

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТЕРНОПЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

ОПОРНИЙ КОНСПЕКТ ЛЕКЦІЙ

з дисципліни

ЛЮДИНО-КОМП'ЮТЕРНА ВЗАЄМОДІЯ

для студентів напрямку підготовки:
Комп'ютерні науки

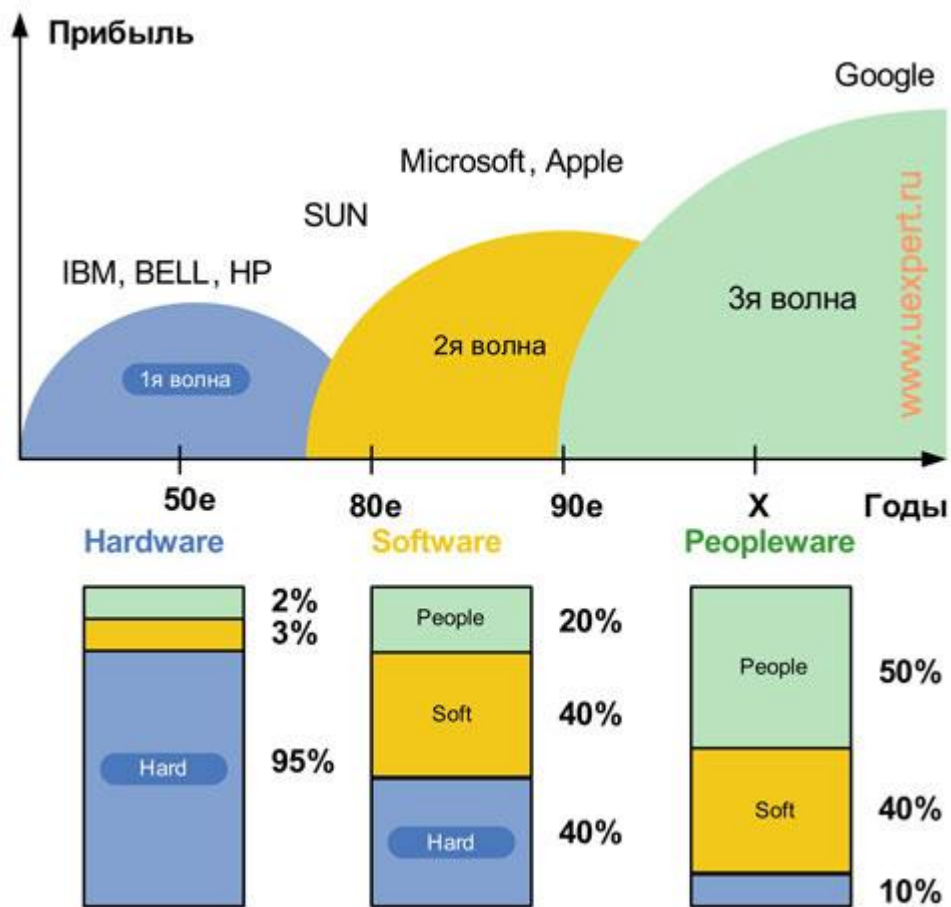
ТЕРНОПЛЬ – 2016

ЗМІСТ

Вступ	4
1. Психологічні принципи людино-машинної взаємодії	7
2. Аналіз, проектування та прототипування людино-машинного інтерфейсу	19
3. Функціональні компоненти та властивості людино-машинного інтерфейсу	25
4. Засоби розробки людино-машинного інтерфейсу	31
5. Оцінювання якості людино-машинного інтерфейсу	39
Література	41

ВСТУП. ІСТОРИЧНІ АСПЕКТИ ЛЮДИНО-МАШИНОЇ ВЗАЄМОДІЇ

У 40-60-і роки ХХ століття, коли лише з'явилися перші комп'ютери, велика частина коштів вкладалася в «залізо», і зовсім мало – в «софт» і людський чинник. Це була перша хвиля комп'ютерної індустрії. Компанії, які вчасно відчували віяння часу і вклали свої гроші в створення перших комп'ютерів, отримали великі дивіденди. На цій хвилі зійшли такі компанії, як IBM, Bell і Hewlett-Packard. В цей час важливо було створити інструмент для вирішення комбінаторних завдань і складних обчислень. Комп'ютери були розміром з двоповерхову будівлю і ними користувалися наукові інститути, військові і уряд.



Ілюстрація трьох хвиль розвитку комп'ютерної індустрії

У 70-80-і роки «залізо» стало дешевшати, і з'явилися персональні комп'ютери. Найбільш актуальним став розвиток програмного забезпечення, націлений на автоматизацію виробничих і особистих завдань людини. Саме на

другій хвилі комп'ютеризації нестримно вирости такі компанії, як Microsoft, Sun і Apple. Вони вірно уловили тенденції розвитку комп'ютерної індустрії і вклалися в створення «софтвера».

Починаючи з 90-х років і «залізо» і «софтвер» стали ще доступнішими, і на перший план вийшов «людський чинник». Фахівці відзначили, що в цей час почалася третя хвиля розвитку комп'ютерних технологій, коли вигравати стали продукти, що володіють властивостями простоти, зрозумілості і зручності використання. На цій хвилі усесвітнього успіху добилися такі компанії як Google і Apple.

Зараз, коли рівень конкуренції в ІТ - секторі досить високий, виграють в першу чергу ті продукти, які не лише представляють багатий список функціональності, але дозволяють користувачам досягати їх цілей максимально простим, зрозумілим і зручним способом. Саме з цим пов'язаний бум проектування інтерфейсів з врахуванням вимог юзабіліті на Заході. І хоча Україна в даному секторі відстає від західного світу, багато передових вітчизняних компаній також вже зрозуміли тенденцію часу. Вони почали активно упроваджувати в свої бізнес-процеси методи проектування ефективної взаємодії з користувачами.

Алан Купер, визнаний світовий експерт в питаннях створення інтерфейсів, в своїй книзі “Про інтерфейс” також говорить про те, що при створенні продуктів важливо враховувати не лише кінцеву мету користувачів, але і їх емоційні цілі. Емоційні цілі виражають те, як людина хоче себе відчувати, працюючи з продуктом. Коли продукт заставляє користувачів відчувати себе безглуздо або незручно, їх самоповага і продуктивність праці знижуються, а незадоволеність зростає. Варто трохи переборщити з таким відношенням до користувачів – і вони скористаються першим же шансом, аби позбавитися від цього продукту. Такий продукт потерпить невдачу – незалежно від того, наскільки добре він дозволяє користувачам досягати всіх інших цілей.

Дональд Норман, ще один відомий юзабіліті-спеціаліст, професор когнітології і психології, автор книги “Емоційний дизайн”, в результаті своїх досліджень прийшов до виводу, що для людини цінність багато в чому визначається не лише корисним результатом, але і емоційним компонентом взаємодії.

Отже, «емоційний дизайн» або, якщо коректно перекласти термін «дизайн» на українську мову, «емоційне проектування» – це проектування

продуктів, які не лише дозволяють користувачам ефективно досягати кінцевої мети, але і викликають позитивні емоції при взаємодії з ними.

Це означає не лише забезпечити наявність корисних функцій і зручність їх використання, але і позитивне емоційне забарвлення взаємодії людини з комп'ютерним продуктом. Взаємодія людини з комп'ютером так само, як і взаємодія людини з людиною, має на увазі емоційно-психологічні аспекти – чекання один від одного, здібність до сприйняття інформації, що поступає від протилежної сторони, емоційний фон спілкування. Ви напевно неодноразово бачили, як люди розмовляють зі своїм комп'ютером, умовляють машину завестися або свій модем не переривати з'єднання при передачі важливого файлу. Люди чекають від складних технічних пристроїв поведінки, властивої людині, – чекають розуміння, турботи і навіть співчуття.

Тому для створення успішного комп'ютерного продукту необхідно потурбуватися про три важливі складові: як технічно забезпечуватиметься функціонування продукту, як передаватиметься інформація (за допомогою візуальної мови, втіленої в інтерфейсі) і як сприйматиметься інформація живою людиною-користувачем.

За кожною програмою і веб-сайтом теж стоять люди – команда розробників. Але оскільки вони не спілкуються безпосередньо з користувачами, необхідно на етапі посередництва (яке відбувається через інтерфейс) спроектувати таку поведінку програми, яка викликатиме позитивний емоційний відгук у користувачів.

1. ПСИХОЛОГІЧНІ ПРИНЦИПИ ЛЮДИНО-МАШИНОЇ ВЗАЄМОДІЇ

1.1 Класифікація людино-машинних систем

Структура системи людина-машина-середовище (СЛМС) складається з:

- машини (М) – все те, що штучно було створене руками людини для задоволення своїх потреб (технічні пристрої, інформаційне забезпечення і т.д.);
- людини (Л) – оператора, який при взаємодії з машиною виконує певні функції керування для досягнення поставленої мети;
- середовища, яке умовно можна розбити на два види – навколишнє середовище (НС) і соціальне середовище (СС).

Навколишнє середовище характеризується такими основними параметрами: мікроклімат, шум, вібрація, освітленість, заповишеність, загазованість і т.д.

Соціальне середовище характеризується соціально-економічними і політичними відносинами в суспільстві.

Людина і машина, при своїй взаємодії, складають підсистему в рамках СЛМС, яка називається система людина-машина – СЛМ.

Оснору класифікації СЛМ складають чотири групи ознак:

1. цільове призначення системи;
2. характеристики людської ланки;
3. тип машинної ланки;
4. тип взаємодії компонентів системи.

За цільовим призначенням СЛМ діляться на:

- керуючі, в яких основною задачею людини є керування машиною;
- обслуговуючі, в яких задачею людини є контроль за станом машини;
- навчальні – вироблення у людини певних навиків;
- інформаційні – пошук, накопичення або отримання необхідної інформації;
- дослідницькі – аналіз тих або інших явищ.

За характеристиками людської ланки СЛМ діляться на:

- моносистеми, до складу яких входить одна людина;
- полісистеми, до складу яких входить цілий колектив і взаємодіючий з ним комплекс технічних пристроїв.

Полісистеми можна розділити на паритетні та ієрархічні (багаторівневі).

В паритетних системах між членами колективу немає підлеглості та пріоритетності. В ієрархічних СЛМ встановлюється організаційна або пріоритетна ієрархія взаємодії людини з технікою.

