

РОЗДІЛ 5. ПРОЕКТУВАННЯ ІНТЕРФЕЙСУ КОРИСТУВАЧА

5.1. Вимоги до інтерфейсів інформаційних систем

Оскільки якість процесу інтерактивної взаємодії користувача із системою (швидкість, зручність, низький рівень втоми) пов'язана з такими психологічними характеристиками людини як короткострокова та середньострокова пам'ять, час реакції, можливості сприйняття візуальної інформації, то при розробці інтерфейсу необхідно пам'ятати, що [17, 46]:

- інтерфейс – *сама важлива частина СППР* з точки зору її реклами з метою продажу і з точки зору безпосереднього користувача системи, який може працювати з нею по декілька годин поспіль;
- інтерфейс впливає на характер рішень, які приймає ОПР, він може *прискорювати* час прийняття рішення та *покращувати* або *погіршувати* їх якість;
- який саме конкретний тип інтерфейсу **можна** створити за допомогою вибраних інструментальних засобів і які принципові можливості може надати інструментальна система.

Основними властивостями, яким повинні задовольняти інтерфейси, є такі [46]:

- А** – **Адаптованість** означає, що інтерфейс повинен бути:
- сумісним з потребами та можливостями користувача;
 - забезпечувати простоту переходу від виконання однієї функції до іншої;
 - *забезпечувати користувача на високому рівні вказівками* стосовно його можливих дій, а також генерувати належний зворотний зв'язок на його запити;
 - надавати користувачу можливість відчувати себе повноправним керівником ситуації при розв'язанні всіх типів задач, тобто, забезпечувати його всією необхідною інформацією; користувач повинен бути впевненим, що *він сам розв'язує поставлену задачу*;
 - забезпечувати користувача різними, взаємно доповнюючими формами представлення результатів в залежності від типу запиту або від характеру отриманого рішення;
 - враховувати особливості користувачів різних рівнів; наприклад, для *керівника підприємства* зручнішим є узагальнене графічне пред-

ставлення результатів роботи СППР у вигляді діаграм та графіків, а інженеру-економісту потрібні конкретні цифри у їх часовій послідовності.

Б – **Достатність інтерфейса** означає таке:

– допустимі запити користувача повинні бути чіткими і однозначними для користувачів всіх рівнів, а також для прикладних задач всіх типів;

– реакція системи на всі типи запитів також повинна бути однозначною і зрозумілою і, по можливості, простою.

В – **Дружність інтерфейсу**

Це максимальна простота його використання і готовність в повній мірі задовольнити запити користувача при розв'язанні визначеного класу задач.

Г – **Гнучкість інтерфейсу**

Гнучкість інтерфейсу – це можливість його адаптування до розв'язання конкретної задачі. Якщо розв'язувана задача дуже складна, то інтерфейс повинен полегшувати формулювання запитів і видавати результати у формі, яка легко і швидко сприймається користувачем. Тобто *інтерфейс повинен буди максимально простим* навіть у випадку, коли розв'язується дуже складна задача.

При цьому *простота означає* таке:

– інтерфейс не повинен бути перевантажений деталями щодо представлення розв'язку поставленої задачі – користувач може не охопити всіх подробиць (і в цьому, як правило, немає потреби) – тобто *нічого зайвого, крім того, що необхідно для розуміння результату*;

– він не повинен містити зайвих *декоративних деталей*, які відволікають від головної задачі;

– інтерфейс повинен бути консистентним, тобто, ґрунтуватись на використанні відомих, загальноприйнятих методів і засобів представлення інформації;

– в ідеалі процес взаємодії користувача з системою не повинен представляти ніяких труднощів.

5.2. Характеристики інтерфейсу користувача та принципи його формування

Під інтерфейсом взаємодії розуміють сукупність домовленостей про форми, способи, процеси, правила взаємодії користувача з ПЕОМ [17, 46]. Одним із перспективних напрямів створення інтерфейсів

користувача вважаються адаптивні інтерфейси. Адаптивним інтерфейсом користувача (AI) називають сукупність програмних та технічних засобів, які дозволяють користувачу ефективно використовувати всі можливості, які надає система, та задаються конкретні налагодження для кожного користувача. Одним з головних критеріїв ефективності функціонування програмних продуктів є максимальне використання для роботи з ними людських ресурсів. Це пов'язано перш за все з тим, що вартість людських ресурсів для роботи з програмним забезпеченням вже досягла вартості системних компонент, а в багатьох випадках суттєво їх перевищує.

