

# **Проектирование автоматизированных систем обработки и управления**

Гутгарц Римма Давыдовна

## **План занятий**

2 семестра. 1 – экзамен. 2 – зачет и курсовой проект.

Лабораторные – ТЗ, тех. проект, рабочее проектирование. Курсовая – предпроектная подготовка и эскизное проектирование.

Сдать самостоятельные и лабораторные до середины мая (13 мая).

## **Общие сведения**

Этапы проектирования:

- предпроектная подготовка
- эскизный проект
- техническое задание
- технический проект
- рабочее проектирование

Конечный результат – программный продукт.

## **Основные понятия проектирования**

### **Внутренние и внешние источники идей**

Внутренние источники идей:

- совершенствование существующих изделий
- выпуск побочной продукции
- организованное творчество в группе конструкторов и разработчиков
- система поощрения идей и предложение, исходящих от персонала

Внешние источники идей:

- изучение возможности и последствий конкуренции
- деятельность и новая продукция фирмы, вне сферы действия рынка сбыта
- идеи от существующих и потенциальных клиентов
- перечни изобретений и заявок на патенты
- консультации инженеров и научных работников
- университеты и частные лаборатории

- специализированные книги и журналы, интернет
- собрание профессиональных сообществ
- научные конференции

## Особенности творческого процесса

Причины, по которым не каждый может придумывать идеи:

- боязнь критики
- отсутствие гибкости мышления
- сила привычки
- узко практический подход
- чрезмерная специализация
- влияние авторитетов

Препятствия к творчеству организационного порядка:

- стремление к немедленному использованию идей
- стремление руководства указывать творческим людям что и как делать
- отсутствие долгосрочных целей
- неудача эксперимента
- частое изменение основных решений
- отсутствие контакта между проектировщиками и руководителями
- отрицательное отношение руководства ко всему новому, как следствие нежелание рисковать

## Процесс инженерного проектирования

1. Определение потребности (необходимости) качественно новой системы, в которой будут:

- отсутствовать недостатки, присущие настоящей системе
- присутствовать новые характеристики, приобретаемые системой по причине использования новых технологий

2. Определение целей

3. Устранять недостатки можно следующими путями:

- модификация (изменение внешних атрибутов системы)
- принципиальное изменение размеров (увеличение – укрупнение, удлинение, увеличение веса, прочности; уменьшение – компактность, легкость, удаление некоторых элементов)
- изменение внешнего вида (дизайн, эстетические аспекты систем)
- сохранение старого (новые технологии для старых конструкций)
- перестановка (замена одних элементов другими или изменение порядка)
- замена (замена материалов или формы системы)
- сочетание
- удобство обращения (изменение эргономических свойств)
- безопасность

4. Научные исследования

Связаны с анализом соответствующей информации.

5. Формулировка задания

Определение исходных данных, которые позволят достичь

поставленной цели.

6. Формирование самих идей

Идеи могут формироваться следующим образом:

- мозговой штурм
- синектика
- морфологический анализ
- деловые игры
- метод сценария
- метод экспертных оценок
- метод делфи (итеративная связь с этапом научного исследования и формулировка задания)

7. Выработка концепции (эскизное проектирование)

Изображение системы в укрупненном виде, без ее детализации, без выполнения инженерных расчетов.

8. Анализ

Проверка концепции на возможность ее физической реализации. При обнаружении несоответствия существующих знаний и технологий и теми, которые необходимы для реализации проекта, происходит итеративный переход к предыдущему этапу.

9. Эксперимент

Создание опытного образца и проверка его работоспособности в лабораторных условиях

10. Решение

Подготовка всей необходимой технической документации и все экономические расчеты.

11. Производство

Производство и управление этими процессами

12. Распределение продукции

Сбытовые процессы, маркетинговая деятельность

13. Потребление

Гарантийное обслуживание

## Особенности инженерного проектирования

### АСОИУ

1. Определение потребности создания качественно новой системы

Выявление проблем с точки зрения автоматизации. Для сбора информации могут быть использованы: анкетирование, интервью, наблюдение, анализ документов.

В результате этого этапа у аналитика должно сформироваться полное представление об информационной деятельности объекта и о проблемах, которые в ней существуют.

