



Санкт-Петербургское государственное бюджетное
профессиональное образовательное учреждение
"Колледж электроники и приборостроения"

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к итоговому проекту

по дисциплине «Технология разработки и защиты баз данных»

Тема: Умный склад на строй-площадке

Студент гр. 21 ИТ	_____	Ильин И.В
Студент гр. 21 ИТ	_____	Белов А.А.
Студент гр. 21 ИТ	_____	Будяков М.А.
Студентка гр. 21 ИТ	_____	Поповская А.О.
Студент гр. 21 ИТ	_____	Бойков Е.Р.
Преподаватель	_____	Сучков А.И.

Санкт-Петербург

2023

Цель прохождения практики: содержание заданий практики позволим нам сформировать первоначальный опыт практической деятельности и отработать необходимые умения, а также подготовиться к освоению ПК, будет способствовать формированию общих компетенций (ОК) в процессе производственной практики по виду профессиональной деятельности сопровождения и обслуживания программного обеспечения компьютерных систем.

Задачи практики:

- Создать удобное приложение.
- Выполнять оформление документационной части.
- Соблюдать дисциплину, правила техники безопасности, пожарной безопасности, учебной санитарии, выполнять требования внутреннего распорядка колледжа.
- Ежедневно вносить записи в свои дневники практики в соответствии с выполненной работой за учебный день.
- Выполнить итоговую работу.

Содержание практики, вопросы, подлежащие изучению и отрабатываемые элементы практической подготовки:

- Разработка и оформление технической документации.
- Разработка перечня обучающей документации.

Задание принято к исполнению:

Студент гр. 21 ИТ	_____	Ильин И.В
Студент гр. 21 ИТ	_____	Белов А.А.
Студент гр. 21 ИТ	_____	Будяков М.А.
Студентка гр. 21 ИТ	_____	Поповская А.О.
Студент гр. 21 ИТ	_____	Бойков Е.Р.
Преподаватель	_____	Сучков А.И.

Содержание:

1. Техническое задание.....	4
2. Конструкторский раздел.....	10
3. Технологический раздел.....	12
4. Экспериментальный раздел	23
Список литературы	24

1. Техническое задание

1. Введение

1.1. Наименование продукта

Наименование продукта: «Умный склад на строй площадке»

1.2. Назначение и область применения

Продукт предназначен для наблюдения за оборотом стройматериалом и карточками сотрудников. С помощью данного продукта легко осуществлять учет материалов.

Продукт представляет собой веб-приложение с различными страницами, содержащими различную информацию из базы данных.

2. Требование к программе

2.1. Требование к функциональным характеристикам

2.1.1. Просмотр

2.1.2. Переключения страниц

2.2. Требование к надежности

2.2.1. Требование к обеспечению надежного функционирования приложения

Надежное(устойчивое) функционирование приложения должно быть обеспечено выполнением заказчиком совокупности организационно-технических мероприятий, перечень которых приведен ниже:

А) Организация бесперебойного электропитания всех технических устройств

Б) Организацией постоянной работы сервера

В) Обеспечением защиты от различных атак сайта

2.2.2. Время восстановления после отказа

Время восстановления после отказа, вызванного сбоем электропитания технических устройств (иными внешними факторами), не должно превышать 15-ти минут при условий эксплуатации технических и программных средств.

Время восстановления после сбоя, вызванного выходом из строя технических средств, не должно превышать времени, требуемого на восстановление технических средств.

2.2.3. Отказы из-за некорректных действий пользователя

Отказы приложения из-за некорректных действий пользователя при взаимодействии с приложением недопустим.

3. Условия эксплуатации

3.1. Климатические условия эксплуатации

Не регламентируется.

3.2. Требования к составу и параметрам технических средств

Наличие браузера и интернета.

3.3. Требования к информационный и программной совместимости

4. Требования к программной документации

4.1. Предварительный состав программной документации

4.1.1. Техническое задание

4.1.2. Исходный код программы и методики испытаний

4.1.3. Руководство пользователя

4.1.4. Руководство оператора

5. Техничко-экономические показатели

5.1. Экономические преимущества разработки

Ориентировочная экономическая совместимость не рассчитывается.

6. Стадии и этапы разработки

6.1. Стадии разработки

Разработка должна быть проведена в три стадии:

1. Разработка технического задания.

2. Рабочее проектирование.

3. Внедрение.

6.2. Этапы разработки

На стадии разработки технического задания должен быть выполнен этап разработки, согласование и утверждение настоящего технического задания.

