МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ

ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО

ОБРАЗОВАНИЯ

РУТ (МИИТ)

Кафедра «Автоматизированные системы управления»

**КУРСОВАЯ РАБОТА**

**По дисциплине «Технологии программирования»**

**По теме: «Проектирование и разработка информационной**

**системы завода»**

Группа: УВВ-211

Студенты:

Гущин Е.А., Сибилев И. А.,

Залунин А.В., Лопатников В..

Преподаватель: Давыдовский М.А.

Москва

2021 г.

**Задания по курсовой работе для дисциплины «Технологии**

**программирования»**

Задание выдается на группу из 2-4 человек и выполняется на лабораторных занятиях и

занятиях по курсовой работе.

Язык программирования - Python.

Описание проекта разрабатываемой информационной системы должно быть выполнено на

языке UML.в среде UML Designer.

Документация по заданию должна содержать:

1. Задание по курсовой работе.

2. Диаграммы на языке UML.

3. Краткое описание каждого класса и метода.

4. Руководство пользователя со скриншотами интерфейса программы.

5. Набор тестов и результатов для отладки отдельных методов, модулей и

комплексной отладки задания.

6. Тексты программ с комментариями

Диаграммы языка UML:

• Диаграмма вариантов использования (Use Case Diagram)

• Диаграмма классов (Class Diagam)

• Диаграмма деятельности (Activity Diagram)

• Диаграмма состояний (State Machine Diagram)

• Диаграмма последовательности (Sequence Diagram)

• Диаграмма компонентов (Component Diagram)

• Диаграмма развертывания (размещения) (Deployment Diagram)

Необходимо разработать информационную систему в соответствии с заданием. В задании

перечислены пользователи системы (актеры) и варианты использования системы

актерами. Можно скорректировать список актеров и вариантов использования по

согласованию с преподавателем.

Предлагаемый перечень заданий может быть дополнен любым другим заданием по

согласованию с преподавателем. Приветствуется разработка проекта на основе реальных

заданий.

Общие требования по выполнению задания:

• Каждый пользователь информационной системы должен регистрироваться в

системе сам или это делает администратор

• При входе в систему каждый пользователь вводит свой логин и пароль

• Данные в системе хранятся в файлах

• При запуске системы все данные из файлов считываются в оперативную память и

хранятся в форме таблиц.

• По завершению работы с системой данные из таблиц записываются обратно в

файлы. Файлы при этом не обновляются, а создаются заново на основании данных

из таблиц

1. **Интернет-магазин. Доставка товаров**

Администратор

* Составление списка товаров

Кладовщик

* Формирование заказа (подбор товаров по заказу пользователя)

Диспетчер

* Формирование набора заказов для доставки курьером
* Слежение за доставкой заказов

Курьер

* Доставка заказов. После доставки заказа информация сразу поступает к диспетчеру

**Распределение задач:**

Гущин Е.А.:

• Построение UML-диаграмм;

• Создание графического интерфейса приложения;

• Написание классов-контроллеров к графическому интерфейсу;

Сибилев И. А.:

• Построение UML-диаграмм;

• Написание тестов к методам классов;

• Руководство пользователя;

Залунин А.В.:

• Построение UML-диаграмм;

• Написание классов-списков;

• Работа с базами данных;

Лопатников В..:

• Построение UML-диаграмм;

• Написание классов-сущностей;

• Тестирование работы программы;

|  |  |
| --- | --- |
| **Компонент “администратор ”** | |
| Действие исполнителя | Отклик системы |
| Поток “Регистрация нового работница” | |
| 1. Заполнение информации о  работнике, ввод логина и пароля, выбор должности | 1. Проверяет, нет данного логина в базе |
| 2. Подтверждение | 4.1 Записывает работника в базу данных |
|  | 4.2 Просит проверить правильность введённых данных |
| Поток “Удаление работника” | |
| 1. Выбор сотрудника для удаления | 1. Выводит список сотрудников |
| 3. Подтверждение | 4. Удаляет выбранного сотрудника из базы данных |
| Поток “Добавление новой категории товаров” | |
| 1. Заполнение информации о новом товаре | 2. Проверяет на наличие повторов |
|  | 3.1 Заносит новый товар в базы данных |
|  | 3.2 Просит проверить введённые данные |
| Под поток “Удаление категории товаров” | |
| 2. Выбор товара из списка | 1. Выводит список товаров |
|  | 3. Удаляет категорию |

|  |  |
| --- | --- |
| **Компонент “кладовщик”** | |
| Действие исполнителя | Отклик системы |
| Поток “Сборка заказа” | |
| 1. Выбор заказа для сборки | 2. Удаление нужного количества товаров из базы данных склада |
|  | 3. Добавление нового заказа в список |
| Поток “Удаление заказа” | |
| 2. Выбор заказа для отмены | 1. Показывает список заказов |
| 3. Подтверждение | 4. Добавление нужного количества товаров в базу данных склада |
|  | 5. Удаление заказа из списка |
| Под поток “Списание товаров” | |
| 1. Ввод id товара и его количества | 1. Удаляет нужное количество товаров из базы данных |
| Поток “Учёт поступлений товара” | |
| 1. Ввод id товаров и их количество | 1. Добавляет нужное количество товаров в базы данных |
| Поток “Обработка возврата” | |
|  | 1. Выводит сообщение о возврате |
| 2. Подтверждение возврата | 3. Добавляет нужное количество товаров в базы данных |

