

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Таныгин Максим Олегович
Должность: и.о. декана факультета фундаментальной и прикладной информатики
Дата подписания: 21.09.2023 12:41:55
Уникальный программный ключ:
65ab2aa0d384efe8480e6a4c688eddbc475e411a

МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Юго-Западный государственный университет»
(ЮЗГУ)

Кафедра программной инженерии

УТВЕРЖДАЮ
Проректор по учебной работе



О.Т. Дюконинова

« 21 » 09

**РАЗРАБОТКА ТРЕБОВАНИЙ К ИНФОРМАЦИОННОЙ ЧАСТИ
ПРОГРАММНО-ИНФОРМАЦИОННОЙ СИСТЕМЫ**

Методические указания к практическим занятиям
по дисциплине «Разработка и анализ требований» для студентов всех
форм обучения направления подготовки бакалавров
09.03.04 «Программная инженерия»

Курск 2023

УДК 004.414

Составитель: Е.А. Петрик

Рецензент

Кандидат технических наук, доцент Т.И. Лапина

Разработка требований к информационной части программно-информационной системы: методические указания к практическим занятиям по дисциплине «Управление программными проектами» для студентов всех форм обучения направления подготовки бакалавров 09.03.04 «Программная инженерия» / Юго-Зап. гос. ун-т; сост.: Е. А. Петрик. Курск, 2023. 12 с. Библиогр.: с.12.

Изложена последовательность действий при разработке требований к информационной части программно-информационной системы. Рассмотрено определение функциональных возможностей системы и составление перечня требований на основе необходимого функционала.

Предназначены для студентов всех форм обучения направления подготовки бакалавров 09.03.04 «Программная инженерия»

Текст печатается в авторской редакции

Подписано в печать Формат 60x84 1/16.
Усл. печ. л. Уч. – изд. л. .Тираж экз. Заказ . Бесплатно.
Юго - Западный государственный университет.
305040, г. Курск, ул. 50 лет Октября, 94.

Теоретическая часть

1.1 Описание предметной области

Проводится исследование предметной области разрабатываемой системы. Составляется словесное описание действующих лиц, их характеристик и процессов взаимодействия, а также других объектов и их свойств. Взаимодействие действующих лиц описывается поверхностно, без серьёзной детализации. Результатом исследования является неформальное описание предметной области.

1.2 Составление словаря предметной области

Словарь предметной области – документ, в котором содержатся определения всех терминов, использующихся в рамках данной предметной области. Словарь предметной области составляется на основе её неформального описания.

1.3 Определение концептуальных классов

На основе неформального описания предметной области и словаря предметной области определяются концептуальные классы системы. Концептуальные классы описывают объекты реального мира и их свойства, которые описываются атрибутами классов.

1.4 Построение статической модели предметной области

Объекты, моделями которых являются определённые концептуальные классы, в реальном мире взаимодействуют между собой. Поэтому после определения концептуальных классов необходимо при помощи неформального описания предметной области определить зависимости между ними. Совокупность концептуальных классов и зависимостей между ними составляет статическую модель предметной обла-

сти, которая визуально представляется в виде диаграммы концептуальных классов.

1.5 Формирование требований к информационной части системы

На основе статической модели предметной области формируются требования к информационной части системы. Они определяют модель данных, которая будет использоваться в системе. Требования следует записывать в декларативной форме, то есть каждое требование определяет, что должна делать система или другие действующие лица. Все требования собираются в документ, на основе которого в дальнейшем происходит разработка программного продукта.

Пример разработки требований

Тема: «Программа для моделирования работы лифта».

Неформальное описание предметной области

Моделирование работы лифта является достаточно популярной задачей в программировании. Система управления лифтом представляет собой сложную взаимосвязь многих элементов: пассажир, этаж отправления, направление, этаж назначения и, собственно, сам лифт, который в свою очередь можно разбить на множество мелких деталей: кабина лифта, двигатель, двери, противовесы, с помощью чего лифт может опускаться и подниматься. Кроме того, управление лифтом может осуществляться как извне пассажирами на этажах, так и изнутри с помощью панели управления.



Рисунок 1 - Кнопочная панель управления

Лифт - разновидность грузоподъёмной машины, предназначенная для вертикального перемещения грузов.

При работе лифта нужно учитывать количество этажей в здании, их высоту, грузоподъемность лифта. Перемещение в реальном времени можно отследить с помощью дисплея, отображающего состояние лифта.



Рисунок 2 - Изображение лифта

Словарь объектов предметной области приложения

На основе анализа описания предметной области были определены основные понятия и разработан словарь объектов предметной области, представленный в таблице 1.