На стадии рабочего проектирования должны быть выполнены перечисленные ниже этапы работ:

1. Разработка программы
2. Разработка программной документации
3. Испытание программы

6.3. Содержание работ по этапам

1. Постановка задачи
2. Определение и уточнение требований к техническим средствам
3. Определение требований к программе
4. Определение стадий, этапов и сроков разработки программы и документации на нее
5. Согласование и утверждение технического задания

На этапе разработки программы должна быть разработана программа, выполнена работа по программированию (кодированию) и отладке программы.

На этапе разработки программной документации должна быть выполнена разработка программных документов в соответствии с требованиями к составу документации.

На этапе испытаний программы должны быть выполнены перечисленные ниже виды работ:

1. Разработка, согласование и утверждение методики испытаний
2. Проведение приемо-сдаточных испытаний
3. Корректировка программы и программной документации по результатам испытаний.

На этапе и передачи программы должна быть выполнена работа по подготовке и передаче программы и программной документации в эксплуатацию на компьютер заказчика.

7. Порядок контроля и приемки

7.1. Виды испытаний

Приемо-сдаточные работы должны проводиться на компьютере заказчика в определенные сроки

7.2. Общие требования к приемке работы

На основании протокола проведения испытаний Исполнитель совместно с Заказчиком подписывает Акт приемки-сдачи программы в эксплуатацию.

Сведения о стадиях разработки приведены в табл.1.

Таблица 1 – Стадии разработки

№ п/п	Этап	Срок, дней
1	Техническое задание	1
2	Эскизный проект	2
3	Технический проект	6
4	Рабочий проект	2
5	Внедрение	2

В соответствие с планом работ данных о затратах на заработную плату приведены в табл. 2.

Таблица 2 – Затраты на заработную плату

№ п/п	Этап	Кол-во разработчиков	Время разработки(часы)	Ср. стоимость, часа, руб.	Итого, руб.
1	Техническое задание	1	4	120	480
2	Эскизный проект	1	2	150	300
3	Технический проект	1	16	200	3200
4	Рабочий проект	1	10	160	1600
5	Внедрение	1	5	213	1065
Итого					6645

В соответствие с табл. 1, данных о затратах, связанных с эксплуатацией ЭВМ, приведены в табл. 3.

Таблица 3 – Затраты на эксплуатацию ЭВМ

№	Этап	Кол-во разработчиков	Затраты Комп. Времени	Ср. стоимость, комп. Часа, руб.	Итого, руб.
1	Техническое задание	1	4	80	320
2	Эскизный проект	1	2	80	160

3	Технический проект	1	16	80	1280
4	Рабочий проект	1	10	80	800
5	Внедрение	1	5	80	450
Итого					3010

Накладные расходы, связанные с затратами на содержание помещений и затратами на расходные материалы, принимаются в размере 10% от фонда заработной платы, что составит 664,5 руб.

Капитальные затраты на разработку программного продукта составят: $= 6645 + 3010 + 664,5 = 10319,5$ руб.

2. Конструкторский раздел

Функциональная схема работы программы (см. рис. 1)

