

Лабораторная работа №2  
Низкоуровневое проектирование интерфейса:  
количественная оценка и построение прототипа

### 1 Цели работы

- 1.1 Закрепить теоретические знания по разработке пользовательского интерфейса.
- 1.2 Развить навыки создания вариантов прототипов интерфейса пользователя.
- 1.3 Получить практические навыки по количественной оценке интерфейса на этапе низкоуровневого проектирования.
- 1.4 Закрепить принципы обоснования выбора прототипа интерфейса по его количественной оценке.

### 2 Краткие теоретические сведения

Для анализа качества интерфейсов используется множество количественных и эвристических методов. Одним из лучших подходов к количественному анализу моделей интерфейсов является **классическая модель GOMS** (goals, objects, methods and selection rules).

Метод, использующий модель GOMS, основан на **разбиении всех действий пользователя на отдельные составляющие**. Для каждой из них с помощью тщательных лабораторных исследований получен **набор временных интервалов, необходимых для ее выполнения**. В таблице 1 приведена **номенклатура элементарных действий и соответствующие временные интервалы**.

Таблица 1

Нажатие <b>клавиши</b> клавиатуры, включая клавиши Alt, Ctrl, Shift	0,28 с	К
Нажатие клавиши <b>мыши</b>	0,1 с	М
Указание – <b>перемещение</b> курсора мыши, чтобы указать какую-либо позицию на экране монитора	1,1 с	П
Перемещение- перенос руки пользователя с клавиатуры на мышь или обратно	0,4 с	В
Ментальная подготовка – мысленный выбор пользователем своего следующего элементарного <b>действия</b>	1,2	Д
Ответ – <b>реакция</b> системы на элементарное действие пользователя	-	Р

Широкая изменяемость каждой из представленных мер объясняет, почему **эта модель не может использоваться для получения абсолютных временных значений с высокой степенью точности**. Но этот метод вполне пригоден для проведения сравнительной оценки между какими-либо двумя моделями интерфейса по уровню эффективности их использования.

Расчет времени, необходимого для выполнения некоторого действия начинают с разбиения его на элементарные действия, которые соответствуют номенклатуре приведенной в таблице 1. Проще всего выделить движения К, М, П, В. Проблему составляет **определение моментов, когда пользователь должен остановиться, чтобы выполнить бессознательную ментальную операцию**. Рассмотрим основные правила по выявлению этих моментов (таблица 2).

Таблица 2

<b>Правило 0</b> Начальная расстановка операторов Д	<b>Операторы Д следует устанавливать перед всеми операторами К и М (нажатие клавиши), также перед всеми операторами П, предназначенными для выбора команд</b> . Но перед операторами П, предназначенными для указания на аргументы этих команд, ставить оператор Д не следует.
<b>Правило 1</b> Удаление ожидаемых операторов Д	<b>Если оператор, следующий за оператором Д, является полностью ожидаемым с точки зрения оператора, предшествующего Д, то этот оператор Д может быть удален</b> . Если пользователь перемещает мышь с намерением нажать на ее

	кнопку по достижении цели движения, то в соответствии с этим правилом следует удалить оператор Д, установленный по правилу 0. Так последовательность действий П Д К преобразуется в П К.
<b>Правило 2</b> Удаление операторов Д внутри когнитивных единиц	Если строка Д К Д К Д К... принадлежит когнитивной единице, то следует удалить все операторы Д, кроме первого. Когнитивной единицей является непрерывная последовательность вводимых символов, которые образуют название команды или аргумент. Например У, переместить, 4564.23 – это когнитивные единицы.
<b>Правило 3</b> Удаление операторов Д перед последовательными разделителями	Если оператор К означает лишний разделитель, стоящий в конце когнитивной единицы (например, разделитель команды, следующий сразу за разделителем аргумента этой команды), то следует удалить оператор Д, стоящий перед ним.
<b>Правило 4</b> Удаление операторов Д, которые являются прерывателями команд	Если оператор К является разделителем, стоящим после постоянной строки (например, название команды или любая последовательность символов, которая каждый раз вводится в неизменном виде), то следует удалить оператор Д, стоящий перед ним. (Добавление разделителя станет привычным действием, и поэтому разделитель станет частью строки и не будет требовать специального оператора Д.) Но если оператор К является разделителем строки аргументов или любой другой изменяемой строки, то оператор Д следует сохранить перед ним.
<b>Правило 5</b> Удаление перекрывающих операторов Д	Любую часть оператора Д, которая перекрывает оператор Р, означающий задержку, связанную с ожиданием ответа компьютера, учитывать не следует.