Таблица 1 – Словарь предметной области

Понятие	Описание
Лифт	Лифт - разновидность грузоподъёмной машины, предназначенная для вертикального или наклонного перемещения грузов.
Этаж	Этаж - часть пространства здания между двумя горизонтальными перекрытиями (между полом и потолком).
Запрос	Запрос - обращение на предмет выполнения устанавливаемых требований.
Кабина	Кабина - закрытое грузонесущее устройство, предназначенное для транспортировки пассажиров и грузов
Грузоподъёмность	Грузоподъёмность — масса груза, на перевозку которого рассчитано транспортное средство.
Направление	Направление движения — направление (возможно, усреднённого) вектора скорости объекта
Пассажир	Пассажир – человек, совершающий поездку.
Панель управления	Панель управления лифта – это специальное устройство, которое расположено внутри кабины лифта и которое предназначено для управления и контроля движения кабины лифта.

Моделирование предметной области с помощью диаграммы классов

На основе анализа словаря предметной области приложения и описания вариантов использования были определены классы, атрибуты и методы классов и отношения классов, представленные на рисунке 3 в виде диаграммы концептуальных классов.

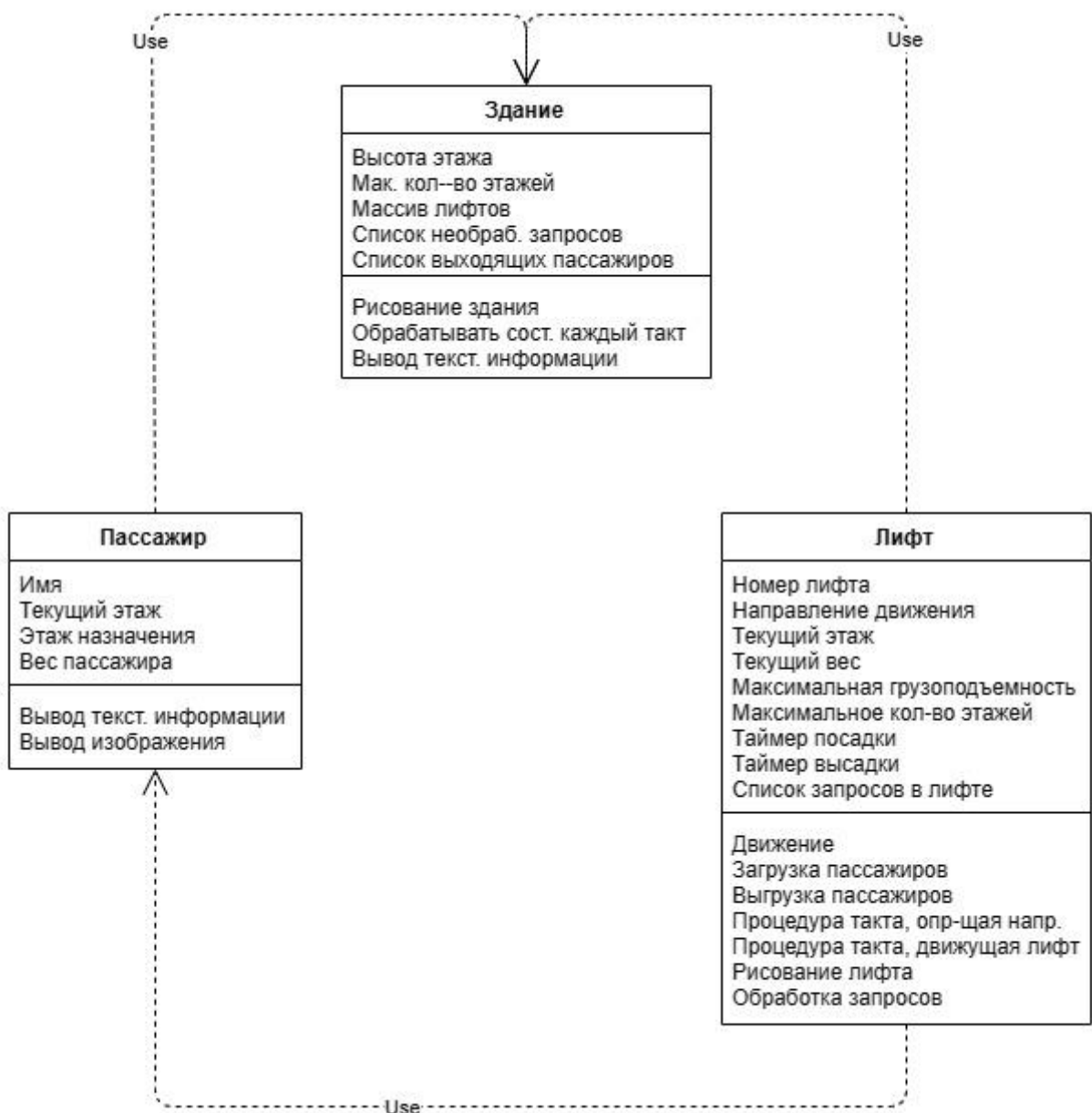


Рисунок 3 – Диаграмма концептуальных классов предметной области программы для моделирования работы лифта

Описание класса Здание:

– Общедоступные поля класса:

int height_floor – высота этажа.

int max_floor – максимальное количество этажей.