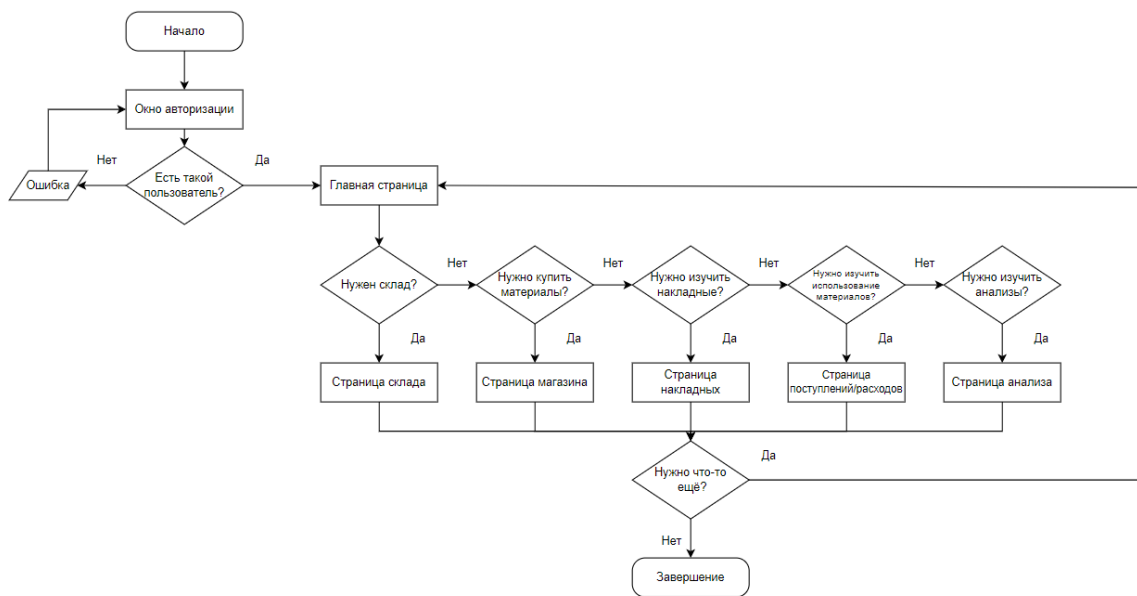


Рисунок 1 – Функциональная схема работы программы

Языком программирования был выбран Python, так как разработка приложения с базой данных MongoDB более удобная на языке Python с фреймворком Flask. С помощью данного фреймворка и языка Python можно написать нужное разработчикам веб-приложение с базой данных MongoDB.

Организация данных и проектирование интерфейсов обмена данными в программной системе.

В данном веб-приложении выводятся различные данные из БД, под определенный набор данных создана своя веб-страница. То есть выводимые данные (см. рис. 3) – это то, что видит пользователь на веб-страницах. Вводимые данные (см. рис. 2 – это то, что пользователь передает программе. В данном веб-приложении вводимые данные используются только на одной странице: “Вход”