Діяльність людини-оператора є процесом досягнення поставлених перед СЛМ цілей.

Розрізняють декілька типів операторської діяльності:

- оператор-технолог – людина безпосередньо була включена в технологічний процес;
- оператор-маніпулятор – основна роль діяльності людини це сенсомоторна регуляція (керування маніпуляторами, залізничним складом і т.д.);
- оператор-спостерігач – класичний тип оператора (диспетчер транспортної системи, оператор станції радіолокації і т.д.);
- оператор-дослідник – дослідники будь-якого профілю;
- оператор-керівник – організатори, керівники різних рівнів, особи приймаючі відповідальні рішення.

За типом машинної ланки умовно можна виділити два види ознак:

- інформаційні – машини, що забезпечують обробку інформації;
- матеріальні – машини, оброблювальні матеріальні носії.

По типу взаємодії компонентів системи в СЛМ виділяють два види:

- інформаційне – взаємодія, обумовлена передачею інформації від машини до людини;
- сенсомоторне – взаємодія, направлена від людини до машини для виконання поставленої мети.

Майже у всіх областях життя в даний час взаємодія людини з технікою є звичайним процесом. Назвемо деякі області використання таких соціально-технічних систем:

1) керування процесами крупних технічних об'єктів: технологічні системи, енергетичні системи, комунікаційні системи;

2) навігація, керування залізничним, наземним, повітряним транспортом, та ін.;

3) медико-технічні установки, військово-технічні, космічні, підводні і робототехнічні системи.

Стимулом для автоматизації багатьох систем звичайно є те, що відповідні функції можуть бути реалізовані машинним шляхом дешевше, швидше,

точніше, надійніше, безпечніше. Часто береться також до уваги, що автоматизація складних систем спрощує маніпулювання такими системами. Проте, за сучасним станом техніки, не всі можливі функції піддаються автоматизації, тому ще неможлива комплексна автоматизація в повному обсязі. Багато складних підфункцій керування об'єктами все ще залишаються в ручній обробці.

Функції в людино-машинних системах повинні виконуватися машинним або ручним способом. Автоматизація функцій має обмеження як із сторони вживаної техніки, так і нетехнічних факторів. Ручне керування має обмеження, обумовлені можливостями людини.

Обмеження автоматизації. В принципі всі задачі, які мають точне визначення, можуть бути автоматизовані. Проте, на сьогодні неможливо повністю виключити людину. Це обумовлено в основному наступними причинами:

1. Недостатньою автоматизацією збору інформації. Збір даних про ситуацію на окремих ділянках утруднений через нестачу сенсорики (наприклад, дорожнього руху), недостатньої надійності сенсорів (наприклад, нестабільності, дрейфу) або через велику мінливість параметрів стану (наприклад, вітру).
2. Недостатньою гнучкістю наявної автоматики. Сучасні автомати особливо добре підходять для використання за умови детермінованих обмежень. В невідомому, мінливому середовищі наявні планові методи часто недостатні для автоматичної роботи. Тут необхідна здібність до навчання і адаптації.
3. Недостатньою надійністю програмного забезпечення (ПЗ). В той час, як надійність апаратних засобів може бути підвищена надмірністю, безпомилкова робота комплексного ПЗ не може бути гарантована за всіма функціями.
4. Нетехнічні обмеження автоматизації. Можливими причинами є:
 - недостатня економічна ефективність;
 - відсутність соціологічних передумов (наприклад, ліквідація робочих місць);
 - відсутність психологічної прийнятності;
 - неясність в питаннях відповідальності при повністю автоматизованому виробництві або в питаннях втручань людини.

Обмеження людських втручань. Можливості людини при розпізнаванні образів і обробці зображень технічно реалізовані лише в окремих областях. Це стосується також і процесів прийняття рішень при неповній інформації і

граничних ситуаціях. При цьому людина на відміну від автоматики не поводиться раціонально. Більш того, вона здійснює інтуїтивно пошук рішень, які задовольняють статистичним критеріям, або проводить заміщення підлягаючого рішенню процесу асоціацією з деякою ситуацією, що відбулась раніше, для якої у неї є в пам'яті рішення по колишньому досвіду. Таке рішення, проте, можливо тільки на базі відповідної ментальної моделі поведінки системи.

Разом з названими здібностями людина як компонент системи виявляє істотні недоліки, такі, як:

- обмеженість пам'яті;
- відсутність ментальної моделі, або помилками в ній;
- обмежена надійність;
- недостатня постійність працездатності при тривалій роботі;
- упередженість і фіксованість;
- недостатня ментальна арифметика;
- обмежена смуга пропускання при ручних реагуваннях.

Інтерфейс користувача – видима частина системи керування і єдина, з якою реально взаємодіє людина. Його важливість виявляється в безпосередніх діях по керуванню складним технічним процесом. Оператори більше не повинні читати покази приладів і задавати опорні значення, безпосередньо контактуючи з керованим технічним процесом. Натомість, вони цілком відповідають за процес, стан якого вони спостерігають на екранах моніторів в центрі керування. Кількість персоналу зменшується, а відповідальність при цьому зростає. Інтерфейс користувача повинен сприяти підвищенню відповідальності, полегшувати роботу, знижувати стрес і зменшити імовірність завдання шкоди від людських помилок.

Користувач як і раніше добивається досягнення мети, але може зробити це тільки опосередковано, за допомогою технічної системи. Але і на саму технічну систему неможливо впливати прямо, а тільки за допомогою керуючого устаткування. Якщо керуюче устаткування або ЕОМ точно не відповідають технічній системі, то працювати користувачу складно, оскільки він повинен мати уявлення про саму технічну систему, інструмент керування і їх взаємодію. Керуюча система є "інструментом для використання інструменту" і не повинна вступати в суперечність з цілями процесу. Проте протягом достатньо довгого часу через неадекватні і недосконалі технології оператори нерідко повинні були

надавати більше уваги обчислювальним системам, ніж технічним процесам, якими ці системи повинні управляти. Звичайно користувач хоче бачити виконання цільових функцій і не бажає вникати в проблеми керуючого устаткування. Взаємодія всіх складових елементів системи повинна бути організована так, щоб користувач врешті-решт добивався мети.

Якщо є невідповідність між інструментом керування і керованою системою, наприклад через неправильний вибір сенсорів і виконавчих механізмів, то це приведе до великих навантажень на оператора або до складного інтерфейсу. Таким чином, вимоги технічної системи і знання про пізнавальні здібності людини повинні бути основою, на якій будується інтерфейс.

1.2 Фізіологічні особливості людини

Первинна інформація про стан зовнішнього середовища поступає людині за допомогою аналізаторів. Ця інформація називається сенсорною, а процес її прийому і первинної обробки – сенсорним сприйняттям.

Відчуття – процес, що полягає у відображенні окремих властивостей або явищ матеріального світу, а також внутрішніх станів організму при безпосередній дії подразників на відповідні рецептори.

Залежно від специфіки сигналів, розрізняють аналізатори:

- зоровий (рецептор очей);
- слуховий (рецептор вуха);
- тактильний, больовий, температурний (рецептори шкіри);
- нюховий (рецептор носової порожнини);
- смаковий (рецептори поверхня язика);
- внутрішні: тиску, кінестетический (рецептори в м'язах і сухожиллях), вестибулярний (рецептор в порожнині вуха), спеціальні, розташовані у внутрішніх органах і порожнинах тіла.

При розробці системи “людина-машина вважається, що до 80% всієї інформації людині поступає через зоровий аналізатор, близько 19%, через слуховий і лише 1% через тактильний.

До основних параметрів аналізаторів відноситься:

1. Абсолютна чутливість – мінімальне значення подразника, що викликає початкові відчуття.
2. Гранично допустима інтенсивність сигналу – больовий поріг сприйняття.

3. Діапазон чутливості – зона сприйняття сигналу від абсолютного до больового порогу.
4. Диференціальна чутливість – мінімальна зміна інтенсивності сигналу, що відчувається людиною.
5. Межі спектральної чутливості – абсолютні пороги відчуттів по частоті сигналу.
6. Диференціальна чутливість до зміни частоти сигналу – диференціальний, розрізняльний поріг по частоті.

Специфічною особливістю рецепторів людини є великий діапазон значень інтенсивності сигналів, в межах якого можливо ефективно функціонування аналізаторів, разом з досить високою диференціальною чутливістю до інтенсивності. Таке поєднання виявляється можливим завдяки системі адаптації і сенсibiliзації аналізаторів (пониження і підвищення їх чутливості залежно від середньої інтенсивності сигналів, що впливають протягом деякого часу).

Адаптація – властивість аналізаторів, що полягає в зміні чутливості під впливом їх пристосування до діючих подразників.

Німецький вчений Э. Вебер показав, що величина приросту інтенсивності, що викликає виразну різницю між двома стимулами, знаходиться в постійному відношенні до початкової інтенсивності. Г. Фехнер дав спостереженням Вебера математичний опис.

Закон Вебера – Фехнера: *відчуття людини - L пропорційні логарифму подразнення X:*

$$L = \lg \frac{X}{X_0} \quad (1)$$

де X_0 – поріг сприйняття інтенсивності подразнення людиною; X – інтенсивність подразнення.

Психологія – наука про людську поведінку, досвід і про відповідні розумові процеси. Багато результатів, отриманих в ході психологічних досліджень, напряду використовуються при розробці інтерфейсу користувача. Особливо важливі ті з них, які торкаються пізнання і сприйняття, тобто осягнення чого-небудь за допомогою відчуттів, – запам'ятовування і обробки інформації. Використовуючи ці можливості, людина вчиться, вирішує проблеми і планує майбутнє. Експериментальним шляхом можна зібрати інформацію про психологічні і пізнавальні процеси і побудувати відповідні моделі. Основи

теорії сприйняття і запам'ятовування розроблені і перевірені експериментально. Ця теорія дає фундамент для проектування інтерфейсу користувача.

При взаємодії з ЕОМ і іншим устаткуванням сприйняття є практично повністю зоровим або слуховим. Найважливішим відчуттям є зір. Важливість дотикової інформації виявляється тільки при використуванні деяких типів маніпуляторів (механічні виконавчі механізми, гальма, ручка управління літака), оскільки вони створюють свого роду зворотній зв'язок. З появою керованих ЕОМ силових приводів цей тип відчуттів зникає або замінюється на зорові образи, наприклад комбінацію ламп або символів на екрані.