Для того, щоб адаптивний інтерфейс міг бути коректно вбудованим в процес діалогу для користувача кожного рівня, він повинен враховувати апіорну інформацію про психофізичні, професійні, особисті характеристики користувача. Ця інформація здобувається шляхом попереднього початкового тестування користувача. Але цієї інформації для коректної роботи AI явно недостатньо. Додаткова інформація про користувача, який взаємодіє в даний момент з системою, повинна бути отримана з аналізу дій користувача безпосередньо в процесі роботи. Цей спосіб отримання знань про користувача і є основою динамічної адаптації.

Адаптивний інтерфейс повинен забезпечувати користувачу полегшений режим взаємодії. Адаптивна система повинна комбінувати в собі особливості адаптивних і адаптованих компонентів. Це буде давати користувачам засоби, які надають можливість проводити власні зміни в налагодженнях інтерфейсу, виходячи з їх потреб.

Системи, які дозволяють користувачу змінювати певні системні параметри і відповідно змінювати їх поведінку, називають адаптованими. Системи, які адаптуються до користувача автоматично, ґрунтуючись на припущеннях системи, називаються адаптивними.

Незважаючи на те, що на сьогодні вже досягнуто певний прогрес у побудові інтерфейсів користувача, питання взаємної адаптації функціонування програмних систем (ПС) і користувача ПС, досі залишаються відкритими. Більш того, актуальність їх зростає у зв'язку з розвитком мережі Internet, появою повністю графічних інтерфейсів, розвитком засобів мультимедії, широким застосуванням гіпертекстових документів, розширенням сфер використання та масовістю застосування ПЕОМ.

Розробка природно-мовного (ПМ) інтерфейсу в складі СППР дозволяє забезпечити виконання запиту особи, що приймає рішення (ОПР), або експерта-аналітика до бази експертних знань. Можливість

адаптації системи до рівня професіоналізму користувача здійснюється за рахунок здатності сприймати і виконувати запити на внутрішній формальній мові, що забезпечує більш швидкий доступ до інформації.

Природно-мовний інтерфейс є посередником між людиною і базою даних. Він переводить запити, що надходять природною мовою, у формальне представлення, звертається з ним до бази даних, і представляє результат, використовуючи алгоритми і технології реалізовані в СППР, у вигляді, придатному для перегляду й аналізу (рис. 5.1).

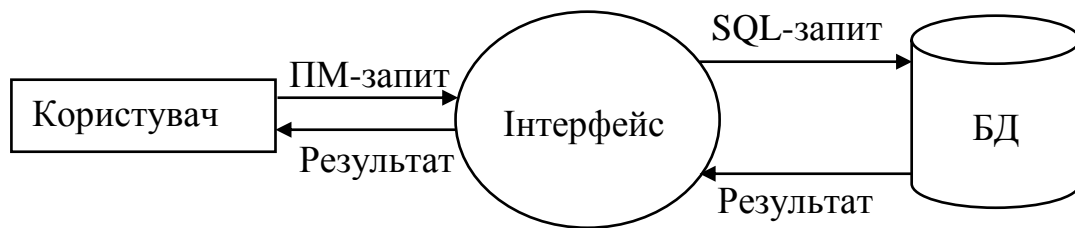


Рис. 5.1. Загальна схема потоків даних ПМ-інтерфейсу

Користувачами ПМ-інтерфейсів можуть бути як співробітники підприємства – менеджери різних рівнів, так і клієнти. Усім, кому потрібна можливість швидкого і прямого доступу до актуальної інформації. Наведемо кілька областей основних застосувань ПМ-інтерфейсів у діяльності організацій:

- каталог товарів і електронний магазин;
- календарне планування;
- керування кадрами і бухгалтерія;
- склад і планування закупівель;
- аналіз продаж;
- банківська діяльність;
- біржова діяльність.

Використання ПМ-інтерфейсу найвигідніше у випадку масового користувача (наприклад, відвідувача сайту), а також як інтегровану з інформаційною системою організації можливість прийняття рішення, при якому доступ до даних є штатною можливістю інформаційної системи організації і, таким чином, є універсальним і, можливо, основним способом пошуку й одержання інформації.

У ситуації, коли успіх організації визначає його здатність адекватно реагувати на зміну зовнішніх і внутрішніх умов, прямий і швидкий доступ до достовірної актуальної інформації, використання ПМ-інтерфейсу

надає можливість значно скоротити витрати організації на різних рівнях:

- можливість формулювати потреби в інформації найбільш простим і разом з тим самим доступним чином – як для співробітників усередині компанії, так і для доступу клієнтів до публічної інформації (наприклад, до каталогу товарів);

- розвантажує фахівців від рутини створення форм і звітів по кожному розрізі інформаційного простору в базі даних. Крім того, оскільки формальне представлення ПМ-запиту доступно, ПМ-інтерфейс може бути корисний для розробки традиційних додатків на основі баз даних, оскільки рятує від ручного етапу побудови складних SQL-запитів;

- при розробці ПМ-інтерфейсів використовуються найсучасніші методи інженерії знань, що дозволяє використовувати технологію розуміння на більш високому рівні абстракції даних, ніж реляційна модель зі збереженням записів у зв'язаних таблицях; це серйозний крок на шляху до керування не просто інформацією, але й корпоративними знаннями.