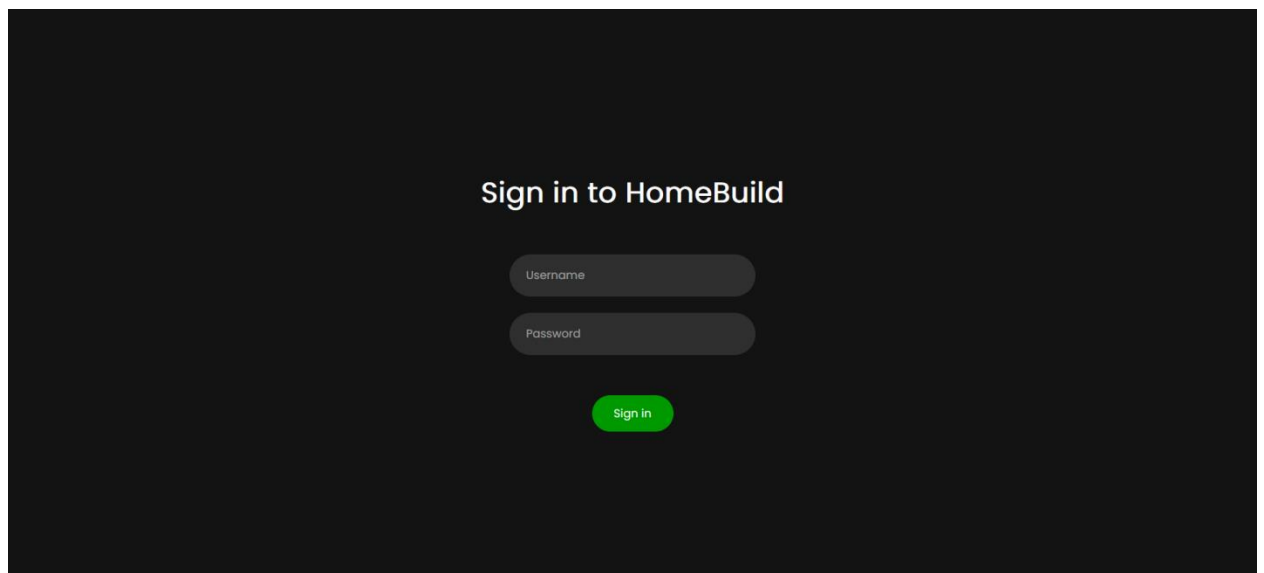


Рисунок 2 – Вводимые данные

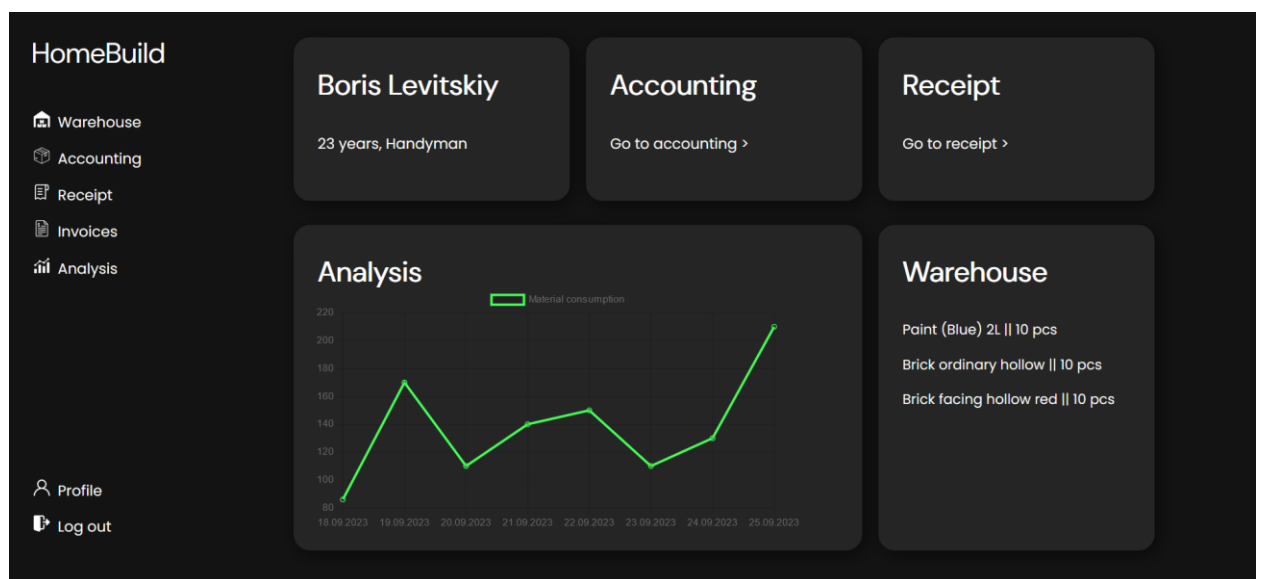


Рисунок 3 – Выводимые данные

3. Технологический раздел

3.1 Определение структуры и состава программной систем

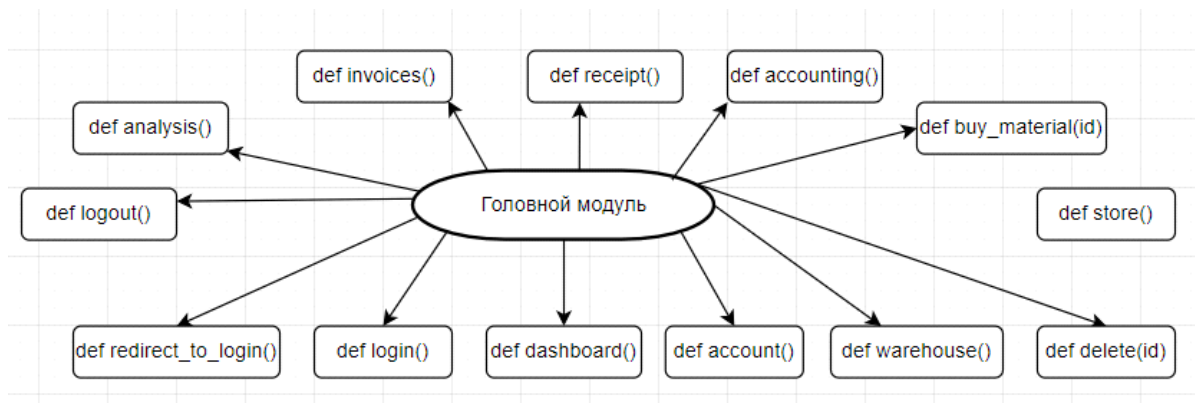


Рисунок 4 – Структура и состав программной системы

3.2 Описание разработанных алгоритмов программы

`from flask import Flask, redirect, render_template, request, session, url_for, jsonify` –

импортируем нужные методы из библиотеки flask

`from flask_bcrypt import Bcrypt` – импортируем метод Bcrypt из flask_bcrypt для шифрования паролей

`from pymongo import MongoClient` – импортируем метод MongoClient из библиотеки pymongo

`from datetime import datetime` – импортируем datetime из datetime

`from bson.objectid import ObjectId` – Импортируем метод ObjectId из bson.objectid

`def redirect_to_login()` – метод, который направляет юзера на форму авторизации, если тот не вошел в учетную запись, иначе направляет на панель управления (dashboard).

`def login()` – метод, который принимает логин и пароль, пароль шифрует в bcrypt, по логину ищет запись в бд и сравнивает зашифрованную полученную строку с зашифрованной строкой поля password. Если строки совпадают, то будет осуществлен вход в аккаунт, иначе выведет ошибку на странице.

`def dashboard()` – метод, который получает необходимые данные из бд и передает на страницу панели управления.