В этих правилах под **строкой** понимается некоторая последовательность символов.

**Разделителем** считается символ, которым обозначено начало или конец значимого фрагмента текста, такого как, например, слово естественного языка или телефонный номер. Так, пробел является разделителем для большинства слов, а точка используется в конце предложений для разделения. В качестве разделителей могут выступать скобки для ограничений пояснений или замечаний и т.д.

Если для выполнения команды требуется дополнительная информация, она называется здесь **аргументом** данной команды.

Правила GOMS позволяют определить время, необходимое пользователю для выполнения любой четко сформулированной задачи, для которой данный интерфейс предусмотрен. Однако этого метода недостаточно, чтобы оценить насколько быстро должен работать интерфейс – его производительность. Рассмотрим подробнее критерий эффективности Раскина, который оценивает информационную производительность интерфейса.

Информационная производительность интерфейса Е определяется как отношение минимального количества информации, необходимого для выполнения задачи, к количеству информации, которое должен ввести пользователь. Параметр Е изменяется в пределах [0, 1]. В параметре Е учитывается только информация, необходимая для задачи, и информация вводимая пользователем. Несколько методов действия могут иметь одинаковую производительность Е, но иметь разное время выполнения. Возможно, что один метод имеет более высокий показатель Е, но действует медленнее, чем другой метод.

Информация измеряется в битах. Один бит представляет собой один из двух альтернативных вариантов (0 или 1; да или нет) и является единицей информации. При количестве  $n$  равновероятных вариантов суммарное количество передаваемой информации определяется как

$$\log_2 n$$

Количество информации для каждого варианта определяется как

$$(1/n)\log_2 n \quad (1)$$

Если вероятности для каждой альтернативы не являются равными и  $i$ -я альтернатива имеет вероятность  $p(i)$ , то информация, передаваемая этой альтернативой определяется как

$$p(i)\log_2(1/p(i)) \quad (2)$$

Общее количество информации является суммой по всем вариантам выражений (1) или (2)

При использовании мыши в качестве устройства ввода информации, количество информации оценивают подобным же образом. Если экран поделен на две равные области: одна соответствует «да», а вторая соответствует «нет», то щелчок мыши в одной из этих областей будет передавать 1 бит информации. Если имеется  $n$  равновероятных объектов, то нажатием на один из них сообщается  $\log_2 n$  бит информации. Если объекты имеют разные вероятности, используется сумма значений количества информации, полученные по формуле (2).

При передаче информации нажатием клавиши ее количество зависит от общего числа клавиш и относительной частоты использования каждой из них. То есть нажатия клавиш могут использоваться как приблизительная мера информации. Если на клавиатуре имеется 128 клавиш, и каждая из них используется с одинаковой частотой, то нажатие любой из них будет передавать

$$\frac{1}{128}\log_2(128)=7 \text{ бит.}$$

В действительности частота использования клавиш существенно изменяется (например, клавиша  $e$  в русской раскладке клавиатуры используется наиболее часто) с учетом этого можно считать, что каждое нажатие клавиши передает  $\approx 5$  бит информации.

Иногда на практике удобнее использовать символьную эффективность вместо информационной производительности. Она определяется как минимальное количество символов, необходимое для выполнения задачи, отнесенное к количеству символов, которое в данной модели интерфейса требуется ввести пользователю.