Определение целей:

- Традиционные (уточнение расчетов, сокращение затрат/времени решения, оперативность принятия решений, оптимизация расходов ресурсов)
- модификация; принципиальное изменение размеров – увеличение/уменьшение количества решаемых задач или наращивание функциональности; изменение внешнего вида (интерфейса);

сохранение старого (новые технологии для старых алгоритмов); перестановка (перегруппировка задач); замена (появление новых технологий); сочетание; удобство обращения (изменение интерфейса и юзабилити); безопасность.

На любом предприятии используются самые разные типы систем (АСУТП, САПР, АСУ и др.)

2. Научные исследования

Создание новых технологий.

3. Формулировка задания

Включает в себя следующие работы:

- анализ типовых решений
- анализ аналогичных систем, выявление их достоинств и недостатков
- определение и обоснование целей создания системы
- определение функциональных возможностей
- выбор и обоснование технических и программных средств проектирования
- определение требований к серверной и клиентской частям
- определение выгод для заказчика
- оценка объема работ
- описание бизнес-процессов до автоматизации
- принципиальные решения для БД
- принципиальные решения для интерфейса

4. Формирование идей Принимаются решения о том, какие технологии буду использоваться при проектировании и разработки АСОИУ. При отсутствии технологий, разработать самостоятельно, либо аутсорсинг.

5. Выработка концепции

Составление технического задания.

6. Решение

Техническое проектирование. Описание постановок всех задач, которые будут программироваться. При создании сайта, требуются постраничное описание с учетом серверной и клиентской частями.

7. Программирование

Создании программ на основе технического проектирования.

8. Анализ

Верификация и валидация созданного программного продукта. Валидация – проверка того, что реализовано то, что должно. При выявлении несоответствий, возможны итеративная связь с предыдущими этапами

9. Эксперимент

Проверка АСОИУ на реальной базе.

10. Распределение продукции

Распространение.

11. Потребление

Эксплуатация и сопровождение.

## Особенности проектов по созданию АСОИУ

1. Сложность описания

Отсутствует какая-либо формализация на этапе обследования.

2. Любая АСОИУ состоит из многих подсистем, которые взаимодействуют между собой.
3. Отсутствие прямых аналогов
4. Необходимая интеграция старых и вновь разрабатываемых приложений
5. Как правило АСОИУ функционирует в неоднородной среде:
  - ПК разных производителей;
  - некоторые функции могут быть вынесены в Интернет; -
  - использование станков с ЧПУ и системы класса АСУТП.
6. Разобщенность и разнородность разработчиков
7. Существенная временная протяженность проекта

## Предпроектное обследование

Случаи, в которых необходимо предпроектное обследование:

- создание системы по индивидуальному заказу;
- внедрение готовой системы;
- внедрение дополнительных модулей;
- модернизация существующей системы;
- внедрение дополнительных систем в дополнение к уже существующим.

Результаты предпроектного обследования:

- описание существующих бизнес-процессов заказчика;
- перечень задач, которые необходимо решить с помощью автоматизации;
- оценка бизнес-процессов заказчика и рекомендации по их изменению или оптимизации, с учетом преимуществ, которые предоставляют информационные системы;
- уникальные конкурентные преимущества заказчика, которые обязательно необходимо учесть при внедрении;
- анализ имеющейся IT-среды на соответствие требованиям внедряемой системы;
- оценка работ по интеграции новой системы в уже существующую;
- описание документооборота в автоматизируемых подразделения и требований к аналитике;
- оценка возможных рисков и минимизация их влияния;
- описание пожеланий заказчика, которые не могут быть реализованы в предлагаемой системе и потребуют доработки;
- предложение по реализации проекта с обоснованием предлагаемого варианта и задание необходимого для заказчика количества автоматизированных рабочих мест;

Какие вопросы должны быть выявлены в процессе предпроектного обследования:

- функционально-организационная структура;
- функциональные обязанности подразделения, внутренняя структура, кадровый состав;
- информационные и временные характеристики данных, которые используются при решении задач;
- сроки решения задач;
- составить перечень документов и привести их образцы;
- составить схему документооборота, при этом учесть:

- описание периодичности каждого документа;
- источник документа и его назначение;
- маршрут;
- правила внесения изменений;
- правила утверждения и исполнения документа;
- формы хранения и представления.

2015-05-23

## Функциональная система

Любая организация может работать без автоматизации.