|  |  |
| --- | --- |
| **Компонент “Диспетчер”** | |
| Действие исполнителя | Отклик системы |
| Поток “Формирование наборов” | |
|  | 1. Загружает все заказы ожидающие доставки из базы данных |
| 1. Формирование набора | 3. Добавление нового набора в список ожидающих доставки |

|  |  |
| --- | --- |
| **Компонент “Курьер”** | |
| Поток “Доставка заказов” | |
| Действие исполнителя | Отклик системы |
| 1. Взятие набора заказов в доставку | 1. Выводит информацию о заказах и адресах доставки |
| 4.1 Подтверждение доставки заказа | 1. Изменяет статус заказа, сообщает диспетчеру |
| 4.2 Сообщить об ошибке |  |
| Под поток “Заказ доставлен” | |
| 4.1 Подтверждение доставки заказа | 5. Изменяет статус заказа, сообщает диспетчеру |
| Под поток “Возникла проблема” | |
| 4.2 Сообщить об ошибке | 5. Изменяет статус заказа, сообщает диспетчеру |
| 1. Возврат заказа |  |

**Диаграммы на языке UML**

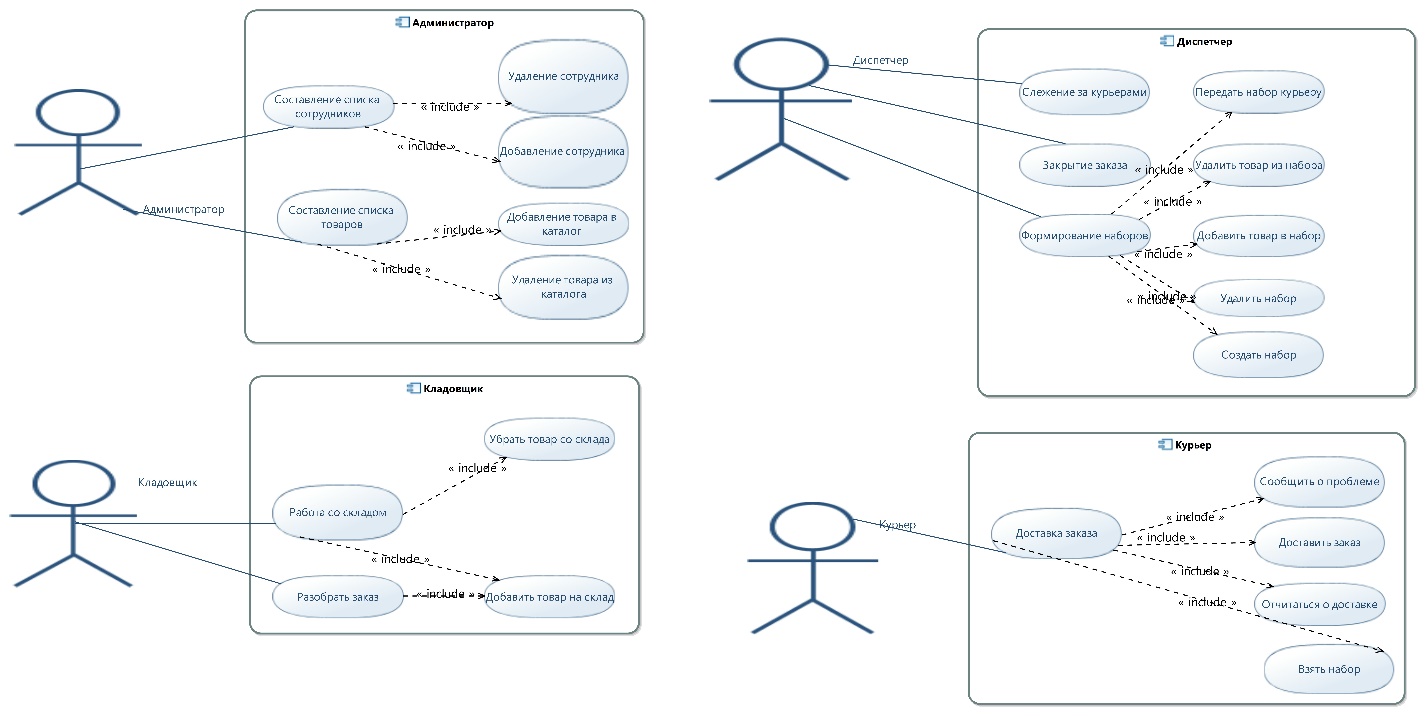
Диаграмма вариантов использования (Use Case Diagram):****

Диаграмма классов(Class Digram):