`def account()` – метод, который получает данные о данном аккаунте из бд и выводит их на страницу

`def warehouse()` – метод, который получает список материалов, данные по ним и выводит это на страницу

`def delete(id)` – метод, вызываемый по нажатию на кнопку возле товара на складе, чтобы взять товар. Этот метод удаляет конкретную запись из документа бд и добавляет в другой документ запись о том, что конкретный товар был взят.

`def store()` – метод, который получает список материала доступного в магазине для заказа и выводит его на страницу

`def buy_material(id)` – метод, вызываемый по нажатию на кнопку возле товара в магазине, для того чтобы купить материал. Данные о данном материале в бд берутся из документа `store` и переносятся в документ `warehouse`. Если такой товар уже есть, то просто прибавляется купленное количество. Также в другую таблицу заносятся сведения о совершенном действии.

`def accounting()` – метод, который занимается учетом материала. Он получает дату, по ней ищет данные в документе бд и выводит их на страницу.

`def receipt()` – метод, принимающий дату и выводящий, данные о полученном материале в эту дату

`def invoices()` – метод, принимающий дату и выводящий данные о выданном материале в эту дату.

`def analysis()` – метод, который получает данные для аналитики из бд и возвращает их на сайт

`def logout()` – метод, позволяющий выйти из своей учетной записи.

