

Лекция 5

Разработка интерфейса программного обеспечения

Вычислительную систему можно оценивать по ряду критериев:

- Точность
- Надежность
- Удобство

Интерфейс – способ, которым выполняется какая-либо задача при помощи специализированных средств. При проектировании интерфейса необходимо учитывать психофизические особенности человека. Психофизические особенности определяются моделью мозга.

Интерфейс пользователя

Ценности:

- Защита материальных ценностей: данные, программу и т.д.
- Интерфейсная защита от потери данных
- Сокращение времени на повседневные операции
- Принципы хороших интерфейсов
 - Простота ментальной модели
 - Доступность основных функций
 - Скорость работы
 - Обратная связь с интерфейсах
 - Минимизация ошибок при использовании интерфейсом
 - Информативность интерфейса

Критерии оценки пользовательского интерфейса

- Простота в освоении и запоминании операций
- Скорость достижения целей задачи, решаемой с помощью системы
- Устойчивость системы к ошибкам пользователя
- Адаптируемость
- Субъективная удовлетворенность

Принципы проектирования интерфейса пользователя

- Учет знаний пользователя
- Согласованность
- Минимум неожиданностей
- Способность к восстановлению
- Руководство пользователя
- Учет разнородности пользователей

Типы и формы диалога

- Типы диалога
 - Управляемый пользователем
 - Управляемый системой
- Форма диалога
 - Директивная. Командная строка
 - Фразовая. Пример: компиляция строчек программ
 - Табличная. Обычные юзер-формы

Процесс проектирования интерфейса пользователя

- 1) Изучение и анализ действий пользователя
- 2) Создание проекта прототипа интерфейса, макета
- 3) Оценка проекта пользователем
- 4) Разработка концепта
- 5) Создание детального дизайна интерфейса
- 6) Исполняемый прототип
- 7) Реализация интерфейса

Структуры взаимодействия пользователя с системой

- Вопрос – ответ. Недостатки: дотошности вопросов, которые пользователь вполне может игнорировать
- Прямое манипулирование. Интуитивно понятное взаимодействие. Довольно сложная реализация с точки зрения программиста.
- Интерфейс типа «Меню». Медленный вариант для опытных пользователей.
- Экранные формы. Очень просто вводятся данные, но занимает пространство на экране
- Командный язык. Мощный и гибкий, но труден для изучения
- Естественный язык. Подходит неопытным пользователям. Требуется наличие ИИ

Требования к пользовательскому интерфейсу:

Оптимальная плотность распределения информации на экране должна быть выбрана исходя из следующих правил:

- Необходимо оставлять пустым половину окна.
- Если вы работаете с таблицами, то необходимо оставлять пустой каждую 5-ю строку и 3-4 пробела между столбцами
- Использовать минимальное количество цветов, не более 5
- Использовать разные цвета для отображения изменения состояния в системе
- Использовать цветовое кодирование: выделение аномальных элементов красным цветом, максимально разрешенных в зеленый
- Не использовать цвета дополнительного цветового диапазона (фиолетовый, люминесцентно-желтый и т.д.)
- При написании текста использовать контрастные цвета.

Оценка эффективности заполнения экрана:

- Метод прямоугольников. Экран разбивается на участки, каждый из которых забивается текстом. Через центр экрана проводится ось, позволяющая оценить сбалансированность загрузки экрана, чем больше получилось маленьких прямоугольников, тем хуже.
- Метод выделенных точек. Позволяет определить область экрана, к которой максимально будет привлечено внимание пользователя. Чем больше точек находится в том месте.

При разработке интерфейса и согласовании его с пользователем необходимо его описывать. Для описания интерфейса могут использоваться различные методики. Одна из наиболее популярных методик – описание интерфейса с помощью сети переходов. Сеть переходов отражает набор состояний, в которых может находиться система и условий перехода из одного состояния в другое. На практике даже небольшие программные системы имеют обширную сеть перехода. Набор состояний – вершина сети, а на дугах условия. Для сокращения сети переходов и облегчения взаимодействия с заказчиком, выполняют агрегацию сети переходов:

- Из сети переходов удаляются вершины являющиеся не функциональными
- Если между двумя вершинами существуют несколько переходов по различным условиям, то переходы объединяются в одну дугу, а условие записывается через запятую.
- Если есть переход по умолчанию, то он заменяется подчеркиванием одного из условий

Для облегчения восприятия сети переходов строят таблицу вектора маршрутизации. Эта таблица содержит определение всех возможных переходов из всех возможных состояний системы. По строкам таблицы располагаются состояния системы. Столбцы:

- Exit. Действия, которые будут выполняться по умолчанию
- Jump. Определяются безусловные действия
- Оставшиеся столбцы – набор условий.

Для описания диалога необходимо определить синтаксис входных и выходных сообщений. Это определяется понятием грамматика. Чаще всего грамматика определяется набором правил продукционной системой. Эта система предназначена для хранения о предметной области в формате «если-то». Таким образом грамматика – набор правил, определяющих конструкции языка. Грамматика включает в себя:

- Терминальные символы. Т.е. основные атомарные символы из которых формируются сложные структуры.
- Не терминальный символ. Символ, используемый для составления составных конструкций.
- Правило вывода. Правило, по которым строятся не терминальные символы определяющие основные конструкции языка.

При проектировании интерфейса необходимо использовать средства поддержки пользователя. Средства поддержки пользователя делятся на 2 категории:

- 1) Сообщения системы. В качестве сообщений системы чаще всего используются сообщения о пользовательских ошибках. Основным требованием является максимальная однозначность и предоставление возможности пользователю устранить ошибку.
- 2) Справочная информация.

Факторы проектирования:

- Содержание. Указание на действий пользователя для их продолжения
- Опыт пользователя. В системе должны поддерживаться как для опытных, так и для начальных пользователей
- Профессиональный уровень. Должны использоваться термины, соответствующие программе и пользователю
- Стил ь общения.
- Культура. Учет культуры страны.

Для поддержки пользователя разрабатывается пользовательская документация. Документация пользователя должна содержать функциональное описание, документы по инсталляции, справочное руководство и руководство по администрированию. Каждый тип документов предназначен для конкретного типа пользователя.

Функциональное описание – набор тестов соответствующие ТЗ.