

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ТЕЛЕКОММУНИКАЦИЙ ИМ. ПРОФ. М.А. БОНЧ-БРУЕВИЧА»
(СПбГУТ)

АРХАНГЕЛЬСКИЙ КОЛЛЕДЖ ТЕЛЕКОММУНИКАЦИЙ
ИМ. Б.Л. РОЗИНГА (ФИЛИАЛ) СПбГУТ
(АКТ (ф) СПбГУТ)

КУРСОВОЙ ПРОЕКТ

НА ТЕМУ

РАЗРАБОТКА ПОДСИСТЕМЫ

«КОФЕЙНЯ: ЗАКАЗЫ»

Л109. 23КП01. 001 ПЗ

(Обозначение документа)

МДК.02.01 Технология разработки

программного обеспечения

Студент	ИСПП-11	06.12.2024	М.К. Абрамов
	(Группа)	(Подпись)	(И.О. Фамилия)
Преподаватель		06.12.2024	Ю.С. Маломан
		(Подпись)	(И.О. Фамилия)

Архангельск 2024

СОДЕРЖАНИЕ

Перечень сокращений и обозначений.....	3
Введение.....	4
1 Анализ и разработка требований.....	6
1.1 Назначение и область применения.....	?
1.2 Постановка задачи.....	?
1.3 Описание алгоритма функционирования системы.....	?
1.4 Выбор состава программных и технических средств.....	?
2 Проектирование программного обеспечения.....	?
2.1 Проектирование интерфейса пользователя.....	?
2.2 Разработка архитектуры программного обеспечения.....	?
2.3 Проектирование базы данных.....	?
3 Разработка и интеграция модулей программного обеспечения.....	?
3.1 Разработка программных модулей.....	?
3.2 Реализация интерфейса пользователя.....	?
3.3 Разграничение прав доступа пользователей.....	?
3.4 Экспорт и импорт данных.....	?
4 Тестирование и отладка программного обеспечения.....	?
4.1 Структурное тестирование.....	?
4.2 Функциональное тестирование.....	?
5 Инструкция по эксплуатации.....	?
5.1 Установка приложения.....	?
5.2 Инструкция по работе.....	?
Заключение.....	?
Список использованных источников.....	?
Приложение.....	?

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ

В настоящем курсовом проекте применяют следующие сокращения и обозначения:

БД – база данных

ИС – информационная система

ОС – операционная система

ПО – программное обеспечение

СУБД – система управления базами данных

API – интерфейс программирования приложения

ER-модель – модель «сущность-связь»

ERD – диаграмма «сущность-связь»

IDE – интегрированная среда разработки

SQL – язык структурированных запросов

UML – язык структурированных запросов

ВВЕДЕНИЕ

Текст введения на 1,5-2 страницы.

Обосновать актуальность создания, применения и защиты баз данных в современном мире.

Обосновать необходимость и практическую значимость выбранной темы, сформулировать цель (что нужно сделать) и задачи (как достигнуть поставленную цель) курсового проектирования.

Актуальность курсового проекта заключается в том, что применение баз данных ...

Целью курсового проектирования является разработка многопользовательской клиент-серверной информационной системы для учета ...

Для достижения поставленной цели требуется решить следующие задачи:

- выполнить сбор требований ...,
- проанализировать ...,
- спроектировать ...,
- разработать ...,
- выполнить тестирование и отладку ...,
-

1 Анализ и разработка требований

1.1 Назначение и область применения

ИС предназначена для автоматизации оформления и обработки заказов в сети кофеен “Commutity” владельцем которой является ИП Лютаревич Алексей Викторович. Основной вид деятельности: деятельность ресторанов и услуги по доставке продуктов питания. Дополнительные виды деятельности: производство хлеба и мучных кондитерских изделий, тортов и пирожных недлительного хранения, сухарей, печенья и прочих сухарных хлебобулочных изделий, какао, шоколада и сахаристых кондитерских изделий.

