**Лабораторная работа № 1**

**Тема работы: «Исследование предметной области на основе анализа поставленной задачи»**

**1. Цель работы**

Сформировать умения анализировать поставленную задачу, исследовать предметную область.

**2. Задание**

Сформулировать и обсудить тему проекта, проанализировать поставленную задачу, уточнить детали. Провести анализ задачи, определить предметную область выбранной темы проекта.

**3. Оснащение работы**

**Примерный перечень тем разработки:**

1. Интернет-магазин. Должны быть реализованы сценарии: покупка товара, поиск товара, добавление нового товара в базу данных магазина, просмотр и обработка заказов покупателей, регистрация нового покупателя.

2. Книжный каталог. Должны быть реализованы сценарии: добавления новой книги, поиск книги по нескольким полям, бронирование книги, списание старых книг, регистрация пользователей каталога.

3. Адресная книга. Должны быть реализованы сценарии: добавление нового абонента, добавление категорий абонентов, поиск абонентов по нескольким полям, добавления администраторе каталога (пользователей, которые имеют право редактировать данные адресной книги), редактирование данных абонента.

4. Расписание занятий. Должны быть реализованы сценарии: добавление новой группы, добавление занятий (с указанием названия предмета, времени, аудитории, группы, недели, преподавателя, типа занятия), просмотр списка занятий на выбранную дату, добавление списка преподавателей, поиск занятий по нескольким полям (предмету, преподавателя, группе, времени, типе занятия).

5. База студентов. Должны быть реализованы сценарии: добавление новой группы, добавление нового студента, поиск студента по различным полям, добавления информации об оценках по различным предметам, отчисление студента.

6. Прайс-лист фирмы. Должны быть реализованы сценарии: добавление новой категории товаров, добавление нового товара, поиск товара по различным полям, добавление администратора прайс-листа (пользователей, которые имеют право редактировать прайс-лист), перемещение товара из одной категории в другую.

7. База склада фирмы. Должны быть реализованы сценарии: добавление нового товара на склад, списание товара, выдача товара, поиск товара по различным полям, изменение месторасположения товара на складе.

8. Аптечная база. Должны быть реализованы сценарии: прием заказа от клиента на изготовление раствора, продажа лекарства, списание просроченных лекарств, добавление новые лекарств в базу данных, поиск заказов по различным полям.

**4. Основные теоретические сведения**

Одним из первых действий при создании программного продукта (ПП) является сбор и систематизация требований к нему.

**Требование** – это условие или характеристика, которым должна удовлетворять система. Требования бывают функциональные и нефункциональные.

**Функциональные требования** определяют действия, которые должна выполнять система, без учета ограничений, связанных с реализацией. Т.е. функциональные требования определяют поведение системы в процессе обработки информации.

**Нефункциональные требования** описывают атрибуты разрабатываемого ПП или атрибуты системного окружения.

На базе требований, предъявляемых к разработке, формируется словарь предметной области.

**Словарем предметной области** будем называть набор понятий для данной задачи.

На начальной стадии разработки разработчикможет и должен:

* получить достаточно информации о предметной области решаемой задачи;
* иметь возможность выяснять и уточнять детали в процессе анализа;
* предоставлять ориентировочные, оценки трудозатрат на каждый из этапов жизненного цикла ПО;

- изучить аналоги с целью их дальнейшего использования в процессе создания ПП.

При разработке ПП на базе аналога можно выделить следующие основные этапы:

* построение каркаса модели;
* изучение уже существующих систем данной предметной области;
* определение схожести и различий между разрабатываемой системой и уже существующими;
* пересмотр существующей системы или ее компонентов на основе предъявляемых требований.

**Сценарии использования**

**Сценарий** – это один из способов описания структуры задачи. Это повествовательный рассказ о совершаемых действиях, это история, эпизод, происходящий в данных временных рамках и в данном контексте. Различные формы сценариев широко применяются при разработке программного обеспечения. Сценарии задач и взаимодействий обычно богаты характеристиками и обладают высокой реалистичностью.