Функции предприятия – функции, не зависящие от специализации предприятия:

1. Рабочие
2. Материалы
3. Помещения
4. Оборудование/инструменты
5. Технологии
6. Деньги
7. Рынок сбыта

Управление

процесс поддержания системы (любого характера) в назначенных параметрах, путем обработки информации о состоянии системы и принятия на этой основе решений о воздействии на систему

Процесс управления осуществляется посредством функций управления. К основным из них относятся:

1. планирование (прогнозирование и расчет)
2. учет (констатация факта)
3. контроль
4. анализ и принятие решений
5. регулирование

Управление сводится либо к выполнению какой-то одной функции, либо любому их сочетанию.

Каждой функцией предприятия необходимо управлять. Автоматизация управления отдельными функциями предприятия реализуется отдельными подсистемами (модулями).

## Методологии создания автоматизированных систем

1. Классическая – все этапы, длительность – 1-1,5 года, проектирование больших систем.
2. Экстремальное программирование – небольшие коллективы, маленькие

## Понятие жизненного цикла информационной системы

### Процесс

цепочка последовательно выполняемых работ.

### Этапы

последовательные отрезки времени для выполнения работ, относящиеся к разным процессам.

Традиционно выделяются следующие этапы АСОИУ:

1. Планирование и анализ требований (системный анализ)
2. Проектирование
3. Реализация
4. Внедрение
5. Эксплуатация

В настоящее время схема проектирования выглядит следующим образом:

1. исследование бизнес-процессов заказчика и постановка задач;
2. разработка ТЗ;
3. создание программного кода и организация тестовой среды отладки;
4. тестирование;
5. разработка документации и устранение недостатков;
6. передача системы в эксплуатацию;
7. сопровождение и поддержка.

## Модели жизненного цикла

1. Каскадная модель – каждый следующий этап выполняется после полного завершения предыдущего.

Особенности:

- На каждом этапе появляется законченный пакет документации;
- Легко планировать время и средства.

2. Итерационная модель.

Особенности:

- На каждом этапе можно вернуться на любой из предыдущих этапов.

3. Спиральная модель.

Каждый виток спирали соответствует созданию фрагмента или версии проекта. На нем уточняются цели и характеристики проекта, определяется его качество и планируется следующий виток спирали.

На каждом витке спирали углубляются и конкретизируются детали проекта.

Каждый виток учитывает особенности и проблемы предыдущих. Возможно развитие и модификация системы в процессе ее проектирования.

# Экстремальное программирование

Экстремальное программирование относится к гибким методологиям разработки программного обеспечения.

Основные выводы:

- должна быть усовершенствована связь между разработчиками;
- упростить проектные решения;
- усилить обратную связь с заказчиком;
- проявлять больше активности.

Ключевые моменты этой методологии:

1. Создавать простую и понятную программу легче, чем сложную и запутанную. В типовом проекте труд программиста дороже в двадцать раз аппаратной части.
2. Важный момент – ошибки программы. Тесты пишутся до разработки программы или во время разработки.
3. Готовность программистов к постоянным изменениям в проекте.
4. Наиболее эффективно для проектов с постоянно изменяющимися требованиями.
5. Ориентация на проекты с высокой степенью риска.
6. Невозможность использования этот подход на больших проектах.
7. Для предметной области должна существовать возможность создания автоматизированного теста.
8. Вместо термина техническое задание используется формулировка пользователя. Каждая формулировка занимает три-четыре предложения в терминах заказчика. Главное отличие ТЗ от формулировок пользователя – уровень детализации.
9. Планирование устанавливает правила того, каким образом вовлеченные в проект стороны принимают решения, поэтому существует план релизов, в котором определяются сроки, в которые будут предоставлены формулировки пользователя. Исходя из этого, формулируется требования для очередной итерации. Каждая итерация может длиться от одной до трех недель.
10. Наличие ежедневных совещаний.
11. Наличие системы метафор. Следует использовать такую систему имен для объектов, которую сможет понять каждый участник без специфических знаний о системе.
12. Парное программирование.
13. Смена поручений – во время очередной итерации следует помещать сотрудников на новые участки работы.
14. Не добавлять функциональность слишком рано – если пользователь не попросил, не добавлять функциональность.