Внедрение ИС позволит оформлять и обрабатывать большее количество заказов. Категории пользователей системы: администратор, работник, клиент

1.2 Постановка задачи

В ИС должны храниться данные о клиентах, работниках, кофейнях, заказах, товарах. ИС должна выполнять задачи оформления и обработки заказов.

Будет разработано клиентское мобильное приложение для просмотра меню, оформления заказов, истории заказов.

Будет разработан телеграм бот для обработки заказов по установленному адресу.

БД предназначена для хранения данных ИС, автоматизированы следующие задачи: расчет стоимости заказа, просмотр всей информации о заказе, экспорт статистики по заказам.

1.3 Описание алгоритма функционирования системы

Функциональность для каждой категории пользователей:

Клиент (Client):

- Просмотр списка товаров
- Оформление заказов
- Просмотр истории заказов

Работник (Worker):

- Просмотр списка новых заказов
- Просмотр списка заказов «В работе»
- Просмотр списка выполненных заказов
- Изменение статуса заказов (новый > в работе > готов)
- Выбор адреса для работы
- Выбор состояний встать в работу / выйти с работы

Администратор (Admin):

- Добавить нового работника
- Удалить работника
- Изменить данные работника
- Просмотреть список работников

Для наглядности представлена диаграмма вариантов использования на рисунке 1.

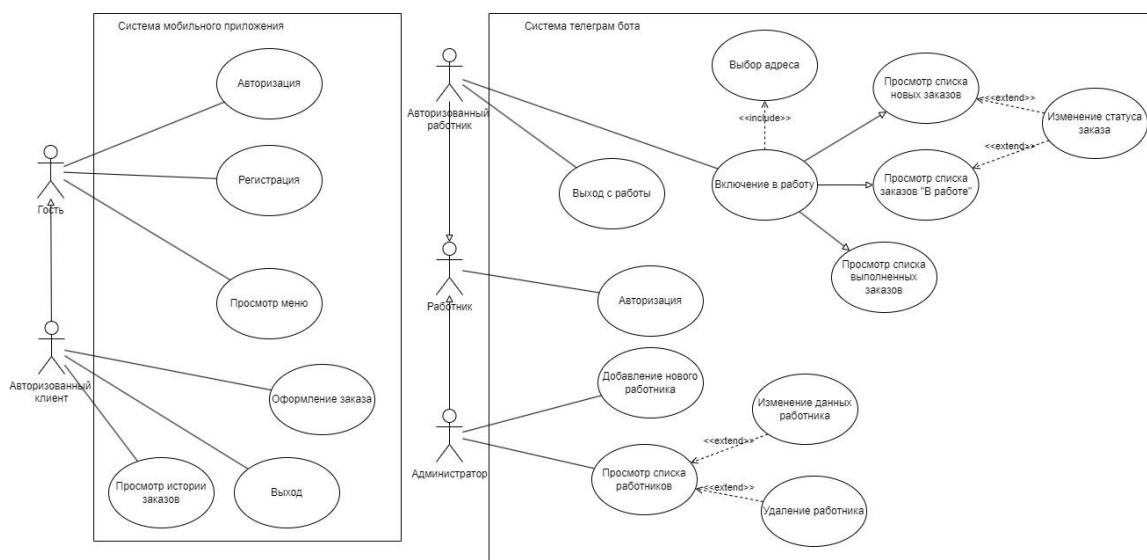


Рисунок 1 - draw.io. Диаграмма вариантов использования

1.4 Выбор состава программных и технических средств

Согласно цели проекта требуется создать многопользовательскую клиент-серверную информационную систему для оформления и обработки заказов.

Работа с информационной системой будет осуществляться на мобильных телефонах с установленной операционной системой Android 6.0/IOS 9.0/IPad 9.0 и выше с выходом в интернет.

В качестве системы управления базами данных выбрана СУБД MySQL 8, так как эта СУБД легка в использовании, предоставляет большой функционал, легко масштабируется и устанавливается на любой сервер

Мобильное приложение будет написано на языке программирования Dart и фреймворке Flutter, так как позволяет разрабатывать кроссплатформенные приложения.