Сценарии при разработке пользовательского интерфейса описывают взаимодействие между пользователем (или типом пользователей) и системой. Обыкновенные сценарии обладают некоторыми серьезными ограничениями при попытке использовать их для проектирования пользовательского интерфейса. В них делается упор на реалистичность и детали, при этом на серьезные проблемы и общую организацию обращается недостаточно внимания. Сценарии включают в себя правдоподобные описания комбинаций отдельных действий и задач, поэтому часто бывает тяжело выделить и понять основную суть взаимодействия.

**Модели use case**

Концепция моделей use case впервые была применена для разработки ПО Айваром Якобсоном в качестве составной части его объектноориентированного подхода к программной инженерии. Успех модели оказался столь значительным, что со временем произошла интеграция элементов use case практически во все основные методы объектно-ориентированного анализа и проектирования. Несмотря на то что модель была разработана специально для проектирования объектно-ориентированного ПО, ничего особенно «объектноориентированного» в элементах use case нет, поэтому их можно применять практически ко всем подходам к проектированию.Элемент use case – это ситуация, вариант использования, то есть некоторый случай применения системы. По сути use case это:

* обеспечение функциональности;
* сугубо внешняя точка зрения (принцип «черного ящика»);
* повествовательное описание;
* описание взаимодействия между пользователем (в какой-то роли) и системой;
* завершенное и понятное пользователю применение системы.

Каждый элемент use case описывает в повествовательной форме завершенное, хорошо определенное взаимодействие, имеющее ясную цель с точки зрения пользователя. При объектно-ориентированном подходе элементы use case могут описывать взаимодействие с другими системами и оборудованием, а не только с живыми пользователями. Тем не менее, когда целью является разработка пользовательского интерфейса, можно совершенно спокойно ограничиться рассмотрением только тех элементов use case, которые относятся к взаимоотношениям человека и системы.

**Сущностные элементы use case**

Сущностный элемент **use case** – это структурированное повествование, выраженное на языке данной предметной области и пользователей системы и содержащее упрощенное, обобщенное, абстрактное, не зависящее от технологии и реализации описание одной завершенной, наполненной смыслом и хорошо определенное с точки зрения пользователей задачи или взаимодействия. Предполагается, что пользователь играет определенную роль по отношению к системе, а в описании воплощаются цели и замыслы лежащего в его основе взаимодействия. Сущностные элементы use case строятся на основе целей и задач пользователя, а не на основе каких-то конкретных механизмов или этапов, ведущих к достижению этих целей. Некоторым кажется значимым включение целей пользователей в модели use case, но это не должно быть связано с упрощениями, присущими сущностному моделированию. При использовании подхода, ориентированного на удобство использования, в сущностных элементах use case, являющихся структурированным описанием, можно выделить три части: изложение общих устремлений пользователя, выраженное в элементе use case, плюс состоящее из двух частей описание, включающее в себя модель пользовательских устремлений и модель обязательств системы. Сущностные элементы use case именуются, причем при помощи этих имен стараются выразить пользовательские намерения в условиях данного варианта использования. В соответствии с соглашением, предложенным Якобсоном, элемент use case изображается в виде эллипса с именем элемента.

**Описание вариантов использования**

Функциональные требования к системе моделируются и документируются с помощью вариантов использования (use case). Вариант использования (use case) – связный элемент функциональности, предоставляемый системой при взаимодействии с действующими лицами. Действующее лицо (actor) – роль, обобщение элементов внешнего окружения системы, ведущих себя по отношению к системе одинаковым образом. В контексте процесса управления требованиями варианты использования трактуются следующим образом (согласно Коберну):

* вариант использования фиксирует соглашение между участниками проекта относительно поведения системы;
* вариант использования описывает поведение системы при различных условиях, когда система отвечает на запрос одного из участников, называемого основным действующим лицом;
* основное действующее лицо инициирует взаимодействие с системой, чтобы добиться некоторой цели. Система отвечает, соблюдая интересы всех участников. Варианты использования – это вид документации, применяемый, когда требуется сконцентрировать усилия на обсуждении принципиальных требований к разрабатываемой системе, а не на подробном их описании. Стиль их написания зависит от масштаба, количества участников и критичности проекта.