## Проектирование пользовательского интерфейса

### Понятие и основные особенности



# пользовательского интерфейса

## Пользовательский интерфейс

интерфейс, в котором одна сторона представлена человеком, а вторая машиной или устройством. Представляет собой совокупность средств и методов, при помощи которых пользователь взаимодействует с различными сложными машинами, устройствами, аппаратурой

К средствам относятся:

- Средства вывода информации из устройства пользователю (монитор, принтер, лампочки, динамики)
- Средства ввода информации в устройство (кнопки, переключатели, датчики)

## Методы

набор правил, которые закладываются разработчиком устройства и согласно этим правилам действия пользователя должны привести к решению задачи

## Юзабилити

понятие в микроэргономике, которое обозначает уровень удобства предметов, в заявленных целях

## Юзабилити

определение, закрепленное в ИСО 9241-11 – степень, с которой продукт может быть использован определенными пользователями, при определенном контексте использования, для достижения определенных целей, с должной эффективностью, продуктивностью и удовлетворенностью

# Концептуальные модели пользовательского интерфейса

1. Пользовательская модель
2. Модель программиста
3. Модель проектировщика

Каждая модель отражает ожидания человека, который работает с компьютером и тот опыт, который он получает в результате работы.

В зависимости от модели, должны учитываться три принципа пользовательского интерфейса:

1. Контроль интерфейса пользователем
2. Уменьшение нагрузки на память пользователя
3. Последовательность пользовательского интерфейса

Принципы пользовательского интерфейса:

1. Обеспечить пользователю полный контроль над интерфейсом:
  - 1.1. Дать возможность пользователю использовать альтернативные способы взаимодействия.
  - 1.2. Обеспечить возможность прерывания работы без потери данных.
  - 1.3. Обязательно показывать поясняющие сообщения (подсказки,

комментарии) с обязательным соблюдением понятных пользователю терминов.

1.4. Создать условия для обратимых действий.

1.5. Обеспечить понятные пути и выходы, т.е. свободную ориентацию пользователя, для этого могут быть использованы контекстные подсказки.

1.6. Предусматривать настройку интерфейса под пользователя с различным уровнем подготовки.

1.7. Сделать пользовательский интерфейс прозрачным – не заметным для пользователя.

1.8. Возможность настройки интерфейса под индивидуальные особенности пользователя.

1.9. Разрешить пользователю напрямую манипулировать объектами интерфейса (?)

2. Уменьшить нагрузку на память пользователя:

2.1. Не загружать кратковременную память пользователя.

2.2. Предоставлять возможность выбора из списка

2.3. Визуальные подсказки интерфейса, которые информируют пользователя, где он находится, что делает.

2.4. Наличие горячих клавиш, для ускорения работы.

2.5. Отображать на экране только те действия, которые пользователь может производить над объектами.

2.6. Использовать приемы как мастер или советчик.

2.7. Увеличить визуальную ясность, т.е. группировать объекты в меню или списке, нумеровать объекты, использовать заголовки или другие объекты для привлечения внимания. При этом избегать случайной группировки и ненужных выделений.

3. Сделать интерфейс совместимым

3.1. Пользователи должны иметь опорные точки при перемещении

3.2. При проектировании одним разработчиком разных систем, ориентированных на разных пользователей, разработчик должен использовать одни и те же элементы интерфейса.

3.3. При выполнении одних и тех же действий в разных местах программы, пользователь должен получать один результат.

3.4. В рамках одной системы, необходимо разработать общую структуру интерфейса, определить используемые шрифты, цветовые схемы, и придерживаться их в течении всего процесса разработки.

3.5. Одним из элементов дружественного интерфейса, является возможность исследовать возможности интерфейса, без страха сделать что-то неправильно.

3.6. Наличие в системе развитой навигации.

Почти любая навигация представляет собой меню. Недостатки:

- Ограничивает выбор
- Любое многоуровневое страдает от каскадных ошибок
- Почти не одно меню не способно показать пользователю, куда ему целесообразно пойти.

2015-04-06

Перед проектированием интерфейса необходимо выяснить следующие вопросы:

- Какая информация необходима пользователю для решения каждой задачи.
- Какую информацию пользователь может игнорировать.
- Совместно с пользователем разделить всю информацию на:
  - Сигнальную
  - Отображаемую
  - Редактируемую
  - Поисковую
  - Результирующую.
- Какие решения пользователю необходимо принимать в процессе работы с программой (или при решении задачи) – просмотр промежуточных результатов.
- Может ли пользователь совершать несколько действий одновременно.
- Какие типовые операции использует пользователь при решении задач (использование справочников).
- Что произойдет, если пользователь нарушит предписанный алгоритм.