Обсяг інформації, що поступає до людини ззовні, оцінюється в 10^9 біт/с, з яких лише близько 100 біт/с обробляються свідомістю. Мозок прагне зменшення кількості оброблюваної інформації. Якщо кількість інформації, що поступає в одиницю часу, зростає, здібність до обробки втрачається і увага концентрується тільки на її частині.

Як правило, більш високий рівень уваги відповідає несподіваному подразнику і знижується, якщо подразник повторюється. Інші чинники, які підвищують увагу, – це інтенсивність, розмір, контрастність і рух. Мозок чудово "фільтрує" образ або звук з набору кольорів і шумів. Наприклад, коли людина дивиться на картину, око прагне зосередитися на найважливіших деталях. Аналогічний ефект існує і для звуків – людина може одночасно чути декілька голосів, але лише один з них сприймає свідомо.

1.3 Модель людської поведінки

Модель поведінки дає спрощену основу для опису взаємодії між людиною і навколишнім середовищем. Відповідно до моделі Йенса Расмуссена людські дії з погляду прийняття рішень і поведінки можна розділити на три рівні (рис. 1.1).

Поведінка, що ґрунтується на навиках – автоматичні сенсомоторні дії, легко виконувані без свідомого контролю. Поведінка, що ґрунтується на правилах виявляється в знайомих ситуаціях і керується набором наперед відомих правил або процедур, сформованих на основі попереднього досвіду.

Очевидно, що індивідуум стикається не тільки із знайомими ситуаціями. В нових умовах, коли попередній досвід або правила відсутні, управління діями повинне відбуватися на ще більш високому рівні, на якому поведінка є цілеспрямованою і основою на знаннях. В цій ситуації найближча задача формулюється на основі аналізу навколишнього середовища і кінцевої мети. На

цьому рівні структура навколишнього середовища представляється уявною моделлю, від якої залежить вибраний спосіб дії.

символи **Поведінка, що ґрунтується на знаннях**

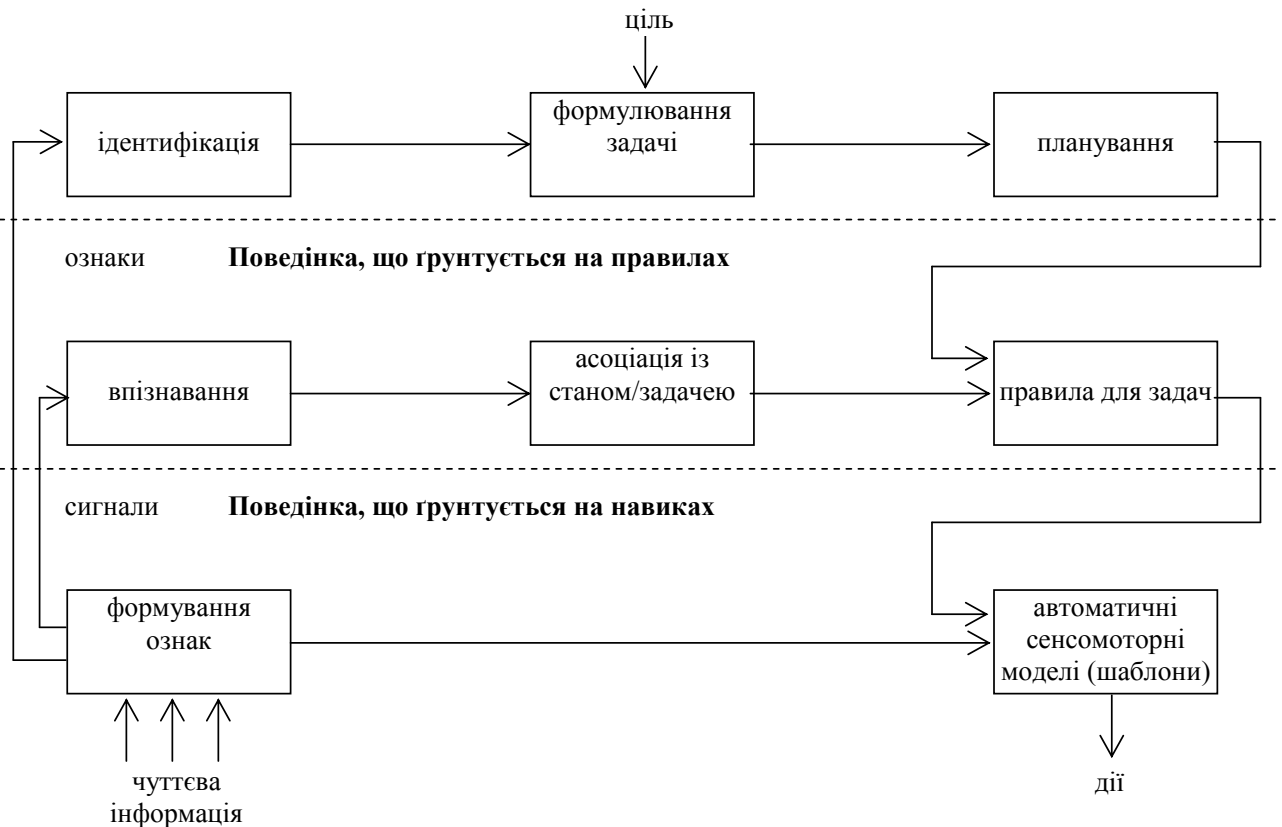


Рис.1.1. Спрощена ілюстрація рівнів людської поведінки – модель дії

Цю концепцію можна підтвердити рядом практичних прикладів. При навчанні складним функціям, наприклад керуванню автомобілем або грі на роялі, спочатку включається рівень знань. Перемикання швидкостей автомобіля або рух руки по клавіатурі вимагає значної концентрації. На цій стадії потрібна підвищена увага. У міру накопичення досвіду дії учня стають все більш і більш автоматичними – спочатку на рівні правил, наприклад учень відразу і без аналізу розуміє, коли потрібно переключити передачу, і потім виходять на сенсомоторний рівень, коли виконуються практично без зусиль свідомості. Для досвідченого водія зовсім нескладно оцінювати дорожню обстановку, переключати передачі і одночасно підтримувати розмову, оскільки перші дві дії виконуються на сенсомоторному рівні і не вимагають спеціальної уваги. Досвідчений музикант не прочитує ноту за нотою, а відразу розпізнає складніші об'єкти – фрази, інтервали, гамми, арпеджіо – і виконує їх відповідним чином.

Дії на самому нижньому рівні є найефективнішими, оскільки не вимагають обдумування в явному виді і реакція слідує відразу за подразником. Дії на рівні правил вибираються з декількох зразків, що зберігаються в пам'яті і, отже, виконуються з деякою затримкою. Для того, щоб визначити нову дію в незвичайній ситуації, необхідне аналітичне мислення і порівняння з наявними знаннями і досвідом, що звичайно вимагає більшого часу і значних розумових зусиль.

Поділ поведінки на три категорії – яке ґрунтується на навиках, правилах і знаннях – полегшує класифікацію задач і визначає, який тип підтримки користувача повинен забезпечувати інтерфейс на кожному рівні.

1.4 Теорія дворівневої пам'яті

Психологи тривалий час досліджували функції пам'яті і механізм запам'ятовування інформації і досвіду. Сучасна психологія розрізняє декілька пізнавальних функцій, тобто способів сприйняття мозком інформації про об'єкти, одержуваної від органів чуття. Ці функції (рис. 1.2):

- накопичення і зберігання інформації від органів чуття (sensory storage);
- короткочасна або оперативна пам'ять (short-term memory);
- довготривала пам'ять (long-term memory).

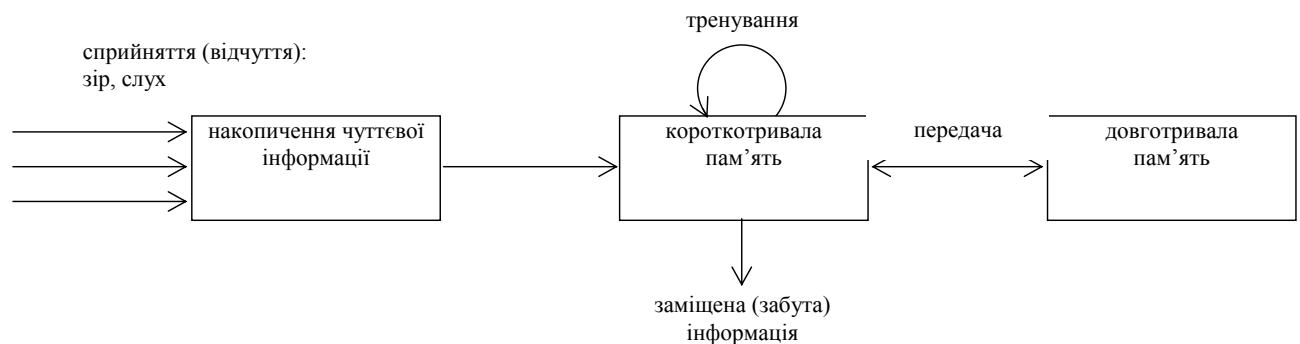


Рис.1.2. Модель дворівневої пам'яті

Етапи обробки інформації мозком – це сприйняття, запам'ятовування в короткочасній і потім в довготривалій пам'яті.

Інформація, накопичена органами чуття, потрапляє в короткочасну пам'ять, де мозок може свідомо звернути на неї увагу. З короткочасної пам'яті інформація передається в довготривалу, в більшості випадків свідомим зусиллям волі. Короткочасна пам'ять є тим, що ми називаємо свідомістю (consciousness). Її зміст це те, про що людина роздумує в даний момент і на

чому ґрунтуються її дії. Дослідження показали, що в короточасній пам'яті є місце приблизно для 7 ± 2 елементів інформації, які називають брусками (chunk). Брусок може бути дуже складним і змістовним. Нова інформація, що поступає, стирає або заміщає існуючі бруски. Інформація, яка не обдумується, швидко "розмивається", і зникає з свідомості. Образи в короточасній пам'яті мають приблизно один рівень абстракції. Інформація з короточасної пам'яті швидко витягується і так само швидко забувається, вона легко осяжна, і на її основі виробляються швидкі реакції.

