**курсова робота**

по предмету

*„Комп’ютеризоване Програмування Інтерфейсів* *Користувачів”*

на тему

*„АРМ – Колекціонера марок”*

**Зміст:**

1. Вступ
2. Загальна характеристика предметної області та мети створення додатку
3. Характеристика задачі що розв’язується
4. Опис процесу взаємодії користувача з додатком
5. Обґрунтування проектних рішень по інтерфейсу користувача при:
   * Проектуванні форм
   * Проектуванні діалогів
   * Проектуванні довідкової системи
6. Використання принципу розробки орієнтованих на користувача
7. Використання принципів організації екранів
8. Рішення щодо кольору, при проектування додатку
9. Перелік форм додатка

10. Класи, підкласи та компоненти що використовуються в додатку

Додаток: конверт з дискетою.

* + - 1. **Вступ**

Сучасні методи проектування діяльності користувачів АСУ склалися в рамках системотехнічної концепції проектування, через що облік людського чинника обмежився рішенням проблем узгодження «входів» і «виходів» людини і машини. Разом з тим при аналізі незадоволеності користувачів АСУ вдається виявити, що вона часто пояснюється відсутністю єдиного, комплексного підходу до проектування систем взаємодії. Використовування системного підходу дозволяє взяти до уваги безліч чинників самого різного характеру, виділити з них ті, які роблять найбільший вплив з погляду наявної загальносистемної мети і критеріїв, і знайти шляхи і методи ефективної дії на них.

Системний підхід заснований на застосуванні ряду основних понять і положень, серед яких можна виділити поняття системи, підлеглості мети і критеріїв підсистем загальносистемній меті і критеріям і т.д. Системний підхід дозволяє розглядати аналіз і синтез різних по своїй природі і складності об'єктів з єдиної точки зору, виявляючи при цьому найважливіші характерні риси функціонування системи і враховуючи найбільш істотні для всієї системи чинники. Значення системного підходу особливо велике при проектуванні і експлуатації таких систем, як автоматизовані системи управління (АСУ), які по суті є людино-машинними системами, де людина виконує роль суб'єкта управління.

Системний підхід при проектуванні є комплексним, взаємозв'язаним, пропорційним розглядом всіх чинників, шляхів і методів рішення складної задачі багаточинника і багатоваріантного проектування інтерфейсу взаємодії. На відміну від класичного інженерно-технічного проектування при використовуванні системного підходу враховуються всі чинники проектованої системи - функціональні, психологічні, соціальні і навіть естетичні. Автоматизація управління неминуче спричиняє за собою здійснення системного підходу, оскільки вона припускає наявність саморегульованої системи, що володіє входами, виходами і механізмом управлінням. Вже саме поняття системи взаємодії указує на необхідність розгляди навколишнього середовища, в якому вона повинна функціонувати. Таким чином, система взаємодії повинна розглядатися як частина більш обширної системи - АСУ реального часу, тоді як остання - системи керованого середовища. В даний час можна визнавати доведеним, що головна задача проектування інтерфейсу користувача полягає не в тому, щоб раціонально «вписати» людину в контур управління, а в тому, щоб, виходячи із задач управління об'єктом, розробити систему взаємодії двох рівноправних партнерів (людина-оператор і апаратно-програмний комплекс АСУ), раціонально управляючих об'єктом управління.

**2. Загальна характеристика предметної області**

**та мети створення додатку**

Процес колекціонування поштових марок пов’язаний з великою роботою по їх систематизації та впорядкуванню по багатьом параметрам. Кожна марка має певний набір характеристик, що являється основоположним при її ідентифікації. Персональні колекції марок досягають розмірів до декількох тисяч штук. Зараз, в час стрімкого розвитку комп’ютерних технологій, недоцільно сидіти з папірцем та папірцем, ретельно перераховуючи свою цінну колекцію. З’являється необхідність створити сучасний інформаційний додаток, який би забезпечував виконання всієї кропіткої роботи по систематизації колекції. Який би забезпечував швидкий пошук необхідної марки, або марок, з наявних в колекції, був би зручний, простий та приємний в користуванні.

1. **Характеристика задачі що розв’язується**

За мету в створенні даного додатку ставилась створити програмний комплекс який би виконував функцію власника колекції марок, по збереженню поточної інформації про колекцію, та по її систематизації.

Даний додаток оперує з базою даних, що зберігає перелік існуючих марок, та їх характеристики. База даних являється реляційною, та підпадає під вимоги третього рівня нормалізації.