Telegram бот будет разработан на языке программирования Python, так как этот язык прост и имеет множество библиотек для работы с telegram API.

Для разработки приложений будет использоваться интегрированная среда разработки программ VisualStudioCode и PyCharm, так как VS Code

позволяет быстро и удобно работать с различными библиотеками для Dart а PyCharm имеет удобный менеджер пакетов и средства отладки.

Для функционирования системы на стороне сервера достаточны следующие программные и технические средства:

- операционная система Ubuntu 24,
- сервер БД: MySQL версии не ниже 8,
- программное обеспечение для конфигурирования, управления и администрирования сервера БД: MySQL Workbench,

Для функционирования системы на стороне клиента достаточны следующие программные и технические средства:

- операционная система Android 6.0/IOS 9.0/IPad 9.0 и выше

Приложение «Telegram» или CoffeeApp,

2 Проектирование ПО

2.1 Проектирование интерфейса пользователя

изображения каркасов (wireframe) и/или мокапов представлены на рисунке(ах) **x**

Выбранные цвета: белый, черный, #EC5E32

Шрифт: **HereTheSuperInterestingFont**

Иконки представлены на рисунке **x**.



Рисунок **x** - draw.io. Иконки мобильного приложения

2.1 Разработка архитектуры программного обеспечения

2.3 Проектирование базы данных

На рисунке **3** показана физическая модель предметной области, разработанная на основе логической модели для СУБД **MySQL 8.0**.