3.3 Диаграммы UML

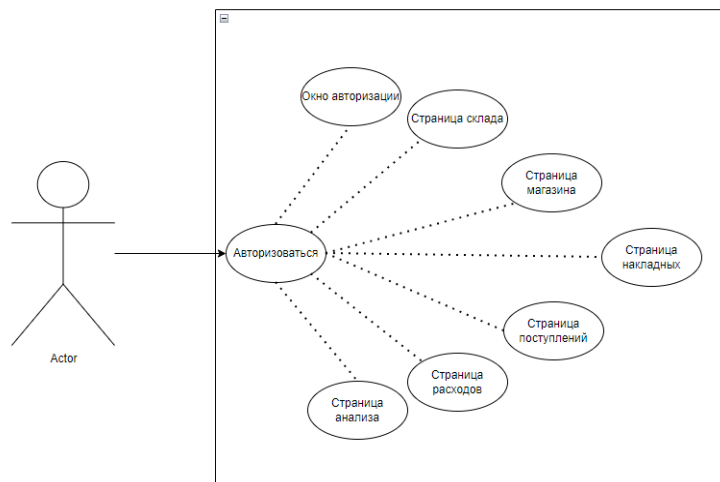


Рисунок 5 – Диаграмма вариантов использования

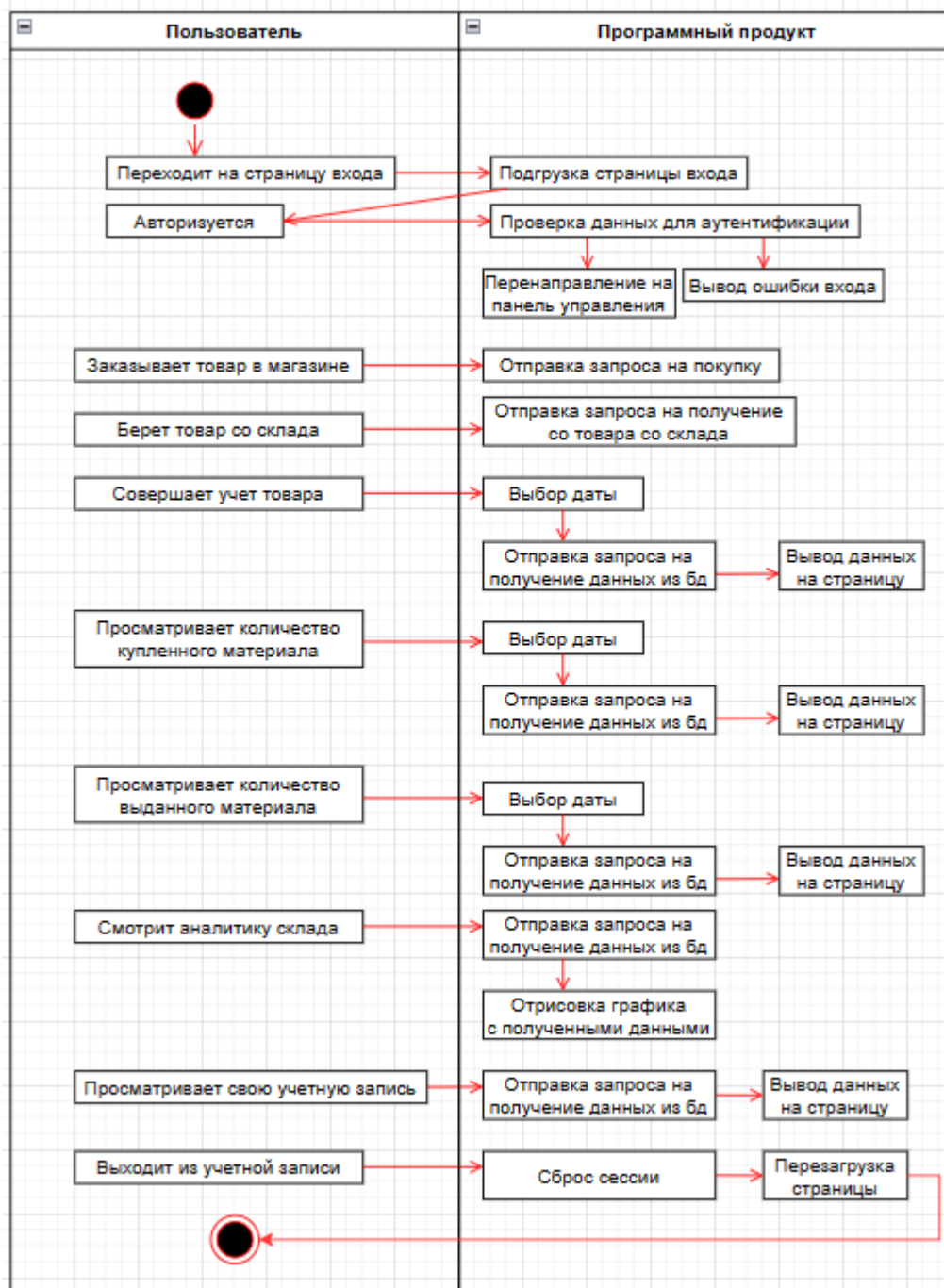


Рисунок 6 – Диаграмма активности

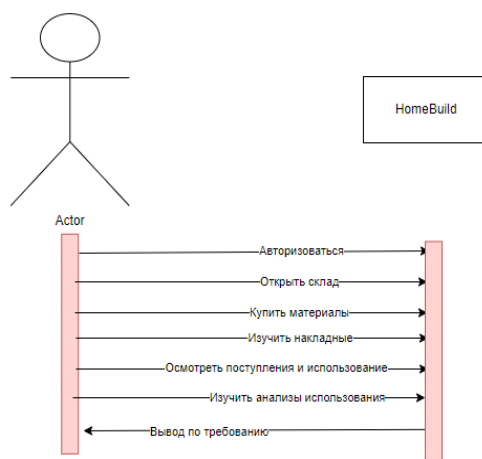


Рисунок 7 – Диаграмма последовательности для прецедента

3.4 Руководство пользователя

1. Введение

Приложение «HomeBuild» – веб-приложение, созданное для наблюдения за оборотом стройматериалов и карточками сотрудников. С помощью приложения легко осуществляется учёт материалов.

1.1 Область применения

Данный проект является компьютерным приложением.

Настоящий стандарт устанавливает основные характеристики электронных приложений, необходимый набор видовых параметров, определяет состав, структуру, функциональность, организационные основы электронного приложения, приводит необходимые термины и их определения. Стандарт предназначен для использования приложениями всех типов и видов, архивными службами, органами НТИ и другими информационными службами при разработке регламентирующих и технологических документов, организации практической деятельности в сфере формирования электронных приложений.

1.2 Краткое описание возможностей

Войти в учетную запись – страница авторизации, которая позволяет войти в учётную запись

Заказать материалы в магазине – страница магазина, позволяющая заказать материалы из каталога

Взять материал со склада – кнопка выдачи материала сотруднику

Учет материалов – страница для учёта материалов на складе

Просмотр купленного материала – страница просмотра поступлений материала

Просмотр выданного материала – страница просмотра выдачи материала

Аналитика склада – страница склада, позволяющая проанализировать использование материалов

Управление учетной записью – страница, позволяющая изучить данные сотрудника

Выход из учетной записи – кнопка выхода из учетной записи сотрудника

1.3 Уровень подготовки пользователя

Рекомендованный возраст пользователей 18+. Данная программа рассчитана на низкий уровень подготовки пользователя.

1.4 Перечень эксплуатационной документации, с которой необходимо ознакомиться пользователю

Материалы, размещенные в приложении «HomeBuild» для свободного использования, допускается копировать, а также цитировать исключительно в некоммерческих целях с соблюдением соответствующих положений действующего законодательства (см. ст. 1274 Гражданского кодекса РФ) с обязательным указанием имени автора произведения и источника заимствования.

На копиях документов должен быть сохранен знак авторского права ©, который запрещается удалять.