Для реалізації збереження інформації про марки в база даних складається з шести таблиць. В основній таблиці утримуються власне записи про марки. В інших, переліки країн, типів паперу, особливостей перфорації, друку, та назви грошових одиниць.

В додатку реалізовані наступні функції навігації по базі даних: послідовна навігація, фільтрація по країні, по способу друку, по типу паперу, по перфорації. Всі перелічені типи навігації можуть використовуватись комплексно. Користувачу надається інформація що до загальної кількості марок в колекцію, та що до кількості марок, що підпадають під встановлений фільтр.

За задачу при створенні даного додатку, в рамках задачі до курсової роботи, також ставилось розробити максимально сприятливий та зручний інтерфейс для користувача. Дане питання буде детально розглянуто в четвертому розділі пояснювальної записки.

1. **Опис процесу взаємодії користувача з додатком**

Процес взаємодії користувача з додатком виконується за допомогою створення інтерфейсу користувача. Тобто набору засобів, які дозволяють реалізувати спілкування користувача з обчислювальними спроможностями комп’ютера. Для цього спілкування користувач може використовувати клавіатуру та графічний маніпулятор – мишу.

Для використання додатку необхідно запустити на виконання виконуваний файл. Він має розширення .exe .

Після запуску додатка, з’являється головне вікно програми. В верхній частині вікна розміщується головне меню додатка. В цьому меню можна знайти опції, за допомогою яких виконується загальне управління роботою додатка.

Дане вікно являється головним вікном додатка, саме в ньому відбувається редагування переліку марок колекції, їх, перегляд. В окремі вікна винесені настройки умов фільтрації відображуваних марок, редагування списків номінальних одиниць, засоби для налагодження інтерфейсу користувача.

1. **Обґрунтування проектних рішень**

**по інтерфейсу користувача**

* 1. При проектуванні форм

В проекті є головна форма form1- вона виконує головну функцію додатку - відображення колекції марок. Разом з цим це вікно виконує адмініструючи функції додатку. З цього вікна можна визвати форму, в якій виконується настройка інтерфейсу користувача. З цієї форми можна викликати форму для встановлення фільтру відображення колекції. З цієї форми можна припинити виконання додатку. Всі останні форми є допоміжними, в рамках даного додатку. В формі спостерігається ієрархія елементів управління. Пріоритет спадає з ліва на право і з гори вниз. В верхній лівій частині вікна утримується головне меню додатка. Воно містить в собі пункти, які виконують загальне керування роботою додатка – настройка інтерфейсу, допомога, коротка інформація про додаток та вихід з додатку. В середній частині вікна розміщується набір сторінок, в якому розміщено дві сторінки – перегляд колекції марок, та внесення/редагування записів про марок. На сторінці перегляду колекції марок є кнопка фільтрації. За її допомогою можливо викликати форму, яка встановить фільтр відображення марок колекції. Ця функція програми відноситься до другорядних, тому вона розміщена справа внизу – згідно з правилами організації ієрархії елементів управління. На кожній з сторінок є кнопка виклику допомоги. При її натисненні буде відображена допомога, що до цієї сторінки. Під набором сторінок розміщено кнопки яки є найменш приоритетні функції на цій формі – допомога, та вихід. При натисненні кнопки допомоги користувачу надається допомога щодо даного додатку в загалом. Кнопка вихід має розміщено тут зображень організації подібності всіх форм програмного комплексу.

Форма form2 забезпечує навігацію по номінальних одиницям, що відносяться до колекції марок. Тут також спостерігається ієрархія елементів управління з гори вниз. В верхній частині є не спадаючий список, який реалізує вибір номінальної одиниці, що переглядається. Нижче знаходиться список в якому відображаються всі наявні одиниці, даної номінальної одиниці. Поряд знаходяться кнопки які забезпечують створення нового запису в списку номінальних одиниць, редагування або видалення існуючого. Найменший пріоритет має кнопка закриття даної форми – „Вихід”.

Форма Form3 забезпечує створення умов фільтрації для відображення списку марок колекції в головному вікні. Тут розміщені поля зазначення номінальних одиниць, та підписи до них, які вирівняні по най правішому. Поля зазначення подібні. Вони знаходяться на однаковій відстані один від одного. Ширина кожного з полів зазначень не перевищує двадцяти відсотків від максимальної довжини строки даних, що в них обираються. Висоти їх однакові. Це забезпечує користувачеві потребу подібності. В самій нижній частині форми розміщуються кнопки виклику допомоги.

