грн.

Запланований баланс	Різниця
Income 1	4 000 грн.
Income 2	1 200 грн.
Додатковий дохід	300 грн.
Загальний дохід за	5 500 грн.

Постановка завдань дослідження

- Для досягнення поставленої мети у роботі варто виконати низку завдань:
- здійснити теоретичний аналіз і обробку практичної і методичної літератури по темі дипломної роботи;
- виконати проектування і реалізацію Personal accounting для моніторингу фінансових витрат на мові C#.
- запропонувати реалізацію отриманої БД у програмному середовищі C#.
- навести обґрунтування економічної ефективності та впровадження Personal accounting для моніторингу фінансових витрат на мові C#.

•

•

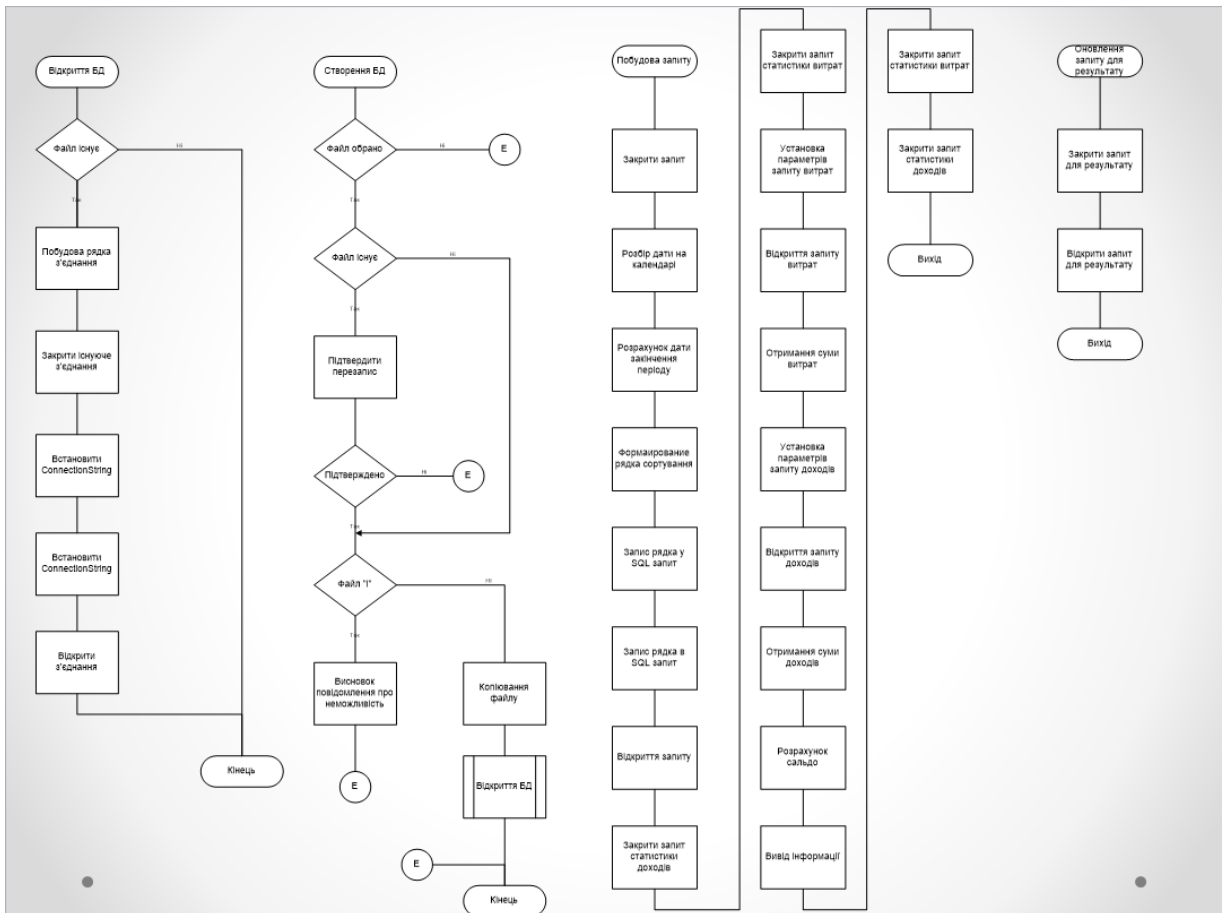
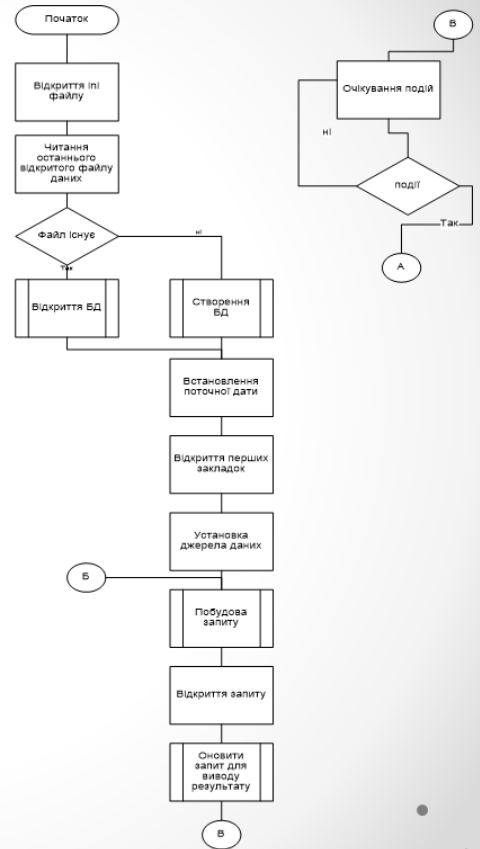
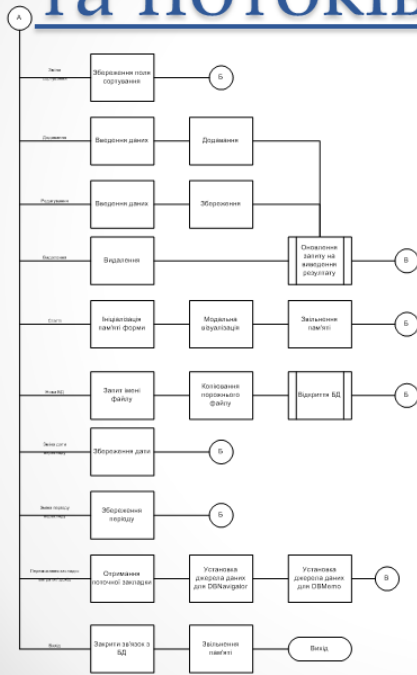
Проектні і технічні рішення щодо розробки