Довготривала пам'ять володіє практично невичерпною місткістю, але запам'ятовування і відтворення інформації займає більше часу. Інформація в довготривалій пам'яті визначає повне знання індивідуума і містить все, від мовних навиків до дитячих спогадів, від таблиці множення до імені короля Рурітанії. Час зберігання інформації в короточасній пам'яті звичайно складає секунди, а в довготривалій пам'яті вона може зберігатися протягом всього життя.

Відмінність між короточасною і довготривалою пам'яттю має і психологічне пояснення. В мозку немає анатомічного розділення на області пам'яті – в роботі обох видів пам'яті бере участь весь мозок. Різниця полягає в типі процесу. Механізм короточасної пам'яті можна описати з позицій розподілу електричного поля, а довготривала пам'ять ґрунтується на взаємодії нейронів і зв'язках, що мають більш постійну, хімічну природу.

Людська пам'ять не працює на основі прямої адресації комірок, як ОЗП ЕОМ. Вона працює на основі аналогій та асоціацій. В цьому відношенні кодування запам'ятовуваної інформації, грає дуже важливу роль. Запам'ятовуються не форми або шаблони, а, поняття і асоціації. Іншими словами, – те, що запам'ятовується, – це значення, а не форма повідомлення або символів; наприклад, після прочитання тексту запам'ятовується зміст, а не дослівно речення або шрифт, яким текст надрукований.

Інформація запам'ятовується легше, якщо її можна співвіднести з інформацією, що вже зберігається в довготривалій пам'яті. Запам'ятовування різних фактів відбувається краще, якщо вони не розрізнені, а знаходяться в причинному взаємозв'язку. Аналогічно, відтворення (витягання) інформації полегшується асоціаціями – "підказками", – тим або іншим чином з нею зв'язаними. Є експериментальні підтвердження того, що інформація запам'ятовується назавжди. Не можна стверджувати, що забута інформація

втрачена остаточно, – вірогідніше, що її не вдається відтворити через невірні або втрачені асоціативні зв'язки.

Нову інформацію простіше сприймати, якщо її можна упорядкувати або пов'язати з вже існуючою в довготривалій пам'яті. Ключем до ефективного використання короткочасної пам'яті є кодування, тобто скільки первинній інформації вміщається в брусок. Розглянемо, наприклад, послідовність чисел 80442296306. Вона виглядає як випадковий набір з одинадцяти цифр. Більшість людей нездатна відтворити її без значних зусиль і, швидше за все, дуже швидко її забудуть. Проте, якщо представити цю послідовність у вигляді 8-044-229-6306, то сприймати її значно простіше – в ній легко визначити телефонний номер в Києві. Перегрупування зменшило число брусків з 11 до 4 – кількості, з якою можна справитися без великих труднощів. Більш того, абстрактні цифри пов'язані із знайомими поняттями – назва міста, номер телефону. Щось схоже відбувається з шахістами: майстер може запам'ятати позицію з 20 фігур, новачок – помітно менше. Причина в тому, що новачок бачить 20 різних фігур, а майстер – один або два бруски.

1.5 Помилки людини

Помилки завжди супроводжують будь-яку людську дію, а значить, при розробці інтерфейсу користувача необхідно передбачити спосіб боротьби з ними. Помилки можна розглядати як дії, які приводять до небажаного результату, якщо людина, що вчинила помилку, могла ухвалити інше рішення, відповідне ситуації і рівню своєї компетентності. Теоретично помилок можна уникнути. Неправильні дії, викликані ситуацією або недостатньою компетентністю для виконання роботи, не можна вважати помилкою.

Розрізняють два типи помилок – промахи і власне помилки. Промах (slip) – це неправильно реалізований вірний намір. Помилка (mistake) – правильно виконана дія на основі невірних передумов.

Промахи відносяться до сенсомоторного рівня, а помилки – до більш високого рівня моделі поведінки. Приклад промаху – друкарська помилка або включення не тієї передачі автомобіля. Помилка може відбутися на рівні правил – невірне написання слова – або на рівні знань – використання правильно написаного, але не відповідного контексту слова. При керуванні автомобілем помилкою є неправильна оцінка нахилу дороги, яка веде до включення передачі, що не відповідає ситуації.

Інтерфейс технічної системи повинен допомагати людині приймати рішення в ситуаціях, пов'язаних з помилками. Він повинен зменшувати число помилок і полегшувати їх наслідки. Боротьба з помилками складається з двох задач – попередження помилок і їх виправлення.

Попередження помилок (error avoidance) вимагає, щоб кероване устаткування тим або іншим способом розпізнавало помилкову ситуацію або навіть її передумови і попереджало про це оператора. Устаткування не повинно сприймати команди з можливими небезпечними наслідками.

Виправлення помилок (error correction) означає, що оператор або система виявляє помилку і намагається її виправити. Добре знайомий приклад з області ОТ – це команда "Відмінити" ("Undo"), яка використовується для того, щоб відмовитися від вже виконаної дії, тобто повернутися до попереднього стану. Система запам'ятовує результат виконання команди як якийсь тимчасовий, проміжний стан, при цьому демонструє його користувачу як новий. Наприклад, файл не знищується негайно після виконання відповідної команди – він лише позначається як видалений, а його знищення відбувається пізніше. Якщо користувач раптом схоче відмінити видалення файлу, він може це зробити протягом деякого часу після виконання команди, поки файл фізично не стертий.

Є очевидним, що команда "Відмінити" спрацює, якщо в процесі реально не відбулося ніяких змін. Проте, в системі управління не можна буферизувати виконання команд, як це робиться у віртуальному середовищі ЕОМ. Тому в системах моніторингу і управління помилки повинні виключатися – наскільки це можливо із самого початку, тобто система керування процесом повинна включати інформацію про можливі небезпечні стани або навіть процедуру, що моделює такі стани, для того, щоб наперед передбачити наслідки і необхідну протидію.

В складних системах не доцільно прагнути повністю уникнути всіх помилок – саме помилки є прекрасним джерелом досвіду. Не випадково досить важливий метод навчання називається методом "проб і помилок". Власний досвід дозволяє виконати більшість дій на сенсомоторному рівні, який неможливо замінити теоретичними заняттями.

Якщо експерименти та "ігри" з технічною системою небажані або просто неможливі, наприклад при керуванні ядерним реактором, процедури моделювання повинні допомогти набрати необхідний досвід без зайвого ризику. Пілот початківець, який кілька разів вривався в землю на тренажері, найімовірніше, краще керуватиме машиною в повітрі, ніж той, хто взагалі не знайомий з технічними можливостями і обмеженнями свого літака. Моделі постійно використовуються в електроенергетиці для того, щоб заздалегідь оцінити ефект від зміни потоків потужності і конфігурації, а також інших операцій, не піддаючи ризику реальну систему.

2. АНАЛІЗ, ПРОЕКТУВАННЯ ТА ПРОТОТИПУВАННЯ ЛЮДИНО-МАШИННОГО ІНТЕРФЕЙСУ

Уявні моделі мають вирішальний вплив на експлуатацію складної системи і розробку інтерфейсу користувача. При керуванні технічними процесами, які в більшості є складними системами, оператор приймає рішення на основі деякої уявної моделі. Уявну модель (mental model) можна визначити як набір абстрактних уявлень, що сформувалися в оператора, про те, як технічна система працює і реагує на команди. Поведінка на рівні знань припускає наявність уявних моделей.

Розробники користуються припущенням, що кожен оператор має в довготривалій пам'яті модель процесу, яким він керує. Відповідно до цього припущення інженери-технологи і оператори вивчають всі інструкції і креслення напам'ять і завжди знають, що робити для досягнення потрібного результату, тобто людина вважається системою попереджувального керування, яка знає, як в аварійній ситуації впливати на роботу устаткування.

На основі психологічних дослідів і аналізу взаємодії операторів з складними технічними системами були отримані наступні загальні закономірності:

- більшість людей в першу чергу цікавиться не тенденцією і характером розвитку процесу, а лише існуючою ситуацією і схильні реагувати тільки на поточні обставини, не плануючи свої дії в майбутньому;
- затримка реакції негативно позначається на керованості системи; люди звичайно не розпізнають нестійкість, викликану запізненням керуючої дії;
- у людей звичайно виникають труднощі, якщо зміни відбуваються по експоненціальному закону, вони, як правило користуються лінійною екстраполяцією;
- для людського мислення характерні взаємно-однозначні (одна подія спричинює іншу), а не множинні (складні поєднання подій спричинюють інші події) причинно-наслідкові зв'язки;
- люди прагнуть уникати труднощів, тобто займатися якнайменш проблемною і, відповідно, менш важливою діяльністю;
- люди прагнуть спростити проблеми, зводячи їх до мінімально можливого числа причин.

Формування уявної моделі – це тривалий процес, що супроводжується постійними змінами: додається нова і переосмислюється стара інформація.

Уявна модель процесу, властива поведінці на основі правил має формально-теоретичний характер, а необхідна інформація нагромаджується за допомогою вправ і будується головним чином на основних фактах з досвіду. На рівні правил модель складається із тверджень виду "якщо-то" стосовно передбачуваних режимів процесу. Модель сенсомоторного рівня є результатом досвіду і практики. Тип уявної моделі, залежить від типу керованої системи, наприклад, відчуття фізичного злиття з машиною, необхідне пілоту, надмірне для диспетчера електростанції.

Загалом, дії, виконувані на сенсомоторному рівні, є більш швидкими і набагато ефективнішими, ніж які вимагають будь-якого обдумування. Мислення високого рівня необхідне при зіткненні з новими ситуаціями, наприклад для з'ясування причин відмови якого-небудь устаткування і прийняття рішення про те, як вийти із ситуації. Уявні моделі, що відносяться до однієї і тієї ж технічної системи, можуть бути різними залежно від їх призначення. Уявна модель автомобіля у автомеханіка відрізняється від моделі, існуючої у автогонщика.

Керування складними системами в більшості випадків – це виконання наперед розроблених процедур. Штатні і нештатні ситуації описані в керівництвах оператора і в більшості випадків не вимагають, щоб оператор проявляв власну ініціативу. Керівництва оператора відображають підхід на основі правил.

Інший підхід до врахування уявних моделей – це аналіз інформації, необхідної для виконання задачі. Розрізняють синтаксичну інформацію (syntactic information), яка є формою уявлення, і семантичні знання (semantic knowledge), тобто зміст. Наприклад, географічна інформація включає назви світових столиць або висоту гір; географічні знання були пов'язані із здатністю орієнтуватися на місцевості і робити висновки про ландшафт або економіку регіону по карті.