Lift[] mas_lift – массив лифтов.

List<Request> List_Request0 – список необработанных запросов (пассажиры ждут лифт на своих этажах).

List<Request> List_Request_2 – список запросов, обработанных лифтом, пассажиры на выходе из лифта.

– Общедоступные методы класса:

Draw(Graphics g) – рисует здание, пассажиров, информацию о них.

Метод ничего не возвращает.

Master_tick() – процедура обработки лифтов каждый такт, использует методы Lift_tick1() и Lift_tick2(). Метод ничего не возвращает.

String_To() – метод, который позволяет преобразовать в строку и количество необработанных запросов. Возвращает строку s.

Описание класса Пассажир:

– Внутренние поля класса:

string name – имя пассажира.

– Общедоступные поля класса:

int first_floor – Текущий этаж – местоположение вызова.

int last_floor – этаж назначения.

int cur_weight – вес пассажира.

int time_counter – счетчик времени.

– Общедоступные методы класса:

`String_To()` – метод, который преобразует в строку параметры запросы с возможностью дальнейшего вывода на экран. Возвращает строку `s`.

`Draw(Graphics g, Zdanie zdanie, int k)` – отрисовка пассажиров с их параметрами, основываясь на списке запросов. Метод ничего не возвращает.

`Draw_2(Graphics g, Zdanie zdanie, int k, int time)` – метод, позволяющий нарисовать динамическое отображение пассажира. Метод ничего не возвращает.

Описание класса Лифт:

– Внутренние поля класса:

`int number` – номер лифта.

`enum Direction` – направление движения лифта.

`int cur_floor` – текущий этаж лифта.

`byte cur_direction` – текущее направление движения.

`int cur_weight` – текущий вес груза.

`int max_weight` – максимальная грузоподъемность.

`int max_floor` – максимальное количество этажей.

`int loading_timer` – таймер во время посадки.

`int unloading_timer` – таймер во время высадки.

`List<Request> List_Request` - список запросов в лифте (лифт их забрал, но еще не отвез).

– Общедоступные методы класса:

`Move1()` – Метод, изменяющий текущий этаж лифта на 1 единицу.

Метод ничего не возвращает.

`Load_pass(Request req)` – Загрузить пассажиров, при этом увеличить вес груза, добавить в список запросов в лифте. Возвращает переменную логического типа. Если «true» пассажир заходит, в противном случае остается на этаже.

`Unload_pass(Request req)` – Выгрузить пассажира, снизить вес груза, добавить в список пассажиров, выходящих из лифта. Метод ничего не возвращает.

`String_To()` – метод, который преобразует в строку параметры лифта с возможностью дальнейшего вывода на экран. Возвращает строку `s`.

`Draw(Graphics g, Zdanie zdanie)` – рисовать лифт. Метод ничего не возвращает.

`Lift_tick1()` - Процедура такта, которая использует метод `Work()`. Метод ничего не возвращает.

`Lift_tick2()` – Процедура такта, которая использует метод `Move1()`, т.е. двигает лифт. Метод ничего не возвращает.

`Work()` – Метод, в котором заключается основная логика программы, по которой лифты принимают то или иное решение о движении.

Задание

По описанной задаче разработать требования к информационной части программно-информационной системы (отчёт в виде документа MS Word):

Предпринимателю необходимо в ближайшее время перенести часть своих торговых площадей в интернет-пространство, обеспечив функционирование всех торговых точек в рамках единой информационной системы.

Профиль организации:

1. Спортивный магазин.
2. Магазин фермерских продуктов.
3. Магазин детских товаров.
4. Магазин товаров для животных.
5. Магазин праздничной продукции.
6. Магазин канцелярских товаров.
7. Магазин туристических путевок.
8. Продуктовый магазин.
9. Магазин техники.
10. Магазин услуг.

Список использованных источников

1. Буч, Г., Рамбо, Д., Якобсон, И. Язык UML. Руководство пользователя. 2-е изд.: Пер. с англ. Мухин Н. / Г. Буч, Д. Рамбо, И. Якобсон. – М.: ДМК Пресс, 2006. – 496 с. – ISBN 5-94074-334-X. – Текст: непосредственный.

2. ГОСТ 19.201-78. Единая система программной документации. Техническое задание. Требования к содержанию и оформлению : межгосударственный стандарт : дата введения 1980-01-01.

3. Чаплыгин, А. А. Основы дипломного проектирования : учебное пособие для студентов направления подготовки 09.03.04 Программная инженерия, направленность (профиль) «Разработка программно-информационных систем» / А. А. Чаплыгин, В. В. Апальков, А. В. Малышев ; Юго-Зап. гос. ун-т. - Электрон. текстовые дан. - Курск : ЮЗГУ, 2022. - 102 с. - ISBN 978-5-907407-65-7 : -Текст : электронный.