- Технічні рішення
- - 32- розрядний (x86) або 64 - розрядний (x64) процесор з тактовою частотою не менш 1 гігагерц (ГГц);
- - Оперативна пам'ять повинна бути обсягом не менше 1 ГБ;
- - Жорсткий диск об'ємом не менше 20 ГБ;
- - Матричний або лазерний принтер;
- - Пристрої периферії (миша, клавіатура);
- - Монітор;
- У якості інструментального засобу для створення Personal accounting для моніторингу фінансових витрат на мові C#, було обрано мову програмування C#.

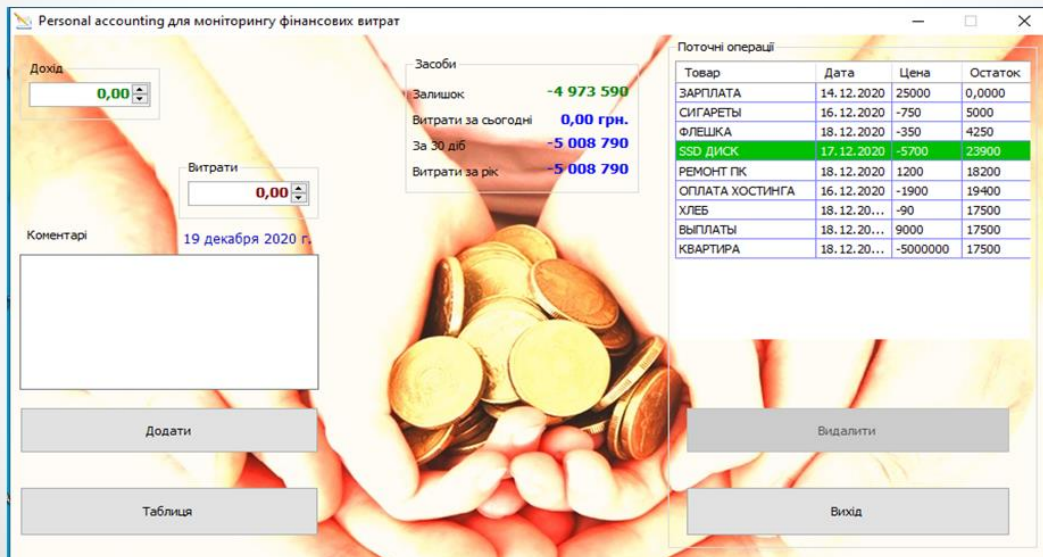
•

•

Схема даних та потоків



Опис роботи програми та її компонентів



Інструкції по роботі

- Програма «Personal accounting для моніторингу фінансових витрат на мові С» розроблена так, що будь-який користувач зміг розібратися в роботі з додатком. У даній програмі можна зберігати такі дані: назва покупки, вартість, дата покупки, дату отримання доходів і в якій кількості. Додаток може використовуватися для контролю фінансів сім'ї, і планування покупок.
- Дана програма не вимагає установки на жорсткий диск комп'ютера. Для того, щоб її відкрити, досить запустити файл програми. Додаток має два робочих вікна.
- Взаємодія користувача з програмним середовищем здійснюється за допомогою 4 кнопок для додавання доходів, витрат, перегляд таблиці, перегляд за датою (діапазон дат), вихід.
- Запуск програми здійснюється натисканням на знак . На екрані з'являється головне вікно програми і користувач може вносити свої витрати та доходи.
- Для того щоб внести суму доходів варто внести суму у відповідне вікно і натиснути на кнопку додати, внесена сума відобразиться у таблиці праворуч
- Для того щоб внести суму витрат варто внести суму у відповідне вікно і натиснути на кнопку додати, внесена сума відобразиться у таблиці праворуч

Відомості про програму

- Пріоритетним завданням є створення бази даних для маленької власної бухгалтерії, щоб швидко і точно враховувати всі доходи і витрати.
- Отже, до переліку обов'язкових повинні увійти такі вимоги:
- облік всіх надходжень за сумами;
- облік джерел надходжень;
- облік поточного курсу, наприклад до долара, так як високі темпи інфляції в країні не дозволяють коректно порівнювати навіть ціни на масло тримісячної давності;
- облік найменувань статей витрат по групах товарів;