За допомогою форми Form4 користувач має можливість налагодити деякі елементи інтерфейсу додатку. Дане вікно має дві вкладки. На першій з них можна настроїти наявність та порядок відображення інформації що до марок в списку марок колекції. Зробити це можна безпосередньо в списку опцій відображення. Така можливість зробити одне і те саме різними способами покликано забезпечити у користувача відчуття повного контролю над додатком. На другій вкладці користувач отримує можливість настроїти за власними вподобаннями шрифти, що використовуються в додатку. Всі шрифти додатку розподілені на дві групи: шрифти надписів та шрифти даних. Користувач має можливість налагодити будь який з них. Поряд з кнопками, що викликають діалоги зміни шрифтів розміщуються інформаційні надписи, які описують який саме шрифт являється поточним, тобто шрифт, якім відображаються дані або надписи – назва шрифту та його розмір. Це вікно, взагалі, реалізує необхідність користувача в налагодженні інтерфейсу. Маючи таку можливість користувач, при роботі з програмою відчуває себе комфортно і захищено.

Форма Form5 надає користувачу коротку інформацію щодо додатку – ким для чого та коли створено даний додаток. Єдина кнопка „Вихід” цієї форми розмішується з права внизу, як і на будь якій формі з даного програмного комплексу. Це реалізує подібність та дружність форм до користувача.

* 1. При проектуванні діалогів

Принцип: командує користувач. Даний додаток відповідає цьому принципу, згідно того, в силу того, що при його проектуванні розробник намагався максимально забезпечувати користувача можливість керування додатком. Можливо змінювати порядок відображення даних про марки колекції, можливо окремо настроювати шрифти відображення даних та шрифти відображення надписів. Користувачу прививається думка, що він може звертатися з системою як з продовження самого себе. Даний програмний комплекс є гнучким і настоюваним.

Принцип: наочність. Даний принцип реалізується в додатку в тому, що: на кнопках хоча і розмішуються короткі надписи, тим не менш зміст їх дій становиться очевидним в зв’язку з їх розташуванням. Наприклад кнопка „Зберегти” розмішується під полями вводу даних. Користувачу зрозуміло, що дана кнопка зберігає значення, які утримуються в цих полях. Кольори додатку витримані в одному тоні – це сприяє наочності. Внесення або редагування даних пов’язано з активацією полів вводу. При своїй активації вони змінюють колір з сірого на білий. Сірий – колір пасивності. Білий – колір активності. При неможливості видалити запис номінальної одиниці користувача видається повідомлення про це. Все це наочність.

Принцип: однаковості. Всі форми додатку подібні між собою: їх розміри відповідають золотому перетину, кнопка „Вихід” на кожній формі знаходиться в однаковому місті. Завдяки належності додатку до даного принципу, користувач знаходить інформацію, або елементи керування завжди в очікуваних місцях форми. Кнопки по навігації по базі даних розміщуються поряд – це відповідає принципу „подібне лежить поряд”. Це відповідає принципу однаковості. Додаток подібний до операційного середовища, в якому він запускається - до Windows.

Принцип: терпимість до користувача. Додаток спроектований таким чином, що користувач може робити все що завгодно всередині додатка. Однак при дії користувача, яка приводить до зміни інформації в базі даних, спливає діалогове повідомлення, яке попереджує користувача про можливі наслідки його дій. Це вікна запитів при видалені записів номінальних одиниць, чи при видалені марок. Хоча в додатку і витримується принцип „командує користувач”, тим не менш його дії не приводять до знищення додатка чи до значного його псування. До більшості елементів додатку є контекстні підказки, які з’являються, коли указник миші знаходиться над цим елементом.

Принцип: зворотній зв’язок. Не роблячи ніяких зайвих рухів мишею та не натискаючи клавіші клавіатури можна в будь який момент часу дізнатися в якому стані зараз знаходиться додаток. Реалізується це за допомогою статусного рядка в нижній частині головного вікна.

Принцип: естетика. Додаток спроектований таким чином, що користувач, працюючи з ним не погіршує своє здоров’я. Додаток має привабливий зовнішній вигляд. Розміри всіх форма додатку відповідають золотому перетину. Розміщення і вигляд елементів управління у всіх вікнах подібне.

