Учреждение образования

«Белорусский государственный технологический университет»

**Кафедра информатики и веб-дизайна**

**Лабораторная работа №6**

Разработка пользовательского интерфейса:

этапы первоначального проектирования

Выполнил:

студент 2 курса 2 группы ФИТ

Аникеенко Егор Вячеславович

**Разработка пользовательского интерфейса: этапы первоначального проектирования**

**Цель работы:** закрепить теоретические знания по разработке пользовательского интерфейса. Получить практические навыки по разработке целей проекта, созданию профилей пользователя, составлению сценариев, по определению функциональных блоков приложения, составлению навигационной схемы системы и диаграммы вариантов использования.

**Описание предметной области:**

Одним из существенных видов деятельности любого предприятия является бухгалтерский учет. Моей задачей является отслеживание внутриофисных расходов. В зависимости от вида расходов сотруднику выдается определенный бюджет, который можно потратить. Каждый вид расходов имеет название, некоторое описание и предельную сумму средств, которые могут быть потрачены по данному виду расходов в месяц. При каждой покупке сотрудник оформляет документ, где указывает вид покупки, свой код сотрудника, вид расхода, дату, сумму и отдел, где он работает. Нормативы по расходованию средств устанавливаются не в целом, а по каждому отделу за каждый месяц. Есть таблица с информацией по отделам с лимитом средств на определенные внутриофисные расходы.

Нужно хранить данные о расходах не только в целом по отделу, но и по отдельным сотрудникам.

Необходимо предусмотреть формирование отчета о состоянии дел на фирме по внутриофисным расходам.

**Описание целей проекта:**

Цели создания системы внутриофисных расходов:

• Оптимизация управления финансовыми и информационными потоками компании;

• Сокращение расходов за счёт значительного снижения общего числа рутинных операций, выполняемыми сотрудниками компании;

• Необходимо спроектировать удобный для сотрудников фирмы интерфейс.

**Описание целевой аудитории и основные профили пользователей:**

Бухгалтеры: бухгалтеры анализируют, планируют и оценивают расходы и доходы компании. Главный бухгалтер, в свою очередь, руководит всем бухгалтерским отделом, осуществляет контроль за бухгалтерским учетом, а также за экономичным использованием материальных, трудовых и финансовых ресурсов компании.

Пол: любой. Возраст: от 22 до 60 лет. Места проживания: русскоязычные страны (Беларусь, Россия, Украина и др.). Уровень владения компьютером: выше среднего. Требования к системе: удобство и простота в использовании (внос информации о стоимости продуктов, веса всех продуктов в отдельности, заполнение форм и др.).

Сотрудники фирмы: сотрудники фирмы могут осуществлять покупки в соответствии с видом расходов.

Пол: любой. Возраст: от 18 до 60 лет. Места проживания: русскоязычные страны (Беларусь, Россия, Украина и др.). Уровень владения компьютером: любой. Требования к системе: удобство и простота в использовании (просмотр информации о необходимых блюдах для приготовления, продуктах необходимых для блюда, общего выхода блюд и количества порций конкретного блюда).

**Описание сценариев пользователей:**

Персонаж первой целевой аудитории:

1) Бухгалтер принимает отчеты о внутриофисный покупках сотрудников фирмы для дальнейшей их обработки.

2) Бухгалтер имеет доступ к таблицам с внутриофисными расходами, как всего отдела в общем, так и по каждому сотруднику в отдельности для дальнейшей обработки данной информации.

Персонаж второй целевой аудитории:

1) Для покупки какого-либо товара, либо получения услуги от лица фирмы сотрудник должен заполнить документ, где указывается вид покупки, свой код сотрудника, вид расхода, дату, сумму и отдел, где он работает.

**Описание функциональности приложения, функциональных блоков и операций, выполняемых пользователями в рамках возможностей, предоставляемых ему приложением:**

Описанным выше персонажам в приложении необходимо, чтобы можно было выполнять следующие функции:

1) Создать заявку для покупки;

2) Поиск информации об определенном сотруднике;

3) Поиск информации об определенном отделе;

4) Просмотр таблицы с информацией о покупках;

5) Установка фильтров для таблицы;

6) Сформировать отчет о покупках всех отделов за месяц;

7) Сформировать отчет о покупках определенного отдела за определенный промежуток времени;

8) Сформировать отчет о покупках определенного сотрудника за определенный промежуток времени;

9) Редактирование ошибочных заявок;

10) Удаление ошибочных заявок;

11) Установка лимита на определенный вид покупок;

12) Печать необходимого отчета.