При разработке прототипа пользовательского интерфейса необходимо учитывать следующее:

- Особенности устройства ввода-вывода информации.
- Выбор технологий и методов ведения диалога программы с пользователем:
  - степень активности пользователя при взаимодействии с системой;
  - степень учета ситуации;
  - соответствие ожиданиям пользователя;
  - устойчивость и терпимость к ошибкам пользователя, путем исправления типовых ошибок;
  - настройка пользовательского интерфейса на различный уровень подготовки пользователя;
  - степень адаптивности пользовательского интерфейса под предпочтения пользователя (цветовая палитра);
- При композиции экрана необходимо учитывать ограниченные пространства экрана, в связи с чем возникает задача оптимального расположения максимально возможного объема информации.
- Формирование обратной связи между пользователем и приложением:
  - Показ актуального состояния системы
  - Вывод отдельных важных для работы данных и показателей
  - Отражение действий пользователя
  - Ясность и информативность сообщений системы
- При проектировании панели меню и инструментов, а также выбора пунктов в них следует учитывать:
  - Логическую и смысловую группировку пунктов
  - Фиксированная позиция панелей на экране
  - Ограничение на ширину списка выбора и глубины меню
  - Использование привычных названий, распространенных иконок
  - Аккуратное использование сокращений
- Разработка средств ориентации и навигации - удобный переход между режимами работы, быстрый поиск в списке
- Создавать формы для ввода данных

Дополнительные требования к интерфейсу:

- Не следует помещать в главное меню пункты, которые срабатывают сразу после нажатия
- Делать действия пользователя обратимыми
- Всегда предоставлять отмену
- Предоставлять запасной выход
- Предлагать пользователю постоянные наглядные подсказки
- Стремиться к единообразию интерфейса в различных подсистемах или частях программы
- Обеспечить защиту работы пользователя

## Стили взаимодействия пользователя с системой

Стили взаимодействия	Плюсы	Минусы	Примеры
Прямое манипулирование	<ul style="list-style-type: none"> <li>- быстрое и интуитивно понятное взаимодействие</li> <li>- легкость в изучении</li> </ul>	<ul style="list-style-type: none"> <li>- сложная реализация</li> <li>- подходит только для систем, где есть зрительные образ объекта</li> </ul>	компьютерные игры
Выбор из меню	<ul style="list-style-type: none"> <li>- сокращение ошибок пользователя</li> <li>- минимальный ввод с клавиатуры</li> </ul>	<ul style="list-style-type: none"> <li>- медленно для опытных пользователей</li> <li>- может быть сложным, если меню состоит из большого количества вложенных пунктов</li> </ul>	программы общего назначения
Заполнение форм	<ul style="list-style-type: none"> <li>- легкость изучения</li> </ul>	<ul style="list-style-type: none"> <li>- занимает пространство на экране</li> </ul>	формализованные документы
Командный язык	<ul style="list-style-type: none"> <li>- мощный и гибкий</li> </ul>	<ul style="list-style-type: none"> <li>- сложность изучения</li> <li>- возможны ошибки</li> </ul>	cmd
Естественный язык	<ul style="list-style-type: none"> <li>- подходит неопытным пользователям</li> </ul>	<ul style="list-style-type: none"> <li>- требует большого ручного набора</li> </ul>	www

2015-04-07

## Представление информации

## Представление отчетов

Получение регламентированных отчетов.

**Преимущество:** пользователю не нужно ничего знать, кроме того, какой отчет ему нужен.

**Недостаток:** количество отчетов ограничено.

Отчетность нерегламентированного характера. Запросная система.

**Преимущество:** количество отчетов неограниченно.

**Недостаток:** сложность освоения.

## Представление графической информации.

Представляемые данные  $\Rightarrow$  ПО для представления данных  $\Rightarrow$  Отображение

В качестве представляемых данных могут отображаться ход решения задачи, промежуточные результаты или конечная отчетность.

Принимая решение по представлению данных, разработчик должен учитывать следующие факторы:

1. Что нужно пользователю – точное значение данных или соотношение между ними.
2. Насколько быстро будут происходить изменения значений данных. Нужно ли пользователю немедленно показывать изменения значений.
3. Должен ли пользователь предпринимать какие-либо действия в ответ на изменение данных.
4. Нужно ли пользователю взаимодействовать с отображаемой информацией путем прямого манипулирования.
5. В каком виде должна отображаться информация: в текстовом, графическом, табличном.
6. Необходимо ли показывать значения элементов отображаемых данных.