Принцип: простота. Загалом додаток досить прозорий, і не має обтяжливих елементів. Від головної форми до дочірніх модальних форм спостерігається тенденція спрощення, та зменшення кількості інформації, що утримується. Кількість інформації що утримується в головній формі - найвища. Додаток моє баланс між простотою інтерфейсу і доступністю всіх необхідних функції. Використовується принцип мінімальної кількості слів і принцип послідовності розкриття даних.

* 1. При проектуванні довідкової системи

Довідкову систему даного додатку було спроектовано з міркувань забезпечення найвищого рівня допомоги користувача в роботі з додатком.

В програмному комплексі використовується два типа довідки. Один з них – основний. Ця довідка розміщується в окремому файлі з розширенням .chm. Це стандартний тип файлів допомоги системи Windows. Ця допомога складається з розділів. Для кожної форми додатку в допомозі мається окрема сторінка допомоги. Дана довідка є контекстно залежною. При бажанні отримати довідку на будь якому вікні додатку натисніть кнопку „Допомога”, або натисніть кнопку F1. Буде відкрито розділ довідки, який описує дане вікно, або сторінку в вікні.

Інший типі довідки, представлений спливаючими підказками. Практично на кожному елементі додатку є спливаюча довідка. Для того, щоб викликати цей тип допомоги, підведіть мишу до необхідного елементу додатку. Через декілька секунд спливе вікно з довідкою. Цей вид допомоги можна відключити вимкнувши відповідний вимикач на сторінці „Інше” вікна „Настройка інтерфейсу користувача”.

1. **Використання принципу розробки**

**орієнтованих на користувача**

При розробці даного додатку були використані наступні принципи розробки, орієнтовані на користувача:

А) Командує користувач

Б) Наочність

В) Однаковість

Г) Терпимість до користувача

Д) Зворотній зв’язок

Е) Естетика

Ж) Простота

Робота додатку організована таким чином, що у користувача складається враження, що все, що робиться залежить тільки від нього, що він може все, курує всім і може в будь який момент змінити поточний стан додатку. Для цього в додатку передбачена можливість налагодження середовища. Користувача може зробити зовнішній вигляд додатку такий, який йому найбільше до вподоби.

Всі форми, що використовуються в додатку мають принцип наочності, тобто користувач витрачає мінімум часу для визначення з тим, що йому необхідно робити. Максимальному сприйняттю інформації, розміщеній на екрані сприяє правильний підбір кольору графічних об’єктів, шрифти означальних текстів. Додаток реагує на кожну дію користувача зміною режиму доступності об’єктів за допомогою кольору і обрамлення.

У відношення однаковості проектування додатку можливо помітити, що від вікна до вікна елементи управління ті інформації розміщуються в одних і тих же місцях у вікні. Використовуються однакові елементи навігації, піктограми, кнопки та коментарі для реалізації подібних дій користувача.

Додаток спроектований таким чином, що користувач може робити все що захоче в середовищі додатку. Хоча в додатку і використовується принцип „командує користувач”, тим не менш його дії не можуть призвести до наслідків, які пов’язані з пошкодженням даних або безпосередньо самого додатка.

Весь програмний комплекс в цілому має збалансований привабливий вигляд. Додаток створено з урахуванням мінімізації погіршення зору у користувача і мінімізацією втоми користувача при роботі з додатком. Весь додаток збалансований по кольорам, контрастності, типу шрифтів та використанню графічних елементів.

В цілому весь додаток „прозорий” і не має перевантажуючи елементів, що можуть стати на перепоні простій та елегантній роботі з додатком. Використовується підхід в зв’язку з яким користувач може відключити додаткові спливаючі підказки для елементів додатку. Якщо на початку роботи з додатком функції деяких елементів ще не зовсім зрозумілі користувачу і спливаючі підказки досить велика порада при роботі, то через деякий час, коли користувач все впевнено користується додатком ці спливаючі елементи лише лютують користувача. Тому їх можна відключити.

Використовуються принципи мінімальної кількості слів і послідовного розкриття змісту.

1. **Використання принципів організації екранів**

В даному додатку було прийнято рішення створити ієрархічну структуру вікон. Головне вікно – вікно що найбільше насичено інформацією корисною для користувача. Тут відображається список самої колекції марок, саме в цьому вікні можна реєструвати нові марки. Інші вікна мають допоміжний характер в порівнянні з головним вікном. Всі екрану мають чітку підпорядкованість елементів. Послідовне розміщення елементів реалізується з ліва на право і згори вниз. Також використовується принцип послідовності подій, наприклад вікно з’являється тільки після того, як було натиснуто кнопку з відповідним контекстом. Для об’єднання елементів, що приймають участь в вирішення якої-небудь задачі для цих елементів використовуються об’єкти групування. Основні елементи виділяються – кольором, контуром, іншим шрифтом, це зроблено для того, щоб користувач розумів що саме вони являються головними.