- облік витрат по найменуванню товарів;
- облік залишків коштів;
- облік різних варіантів вкладень коштів у різні фінансові інструменти, наприклад в валюту, на депозити і так далі.

Тестування додатку

The screenshot displays a software interface for personal accounting. It features several windows and panels:

- Top Left:** A window titled "Personal accounting for monitoring financial expenses" showing a balance of 11 114.00 and a date of 19 December 2020.
- Top Center:** A "Поточні операції" (Current Operations) table with columns: Товар, дата, Цена, Остаток. It lists transactions such as ЗАРПЛАТА (25000), СИГАРЕТЫ (5000), ФЛЕШКА (4250), and others.
- Top Right:** An empty "Таблица" (Table) window.
- Bottom Center:** A larger "Таблица" window showing a detailed list of transactions with columns: Товар, Дата, Цена, Остаток. The total balance is -4 962 476 грн.
- Bottom Right:** A calendar for December 2020 and a "Витрати" (Expenses) filter.

The main table in the bottom center window contains the following data:

Товар	Дата	Цена	Остаток
ЗАРПЛАТА	14.12.2020	25000	0,0000
ОПЛАТА ХОСТИНГА	16.12.2020	-1900	19400
СИГАРЕТЫ	16.12.2020	-750	5000
SSD ДИСК	17.12.2020	-5700	23900
РЕМОНТ ПК	18.12.2020	1200	18200
ФЛЕШКА	18.12.2020	-350	4250
ХЛЕБ	18.12.2020 13:36	-90	17500
ВЫПЛАТЫ	18.12.2020 14:14	9000	17500
КВАРТИРА	18.12.2020 14:16	-5000000	17500
СТЕПЕНДИЯ	19.12.2020 7:46	11114	17500

Висновки

- У межах даної дипломної роботи здійснено розробку Personal accounting для моніторингу фінансових витрат на мові С#.
- У роботі варто виконано низку завдань:
- здійснено теоретичний аналіз і обробку практичної і методичної літератури по темі дипломної роботи;
- виконано проектування і реалізацію Personal accounting для моніторингу фінансових витрат на мові С#.
- запропоновано реалізацію отриманої БД у програмному середовищі С#.
- наведено обґрунтування економічної ефективності та впровадження Personal accounting для моніторингу фінансових витрат на мові С#.
- В роботі уперше розроблено Personal accounting для моніторингу фінансових витрат на мові С#, на основі модифікованого алгоритму. Запропоновано структуроване подання всіх етапів розробки з описом відповідних функцій.