### 3 Задание на лабораторную работу

3.1 Имея результаты проектирования интерфейса на высоком уровне, разработать варианты моделей – прототипы экранных форм для каждого из функциональных блоков.

3.2 Используя разработанные прототипы форм, провести количественную оценку элементов интерфейса по указанию преподавателя. Метод количественной оценки – GOMS, информационная производительность, символьная эффективность - задается преподавателем.

3.3 По результатам количественной оценки сделать выводы о возможности усовершенствования интерфейса.

3.4 При возможности внести необходимые усовершенствования в модели форм и реализовать их в среде разработки приложения. Каждую форму следует снабдить описанием навигации по ней.

### 4 Порядок выполнения работы

Рассмотрим пример с разработкой пользовательского интерфейса приложения создаваемого для мелкой фирмы (лабораторная работа №1). Программное обеспечение предполагает наличие справочников:

1) со списком клиентов фирмы, причем пользователь – в рассматриваемом примере это представитель обслуживающего персонала – может вносить в него изменения и добавлять новых клиентов;

2) со списком товаров, реализуемых фирмой. Пользователь не может вносить в него изменений.

Кроме того, создается список оформленных заказов – журнал заказов. Предположим, что такой список включает заказы за определенный период, например месяц. На следующий месяц создается новый журнал. Пользователь может вносить изменения в имеющиеся заказы, а также оформлять новые заказы.

Основные экранные формы известны из результата выполнения лабораторной работы

№1:

- Журнал заказов;
- Текущий заказ;
- Список клиентов;
- Карта клиента;
- Список товаров;
- Карта товара.

Работа пользователя начинается с первой из них. Эта форма должна обеспечивать выполнение следующих операций:

- создать новый заказ;
- задать атрибуты поиска заказа;
- найти заказ по текущим атрибутам поиска;
- открыть текущий заказ на редактирование;
- просмотреть список клиентов;
- просмотреть список товаров;

Все эти операции пользователь может выполнить с помощью команд, собранных в разделах главного меню.

Выполним на примере первой из операций, связанной с созданием нового заказа, расчет времени по правилам GOMS.

При использовании команды меню эта операция раскладывается на следующие действия:

- 1) перемещение руки к мыши, В;
- 2) указание на раздел меню Действия, П;
- 3) нажатие клавиши мыши, М;
- 4) указание на команду Создать, П;
- 5) нажатие клавиши мыши, М.

В соответствии с правилом 0 расстановки ментальных операторов Д получим следующую последовательность операторов

В Д П Д М Д П Д М

В соответствии с правилом 1 следует удалить ожидаемые операторы Д

В Д П М Д П М

Остальные правила в этом примере не используются. Складывая соответствующие значения операторов получим общее время

$0,4+1,2+1,1+0,1+1,2+1,1+0,1=5,2$  с.

Очевидно, что это время велико. Необходимо предусмотреть другие варианты выполнения команды. Например, использование «горячих» клавиш или использование командной кнопки, запускающей команду на выполнение. Произведем оценку этих вариантов.

При использовании командной кнопки операция по созданию нового заказа раскладывается на действия:

- 1) перемещение руки к мыши, В;
- 2) указание на командную кнопку Создать, П;
- 3) нажатие клавиши мыши, М.

При расстановке операторов Д здесь также используются правила 0 и 1. Результирующая последовательность выглядит как

В Д П М

Общее время на выполнение операции составляет

$0,4+1,2+1,1+0,1=2,8$  с.

Это почти в 2 раза быстрее, чем предыдущий вариант выполнения команды.

При использовании «горячих» клавиш, например традиционного для создания нового объекта сочетания клавиш Ctrl+N, операция состоит из следующих действий:

- 1) перемещение руки к клавиатуре, В;
- 2) нажатие клавиши Ctrl, К;

3) нажатие клавиши N, K.