Состав программной документации должен включать в себя:

- техническое задание;
- исходный код программы и методики испытаний;
- руководство пользователя;
- руководство оператора;
- восстановление системы;
- защита;
- лицензирование;
- технико-экономические показатели;

- экономические преимущества разработки;

2. Назначение и условия применения

2.1 Виды деятельности, функции, для автоматизации которых предназначено данное средство автоматизации

Учёт поступлений материалов

Учёт использования материалов

Закупка материалов из магазина

2.2 Условия, при соблюдении (выполнении, наступлении) которых обеспечивается применение средства автоматизации в соответствии с назначением (например, вид ЭВМ и конфигурация технических средств, операционная среда и общесистемные программные средства, входная информация, носители данных, база данных, требования к подготовке специалистов и т. п.)

Минимальные требования:

- ОС: Windows 7/8/10
- Процессор: 11th Gen Intel (R) Core (TM) i3-1115G2 @ 3.00GHz
- Оперативная память: 1Gb
- Жесткий диск:128 Mb свободного пространства
- Клавиатура
- Мышка

3. Подготовка к работе

3.1 Состав и содержание дистрибутивного носителя данных

Не предусмотрен дистрибутивный носитель данных

3.2 Порядок загрузки данных и программ

Предварительно необходимо выполнить установку приложения.

Для того, чтобы запустить «HomeBuild», откройте папку, в которую была установлена программа, и запустите файл «app.py» (см. рис. 8)



Рисунок 8 – Файл программы

3.3 Порядок проверки работоспособности

Программное обеспечение работоспособно, если в результате действий пользователя, изложенных в п.п.3.2 на экране монитора отобразилось окно авторизации пользователя без выдачи пользователю сообщений о сбое в работе.

4. Описание операций

4.1 Описание всех выполняемых функций, задач, комплексов задач, процедур

Функции:

Авторизация – пользователь входит в учётную запись сотрудника.

Заказать материалы в магазине – сотрудник заказывает материалы из каталога

Взять материал со склада – сотрудник берет материалы со склада

Просмотр купленного материала – сотрудник смотрит поступления материала

Просмотр выданного материала – сотрудник смотрит выдачи материала

Аналитика склада – сотрудник смотрит анализ использование материалов

Управление учетной записью - сотрудник изучает данные учетной записи

Выход из учетной записи - пользователь выходит из учетной записи сотрудника

4.2 Описание операций технологического процесса обработки данных, необходимых для выполнения функций, комплексов задач(задач), процедур

Не предусмотрены

5. Аварийные ситуации

5.1 Действия в случае несоблюдения условий выполнения технологического процесса, в том числе при длительных отказах технических средств

При сбое работы аппаратуры восстановление нормальной работы системы должно производиться после:

- перезагрузки программы
- перезагрузки оперативной системы

5.2 Действия по восстановлению программ и/или данных при отказе магнитных носителей или обнаружении ошибок в данных

Не предусмотрено.

5.3 Действия в случаях обнаружении несанкционированного вмешательства в данные

Не предусмотрено.

5.4 Действия в других аварийных ситуациях

- перезагрузить приложение.
- перезагрузить операционную систему.

6. Рекомендации к освоению

1. Умение пользоваться компьютером
2. Умение читать
3. Загрузить программу
4. Запустить программу

3.5 Руководство оператора

1. Назначение программы

Приложение «HomeBuild» - веб-приложение, созданное для наблюдения за оборотом стройматериалов и карточками сотрудников. С помощью приложения легко осуществляется закуп и учёт материалов (см. рис. 9).