Найбільшої ефективності можна досягти у випадку використання комбінації традиційних засобів доступу до даних і доступу природною мовою. Якщо на етапі проектування системи в неї буде вбудовано ПМ-інтерфейс, це одночасно підвищить надійність і корисність ПМ-інтерфейсу, а також різко збільшить інтелектуальність і дружність інформаційної системи в цілому.

Розглянемо конкретний випадок використання ПМ-інтерфейсу. Нехай фірма використовує бази даних для збереження внутрішньої бізнес-інформації. На основі баз даних працюють різноманітні додатки – складська програма, бухгалтерія, кадри, планування, прогнозування стану ринку, закупівлі, які часто об'єднані в єдину комплексну систему. Усі ці додатки, як правило, представляють інформацію у виді таблиць і звітів у визначених розрізах з можливістю фільтрації.

Але часто буває так: для прийняття рішення «тут і зараз» необхідно мати такий розріз інформації, таку комбінацію умов, які неможливо задати стандартними засобами, особливо якщо ця інформація знаходиться на стику різних підсистем, наприклад така:

- витрати, зумовлені дефіцитом товару А на складі з початку місяця;

- коли очікується партія товарів фірми X, і що в ній буде;
- кількість проданих холодильників Bosh за кольорами;
- прибуток від продажу відеомагнітофонів у залежності від їхньої ціни;
- наявність на складі 10 самих ходових товарів за минулий місяць.

Як правило, уся необхідна інформація для одержання таких цифр є у корпоративній базі даних. Але ні конструкцію бази, ні мови запитів до неї ви не знаєте – цим володіють розробники БД та інформаційної системи, побудованої на її основі. Можна, звичайно, попросити їх обчислити ці цифри вручну (з використанням мови SQL, якщо база реляційна), або додати в систему нові функціональні можливості.

Це робиться просто, якщо розробники системи і БД працюють на даному підприємстві. Ситуація складніша, якщо розробники – це стороння організація. Але у будь-якому випадку наступного разу вам буде потрібна яка-небудь інша інформація і все почнеться спочатку.

В наявності проблема – і є потреба в оперативному одержанні інформації в різноманітних розрізах для аналізу і прийняття рішень, є сама вихідна інформація, але доступ до неї вимагає знання особливостей побудови бази даних і досвіду розробника. Це рішення полягає у побудові природно-мовного інтерфейсу до бази даних і (у більш загальному випадку) – до інформаційної системи.

Це надає можливість:

- мати прямий доступ до будь-яких аналітичних розрізів інформації, що зберігається в базі;
- швидко отримувати необхідну інформацію, що потрібна у поточний момент;
- зосереджуватися на тому, що треба одержати з бази, а не на тому, як це зробити;
- вчасно контролювати правильність занесення інформації в базу;
- доповнити інформаційну систему інтелектуальною технологією, що дозволить в остаточному підсумку скоротити витрати на її експлуатацію і підвищити ефективність роботи з нею в цілому.

У загальному вигляді вимоги до ПМ-інтерфейсів повинні визначати зміст і форми інформації, що надається, а також регламент взаємодії користувача і системи. Ці вимоги повинні впливати з функцій користувачів, що знаходяться на різних рівнях ієрархії, а також стандартних вимог законодавчих і виконавчих органів до форм обліку і звітності,

регламентам діяльності об'єкта, що визначає в сукупності зовнішній і внутрішній документообіг.

Базовою формою опису технологічних і бізнес-процесів повинна бути сама схема процесу, представлена на різних рівнях ієрархії з різним ступенем деталізації, на якій повинні бути відображені взаємозалежні технологічні і функціональні компоненти процесу разом з показниками функціонування, що дозволяють судити про факти і якість виконання функцій.

Сервісними формами представлення інформації можуть бути тексти, таблиці, графіки, гістограми, а також абстрактні образи, що у сполученні з базовими формами представляють повний спектр типів інформування користувача про стан об'єкта діяльності.

Таким чином, програмне забезпечення інтерфейсного модуля повинне забезпечити такі можливості:

- введення запитів до бази знань на внутрішній формальній мові системи;
- редагування запитів користувача;
- виконання запитів до бази експертних знань і вивід результатів на екран комп'ютера в зручному для користувача вигляді;
- перегляд проміжних результатів роботи;
- поповнення і коректування бази експертних знань у режимі діалогу;
- верифікацію відповідей на запити.