Согласно правилу 0 получаем последовательность операторов

В Д К Д К

Правило 1 в данном случае не работает, зато используется правило 2, согласно которому удаляются операторы Д внутри когнитивных единиц, каковой является комбинация клавиш Ctrl+N. Результирующая последовательность представлена как

В Д К К

Это соответствует времени

$$0,4+1,2+2\cdot0,28=2,16 \text{ с.}$$

Последний вариант самый быстрый. Однако рассматриваемая категория пользователей обладает низкой мотивацией к обучению, это означает, что пользователи могут не знать нужного сочетания клавиш, и они не стремятся ни узнать, ни запомнить его. Поэтому в интерфейсе желательно предусмотреть оба «быстрых» варианта выполнения команды.

Рассмотрим другую операцию, связанную с открытием текущего заказа. Очевидно, что она во многом схожа с предыдущей, однако требует задания аргумента – записи в журнале заказов, которая будет признана текущей. Для перемещения по журналу можно использовать клавиши клавиатуры **Home**, **↑**, **↓**, **End** или традиционные команды (командные кнопки) навигации по базе данных:

- перейти к первой записи;
- перейти к предыдущей записи;
- перейти к следующей записи;
- перейти к последней записи.

Рассмотрим оба этих варианта.

При использовании клавиш клавиатуры имеем:

- 1) перемещение руки к клавиатуре, В;
- 2) нажатие клавиши **Home**, К;
- 3) нажатие n раз клавиши **↓**, К.

Результирующая последовательность операторов

В Д К Д К Д К Д К...

Нажатия клавиши не составляют единой когнитивной единицы, операторы Д исключить нельзя. После каждого нажатия пользователь принимает решение, следует ли ему нажать на клавишу повторно. Расчет по времени

$$0,4+1,2+0,28+n\cdot(1,2+0,28)=1,88+n\cdot1,68 \text{ с.}$$

При использовании командных кнопок навигации получим:

- 1) перемещение руки к мыши, В;
- 2) нажатие кнопки **Перейти к первой записи**, М;
- 3) нажатие n раз кнопки **Перейти к следующей записи**, М.

Результирующая последовательность составляется аналогичным образом

В Д П М Д П М Д М Д М...

Расчет по времени

$$0,4+1,2+1,1+0,1+1,2+1,1+0,1+(n-1)\cdot(1,2+0,1)=5,2+(n-1)\cdot1,3 \text{ с}$$

В зависимости от расположения нужной записи в журнале (значения n) предпочтительным может оказаться тот или иной вариант.

Гораздо проще для пользователя вариант с использованием техники прямого манипулирования, с использованием скроллинга - прокрутки. Рассмотрим, какие шаги должен выполнить пользователь в этом случае:

- 1) перемещение руки к мыши, В;
- 2) указание на область прокрутки, П;
- 3) нажатие клавиши мыши, М;
- 4) прокручивание скроллинга, С;
- 5) указание на нужную запись списка, П;
- 6) нажатие на клавиатуру мыши, М.

Здесь необходимо дать некоторые пояснения. При выполнении шага (3), осуществляется нажатие на клавишу мыши и ее удерживание. Время, поставленное в соответствие оператору М, учитывает как нажатие, так и отпускание клавиши, поэтому шаг (3) реально выполняется за 0,05 с (М/2). Но тогда после выполнения шага (4), необходимо учесть время, требуемое для отпускания клавиши мыши, что составляет 0,05 с (М/2). В приведенной выше последовательности действий, нажатие и отпускание мыши объединено в один шаг. Такое объединение становится возможным, потому что, согласно правилу 1 в последовательности

(М/2) С (М/2)

операторы Д не вставляются.

Время необходимое на прокручивание скроллинга можно оценить только экспериментальным путем. Приблизительно можно считать, что для его выполнения необходимо 3 с.

Результирующая последовательность операторов выглядит как

В Д П М С Д П М

Это соответствует оценке времени

$0,4 + 1,2 + 1,1 + 0,1 + 3 + 1,2 + 1,1 + 0,1 = 8,2$  с.