Например, функциональные блоки, соответствующие работе пользователей с:

- заявками (функция 1): через заполнение формы;

- отчетами (функции 6, 7, 8, 12): через заполнение формы;

- базой данных (функции 2, 3, 4, 5, 9, 10): по списку;

- лимитами (функция 11): через заполнение формы.

Таким образом, можно вести речь о наличии в приложении 5 экранных формах:

- Оформить заявку;

- Сформировать отчет;

- База данных;

- Установка лимитов;

- Отчет на печать.

Операции, которые может выполнять пользователь (включая администратора):

1) Отправка заявки;

2) Открыть любую запись базы данных на редактирование;

3) Удаление любой записи из базы данных;

4) Отфильтровать записи;

5) Формирование отчета;

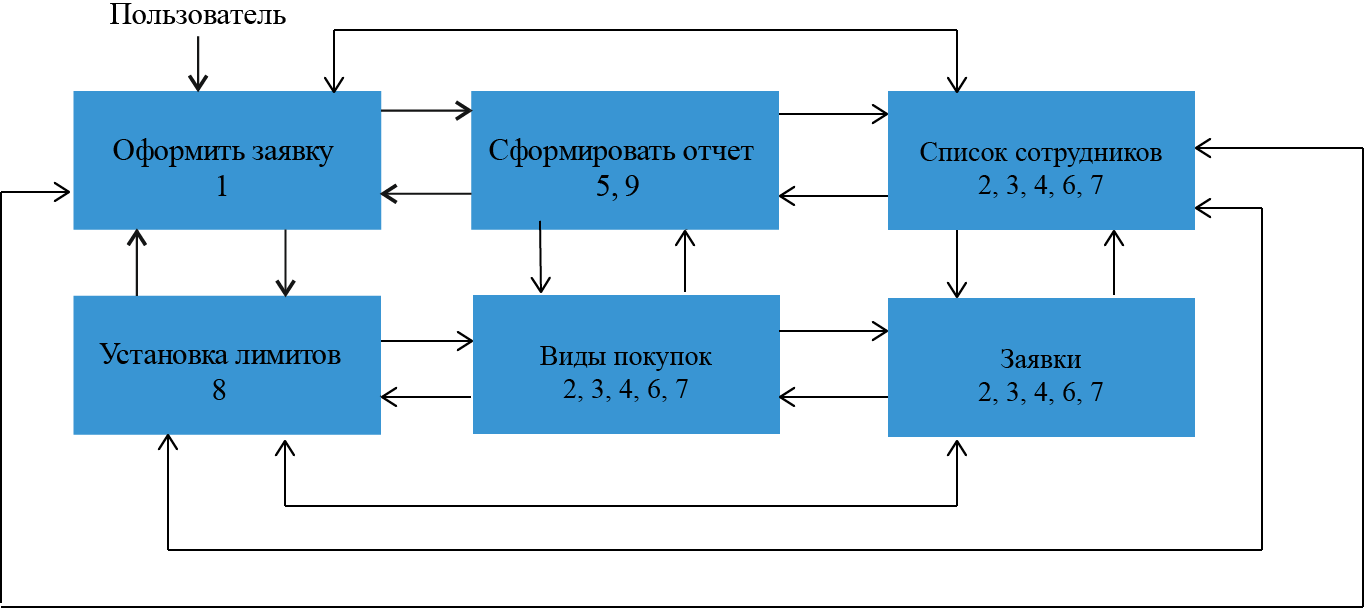
6) Удаление записей в базе данных;

7) Редактирование записей в базе данных;

8) Установка лимита для определенного отдела;

9) Отправка отчета на печать.

Навигационная схема приложения будет следующей:



Структурная схема приложения:



**Ответы на контрольные вопросы:**

1. Из каких этапов состоит работа над пользовательским интерфейсом?

Описание предметной область будущего приложения;

* Цели;
* Определение целевой аудитории вашего приложения;
* Формулировка профилей пользователей;
* Разработка нескольких сценариев для решения задач пользователей;
* Определение функциональности приложения;
* Описание функциональных блоки и операций, выполняемых пользователями в рамках возможностей, предоставляемых ему приложением;
* Составление навигационной схемы приложения;
* Группировка операций, выполняемых пользователями, и составление главного меню;
* Составление структурной схемы приложения.