Основними етапами діяльності користувача в системі, як особи, що приймає рішення, є такі:

1. сприйняття інформації: виявлення об'єкту сприйняття, виділення в об'єкті окремих ознак, ознайомлення з виділеними ознаками і розпізнавання об'єкта сприйняття;
2. оцінювання інформації, її аналіз і узагальнення на основі раніше сформованих критеріїв оцінки; оцінка виробляється на основі співставлення і аналізу сформованої у ОПР інформаційної моделі з внутрішньою образно-концептуальною моделлю об'єкта управління (предметної області);
3. прийняття управлінських рішень.

На основі загальних принципів побудови інформаційних систем і врахування особливостей адаптивного інтерфейсу можна сформулювати загальні принципи побудови адаптивних інтерфейсів.

1) *Принцип відповідності призначення і структури інтерфейсу* поставленим цілям і задачам. Неможливо побудувати універсальний багатофункціональний інтерфейс, який міг би бути використаний і в складі АСУ реального часу, і в складі системи управління фізичним експериментом. Типізація і універсальність можлива тільки в рамках визначених класів систем.

2) *Принцип мінімізації витрат ресурсів користувача:*

– користувач повинен виконувати тільки ту роботу, яка необхідна і не може бути виконана системою, не повинна бути повторенням уже виконаної роботи;

– користувач повинен запам'ятовувати якомога меншу кількість інформації, оскільки це збільшує його можливості приймати оперативні рішення в екстремальних випадках.

3) *Принцип максимального взаєморозуміння та непротиворіччя.* Робота з системою повинна бути простою, не викликати у користувача додаткових ускладнень в пошуках необхідних елементів інтерфейсу. Інформація, що отримана за допомогою інтерфейсу, не повинна вимагати декодування або додаткової інтерпретації користувачем. Якщо в процесі роботи були використані деякі прийоми, то і в інших випадках прийоми роботи повинні бути ідентичні.

4) *Принцип незбитковості.* Користувач повинен вводити тільки мінімальну інформацію та недопустимий ввід уже введеної раніше і отриманої системою інформації.

5) *Принцип безпосереднього доступу до системи підказок.* Система повинна забезпечувати користувача необхідними інструкціями. Система підказок повинна задовольняти трьома критеріям:

- стислість, якість і повнота інформації;
- вичерпний характер повідомлень про помилки;
- наявність повідомлень про стан системи.

6) *Принцип гнучкості.* Адаптивний інтерфейс повинен забезпечити спілкування з системою користувачам різного рівня підготовки. Наприклад, для недосвідчених користувачів інтерфейс може бути організовано як ієрархічна структура меню, а для досвідчених – з команд, комбінацій натиснень клавіш та параметрів командного рядка.

7) *Принцип максимальної концентрації користувача на задачі,* що розв'язується і локалізації повідомлень про помилки. Адаптивний

інтерфейс повинен допомагати концентрувати увагу користувача на конкретній задачі, що розв'язується.

8) *Принцип врахування професійних навичок* конкретного користувача. При роботі з системою на основі збору деяких даних, які отримуються з моделі користувача, проектується «людський фактор», який тісно вплітається в особливості функціонування всієї системи. Формування концептуальної системи означає усвідомлення і оволодіння алгоритмами функціонування інтерфейсу користувача.

10) *Принцип легкості користування і простоти навчання*. Адаптивний інтерфейс не повинен приводити користувача до роздратування, примушувати до необдуманих дій. Адаптивний інтерфейс враховує, що користувач отримує досвід роботи з системою, цілі користувача можуть змінюватись у процесі роботи з системою, Адаптивний інтерфейс характеризується простотою виправлення помилок. Керуванню роботи з системою можна легко навчитись, система забезпечує навчання в процесі роботи.

11) *Принцип надійності*. Система повинна бути надійною з точки зору роботи користувача. Вона готова до роботи завжди, коли в цьому виникає необхідність, відмови трапляються рідко, час відповідей системи не повинен перевищувати встановлених границь. В системі реалізуються можливості захисту інформації та забезпечення необхідного ступеня секретності та різних рівнів доступу до наявної інформації.

5.3. Проектування інтерфейсу на принципах людського фактору

Предметом вивчення «людського фактору» є дослідження факторів, явищ, подій, які впливають на продуктивність праці, ефективність та якість життя всіх тих, хто працює, а також проектування машин та обладнання, що використовується в процесі виконання роботи [46]. Власне наука про людський фактор займається розробкою інженерних принципів проектування (дизайну) в усіх технічних галузях, використанням антропометричних таблиць вимірів людського тіла у процесі дизайну, психологічними принципами навчання, аналізом процесів запам'ятовування візуальної та аудіоінформації, аналізом впливу на людину таких соціальних факторів, як шум навколишнього середовища,

скупчення великої кількості людей, засобів зв'язку, а також проектуванням житлових приміщень, офісів та місць відпочинку (парків, технопарків, майданчиків, і т. ін.). Фактично будь-яка продукція, яка з'являється на ринку сьогодні і буде з'являтися у майбутньому, стає виграшною і конкурентоспроможною завдяки застосуванню при її створенні принципів людського фактору.