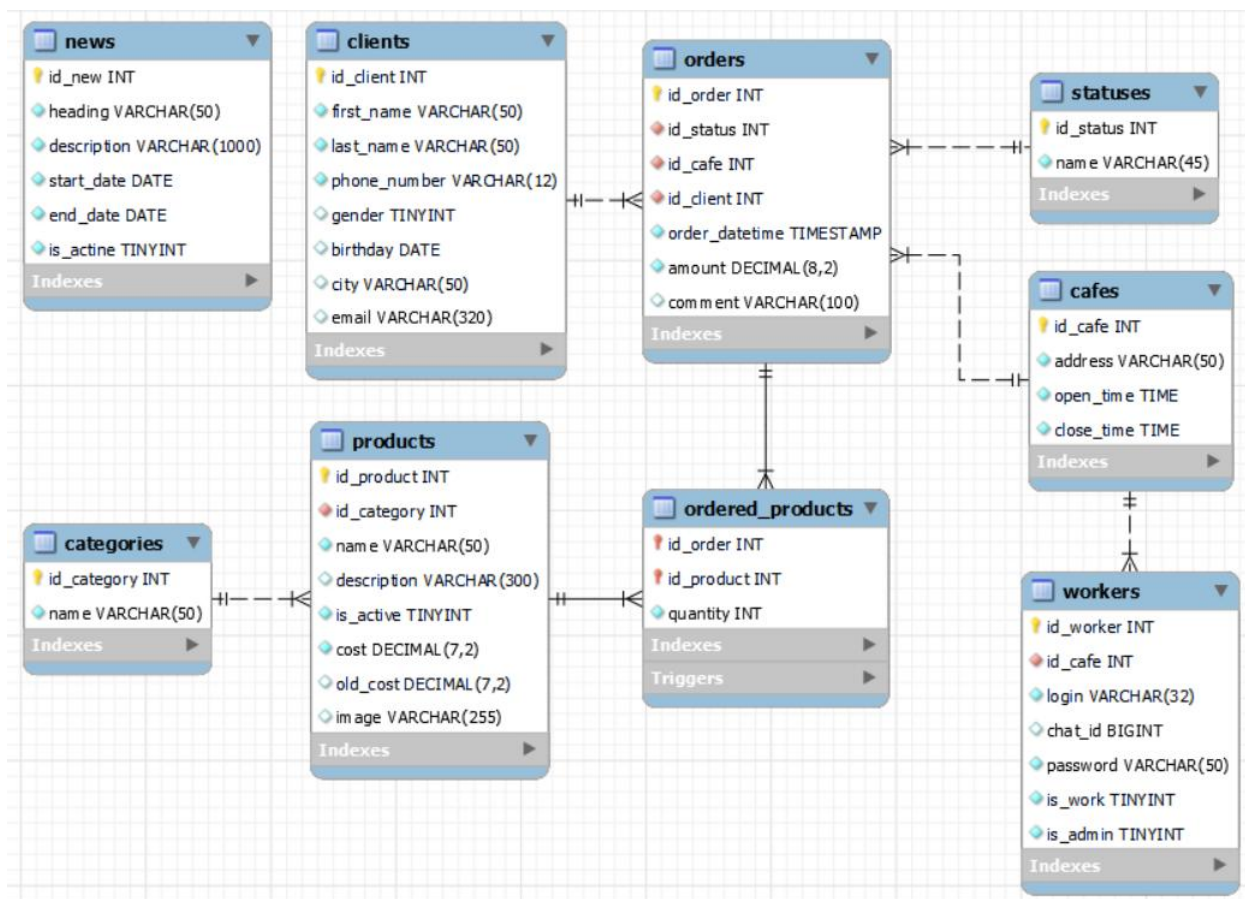


Рисунок 3 – MySQL Workbench. Физическая модель

В таблице 1 в виде словаря данных представлено описание созданных таблиц и ограничений целостности БД.

Таблица 1 – Словарь данных

Ключ	Поле	Тип данных	Обязательное	Примечание
clients				
PK	id_client	INT	+	AI
	first_name	NVARCHAR(50)	+	
	last_name	NVARCHAR(50)	+	
	phone_number	VARCHAR(12)	+	UQ
	gender	BOOL	-	
	birthday	DATE	-	
	city	NVARCHAR(50)	-	

	email	VARCHAR(320)	-	
orders				
PK	id_order	INT	+	AI
FK1	id_status	INT	+	
FK2	id_cafe	INT	+	
FK3	id_client	INT	+	
	order_datetime	TIMESTAMP	+	Значение по умолчанию: CURRENT_ TIMESTTTAMP
	amount	DECIMAL(8,2)	+	По умолчанию: 1.00
	comment	NVARCHAR(100)	-	
statuses				
PK	id_status	INT	+	AI
	name	NVARCHAR(50)	+	
cafes				
PK	id_cafe	INT	+	AI
	address	NVARCHAR(50)	+	
	open_time	TIME	+	По умолчанию: 07:00
	close_time	TIME	+	По умолчанию: 20:00
workers				
PK	id_worker	INT	+	AI
FK	id_cafe	INT	+	
	login	VARCHAR(32)	+	
	chat_id	BIGINT	-	
	password	VARCHAR(50)	+	
	is_work	BOOL	+	По умолчанию: False
	is_admin	BOOL	+	По умолчанию: False
ordered_products				
PK	id_order	INT	+	
FK1				
PK	id_product	INT	+	
FK1				
	quantity	INT	+	По умолчанию: 1

products				
PK	id_product	INT	+	AI
FK	id_category	INT	+	
	name	NVARCHAR(50)	+	
	description	NVARCHAR(300)	-	
	is_active	BOOL	+	По умолчанию: True
	cost	DECIMAL(7,2)	+	По умолчанию: 0.00
	old_cost	DECIMAL(7,2)	-	
	image	VARCHAR(255)	-	
categories				
PK	id_category	INT	+	AI
	name	NVARCHAR(50)	+	

3 Разработка и интеграция модулей

3.1 Разработка программных модулей

В ходе курсового проектирования были разработаны: мобильное приложение на Dart, телеграм бот на Python, Web-API приложение на C#.

Для постоянного получения новых заказов в телеграм боте была разработана функция представленная листингом **x**.

Получение информации из БД осуществляется посредством Web-API приложения, код получения заказа по его ID представлен листингом **x+1**.