1. Какие вопросы рассматриваются при сборе функциональных требований?

Какие цели и задачи мы ставим перед программным продуктом?

Кто является главным пользователем в этом сценарии? Какую роль он играет?

Работал ли пользователь ранее с данным программным продуктом?

Какие срочные потребности заставили обратиться к данной программе?

Что он пытается сделать и зачем?

Участвуют ли в решении задачи другие люди?

Что произойдет, когда пользователь завершит решение своей задачи?

1. Каких два способа существуют для определения функциональности?

Современная наука выдвинула два основных способа определения функциональности, а именно анализ целей и анализ действий пользователей.

1. Какой должна быть цель?

Понятной. Избегайте использования узко специализированной терминологии.

Ясной. Избегайте туманных формулировок; подбирайте выражения, которые были бы уместными при определении приоритетов требований.

Измеримой. Используйте конкретные утверждения, которые можно проверить независимо, чтобы определить степень успешности проекта.

1. Чем отличается цель от задачи?

Цель – это конечное состояние, тогда как задача – переходный процесс, необходимый для достижения цели.

Цель – стабильная сущность. Задачи – преходящи.

1. Какие существуют виды целей?

Существуют следующие виды целей: личные, практические, корпоративные, ложные.

Личные цели:

• не чувствовать себя глупо;

• не совершать ошибок;

• выполнить адекватный объем работы;

• развлечься (или хотя бы не страдать от скуки).

Корпоративные цели:

• увеличить прибыль;

• увеличить рыночную долю;

• победить конкурентов;

• нанять больше сотрудников;

• предложить новые продукты и услуги;

• выпустить акции компании в свободное обращение.

Практические цели:

• удовлетворять требованиям клиента;

• сохранять информацию о заказах клиента;

• создавать математические модели бизнеса.

Ложные цели:

• экономия памяти;

• уменьшение потребности в клавиатурном вводе;

• поддержка работы в браузере;

• простота в освоении;

• обеспечение целостности данных;

• ускорение ввода данных;

• увеличение скорости исполнения программы;

• применение супертехнологии или супервозможностей;

• улучшение внешнего вида;

• сохранение единообразия интерфейса на различных платформах.

1. В чем состоит сущность качественного проектирования взаимодействия?

Сущность качественного проектирования взаимодействия состоит в том, чтобы позволить пользователям достигать практических целей, не отказываясь от целей личных.

1. Каковы правила определения функциональности?

«Чем меньше действий требуется от пользователя, тем лучше». Не стоит забывать и про другое правило: чем меньше функций, тем легче их сделать.

1. Каким должен быть Персонаж?

Персонаж должен быть конкретным. Чем более конкретными мы делаем персонажи, тем более эффективными инструментами проектирования они становятся. Для этого мы выбираем ему имя.

Персонаж должен быть воображаемым. Описание его должно быть подробным, а не идеальным. То есть важнее определить персонаж насколько возможно подробно и конкретно, чем создать абсолютно правильный персонаж.

1. Какие типы сценариев, основанных на персонажах, используются на различных этапах проектирования?

Чтобы создать сценарий, проанализируйте следующие вопросы:

• Кто является главным пользователем в этом сценарии?

• Посещал ли выбранный пользователь этот сайт ранее (работал ли он ранее с данным программным продуктом)?

• Какие срочные потребности привели пользователя на сайт (либо заставили обратиться к данной программе)?

1. Из чего состоит проектирование общей структуры?

Проектирование общей структуры состоит из двух параллельно происходящих процессов: выделения независимых блоков и определения связи между ними. Если проектируется сайт, в завершении необходимо также создать схему навигации.

1. Какие виды связи существует между блоками?

Логическая связь определяет взаимодействие между фрагментами системы с точки зрения разработчика (суперпользователя). Данная связь очень существенно влияет на навигацию в пределах системы (особенно, когда система многооконная). Чтобы не перегружать интерфейс стоит избегать блоков, связанных с большим количеством других – оптимальным числом связей является число три.

Связь по представлению пользователей. В информационных системах, когда необходимо гарантировать, что пользователь найдет всю нужную ему информацию, необходимо устанавливать связи между блоками, основываясь не только на точке зрения разработчика, но и на представлениях пользователей.

Процессуальная связь описывает пусть не вполне логичное, но естественное для имеющегося процесса взаимодействия: например, логика напрямую не командует людям сначала приготовить обед, а потом съесть его, но обычно получается именно так.