Фахівці з людського фактора використовують всю наявну інформацію із психології, математики та технічних наук з метою удосконалення дизайну машин, механізмів, обчислювальних систем і підвищення, таким чином, ефективності роботи людини-оператора, а також поліпшення умов її життя та відпочинку. Набагато ефективніше та дешевше розробити і впровадити елементи «людського фактора» в процес проектування систем, ніж переробляти готову продукцію і відповідності до реклаमाцій та скарг споживачів.

Можна сказати, що *людський фактор* – це метод проектування, дослідження та пояснювання інтерфейсу в широкому сенсі (тобто, проміжного середовища) між людиною та машиною або знаряддями, які створюються людиною. З точки зору психології людини ми розглядаємо характеристики оператора і машини або знаряддя праці з метою визначення кращих методів чи способів конструювання цих машин та знаряддя, які підвищують ефективність праці та полегшують користування ними. В результаті такого аналізу проекту може з'явитись необхідність змінити конструкцію, щоб вона була доступною та зручною (комфортною) для оператора.

Вимоги та рекомендації *науки про людський фактор* є надзвичайно важливими з точки зору організації взаємодії користувач-комп'ютер. Всіх користувачів комп'ютерних систем можна розділити на дві основні групи: *фахівці* (постійно користуються комп'ютером) та *тимчасові користувачі* (користуються комп'ютером час від часу – керівники різних рівнів, бухгалтери, користувачі-початківці). Фахівці проводять, як правило, за комп'ютером значний час, користуючись програмними системами для розв'язання своїх задач. Такі користувачі не мають проблем з користуванням клавіатурою та монітором незалежно від типу програмного продукту. Тимчасові користувачі можуть

час від часу використовувати конкретну програмну систему для знаходження розв'язку спеціалізованих задач, наприклад, редактор текстів для набору текстових матеріалів або Excel для підготовки бухгалтерських звітів. Такий користувач може ніколи не почуватись повністю комфортно з терміналом та програмою. Зважаючи на те, що на сьогодні кількість тимчасових користувачів надзвичайно велике, *інтерфейс програмної системи повинен бути максимально дружнім, щоб забезпечити комфорт та якість результатів взаємодії.*

Загальні принципи людського фактору

При проектуванні інтерфейсу програмна система – користувач необхідно враховувати такі фактори, які є важливими з точки зору інженерії людини.

1. Користувач повинен завжди знати, що робити далі

Тобто, система повинна давати йому інструкції стосовно того, як продовжити роботу, створити резервний файл результатів, вийти із системи, і т. ін. Існують декілька ситуацій, які вимагають зворотного зв'язку від системи (пояснення для користувача):

– *Користувач повинен знати, що система чекає від нього. Це може бути зроблено у вигляді простого повідомлення, такого як ГОТОВО (READY), ВВЕДІТЬ КОМАНДУ (ENTER COMMAND), ВВЕДІТЬ ВИБІР (ENTER CHOICE) або ВВЕДІТЬ ДАНІ (ENTER DATA).*

– *Користувач повинен знати, що дані введено коректно. Наприклад, це можна зробити переміщенням курсора у поле, де написано ДАНІ КОРЕКТНІ або ВВЕДЕННЯ ЗАВЕРШЕНО (INPUT OK).*

– *Користувач повинен знати, що дані не були введені коректно. Система повинна дати просте коротке повідомлення щодо необхідного формату даних. Більш докладні рекомендації щодо можливих форматів даних можна надати за допомогою функції ДОПОМОГА (HELP).*

– *Якщо має місце затримка з обчисленнями, то користувач повинен про це знати. Для виконання деяких видів обчислень (наприклад, при виконанні оптимізаційних розрахунків) необхідно витратити кілька десятків секунд або навіть хвилин. У такому випадку система повинна давати короткі повідомлення, наприклад: ПОШУК ОПТИМУМУ – БУДЬ-ЛАСКА ЗАЧЕКАЙТЕ або СОРТУВАННЯ – ЦЯ*

ОПЕРАЦІЯ МОЖЕ ТРИВАТИ ДЕКІЛЬКА ХВИЛИН, БУДЬ-ЛАСКА ЗАЧЕКАЙТЕ. Користувач буде впевнений, що система продовжує працювати.