Синтаксична інформація умовна, неструктурована і повинна заучуватися. Семантичне знання є проблемно-орієнтованим і має структуру, з якої можна вивести нові факти. В моделі поведінки, приведеній раніше, синтаксис був визначений на рівні правил, а семантика – на рівні знань.

Різницю між синтаксичною інформацією і семантичними знаннями необхідно враховувати при проектуванні системи керування, оскільки від цього залежать багато підказок і вказівок, які вона повинна видавати оператору.

Психологічні експерименти показали, що вивчення нової задачі простіше,

якщо вона по набору дій – синтаксично – схожа на іншу задачу, ніж у випадку, якщо аналогія є понятійною – семантичною. Аналогія, між діями на сенсомоторному рівні важливіша, ніж аналогія між цілями. Це пояснює успіх інтерфейсу Windows із стандартним набором символів для аналогічних операцій, незалежно від прикладної програми.

Синтаксична інформація базується на правилах, а значить, легко запам'ятовується, при регулярному використанні її можна використовувати негайно. Звідси слідує важливий висновок, що стосується інтерфейсу користувача, – існуючі способи представлення інформації і введення команд не повинні змінюватися, будь-які зміни повинні бути очевидними і зрозумілими.

Часто фахівцями вважаються люди, які знають деякі тонкощі процесу або властивості устаткування, тобто володіючі синтаксичною інформацією, або люди з хорошою сенсомоторною реакцією, а не ті, які уміють творчо вирішити проблему.

Системи автоматичного керування повинні, з одного боку, полегшувати роботу операторів, а з іншого – оберігати технічний процес від їх помилок. Існує прямий зв'язок між складністю системи і зусиллями, які тратяться на керування, тому, чим простіший інтерфейс, тим легше керувати. Тут необхідно досягти непрості рівноваги. Інтерфейс користувача ніколи не повинен наближатися або перевершувати межі людського сприйняття стосовно пам'яті і концентрації уваги. Безглуздо виводити на екран сотні характеристик процесу, покладаючи на користувача відповідальність за їх аналіз і висновки. Складність повинна бути зменшена до рівня, що не перевершує межу сприйняття. З другого боку "дуже простий" інтерфейс не дає оператору можливості набиратися досвіду і підтримувати навички, необхідні для роботи в непередбаченій ситуації.

В практичних задачах складність можна розглядати не як об'єктивну і вимірювану властивість технічної системи, а швидше як результат структури системи і уявлень про неї користувача. Це уявлення визначається способом взаємодії користувача з системою.

Складність фізичної системи може бути пов'язана з числом ступенів свободи або числом незалежних змінних, що описують систему, і числом керуючих змінних. Узагальнені параметри можуть спростити уявлення. У фізиці модель ідеального газу має різні рівні складності залежно від того, чи розглядається кожна молекула, описувана динамічними змінними, або маса газу в цілому, описувана термодинамічними параметрами на основі фізичних і статистичних властивостей.

Складність зростає, якщо деякі з параметрів системи зв'язані, але не очевидною залежністю. Дві незалежні змінні, наприклад температуру і тиск, легше спостерігати, якщо вони змінюються однаково, тобто збільшуються або зменшуються одночасно, ніж якщо вони змінюються непередбачуваним чином, як у випадку, якщо зростання температури іноді пов'язано із збільшенням тиску, а іноді і з його пониженням.

Технічний процес розглядається як складна система, а інтерфейс системи керування як інструмент, за допомогою якого складність можна регулювати. При такому підході немає необхідності явно посилатися на уявні моделі процесу, існуючі у операторів. Інтерфейс користувача якраз і є інтерфейс між технічним процесом і уявною моделлю оператора. Проблема полягає в тому, що всі оператори мають різні уявні моделі, і немає гарантій, що їх всіх можна задовольнити. Перевага підходу з позицій складної системи в тому, що оператор може вивчити і використати певний інтерфейс, при цьому деталі конкретної моделі неістотні.

Проблема взаємодії між людиною і машиною розглядається як проблема сприйняття при функціонуванні невідомої, часто досить складної, системи. Користувач слідує уявній моделі, яка більш менш відображає властивості реального технічного процесу. Як-правило, користувач в змозі керувати технічним процесом тільки до тих пір, поки його складність не перевершує здатності сприйняття.

Щонайпершою метою системи керування процесом є зниження рівня складності (complexity reduction) технічного процесу. Іншими словами, процес, "видимий" через інтерфейс автоматизованої системи керування, повинен бути простішим, ніж процес, спостережуваний за допомогою звичайних вимірювальних засобів. Автоматизація не повинна додавати складності, а обслуговування відповідного устаткування – приводити до перевантаження здатності операторів до сприйняття. Складна система керування погіршує керуваність процесу, і навпаки.

Другою важливою характеристикою разом з рівнем складності інтерфейсу є його адекватність (complexity matching). Вже на апаратному рівні інтерфейс повинен відповідати кількості і точності передаваних даних. Саме кількість і тип даних диктують вид інтерфейсу, а не навпаки.

Якщо процес характеризується всього лише декількома подіями протягом години, а число вхідних і вихідних змінних – мале, то для керування цілком підійде невеликий друкуючий пристрій або панель керування. Використання

для керування таким процесом складного інтерфейсу не зменшить складності, швидше навпаки, оскільки до складності самого процесу буде додана складність системи керування.

Важливим параметром при розробці інтерфейсу є межі відповідальності. Необхідно оцінити, наскільки оператор повинен слідувати розпорядженням і наскільки – приймати власні рішення. Інтерфейс повинен бути орієнтований на необхідну ступінь компетентності користувача (рівень правил або рівень знань) і на тип дій при виконанні роботи. Складний інтерфейс повинен розглядатися і з позицій необхідності навчання оператора роботи з ним.

Найприроднішою структурою є ієрархія. В більшості прикладних задач можна визначити, ієрархію так, що елементи з'єднані в структури, описувані спільними параметрами. Рівні ієрархії більш-менш відповідають класам рішень, що приймаються при керуванні процесом. Звичайно всі об'єкти, розташовані на одному рівні ієрархії, характеризуються інтенсивним взаємним обміном даними; обмін даними між рівнями, як правило, менший і не є критичним за часом.

Наприклад, в хімічному виробництві стан реактора характеризується десятками відстежуваних параметрів. Якщо увага сконцентрована власне на реакторі, тоді інтерес представляють величини опорних і поточних значень температури, витрати, концентрації і т.д. Якщо цей же реактор розглядається як елемент виробничого ланцюга, то істотні тільки вхідні і вихідні потоки та індикація штатного або нештатного режиму. Тільки коли система подасть сигнал про збій, має сенс проглянути більш детальну інформацію про реактор, щоб знайти його причину.

Ієрархічна структура спрощує спостереження за складним процесом, але приховує частину даних. Наприклад, перевищення вихідного сигналу регулятора в порівнянні із звичайним робочим значенням може служити показником неправильної поведінки якоїсь частини системи, але це ніяк не відображається на загальній панелі стану системи – всі лампочки горять зеленим – до тих пір, поки опорні і поточні значення параметрів однакові.

Ієрархічна модель є узагальненою і забезпечує корисний підхід до аналізу і структуризації системи керування.

Структурний підхід застосовний не тільки до виводу на екран, але і до командної мови. Команди можуть бути як дуже низького рівня – для безпосереднього керування кожним виконавчим механізмом, так і високого – для виконання цілих задач. Команди низького рівня рідко представляють

інтерес, особливо в складних установках, де елементи тісно були зв'язані і взаємодіють складним чином. Разом з командами низького рівня хорошим рішенням для виконання складних функцій є підготовка розвинених командних файлів, що складаються з набору простих "елементарних команд". Такі файли можна підготувати для всіх основних операцій.

Оснoву для проектування інтерфейсу дає аналіз цілей (task analysis). Це аналіз технічних процесів і операцій, які необхідні для керування. Аналіз цілей – це пошук відповідей на наступні питання:

- які стандартні задачі;
- як ці задачі виконуються;
- яка інформація потрібна для виконання цих задач;
- який ступінь свободи надається оператору у разі аварійних або непередбачених ситуацій (це не те ж саме, що дії в нештатних умовах, – якщо поведінка в таких випадках була регламентована, то це просто один з можливих режимів роботи).

Аналіз фізичних процесів, з одного боку, і пізнавальних здібностей операторів – з іншою, забезпечує підхід, на основі якого інтерфейс користувача визначається не тільки з апаратної і програмної точок зору, але і з урахуванням ролі, яку грають оператори.

3. ФУНКЦІОНАЛЬНІ КОМПОНЕНТИ ТА ВЛАСТИВОСТІ ЛЮДИНО-МАШИННОГО ІНТЕРФЕЙСУ

3.1. Обладнання для інтерфейсу користувача

В керуванні процесами звичайно застосовуються повноекранні термінали, що складаються з монітору і клавіатури, а також панелі з вимикачами, контрольних ламп.

Розповсюдження ПК і постійно зростаючі вимоги привели до помітного поліпшення якості моніторів. Розмір екрану характеризується довжиною його діагоналі, оскільки відношення ширини до висоти завжди однакове (4:3). Роздільна здатність монітора, вимірювана кількістю точок зображення, пікселів, пов'язана з розміром екрану. Роздільна здатність повинна забезпечувати хорошу якість представлення тексту і графіки. Монітор не обов'язково повинен бути кольоровим. З технічних причин якість і різкість зображення у монохромних моніторів краща, ніж у кольорових, тому перші можуть у ряді випадків бути дешевшими і кращими.

Важливим ергономічним показником монітора є швидкість регенерації екрану, тобто частота, з якою електронний промінь повністю обходить екран. Ця частота повинна бути не менше 75 Гц. Інший показник – це рівень випромінювання монітора. Визначення не цілком коректно, оскільки монітор насправді породжує електростатичне поле, електромагнітні хвилі і рентгенівське проміння. Сучасні монітори мають дуже низький рівень випромінювання.

Клавіатура – найпоширеніший пристрій введення. Буквені і цифрові клавіші мають скрізь однакове значення, за винятком деяких національних особливостей. Значення керуючих клавіш не стандартизовано.

Звичайні монітори і клавіатури сконструйовані для використання в офісі і, отже, не придатні для виробничих умов з високою вологістю, заповненою і вібрацією. Для використання в промислових умовах монітори і клавіатури випускаються в спеціальному міцному (промисловому) виконанні.