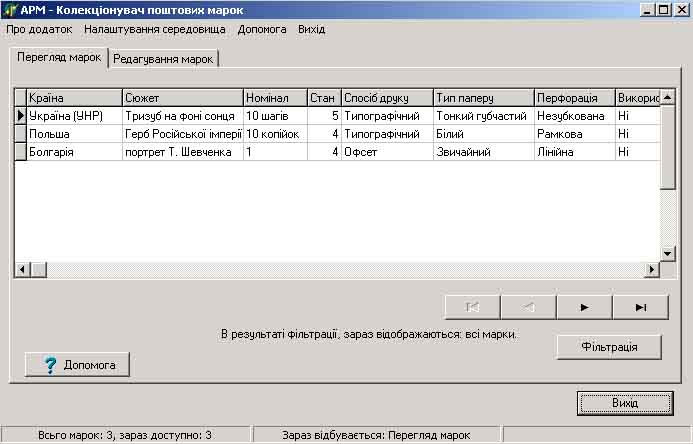
Жодне з вікон не випадає з загального ряду вікон з якими користувач чає справу працюючи з додатком.

**8. Рішення що до кольору, при проектування додатку**

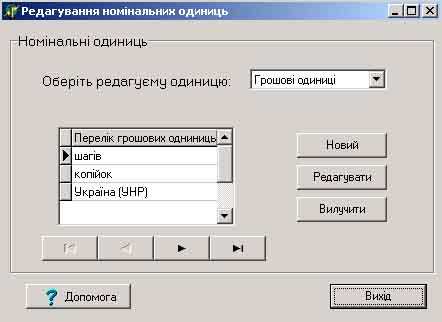
В додатку за допомогою кольору відбувається зосередження уваги користувача до частини екрану. Колір, що використовується для зображення фона форм не навантажує очі – використовується сіро-зелений колір. Це важливо, оскільки користувач може довгий час проводити за роботою з додатком. Для візуального наближення інформації до користувача використовуються більш яскраві відтінки кольорів. Наприклад: активний/неактивний стан поля вводу. Простір вікна на якому не розміщуються які-небудь елементи забарвлений в самий пасивний м’який колір.

**9. Перелік форм додатка**

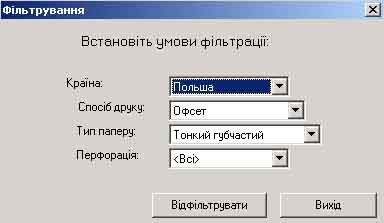
Головна форма додатку



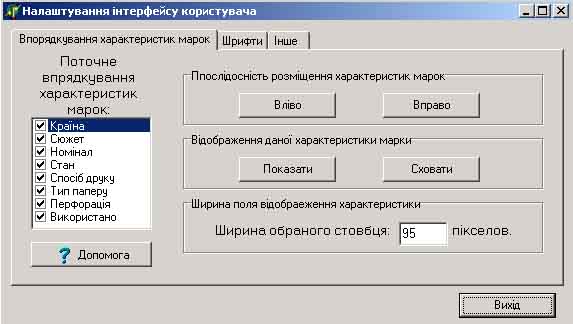
Форма редагування номінальних одиниць



Форма фільтрації марок



Налагодження інтерфейсу



1. **Класи, підкласи та компоненти**

**що використовуються в додатку**

В додатку використовувались наступні візуальні компоненти: для створення форм використовувався клас TForm. Для створення головного меню – клас TMainMenu. Для створення сторінок на формі клас TTabSheet в зв’язку з контейнером TPageControl. Для створення надписів – TL abel. Для створення кнопок - класи TButton та TBitBtn. Для створення полів вводу – класи TEdit, TDBEdit, для створення ниспадаючих списків TComboBox, TDBLookupComboBox. Для створення кнопок навігації по записам – TDBNavigator. Для створення вимикачів – TDBCheckBox. Для створення списків – TDBGrid, TChekListBox. Для групування елементів – TGroupBox, TPanel. Для відображення малюнків – Timage.

Не візуальні компоненти, що використовувались: Для доступу до даних бази даних – компоненти TDatabase, TTable, TQuery, TDataSource. Для запиту шрифту, що застосовується для відображення – TFontDialog.