В общем случае, рекомендуется придерживаться следующих правил:

* названия вариантов использования должны быть деловыми (нетехническими) терминами, имеющими значение для заказчика;
* каждый вариант использования должен представлять собой завершенную транзакцию между пользователем и системой, представляющую для первого некоторую ценность;
* хорошо написанный вариант использования легко читается и состоит из предложений, написанных в единой грамматической форме. На обучение чтению варианта использования не должно уходить больше нескольких минут.

**Формат описания варианта использования (по Коберну):**

1. Имя – цель в виде краткой активной глагольной фразы.

2. Контекст использования – более длинное описание цели.

3. Область действия.

4. Уровень точности.

5. Основное действующее лицо.

6. Другие участники и их интересы.

7. Предусловие (определяет, выполнение какого условия гарантирует система перед тем, как разрешить запуск варианта использования).

8. Минимальные гарантии (наименьшие обещания системы участникам, в частности, когда цель основного действующего лица не может быть достигнута).

9. Гарантии успеха (или постусловие – postcondition – устанавливает, что интересы участников удовлетворяются по успешном завершении варианта использования в конце основного сценария).

10.Триггер (событие, которое запускает вариант использования).

11.Основной сценарий или поток (простой для понимания типичный сценарий, в котором достигается цель основного действующего лица и удовлетворяются интересы всех участников). Каждый шаг основного сценария описывает:

* взаимодействие двух действующих лиц ("Клиент вводит адрес");
* шаг подтверждения для защиты интереса участника ("Система подтверждает PIN-код");
* внутреннее изменение для удовлетворения интереса участника ("Система выводит сумму из баланса").

12.Расширения (запускаются при возникновении определенного условия, содержат последовательность шагов, описывающих, что происходит при этом условии, и заканчивается достижением цели или отказом от неё).

13.Список изменений в технологии и данных.

14.Вспомогательная информация.

**5. Порядок выполнения работы**

1. Выбрать тему разработки.

2. Описать цели, задачи, результат выбранной темы проекта.

3. Описать предметную область задачи.

4. Составить 3 сценариев использования программного обеспечения пользователем согласно формату описания Коберна.

**6. Форма отчета о работе**

*Лабораторная работа № \_\_\_*

*Номер учебной группы \_\_\_\_\_\_\_\_\_\_\_\_*

*Фамилия, инициалы учащегося*

*Дата выполнения работы*

*Тема работы:*

*Цель работы:*

*Оснащение работы:*

*Результат выполнения работы:*

**7. Контрольные вопросы и задания**

1. Дать определение понятий: требования, функциональные требования, нефункциональные требования.

2. Определить словарь предметной области задачи.

3. Каким образом производится моделирование задач?

4. Что такое сценарий использования?

5. Что такое элемент use case?

6. Что такое сущностные элементы use case?

7. Чем отличаются сценарии использования от модели use case?

8. Каким образом можно описать варианты использования?

9. Приведите пример описания варианта использования по Коберну?

**8. Рекомендуемая литература**

**Гагарина, Л. Г., Кокорева, Е. В., Виснадул ,Б. Д.** Технология разработки программного обеспечения: учебное пособие / под ред. Л. Г Гагариной. – М.: ИД «ФОРУМ»: ИНФРА-М, 2008.

**Орлов, С. А.** Технология разработки программного обеспечения / С. А. Орлов. – СПб.: Питер, 2002.