Недорогим і практичним пристроєм введення-виведення для спостереження і керування повільним процесом є друкуючий пристрій або телетайп. Він використовується в тих випадках, коли обмін інформацією проходить поволі (1–2 події в хвилину) і кожне повідомлення є закінченим, тобто не пов'язане з іншою інформацією, і користувач не повинен дуже довго чекати на завершення друку, щоб отримати повну картину ситуації. Перевага

друкуючого пристрою в тому, що носій інформації (папір) можна зберігати в архіві як протокол без якої-небудь обробки.

Панелі (щити) керування були поширені в основному в 1950-70 роки, коли пристрої інтерфейсу процесу підключались кожен до свого індикатора і пристрою керування, розташованого в єдиному центрі. Панелі керування мали індикатори у вигляді ламп і стрілкових приладів для виводу інформації і вимикачі або цифрову клавіатуру як пристрої введення. Панелі керування можна використовувати при досить обмеженій кількості вхідних/вихідних даних і жорстко визначених командах. Якщо панелі керування використовуються в основному як пристрої відображення, їх називають мозаїчними або імітаторними панелями.

Мозаїчні панелі застосовуються в даний час для відображення процесів із значною просторовою протяжністю, наприклад залізничних колій або трубопроводів, оскільки у багатьох випадках розглядати таку систему по частинах на екрані монітора неприродно.

Панелі керування обладнуються візуальними і звуковими індикаторами аварійних станів, які звичайно керуються ЕОМ. Вони використовуються, щоб негайно привернути увагу оператора, навіть якщо він був зайнятий чимось іншим. Такі пристрої повинні обов'язково мати швидку і доступну команду скидання, щоб надмірно не дратувати оператора після того, як він прийняв аварійний сигнал.

Вказівні пристрої прямої дії – світлові пера і сенсорні екрани – особливою популярністю поки не користуються. Переміщення руки від робочого столу до екрану і назад втомлює, вимагає часу, а точність руху обмежена. Вибір певної точки на екрані пальцем або світловим пером вимагає більше часу, ніж за допомогою непрямого керуючого пристрою – мишки або трекбола. До того ж в першому випадку є ефект "відбитка пальця". Світлове перо і сенсорний екран корисні у випадках, коли використання повної клавіатури невиправдано або неможливо, а діалог можна організувати у виді невеликого набору меню з обмеженим числом альтернатив. Сенсорні екрани досить популярні в системах керування верстатами і в загальнодоступних інформаційних системах, наприклад в банкоматах або довідкових кіосках.

Миші, трекболи і джойстики є вказівними пристроями непрямої дії для керування положенням маркера або курсора на екрані терміналу. Вибраний об'єкт або поле підсвічуються, і в результаті виникає негайний візуальний зворотний зв'язок. Вибраний варіант підтверджується натисненням клавіші,

наприклад клавіші мишки. Точність позиціонування набагато вища, ніж у сенсорного екрану або світлового пера.

Виробники ОТ вклали великі кошти в розвиток систем розпізнавання мови. Доступні сьогодні системи можуть розпізнати слова, які вимовляються роздільно і відповідають заздалегідь записаним зразкам. Доти поки ЕОМ не зможе декодувати нормальну мову з різними інтонаціями і акцентами, особливо в несприятливих умовах виробництва з гучними сторонніми шумами, використання інтерфейсів з розпізнаванням мови буде, швидше за все, обмежено.

Синтез мови технічно простіший, ніж розпізнавання, і на ринку пропонуються відповідні продукти. Недоліком таких систем є те, що їх повідомлення вимовляються раптово, коли користувач їх не чекає і, отже, може не звернути на них увагу.

3.2 Загальні принципи проектування

Проектування діалогу визначає спосіб, яким система спонукає користувача вводити інформацію і впливає на все різноманіття керуючих функцій. Діалог повинен бути влаштований так, щоб бути корисним користувачу і не завантажувати його додатковою роботою, пов'язаною з особливостями системи.

Проектування системи "людина – ЕОМ" ґрунтується на вивченні діяльності користувача (рис. 1).

Задачею проектувальника є визначити концептуальний образ системи, що відповідає задачі і типу користувачів, потім сконструювати її так, щоб образ системи привів користувача до відтворення такої моделі, яка відповідає концептуальній моделі системи у проектувальника.

Класифікаційна схема принципів проектування складається з таких основних розділів:

- організація даних (структуризація інформації на екрані дисплея при роботі в інтерактивному режимі);
- режими діалогів;
- пристрій введення;
- організація зворотного зв'язку і виправлення помилок;
- захист даних і запобігання аварійних ситуацій (ненавмисне знищення файлу або передчасне припинення сеансу роботи з системою);
- багатокористувацький режим роботи.

Аналіз	Проектування	Оцінка
Можливості, здібності і навички людини, які використовуються в інформаційній діяльності	Розподіл функцій в системі людина – ПК	Стресові і напружені ситуації
Робочі задачі	Функціонування системи людина – ПК	Облік помилок, якості роботи і часових обмежень
Поточні і прототипні методи	Представлення інформації	приймальні випробовування у діяльності кінцевого користувача
	Керування діалогом	

Рис. 3.1 Вивчення діяльності користувача

Організація діалогового режиму припускає такі основні типи взаємодії:

- режим форматowanego введення за допомогою заповнення форм;
- машинний запит;
- вибір з меню;
- командні мови;
- формальні мови запитів;
- обмежена природна мова.

Перші три діалогові режими керуються головним чином ЕОМ, тоді як три інших – користувачем (рис. 2).

4. Основну увагу слід приділити взаємодії компонентів системи та формату і змісту повідомлень і команд.

Три основні принципи справедливі для будь-якого прикладного або функціонального проектування і, отже, для інтерфейсів користувача. Це – простота, наочність і послідовність.

Простота.

Простота – найважливіший принцип для всіх видів проектування. В задачах керування процесами, в яких використовуються ЕОМ, кожна картинка на дисплеї відображає модель фізичного процесу і його роботу. Простота

означає, що разом з важливими даними не виводиться даремна, неістотна або надмірна інформація. З другого боку, простота зовсім не означає вбогість образотворчих засобів.

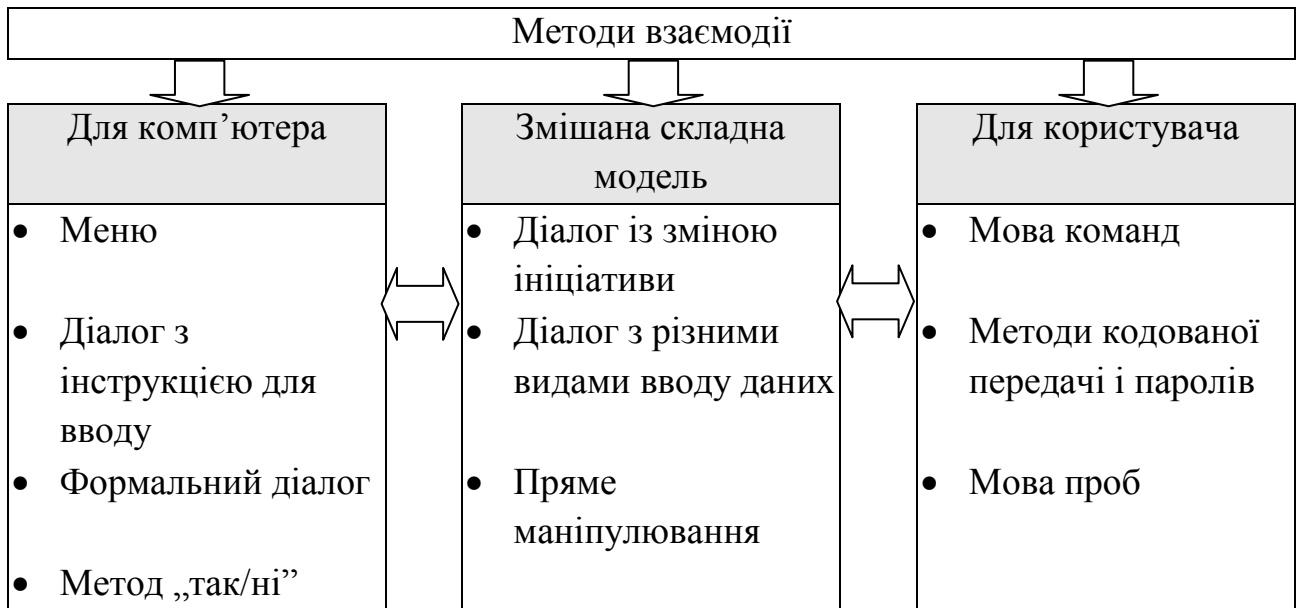


Рис.3.2 Методи взаємодії

Наочність.

Наочність – це ступінь прозорості функціонування системи. В ідеалі користувач повинен мати відчуття прямого контакту з технічним процесом, а не з автоматизованою системою керування. Наочність дозволяє пізнавати цілі і функції пристроїв по деяких візуальних образах інтерфейсу (колір, форма, вигляд). Наочність повинна забезпечувати зв'язок між технічним процесом, його режимами і уявною моделлю в користувача. Система керування повинна підтримувати і покращувати наочність керованого процесу.

Приклад простого і наочного відображення наведений на рис. 3.3. Очевидно, що немає необхідності звертати увагу на конкретну цифру або використовувати керівництво користувача, щоб зрозуміти, чи знаходиться контрольована величина в дозволеному діапазоні.

Сучасні засоби відображення вимірювальної інформації – результат декількох десятиріч розвитку. Стрілкові прилади дають негайну інформацію про відносну величину параметрів і тенденції їх зміни; цифрові прилади показують інформацію з більшою точністю, але вона не сприймається так само швидко.

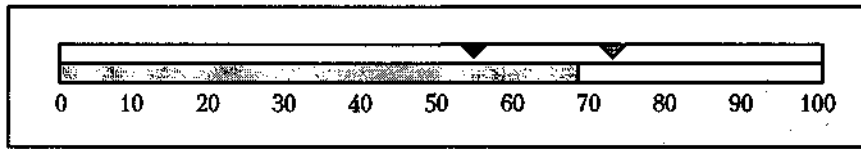


Рис. 3.3 Приклад інтуїтивного представлення інформації

В деяких системах наочність досягається за допомогою графічних метафор – піктограм, пов'язаних з повсякденним досвідом. Піктограма на екрані кодує операцію на основі очевидних асоціацій, наприклад, ручка позначає що-небудь, пов'язане із записами, а збільшувальне скло – пристрій масштабування для детального перегляду інформації. Зображення не обов'язково точно відповідає тому, що стоїть за ним в звичайному житті, воно створює асоціації між призначенням відомих об'єктів і схожими операціями в комп'ютерному середовищі – символи дають візуальні підказки про функції і операції.