– *Користувач повинен знати, що система завершила (не завершила) виконання завдання.* Це важливо завжди, але особливо важливо у тих випадках, коли обчислення можуть тривати відносно довго. Повідомлення може мати такий характер: ДРУК ЗАВЕРШЕНО або ПРИНТЕР НЕ ГОТОВИЙ – ПЕРЕВІРТЕ НАЯВНІСТЬ КАРТРИДЖУ, або ОПТИМАЛЬНА ТРАЄКТОРІЯ ЗНАЙДЕНА.

2. Форматування екрану монітора необхідно робити таким чином, щоб різні типи інформації, команди, повідомлення завжди з'являлись в одній і тій же області.

При такому підході до форматування екрану користувач завжди буде хоча б приблизно знати, де шукати необхідну інформацію. Наприклад, екран можна розділити на такі зони (області):

– *Вікно з назвою (титолом) сторінки на екрані (title window).* Титул (назва) необхідні для ідентифікації вікна конкретного типу з точки зору користувача.

– *Вікно з прапорцями (flag window).* Це вікно містить вказівник (вказівники) на інші конкретні вікна екрану з метою показати, що виникла помилка чи проблема з виконанням завдання. Наприклад, якщо користувач зробив помилку при введенні команди чи даних, то у вікні з прапорцями повинно з'явитись повідомлення «...», яке показує в якому рядку чи полі має місце помилка. Для того щоб взнати конкретну причину чи суть помилки, користувач повинен подивитись у вікно повідомлень (розглядається нижче). Замість вікна з прапорцями можна скористатись такими атрибутами як *мигання* або *інверсним зображенням*.

– *Вікно повідомлень (message window).* В це вікно система посилає повідомлення для користувача. Наприклад, повідомлення про помилки або пропозиції щодо продовження обчислень, ведення діалогу. Це вікно може бути порожнім на протязі більшої частини часу роботи з системою.

– *Вікно виходу (escape window).* Воно призначене для того щоб користувач міг завершити поточний сеанс роботи з системою або підсистемою. Наприклад, вікно виходу може містити інформацію стосовно того як повернутись до головного меню системи або до попереднього меню опцій (вибору).

– *Операційне вікно (body window)*. Операційне вікно – найбільше вікно діалогу. Ця область використовується для вводу даних, виводу результатів роботи, виводу пояснюючих повідомлень за допомогою ведення діалогу з системою, висвітлення опцій меню та ін.

Зони не обов'язково повинні мати візуальні границі, але, якщо вони є, то покращують сприйняття інформації. Приклад можливого розділення екрану на зони наведено на рис. 5.2. В іншому варіанті розподілу екрану повідомлення, допомога та групування клавіш можуть розміщуватись в нижній або верхній частині екрану.

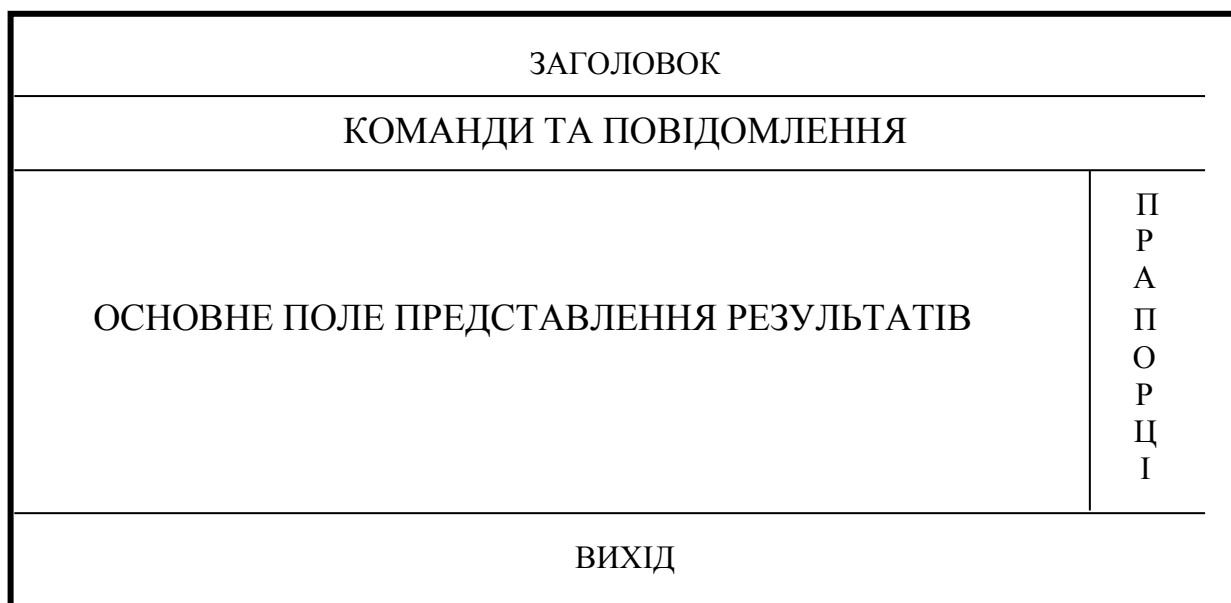


Рис. 5.2. Приклад розподілу екрану терміналу на зони

3. При проектуванні функцій пейджингу чи скролінгу у межах операційного вікна діалог повинен обмежуватись однією ідеєю на фрейм.