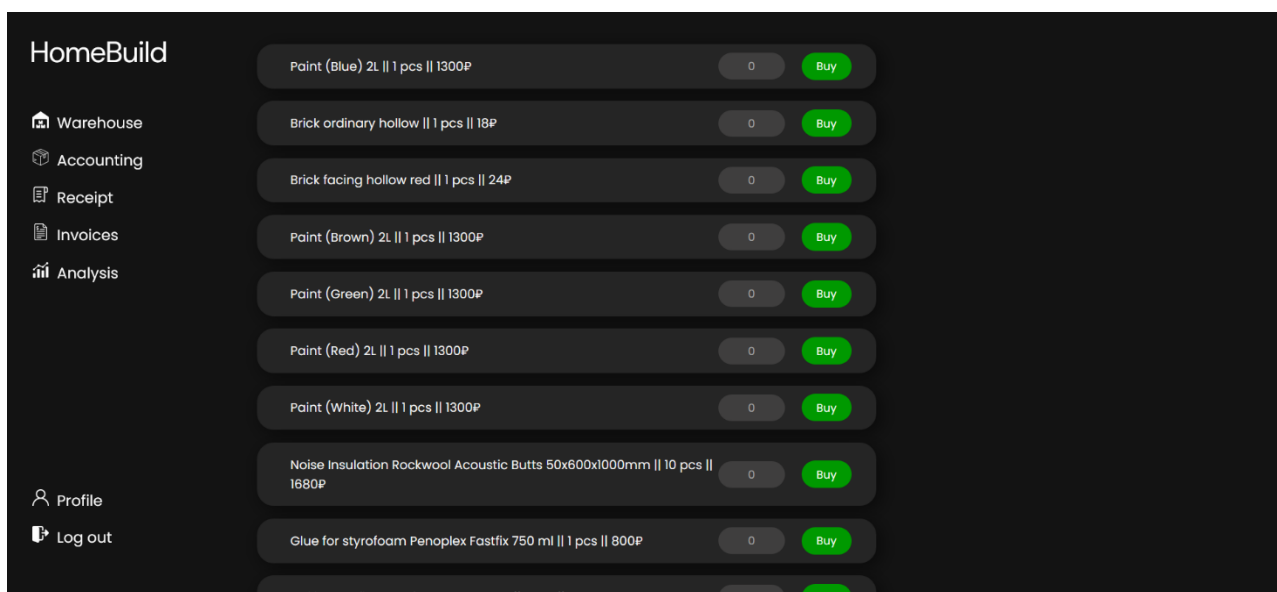


Рисунок 9 – Интерфейс программы

2. Условия выполнения программы

Условия выполнения программы, а также состав аппаратных и программных средств:

Минимальные требования:

- ОС: Windows 7/8/10
- Процессор: 11th Gen Intel (R) Core (TM) i3-1115G2 @ 3.00GHz
- Оперативная память: 1Gb
- Жесткий диск: 128 Mb свободного пространства
- Клавиатура
- Мышка

3. Выполнение программы

3.1 Загрузка и запуск программы

Предварительно необходимо выполнить установку файлов программы. Для того, чтобы запустить «HomeBuild», откройте папку, в которую ранее была установлена программа, и запустите файл «app.py» (см. рис. 10)



Рисунок 10 – Файл программы

После корректного входа можно будет запустить программу.

3.2 Выполнение программы

Оператор может взаимодействовать с интерфейсом программы с помощью мыши и клавиатуры.

При нажатии кнопки «Sign in» оператор может войти в учётную запись.

При нажатии на вкладку «Warehouse» оператор переходит на страницу склада.

При нажатии на вкладку «Accounting» оператор переходит на страницу учёта.

При нажатии на вкладку «Receipt» оператор переходит на страницу заказов.

При нажатии на вкладку «Invoices» оператор переходит на страницу накладных.

При нажатии на вкладку «Analysis» оператор переходит на страницу анализа склада.

При нажатии на вкладку «Account» оператор переходит на страницу данных пользователя.

При нажатии на вкладку «Log out» оператор выходит из учётной записи.

3.3 Завершение работы программы

Чтобы выйти из программы достаточно просто закрыть сайт.

4. Сообщение оператору

Отсутствует

4. Экспериментальный раздел

4.1 Виды контроля качества разрабатываемого ПО

Процесс разработки ПО предполагает три стадии тестирования:

- **автономное тестирование** – это тестирование компонентов ПО;
- **комплексное тестирование**;
- **системное (оценочное) тестирование** - тестирование на соответствие основным критериям качества.

4.2 Методика проведения и результаты тестирования

Для тестирования приложения была распространены среди нескольких человек, которые были должны протестировать весь функционал приложения. У каждого из тестеров была своя конфигурация компьютера и программное обеспечение.

4.3 Методы и способы устранения ошибок

Отладка – это комплексный процесс по выявлению и исправлению дефектов в программном обеспечении. Сами же дефекты, обычно, обнаруживаются в процессе тестирования ПО.

Отладка состоит из следующих этапов:

1. воспроизведение дефекта (любым из доступных способов);
2. анализ дефекта (поиск причины возникновения дефекта);
3. дизайн исправления дефекта;
4. кодирование исправления дефекта;

4.4 Отладка выявленных ошибок, обнаруженных при тестировании

Во время выполнения отладки, были использованы следующие методы:

- анализ кода без исполнения программы;
- запуск программы из-под отладчика.

Список литературы

Ст. 1274 ГК РФ

ГОСТ 19.505–79 ЕСПД. Руководство оператора

ГОСТ 19.201–78 Техническое задание

ГОСТ 51188–98 Защита информации