Послідовність.

Послідовність означає, що для відображення однакових або аналогічних елементів системи застосовуються однотипні позначення. Для того, щоб опис або візуалізація системи були послідовними, спочатку необхідно розробити принципи структуризації. Послідовність можна розглядати як наочність, засновану на аналогії. Там, де наочність необхідна для первинного розуміння ідеї, послідовність допомагає перенести існуючі знання в новий контекст.

Послідовність найбільш важко реалізовувана характеристика інтерфейсу користувача. Для її досягнення необхідно класифікувати використовувані сутності, а потім застосовувати однакові правила (мова, скорочення, колір) для ідентифікації об'єктів кожного класу. Принцип послідовності, з іншого боку, вимагає, щоб кількість класів була зведена до мінімуму.

Послідовність важко дається у випадках, коли в проекті беруть участь декілька чоловік.

Труднощі в досягненні послідовності виявляються у разі, коли до певного уявлення застосовні декілька суперечливих правил і не очевидно, яке з них вибрати. Для подолання такої ситуації використовуються два підходи. Один полягає в тому, що кожний елемент зв'язується тільки з однією зоровою ознакою (величина, колір, форма і т. д.), так що різні ситуації можуть відображатися одночасно, наприклад, за допомогою розміру і кольору. Інше рішення полягає у введенні певної ієрархії, коли певний тип інформації перебиває інші.

Важливий спосіб забезпечити послідовність – використання стандартних інтерфейсів.

4. ЗАСОБИ РОЗРОБКИ ЛЮДИНО-МАШИННОГО ІНТЕРФЕЙСУ

Ергономіка включається в процеси розробки і тестування програмного продукту як частина системи якості. Розробка призначеного для користувача інтерфейсу (КІ) ведеться паралельно дизайну програмного продукту в цілому і в основному передує його імплементації.

Процес розробки ІК розбивається на етапи життєвого циклу:

1. Аналіз трудової діяльності користувача.
2. Побудова призначеної для користувача моделі даних і формування робочих місць.
3. Формулювання вимог до роботи користувача і вибір показників оцінки інтерфейсу користувача.
4. Розробка узагальненого сценарію взаємодії користувача з програмним модулем (функціональної моделі) і його попередня оцінка користувачами і Замовником.
5. Коректування і деталізація сценарію взаємодії, вибір і доповнення стандарту (керівництво) для побудови прототипу.
6. Розробка макетів і прототипів ІК і їх оцінка в діловій грі, вибір остаточного варіанту.
7. Імплементація ІІ в код, створення тестової версії.
8. Розробка засобів підтримки користувача (призначені для користувача словники, підказки, повідомлення, допомога і ін.) і їх вбудовування в програмний код.
9. Usability тестування тестової версії ІК по набору раніше визначених показників.
10. Підготовка призначеної для користувача документації і розробка програми навчання.

Ергономічні цілі і показники якості програмного продукту

Застосування розробляється для забезпечення роботи користувача, тобто для того, щоб він за допомогою комп'ютерної програми швидше і якісно вирішував свої виробничі задачі.

З погляду ергономіки, найважливіше в програмі – створити такий призначений для користувача інтерфейс, який зробить роботу ефективною і

продуктивною, а також забезпечить задоволеність користувача від роботи з програмою.

Ефективність роботи означає забезпечення точності, функціональної повноти і завершеності при виконанні виробничих завдань на робочому місці користувача. Створення ІК повинно бути націлено на показники ефективності:

Точність роботи

визначається тим, в якій ступені розроблений користувачем продукт (результат роботи), відповідає пред'явленим до нього вимогам. Показник точності включає відсоток помилок, які зробив користувач: число помилок набору, варіанти помилкових шляхів або відгалужень, число неправильних звернень до даних, запитів і ін.

Функціональна повнота

відображає ступінь використання первинних і оброблених даних, списку необхідних процедур обробки або звітів, число пропущених технологічних операцій або етапів при виконанні поставленого користувачу завдання. Цей показник може визначатися через відсоток застосування окремих функцій в РМ.

Завершеність роботи

описує ступінь виконання виробничого завдання середнім користувачем за певний термін або період, частку (або довжину черги) незадоволених (необроблених) заявок, відсоток продукції, що знаходиться на проміжній стадії готовності, а також число користувачів, які виконали завдання у фіксовані терміни.

Послідовність дій і набір інструментальних засобів користувача в ІК повинні бути підлеглі технологічному процесу виконання виробничого завдання.

Не треба боятися складності системи, треба уникати такого інтерфейсу, який не відповідає алгоритму рішення призначених для користувача задач.

Необхідно ретельно продумати і усвідомити сценарій взаємодії програми з користувачем, привівши його до оптимальної (щодо розглянутих показників) системи виконання завдань, і реалізувати ІК відповідно до цієї системи.

Для того, щоб розібратися в технології рішення задач користувача, розробнику необхідно з'ясувати наступні моменти (досліджуючи діяльність користувача):

1. *Яка інформація необхідна користувачу для вирішення завдання?*
2. *Яку інформацію користувач може ігнорувати (не враховувати)?*
3. *Спільно з користувачем розділити всю інформацію на сигнальну, таку, що відображається, редаговану, пошукову і результуючу.*
4. *Які рішення користувачу необхідно ухвалювати в процесі роботи з програмою?*
5. *Чи може користувач здійснювати декілька різних дій (вирішувати декілька задач) одночасно?*
6. *Які типові операції використовує користувач при рішенні задачі?*
7. *Що відбудеться, якщо користувач діятиме не по наказаному Вами алгоритму, пропускаючи ті або інші кроки або обходячи їх?*

Продуктивність роботи відображає об'єм витрачених ресурсів при виконанні завдання, як обчислювальних, так і психофізіологічних.

Дизайн ІК повинен забезпечувати мінімізацію зусиль користувача при виконанні роботи і приводити:

- скороченню тривалості операцій читання, редагування і пошуку інформації;
- зменшенню часу навігації і вибору команди;
- підвищенню загальної продуктивності користувача, що полягає в об'ємі оброблених даних за певний період часу;
- збільшенню тривалості стійкої роботи користувача і ін.

Скорочення невиробничих витрат і зусиль користувача -
важлива складова якості програмного забезпечення.

Для оцінки продуктивності використовуються відповідні показники, що перевіряються фахівцями з ергономіки в процесі usability тестування робочого прототипу.

Формування таких показників відбувається в процесі визначення вимог до ІК при вивченні наступних питань:

1. *Що від користувача потрібне в першу чергу?*
2. *Скільки інформації, що вимагає обробки, поступає користувачу за період часу?*
3. *Які вимоги до точності і швидкості введення інформації?*

4. *На які операції користувач витрачає більше всього часу?*
5. *Чим ми можемо полегшити роботу користувача при рішенні типових задач?*

Задоволеність користувача від роботи тісно пов'язана з комфортністю його взаємодії із застосуванням, і сприяє збереженню професійних кадрів на підприємстві Замовника за рахунок привабливості роботи на даному робочому місці.

Вимоги до зручності і комфортності інтерфейсу зростають із збільшенням складності робіт і відповідальності користувача за кінцевий результат. Висока задоволеність від роботи досягається у випадку:

- Прозорої для користувача навігації і цільової орієнтації в програмі. Головне, щоб було зрозуміле, куди йдемо, і яку операцію програма після цього кроку проведе.
- Ясності і чіткості розуміння користувачем текстів і значення ікон. У програмі повинні бути ті слова і графічні образи, які користувач знає або зобов'язаний знати по характеру його роботи або посади.
- Швидкості навчання при роботі з програмою, для чого необхідно використовувати переважно стандартні елементи взаємодії, їх традиційне або загальноприйняте їх розташування.
 - Наявність допоміжних засобів підтримки користувача (пошукових, довідкових, нормативних), у тому числі і для ухвалення рішення в невизначеній ситуації (введення за умовчанням, обхід «зависання» процесів і ін.).

Для оцінки необхідного рівня зручності інтерфейсу також використовуються спеціальні опитувальники, формуляри, листи чека, проте до даної роботи краще залучати фахівців з ергономіки.

Зручний інтерфейс допомагає користувачу справитися з втому і напруженням при роботі в умовах високої відповідальності за результат.

Проблеми, що виникають на етапі розробки прототипу GUI і варіанти їх рішення

1. Врахування особливостей пристроїв введення/виведення інформації, використовуваних користувачем, наприклад:
 - розмір екрану монітора
 - дозвіл екрану

- колірна палітра
 - характеристики звукової (якість відтворення мови) і відеокарти (швидкість виведення при анімації)
 - вид миші (з роликком або без)
 - тип клавіатури (“пряма”, “коса”)
 - необхідність додаткового обладнання (декодера штриха, світлового пера сенсорного екрану і ін.).
2. Специфіка інтерактивних елементів, пов'язана з вибором платформи, стандартних бібліотек:
- програмна організація введення/виведення інформації;
 - зміна і створення нових елементів форм (контролова);
 - придбання нестандартних бібліотек у інших фірм.
3. Вибір технології і методів ведення діалогу програми з користувачем:
- ступінь активності користувача при взаємодії (автоматичний режим або перехоплення управління програмою на себе, визарди, забезпечення доступу до всіх засобів інтерфейсу незалежно від дій користувача)
 - ступінь врахування ситуації (контекстні підказки, меню подальших подій або об'єктів, запам'ятовування типових шляхів діалогу)
 - відповідність очікуванням користувача (передбачення, передобробка, передформатування)
 - стійкість, терпимість до помилок користувача шляхом виправлення типових помилок
 - дублювання уручну окремих функцій системи і додаткові контрольні процедури роботи окремих режимів
 - настройка ПІ на різний рівень підготовки користувача (образність або метафорична наочної області на противагу скороченням і гарячим клавішам)
 - ступінь адаптивності ІК під переваги користувача (зміна способу і порядку відображення, перекомпоновування екрану, вибір окремих характеристик (стилю) і ін.)
 - настройка ПІ на специфіку завдання (новий формат даних, зміна набору об'єктів, доповнення атрибутів об'єктів).
4. Розміщення інформації і елементів, що управляють, в полі екрану, у вікні.
При композиції екрану необхідно враховувати обмежені розміри простору