Наприклад, зона повинна висвітлювати одне меню, один результат виконання завдання або відповідь на один запит.

Вибір між пейджингом та скролінгом в зоні операційного вікна залежить від типу інформації, яку необхідно висвітлити. Якщо інформація є неперервною за своєю природою (наприклад, текстова інформація щодо отриманих результатів), то використовують скролінг. Перегляд рядків тексту виконують за допомогою курсора. Пейджинг використовують у випадках, коли інформація висвітлюється у вигляді окремих записів за один раз.

4. Повідомлення, команди або інформація щодо результатів повинні утримуватись на екрані достатньо довго, для того щоб користувач міг прочитати і сприйняти їх.

Наприклад, результати обчислень не повинні швидко зникати з екрану, тобто, до того, поки користувач їх прочитає. Одним із підходів до висвітлення інформації в зоні є висвітлення такого об'єму інформації, який може вмістити зона, а потім «заморозити» висвітлення до наступного запиту користувача (наприклад, до натискування клавіші із стрілкою і т. ін.).

Для економії місця на екрані деякі зони можна тимчасово анулювати і замінювати їх іншими, які необхідні в конкретній ситуації.

5. Економне використання ресурсів монітора

Такі атрибути монітора як мигання, збільшення яскравості та інверсне відео необхідно використовувати економно, оскільки їх надмірне використання може призводити до зворотного ефекту, тобто ускладнення або погіршення характеристик діалогу (надмірне витрачання часу, відволікання уваги від головного). Коректним є використання додаткових ресурсів монітора для притягнення уваги до важливих моментів, наприклад, до поля для вводу даних, екстрених повідомлень або до команд.

6. Спростуйте складні функції і скорочуйте об'єм введення даних з клавіатури за рахунок використання функціональних клавіш.

Функціональним клавішам можуть бути присвоєні, наприклад, такі функції:

- ЗАПУСК (START) програми або функції;
- ДОПОМОГА (HELP): відображення на екрані пояснюючого тексту;
- переміщення курсора на екрані;
- ВИХІД (EXIT) або ЗАВЕРШЕННЯ (TERMINATE) поточного сеансу діалогу;
- переривання поточної операції (ESCAPE), наприклад, вводу або виводу даних; така операція може означати «повторний запуск», якщо була зроблена серйозна помилка оператором;
- використання *комбінацій клавіш*; комп'ютер може мати спеціальне програмне забезпечення для розширення функцій клавіатури; воно

дозволяє присвоювати одній клавіші (або парі клавіш) функції виконання визначеної послідовності команд.

7. Створюйте специфікації значень або повідомлень «за замовчуванням», які спрощують введення даних користувач системи в процесі діалогу.

Наприклад, якщо необхідно ввести дату деякого замовлення, то інформаційний рядок може мати наступну форму:

ДАТА ЗАМОВЛЕННЯ? <СЬОГОДНІ?>

Для вводу поточної дати користувач натискає клавішу *Ввести* <Enter>.

8. Передбачайте помилки, які можуть мати місце з боку користувача системи.

Як правило, користувачі роблять помилки навіть при виконанні самих простих і очевидних операцій. Якщо дії користувача можуть спричинити потенційно небезпечну ситуацію, то він повинен знати про це. Наприклад, при спробі створити таку ситуацію, він отримує повідомлення:

ВИ ВПЕВНЕНІ, ЩО ТАКА ДІЯ НЕОБХІДНА?

Навіть такі прості попередження відіграють значну роль у попередженні потенційно небезпечних дій користувача.

5.4. Тональність діалогу та термінологія

Ще однією важливою характеристикою діалогу реального часу з точки зору людського фактору є *загальна тональність діалогу та термінологія*, яка використовується при цьому. Діалог повинен бути дружнім по відношенню до користувача, а тональність діалогу повинна ґрунтуватись на таких загальних рекомендаціях:

– *Використовуйте прості, граматично правильні речення.* При цьому краще використовувати розмовні фрази, але не говірки (сленг) або загальноприйняті вирази.