При «удачном» расположении записи в журнале, пользователю вообще не придется работать с прокруткой. Последовательность операторов его действий будет выглядеть как

В Д П М

что составляет по времени 2,8 с.

Вариант с использованием прокрутки можно считать более предпочтительным. Но учитывая низкий уровень владения компьютером и минимальный предварительный тренинг, стоит предусмотреть также вариант с использованием клавиш  $\uparrow, \downarrow$ .

Аналогичные рассуждения можно привести относительно других операций.

Примерный вид экранной формы **Журнал заказов** приведен на рисунке 1.

Номер	Дата	Клиент	Сумма
1	06.04.04	АКБ и Т	100
2	06.04.04	ООО НЕФТЕХИМСНАБ	200
3	06.04.04	ООО УРАЛЭЛЕКТРОТЕ	2000
4	06.04.04	АПТЕКА № 30	15000
5	07.04.04	ООО РЕГИОНТОРГ	200
6	07.04.04	ЧП Янбаев	1000
7	07.04.04	ООО ПРИМПРОЕКТ	500
8	07.04.04	ООО ШАТЛЫК	500
9	07.04.04	ООО ТОМАС-Е	400

Рисунок 1 – Экранная форма **Журнал заказов**

Рассмотрим, каким образом дается оценка производительности элементов интерфейса данной формы. В качестве примера выберем предложенную реализацию для операции поиска по данным справочников. Предусмотрено два варианта справочников: клиентов и товаров. Причем, для задания атрибутов поиска предполагается выполнять переход к формам этих справочников. В начальном прототипе формы переход осуществлялся путем нажатия на командную кнопку **Просмотреть** или командную кнопку **Найти**. Будем считать равновероятным выбор одного из двух справочников; вероятность использования кнопки



**Просмотреть** оценим в 30% ( $p=0,3$ ), кнопки **Найти** – 70% ( $p=0,7$ ). Вероятности различных вариантов составят:

- выбор клиента по кнопке **Просмотреть** –  $0,5 \cdot 0,3 = 0,15$ ;
- выбор товара по кнопке **Просмотреть** –  $0,5 \cdot 0,3 = 0,15$ ;
- выбор клиента по кнопке **Найти** –  $0,5 \cdot 0,7 = 0,35$ ;
- выбор товара по кнопке **Найти** –  $0,5 \cdot 0,7 = 0,35$ .

Информационное содержание рассматриваемого фрагмента интерфейса как

$$0,15 \cdot \log_2 \left( \frac{1}{0,15} \right) + 0,15 \cdot \log_2 \left( \frac{1}{0,15} \right) + 0,35 \cdot \log_2 \left( \frac{1}{0,35} \right) + 0,35 \cdot \log_2 \left( \frac{1}{0,35} \right) = 0,41 + 0,41 + 0,53 + 0,53 = 1,88$$

Теоретически, если пользователь решил выполнить функцию поиска, ему минимально необходимо только определить каким из двух справочников он желает воспользоваться. Следовательно, минимальное количество информации необходимое ему для решения этой задачи определяется как

$$\log_2 2 = 1$$

Информационная производительность

$$E = \frac{1}{1,88} = 0,53$$

Повысить информационную производительность можно, для этого следует предусмотреть объединить обе команды или осуществлять переход к соответствующей экранной форме непосредственно при выборе пользователем нужного ему справочника (рисунок 1).

Оценить символьную эффективность можно только для тех элементов интерфейса, в которых осуществляется ввод символьной информации. Этот метод количественного анализа не представляет особых сложностей.

## 5 Требования к оформлению отчета

Отчет должен содержать:

- название и цели работы;
- внешний вид разработанных экранных форм с описанием навигации по ним;
- описание навигации по приложению с использованием элементов управления на спроектированных формах;
- расчеты количественного анализа, выполненного для указанных фрагментов интерфейса;
- выводы относительно необходимости корректировки проектируемых форм; граф состояний меню;
- при необходимости скорректированные формы;
- общие выводы, сделанные в процессе выполнения лабораторной работы.