Рекомендации по формированию аналитической информации:

1. Если аналитической информации много, то не стремиться ее всю включить в один документ.
2. Проектировать новые формы взамен существующих.
3. Шапки документов должны соответствовать содержанию.
4. Желательно называть графы в именительном падеже.
5. Документы в табличной форме целесообразно упорядочивать и сортировать по соответствующему полю.
6. Использовать в таблицах выделение – шрифт, масштаб, заливку.
7. Не включать в таблицы значения, которые являются одинаковыми во всех строках.
8. Не добавлять единицы измерения к числам.
9. Логически связанные поля размещать рядом.

## Средства поддержки пользователя

К средствам поддержки пользователя относятся все сообщения, которые генерируются системой, и система справок. Справочную систему составляют:

1. Сообщения, генерируемые в ответ на действия пользователей

(сообщение об ошибке, проверочные действия, контролирующие действия)

2. Диалоговая система

3. Документация

При проектировании необходимо учитывать:

1. Содержание справочной системы
2. Опыт пользователя
3. Профессиональный уровень пользователя
4. Стилль сообщений (положительный оттенок)

Особенности сообщений об ошибках:

1. Не должны быть агрессивными и грубыми
2. Должны быть верными

### Типы пользовательской документации

1. Функциональное описание – представляет функциональные возможности и их взаимосвязь.
2. Инсталляционное описание.
3. Руководство пользователя – повседневное использование системы.
4. Справочное руководство – возможности системы и их использование, представлен список сообщений об ошибках и возможные причины этих ошибок.
5. Руководство для администратора.

2015-04-21

### Компоновка интерфейсов

*Располагать информацию нужно по горизонтали по центру потому, что в противном случае нет связи с прошлым или будущим*

*Располагать информацию по вертикали нужно по центру потому, что сверху информация давит, а в случае если информация снизу, то ощущается пустота*

Особенности компоновки:

- оставлять примерно половину экрана пустым
- при выдаче табличной информации, выделять строки; примерно четыре-пять пробелом между столбцами
- на экране должна находиться только та информация, которая действительно необходима пользователю на данном этапе работы
- в меню следует помещать список только тех пунктов, из которых пользователь может сделать выбор
- важно, чтобы на экране отображалась **вся** информация, которая относится к решаемой задаче
- один и тот же тип информации должен появляться на экране в одном и том же месте
- на одном экране необходимо размещать все данные, которые необходимы для принятия конкретного решения
- можно изменять структуру любых внутренних документов; нельзя изменять только те, которые имеют статус государственных (в ОКУД)
- вспомогательные сообщения требуют особенно тщательной подготовке,

т.к они должны полностью соответствовать возникшей проблемы.

## Психологические аспекты интерфейса

Существуют следующие способы уменьшения видимости задержки в работе системы (время отклика должно быть сопоставимо со скоростью реакции человека):

- Обеспечение сопровождения щелчков на всех кнопках визуальной или звуковой обратной связью в пределах 50 мс.
- Вывод сообщений, которые показывают потенциальную длительность ожидания результата для любого действия более двух секунд.
- Демонстрация действительной длительности с помощью анимированного индикатора.
- Показ пользователю сообщений, отражающих информацию о процессе.
- Привлечение внимания пользователя звуковым сигналом или заметной визуальной индикации для процессов дольше 10 секунд.

Другие особенности:

- текст для чтения должен иметь высокий контраст, текст должен быть удобочитаемым. Основное требования к сочетанию цветов текста и фона - достаточный контраст между ними.
- Не рекомендуется размещать текст на сером фоне.
- Любой шрифт имеет свое определенное назначение

## Оценка и улучшение пользовательского интерфейса

Существуют четыре основных критерия качества любого интерфейса.

1. Скорость работы пользователя.
2. Количество человеческих ошибок.
3. Скорость обучения.
4. Субъективное удовлетворение.

При борьбе с ошибками необходимо направлять усилия на следующее:

- плавное обучение пользователя в процессе работы;
- снижение требований к бдительности;
- повышение разборчивости и заметности индикаторов;
- блокировка потенциально опасных действий пользователя, до получения подтверждения;
- проверка всех критических важных параметров перед их применением;
- максимальное использование системой необходимых команд и параметров, при котором от пользователя требуется только проверка.

## Причины человеческих ошибок

- Недостаточное знание предметной области;
- опечатки;
- неправильный ввод информации.