– *Не старайтесь жартувати або вживати гострі фрази.* Якщо хтось буде користуватись вашою системою 50 разів на день, то гумор, наявний в діалозі, буде сприйнятись так само як один і той же жарт 50 разів підряд.

– *Не вживайте фраз, які звучать як повчання для користувача.* Наприклад, не потрібно хвалити користувача або говорити, що він ще не все знає з того, що необхідно (це фрази такого типу: О, ВИ РОБИТЕ ВСЕ ПРАВИЛЬНО або ВИ ПОВИННІ ЗНАТИ ЦЕ КРАЩЕ).

Що стосується **термінології**, то фахівці з проектування комп'ютерних систем рекомендують притримуватись таких правил:

- *Не використовуйте комп'ютерний жаргон.*
- *Уникайте аббревіатур*, тобто не використовуйте ВД замість ВВЕДІТЬ ДАНІ.
- *Уникайте символів, які можуть бути невідомими для користувача; наприклад, математичних позначень там, де в цьому немає крайньої необхідності.*
- *Використовуйте просту і зрозумілу термінологію.* Наприклад, краще використати фразу ПОМИЛКА ВВОДУ ДАНИХ З ДИСКА ніж ПОМИЛКА ЗОВНІШНЬОГО ПРИСТРОЮ.
- *Використовуйте одні й ті ж терміни для виконання однакових операцій.* Наприклад, не потрібно використовувати терміни РЕДАГУВАННЯ ДАНИХ і МОДИФІКАЦІЯ ДАНИХ в одному й тому ж значенні.
- *Команди і пояснення користувачу повинні бути продумані і коректно сформульовані з правильним вживанням дієслів.* Краще сказати ВИБЕРІТЬ МЕТОД РОЗРАХУНКУ ПАРАМЕТРІВ замість ВКАЖІТЬ НА МЕТОД. Що стосується дій з клавіатурою, то необхідно говорити НАТИСНІТЬ КЛАВІШУ <F6>, а не СКОРИСТАЙТЕСЬ КЛАВІШЕЮ <F6>.

5.5. Використання кольорів, мигання і клавіатури

Кольори використовують для таких цілей:

- «піднімання» конкретних повідомлень;
- для зображення кластерів даних;
- виділення цифрових даних, графіків або областей екрану.

Сучасні монітори забезпечують можливості використання досить широкого набору відео- та аудіо-атрибутів, наприклад таких:

- подвійна яскравість вибраних полів екрану або окремих повідомлень;
- мигання вибраних полів або повідомлень;
- приховування відображення вибраних полів (наприклад, паролю);

– інверсне відображення вибраних полів, повідомлень або областей екрану.

Використання клавіатури. Клавіатура містить досить широкий набір функціональних клавіш (F1, ..., F12), Alt (alternative), Esc (escape) та інші. Ці клавіші можна використати для ініціалізації виконання деяких загальних операцій, які повторюються в процесі взаємодії з СППР. Наприклад, START, HELP, PAGE UP, PAGE DOWN, EXIT. В деяких стандартах є обов'язковим використання клавіші F1 для реалізації функції отримання допомоги (HELP). Очевидно, що одні й ті ж функціональні клавіші повинні завжди використовуватись для однієї мети.

Вміле поєднання вказаних атрибутів дозволяє суттєво підвищити швидкість сприйняття та глибину розуміння результатів роботи СППР, прискорити процеси вводу/виводу даних.

Контрольні задачі і запитання

1. Сформулюйте вимоги до інтерфейсів інформаційних систем.
2. Поясніть, що таке адаптованість інтерфейсу?
3. До яких характеристик користувача може адаптуватись інтерфейс?
4. Наведіть основні характеристики інтерфейсів користувача СППР.
5. Вкажіть, які можливості повинен забезпечувати інтерфейсний модуль інформаційної системи.
6. Сформулюйте принципи формування інтерфейсу користувача.
7. Поясніть такі принципи побудови інтерфейсів, як принцип мінімізації затрат ресурсів користувача та принцип незбитковості.
8. Які існують напрями розвитку науки про людський фактор?
9. Розкрийте принципи проектування інтерфейсу користувача на засадах людського фактору.
10. Наведіть ситуації, в яких система повинна надавати пояснення для користувача.
11. Зазначте особливості форматування екрану монітора при розробці інтерфейсів користувача.
12. Опишіть аспекти вибору тональності та термінології при реалізації інтерфейсів користувача.
13. Які недоліки і переваги має інтерфейс на основі натуральних мовних конструкцій?

14. Наведіть приклад застосування інтерфейсу на основі безпосереднього маніпулювання графічними об'єктами?
15. Де застосовуються інтерфейси на основі заповнення стандартизованих форм?