Листинг **x** - Код функции непрерывного получения новых заказов

```
async def monitor_new_orders(msg: Message):
    """    Мониторит новые заказы и отправляет сообщения об этих
    заказах пользователю.    :param msg: Сообщение от пользователя.
    """
    logging.info("Посещение функции monitor_new_orders")
    worker = await APIMethods.get_worker_state(msg.chat.id)
    while worker:
        worker_address_id = await
APIMethods.get_worker_address(msg.chat.username.replace("_",
"\_"))
        worker_address_str = await
APIMethods.get_address_dict()
        worker_address_str =
worker_address_str.get(worker_address_id)
        new_orders_items = await
APIMethods.monitor_new_orders(worker_address_str) # Метод для
получения новых заказов
        for order_id, order_msg in new_orders_items.items():
            await asyncio.sleep(0.5) # Задержка между отправками
сообщений
            order_keyboard = await orders_inline_keyboard(order_id)
            # Кнопки для обновления статуса заказа
            await msg.answer(order_msg, reply_markup=order_keyboard)
            await asyncio.sleep(25) # Задержка перед новым запросом
```

Листинг **x+1** - Код получения заказа по его ID

```
// GET: api/order/5
[HttpGet("info/id={id}")]
```

```

public async Task<ActionResult<orders_full_info_view>>
GetOrderInfo(int id)
{
    var order = await
_context.orders_full_info_views.FindAsync(id);
    if (order == null){return NotFound();}
    return order;
}

```

3.2 Реализация интерфейса пользователя

При реализации интерфейса пользователя был использован язык Dart и его Flutter фреймворк, так же библиотека Dio для работы с http запросами.

3.3 Разграничение прав доступа пользователей

Разграничение прав доступа реализовано разделением работников и клиентов по разным таблицам. Права администратора у работника появляются при True значении в поле is_admin

Пример кода разграничения прав пользователей телеграм бота представлен листингом **х**.

Листинг **х** - Код разграничения прав пользователей телеграм бота

```

async def keyboard_menu(msg: Message) -> ReplyKeyboardMarkup:
    """
    Создает клавиатуру для главного меню.
    :return: Объект ReplyKeyboardMarkup с кнопками меню.
    """
    # Определяем кнопки для клавиатуры
    if await
    APIMethods.get_admin_state(msg.chat.username.replace("_",
    "\_")):
        keyboard_list = [[KeyboardButton(text=variables.new_worker)],
        [KeyboardButton(text=variables.read_workers)], [KeyboardButton
        (text=variables.del_worker)]]
    else:
        keyboard_list = [[KeyboardButton(text=variables.address),
        KeyboardButton(text=variables.work)], [KeyboardButton(text=var
        iables.home), KeyboardButton(text=variables.complete)]]
    # Создаем и возвращаем объект клавиатуры

```

```
keyboard = ReplyKeyboardMarkup(keyboard=keyboard_list,  
    resize_keyboard=True, input_field_placeholder="Выберите  
    действие:")  
return keyboard
```

3.4 Экспорт и импорт данных

4 Тестирование и отладка ПО

4.1 Структурное тестирование

4.2 Функциональное тестирование

5 Инструкция по эксплуатации ПО

5.1 Установка программного обеспечения

Минимальные системные требования для серверной части приложения:

ЦП: 2 ядра

ОЗУ: 8 ГБ

Наличие выхода в интернет

Необходимое ПО для серверной части приложения:

MySQL Server 8.0, Docker, Docker-compose, ASP.NET “Coffee-API”,
Python Aioogram bot “main.py”

Процесс создания БД:

- Авторизация на сервере, в СУБД
- Выполнить импорт / запуск скрипта базы данных

CoffeShopDB.sql

Минимальные системные требования для клиентской части приложения:

Администратор:

Android 6.0/iOS 9.0/iPad 9.0 и выше

Приложение «Telegram»

Наличие выхода в интернет

Клиент:

Android 6.0/iOS 9.0/iPad 9.0 и выше

Наличие выхода в интернет

Бариста:

Android 6.0/IOS 9.0/IPad 9.0 и выше

Приложение «Telegram»

Наличие выхода в интернет

Имена входа по умолчанию:

Клиент: логин: Client@mail.ru, пароль: ClientPass

Работник: логин: задан по умолчанию как логин телеграм, пароль: passwd

5.2 Инструкция по работе

Для запуска мобильного приложения “Name” требуется нажать на иконку этого приложения

Процесс авторизации работников в телеграм боте:

- Открыть чат с ботом
- Отправить боту команду “/start”
- Бот установит user_name пользователя как логин, попросит ввести пароль
- Отправить пароль, полученный ранее от администратора
- Получить сообщение об успешной авторизации
- В зависимости от заранее предустановленной настройки ролей происходит вход в систему от имени работника или от имени администратора

Процесс регистрации клиентов в мобильном приложении:

- открытие приложения,
- ввод адреса электронной почты,
- Получение кода авторизации на почту
- Ввод кода авторизации, отправка кода в приложении
- Ввод персональных данных пользователя
- Нажать кнопку подтверждение данных

Возможности пользователей:

ЗАКЛЮЧЕНИЕ

Текст заключения на всю страницу.

Сделать вывод о процессе курсового проектирования на основе данных введения и остальных разделов курсового проекта.

Обязательно написать, что поставленная цель достигнута и в процессе ее достижения решены следующие задачи (перечислить то, что выполнено).

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Голицына, О. Л. Базы данных : учебное пособие / О. Л. Голицына, Н. В. Максимов, И. И. Попов. – 4-е изд., перераб. и доп. – Москва : ФОРУМ : ИНФРА-М, 2020. – 400 с. – URL: <https://znanium.com/catalog/document?id=362825>. – Режим доступа: для зарегистрир. пользователей. – Текст : электронный.
2. Голицына, О. Л. Основы проектирования баз данных : учебное пособие / О. Л. Голицына, Т. Л. Партыка, И. И. Попов. – 2-е изд., перераб. и доп. Москва : ФОРУМ : ИНФРА-М, 2021. – 416 с. – URL: <https://znanium.com/catalog/document?id=364900>. – Режим доступа: для зарегистрир. пользователей. – Текст : электронный.
3. Дадян, Э. Г. Данные: хранение и обработка : учебник / Э. Г. Дадян. – Москва : ИНФРА-М, 2020. – 205 с. – URL: <https://znanium.com/catalog/document?id=346013>. – Режим доступа: для зарегистрир. пользователей. – Текст : электронный.
4. Мартишин, С. А. Базы данных. Практическое применение СУБД SQL и NoSQL-типа для применения проектирования информационных систем : Учебное пособие / С. А. Мартишин, В. Л. Симонов, М. В. Храпченко. – Москва : ИД «ФОРУМ» : ИНФРА-М, 2018. – 368 с. – URL: <http://znanium.com/bookread2.php?book=926871>. – Режим доступа: для зарегистрир. пользователей. – Текст : электронный.
5. Павловская, Т. А. С#. Программирование на языке высокого уровня: Учебник для вузов / Т. А. Павловская. – Санкт-Петербург : Питер, 2021. – 432 с. – URL: <https://ibooks.ru/bookshelf/377952/reading>. – Режим доступа: для зарегистрир. пользователей. – Текст : электронный.
6. Туманов, В. Е. Основы проектирования реляционных баз данных / В. Е. Туманов. – Москва : Национальный Открытый Университет «ИНТУИТ», 2020. – 502 с. – URL: <https://www.iprbookshop.ru/97570.html>. – Режим доступа: для зарегистрир. пользователей. – Текст : электронный.

7. Фленов, М. Е. Библия С# / М. Е. Фленов. – 4-е изд., перераб. и доп. – Санкт-Петербург : БХВ-Петербург, 2019. – 512 с. – URL: <https://ibooks.ru/bookshelf/366634/reading>. – Режим доступа: для зарегистрир. пользователей. – Текст : электронный.