## Естественные цвета и дизайн интерфейсов

*Необходимо использовать естественные цвета, которые мы постоянно видим*

- способность выделяться
- способность направлять
- способность увлекать
- способность вдохновлять

## Особенности проектирования сайтов

Существует три типа факторов:

1. Расположение информации
2. Навигация по сайту
3. Цветовое решение

Существуют три аспекта, которые влияют на восприятие сайта:

1. Страницы сайта, которые выполняют одни и те же функции, должны иметь единый дизайн.
2. Сайт, выдержанный в едином стиле, работает на узнаваемость (например, продвижение торговой марки).
3. Дизайн, структура и содержание сайта должны связываться только концептуально. Технически они должны быть независимы друг от друга.

Особенности навигации:

1. Все функции элементов интерфейса должны быть интуитивно понятны.
2. В рамках одного проекта, должен использоваться только один вариант дизайна интерфейса.
3. Интерфейс - это не реклама, он должен быть четко выражен в рамках общей концепции дизайна сайта и не должен отвлекать от содержания страницы.
4. В проектах с большим объемом информации и сложной структурой, кроме элементов навигации, необходимо использовать элементы индикации, которая показывает местоположение страницы в структуре сайта.

Кроме того, необходимо учитывать требования к дизайну, которые накладываются физиологией человека. К ним относятся:

- видимость (важная информация или элементы управления должны находиться на переднем плане);
  - понимание (меню должно отражать структуру сайта);
  - память (Кратковременная память ограничена  $7 \pm 2$  элементами информации. Из этого следует, что в простом меню количество пунктов не должно превышать семи. Если пунктов больше, то необходимо разбить на группы, при этом количество подпунктов не должно превышать семи);
  - "не множить сущности без надобности" (любая задача должна решаться за минимальное количество действий);
  - текст не должен сливаться с фоном, а заголовки теряться.
- Использование контрастных цветов недопустимо.

**На экзамене будут вопросы из самостоятельного изучения**



### **Технический проект**

1. Описание базы данных.
2. Часть значений должна выбираться из классификатора
3. Сообщения об ошибках
4. Отчеты из программы
5. Интерфейс

## Основные принципы разработки пользовательских интерфейсов

*У пользователя должно складываться впечатление, что он управляет интерфейсом, а не интерфейс пользователем.*

1. Естественность. Естественный интерфейс - интерфейс, который не вынуждает пользователя существенно изменять привычные для него способы решения задачи. Обязательное использование понятных пользователю понятий и образов. Именно это делает интерфейс интуитивно понятным.
2. Согласованность. Единообразие логики и стиля. Согласованность позволяет пользователям переносить имеющиеся знания на новые знания. Согласованность важна для всех аспектов интерфейса, а именно:
  - имена команд;
  - визуальное представление информации
  - поведение интерактивных элементов.

Бывает согласованность следующего вида:

- 2.1. согласованность в пределах одного программного продукта - одна и та же команда должна выполнять одни и те же функции, где бы она не находилась;
- 2.2. согласованность в пределах одной рабочей среды;
- 2.3. согласованность в использовании метафор;
3. Дружественность. Пользователь обычно изучает особенность работы с новой программой методом проб и ошибок. Интерфейс должен это учитывать. Поэтому на каждом этапе работы он должен разрешать только соответствующий набор действий и предупреждать пользователя о тех ситуациях, где они могут навредить системе или данным. Всегда предусматривать возможность отмены действий. Эффективный интерфейс должен предотвращать ситуации, которые могут закончиться ошибками.
4. Принцип обратной связи. Каждое действие пользователя должно получать визуальное и/или звуковое, подтверждающие, что программа получила введённую команду. При обработке задания, полезно предоставлять пользователю информацию о состоянии процесса, чтобы была возможность прервать этот процесс при необходимости (информация о промежуточных результатах).

5. Простота. Простота предполагает легкость изучения и использования. Реализация доступа к широким функциональным возможностям и обеспечение простоты работы противоречат друг другу. Эффективный интерфейс призван сбалансировать эти цели. Например, достичь этого можно следующим путем:

- показывать на экране только ту информацию, которая минимально необходима для решения задачи;
- избегать многословных командных имен или сообщения;
- размещение и представление элементов на экране с учетом их смыслового значения;
- последовательное раскрытие диалоговых окон, разделов меню и т.д.