екрану, у зв'язку з чим виникає завдання оптимального розташування максимально можливого об'єму інформації шляхом:

- логічною ув'язкою даних залежно від алгоритму роботи користувача, а не орієнтацією на структуру і послідовність фізичних таблиць даних
- визначення рівня “детальної – узагальненості” виведення інформації (знаходження компромісу між бажанням вивести багато записів одночасно і/або відразу побачити детальну інформацію по кожній з них)
- виділення важливої інформації на екрані
- чіткого визначення основних і допоміжних блоків інформації
- визначення статичних полів на екрані, а також полів, де інформація періодично змінюється
- уникнення вікон, що перекриваються, на екрані
- застосування принципів гармонії при компонованні екрану (симетрія, балансу мас, дотримання пропорцій, поєднання кольорів)

5. Формування зворотного зв'язку між користувачем і застосуванням:

- показ актуального стану системи, режиму роботи системи (автономного, штатного, захищеного і ін.) і режиму взаємодії (наприклад, відображення, редагування або пошук даних).
- виведення окремих, важливих для робочої операції даних і показників
- відображення дій користувача (натиснення клавіш, запуск процесу, динаміка виконання процесу, здобуття очікуваного і іншого результату)
- ясність і інформативність повідомлень системи.

6. Проектування панелей меню і інструментів (toolbars) і вибір пунктів

в них:

- логічне і смислове угруповання пунктів
- фіксована позиція панелей на екрані
- обмеження на ширину списку виборів і кроків (глибини) меню
- використання звичних назв, широко поширених ікон-пиктограм, традиційних ікон-символів і акуратне введення скорочень
- розміщення найбільш часто використовуваних пунктів (звичайно на початку списку).

7. Розробка засобів орієнтації і навігації:

- легкість визначення свого місцезнаходження і вказівка напряму проходження
- зручний перехід від узагальненого погляду до конкретних деталей (варіювання ступеня деталізації даних об'єктів)
- швидкий пошук в списку або таблиці
- вказівка на додатково існуючу інформацію і спосіб її отримання
- використання засобів перегортання і прокрутки.

8. Створення форм для введення даних:

- використання одного або декількох механізмів введення в рамках режиму (клавіатура, миша, штрих-декодер, світлове перо, ін.)
- визначення способів введення даних (таблиці, списки, проста форма, меню і ін.)
- мінімізація об'єму введення
- виділення редагованих обов'язкових і необов'язкових, а також нередатованих полів
- використання механізмів швидкого введення (за умовчанням, скорочення, з продовженням і ін.)
- Виділення введеної або відредагрованої інформації.

Принципи реалізації призначеного для користувача інтерфейсу

Стильова гнучкість

– можливість використовувати різні інтерфейси з одним і тим же застосуванням, на практиці реалізується у вигляді набору “skins”, для web-інтерфейсів – за допомогою таблиці стилів, зокрема можливість у виборі користувачем власних установок ІК (колір, ікони, підказки і ін.).

Сумісне нарощування функціональності

– можливість розвивати застосування без руйнування (тобто залишаючись в рамках) існуючого інтерфейсу.

Масштабованість

– можливість легко налаштовувати і розширювати як інтерфейс, так і саме застосування при збільшенні числа користувачів, робочих місць, об'єму і характеристик даних.

Адаптивність до дій користувача

– застосування повинне допускати можливість введення даних і команд безліччю різних способів (клавіатура, миша, інші пристрої) і багатоваріативність доступу до прикладних функцій (ікони, «гарячі клавіші», меню), крім того програма повинна враховувати можливість переходу і повернення від вікна до вікна, від режиму до режиму, і правильно обробляти такі ситуації.

Незалежність в ресурсах

– для створення призначеного для користувача інтерфейсу повинні надаватися окремі ресурси, спрямовані на зберігання і обробку даних, необхідних для підтримки користувача (призначені для користувача словники, контекстно-залежні списки, набори даних за умовчанням або по останньому запиту, історії запитів і ін.)

Переносимість

– при переході на іншу апаратну (програмну) платформу, повинен здійснюється автоматично перенесення і призначеного для користувача інтерфейсу, і кінцевого застосування.

5. ОЦІНЮВАННЯ ЯКОСТІ ЛЮДИНО-МАШИННОГО ІНТЕРФЕЙСУ

Тестуванням можна визначити слабкі місця інтерфейсу, але майже неможливо знайти сильні, оскільки вони користувачами просто не помічаються, і зовсім вже неможливо визначити способи поліпшення. Відбувається це через те, що суб'єкти тестування:

- не володіють всією необхідною інформацією про систему
- нічого не знають про проектування інтерфейсів
- їх мотивація істотно відрізняється від необхідної – замість того, щоб прагнути зробити хороший інтерфейс, вони прагнуть залишити в цьому інтерфейсі свій слід.

В той же час нерідко дизайнер інтерфейсу знає про предметну область менше, ніж майбутні користувачі.

Етапи тестування.

Постановка задачі.

Власне тестування. Технічно сеанс тестування досить простий. Потрібно мати декількох користувачів, які жодного разу не бачили поточного стану системи. За винятком окремих випадків, коли ваша система розрахована на просунутих користувачів (power user), потрібно підбирати не дуже досвідчених суб'єктів.

В більшості випадків достатньо пам'ятати наступне:

- Тестування на одному користувачі дозволяє знайти приблизно 60% помилок.
- Потрібно знаходити можливість залишити тестованого одного.
- Ніколи не переривати користувача. Ніколи не вибачатись за недосконалість тестованої системи. Ніколи не говорити „Ми потім це виправимо”. Нікого не звинувачувати. Викинути із назви тестування слово користувач, інакше користувач вирішить, що тестують його і боятиметься.

Методи тестування.

Перевірка через спостереження за користувачем. Користувачу дається завдання, він його виконує, його дії фіксуються для подальшого аналізу якою-небудь програмою запису стану екрану (наприклад Lotus ScreenCam). Щоб користувач не турбувався і не соромився, його краще всього залишити наодинці. Метод є корисним для виявлення неоднозначності елементів інтерфейсу. Оскільки кожна неоднозначність приводить до помилки, а кожна така помилка фіксується, знайти їх при перегляді записаного матеріалу дуже легко. Якщо заміряти час виконання завдання секундоміром, можна оцінити продуктивність

роботи користувачів. Також метод дозволяє порахувати кількість помилок користувача.

Мислимо вголос. Метод досить нестабільний, дуже залежить від балакучості тестованого користувача, але деколи дає цікаві результати. Відповідає перевірці за допомогою спостереження за користувачем, але тестованого при цьому просять усно коментувати свої дії. Потім коментарі аналізуються. Метод дозволяє легко визначити недоліки реалізації конкретних ідей інтерфейсу (невдало розташування елементів, погана навігація). Вимірювати швидкість роботи цим методом недоцільно.

Перевірка якості сприйняття. Тест дозволяє визначити, наскільки легко вивчити інтерфейс. Оскільки існує різниця між поняттями *бачити* і *дивитися*, а запам'ятовується тільки те, що побачено, необхідно щоб користувач побачив якщо не все, то вже хоча б все необхідне. А значить – запам'ятав, завдяки чому в майбутньому йому не доведеться обнишпорювати меню у пошуках „чогось такого, що, я точно знаю, десь тут є”. Сама по собі методика проста. Користувачу дається завдання, пов'язане з яким-небудь окремим діалоговим вікном, він його виконує. Через декілька хвилин його просять намалювати це вікно. Після чого малюнок порівнюється з оригіналом. Це один з випадків, коли спрацьовує обмеження на обсяг короткочасної пам'яті.

Тестування саме по собі має істотний недолік: якщо тестування проблем не виявило, значить воно було проведено даремно і потрібно його повторити. Якщо виявило, доведеться проблеми вирішувати. Таким чином, після тестування у дизайнера завжди додається роботи. Працювати ж, зрозуміло, не хочеться. Саме тому тестування несвідомо переносять на саме закінчення проекту, коли що-небудь виправляти вже пізно. Тоді як сама по собі ідея тестування зовсім інша. На самому початку роботи, коли тільки був створений прототип майбутньої системи, він тестується, після чого знайдені помилки виправляються. А потім прототип тестується знову. При цьому досвідченість дизайнера виявляється виключно в зменшенні кількості ітерацій. Відповідно, тестування повинне йти паралельно з рештою операцій.

ЛІТЕРАТУРА

1. Алан Купер, Роберт Рейман, Дэвид Кронин. Алан Купер об інтерфейсе. Основы проектирования взаимодействия.– Пер. С англ. – СПб.: Символ-Плюс, 2009. – 688 с.
2. Раскин Д. Интерфейс: новые направления при проектировании компьютерных систем, Пер с англ., - СПб: Символ-Плюс, 2006, -272 с.
3. Мандел Т. Разработка пользовательского интерфейса: Пер. с англ. –М.: ДМК Пресс, 2001, -431 с.
4. Человеко-машинное взаимодействие: теория и практика Учебное пособие / О.С. Логунова, И. М. Ячиков, Е.А. Ильина. -Ростов н/Д: Феникс, 2006. -285 с.
5. Норман, Дональд А. Дизайн привычных вещей.: Пер. С англ. – М. Издательский дом “Вільямс”, 2006.– 384 с.
6. Ачкурин Э.А. Человеко-машинное взаимодействие. Учебное пособие. – М.: СОЛОМОН-ПРЕСС, 2008. – 96 с.
7. Константайн Л., Локвуд Л. Разработка программного обеспечения: Пер. с англ. - Питер, 2004, 552 с.
8. Головач В.В. Дизайн пользовательского интерфейса. 146 с.
9. Жарков С.В. Shareware: профессиональная разработка и продвижение программ.– СПб.: БХВ-Петербург, 2002.– 320 с.
10. Людино-машинні системи автоматизації: управління якістю, безпекою і надійністю/ Архангельський В.І., Богаєнко І.М., Грабовський Г.Г., Рюмшин М.О.– К.: НВК “КІА”, 2000.– 296 с.