6. Гибкость. Способность системы учитывать уровень подготовки и производительность пользователя. Свойство гибкости предполагает изменение структуры диалога и/или входных данных. Основная проблема при этом заключается в том, какие признаки нужно использовать, а не как организовать изменения в диалоге.

7. Эстетическая привлекательность. Качество интерфейса сложно оценить количественными характеристиками. Тем не менее некоторое объективную оценку можно дать следующих частных показателей:

- время, необходимое пользователю для достижения заданного уровня знаний и навыков для работы с приложением (например, непрофессиональный пользователь должен освоить команды работы с файлами за 4 часа);
- сохранение полученных навыков по истечению некоторого времени;
- скорость решения задачи с помощью приложения (например, пользователь должен обработать не менее 20 документов за 1 час с ошибкой не более 1%);
- некая субъективная удовлетворенность пользователя после работы с приложением.

2015-05-12

## Обобщение правил интерфейса

- Интерфейс необходимо проектировать и разрабатывать как отдельный компонент программного продукта;
- необходимо учитывать возможности и особенности аппаратно-программных средств, на базе которых реализуется интерфейс;
- необходимо учитывать особенности и традиции той предметной области, к которой относится создаваемое приложение;
- процесс разработки интерфейса должен носить итерационный характер и его обязательным элементом должно быть согласование с потенциальным пользователем;
- средства и методы реализации интерфейса должны обеспечивать возможность его адаптации к потребностям пользователя.

Стандартизация пользовательского интерфейса

К середине 70-х годов осознали необходимость формирования единых подходов к реализации пользовательского интерфейса, но такие возможности были ограничены. В настоящее время все серьезные публикации, посвященные интерфейсу, опираются на знания в таких областях как психология, эргономика, и т.п. В Америке существует национальный институт стандартов (ANSI), в нем есть специальная консультационная комитет стандартов по интерфейсу "человек-компьютер". Существует международный консультативный комитет по телеграфии и телефонии, который изучает особенности интерактивных элементов интерфейса.

В 1986 г. было опубликовано руководство по разработке программного пользовательского интерфейса, содержащее 944 принципа, которые касались:

- ввода и отображения данных;
- поддержки пользователя;
- защиты данных.

Но этот проект не получил статус официального документа, поскольку имел серьезный недостаток, а именно: не учитывались технологические возможности инструментальных средств.

Коренным образом ситуация изменилась в 1987 г., корпорация IBM объявила о намерении создать единую среду разработки приложений. Цели проекта:

- повышение производительности труда программистов и конечных пользователей;
- облегчение эксплуатации и сопровождения ПО;
- повышение эффективности распределенной обработки информации;
- увеличение отдачи от инвестиций в разработку информационных систем.

Исследованиями и практической реализацией графических интерфейсов занимались и другие фирмы: Xerox, Apple, Digital Research, Microsoft. В результате их деятельности были определены основные концепции построения графических пользовательских интерфейсов:

- использование единой рабочей среды пользователя в виде рабочего стола;
- объектно-ориентированный подход к описанию заданий пользователя;
- использование графических окон, в качестве основной формы отображения данных;
- применение средств неклавиатурного ввода, основанного на выборе и указании чего-либо с помощью мыши.

## Этапы проектирования пользовательского интерфейса

1. Определить структуру диалога (вопрос-ответ, выбор из меню, экранные формы, командный язык).
2. Определить возможный сценарий развития диалога.
3. Содержание управляющих сообщений и данных, которыми могут обмениваться человек и приложение (то есть определить семантику приложения).
4. Определить визуальные атрибуты отображаемой информации (синтаксис сообщений).

## Экспертная оценка пользовательского интерфейса

Для проведения экспертной оценки нужно знать:

- разные люди обнаруживают разные ошибки, т.е. этот метод работает, когда количество экспертов больше одного;
- лучше привлекать несколько экспертов не одновременно, но последовательно;
- чем больше информации будет предоставлено эксперту, тем более сложные проблемы он сможет выявить;
- нельзя требовать от эксперта работы "по весу" - описание одной проблемы должно составлять два-три предложения.

В качестве экспертов могут выступать сами пользователи, для этого используются следующие методы:

- анкетирование;
- наблюдение за работой пользователя с последующим обсуждением;
- видеонаблюдение типичного использования системы;
- добавление в программу кода, который отслеживает информацию о наиболее часто используемых системных сервисах и распространенных ошибках.