

МИНИСТЕРСТВО ОБРАЗОВАНИЯ НИЖЕГОРОДСКОЙ ОБЛАСТИ
Государственное бюджетное профессиональное образовательное учреждение
НИЖЕГОРОДСКИЙ РАДИОТЕХНИЧЕСКИЙ КОЛЛЕДЖ

КУРСОВАЯ РАБОТА

УП.04

**Тема: «Мой курсовой проект -- настольное приложение для работы
с базой данных»**

Выполнил студент
группы 4ИСиП-19-1

Проверил преподаватель
Слугин В. Г.

Танцев В. В.
ФИО

2022 г.

Оглавление

Предметная область автоматизации.....	2
Язык программирование и платформа языка.....	2
Библиотека для разработки графического интерфейса.....	3
СУБД с которой работает приложение	3
Схема базы данных (информационно-логическая модель);	4
Тестирование информационной системы.....	8
Скриншоты работы приложения	9
Ссылка на репозиторий исходного кода приложения	24

Предметная область автоматизации

Вымышленный настольный клуб «6 Граней» является небольшой организацией, которая начала работать год назад. Организацией управляет администратор Фёдоров Иван Николаевич, в его подчинение входят 2 менеджера и два консультанта

«6 Граней» также располагает огромным выбором аренды на настольные игры. Цены на аренду выгодные.

Менеджер заходя в свой профиль способен сформировать новый заказ на аренду, добавить новую настольную игру в список или иной товар, и закрыть заказ на аренду, приняв настольные игры обратно.

Администратор после входа в профиль просмотреть информацию об истории входа сотрудников.

Консультант после входа в свой профиль может сформировать новый заказ на аренду, выбрав нужные товары, и закрыть заказ на аренду, приняв настольные игры обратно.

В данной курсовой работе будет разработан модуль автоматизированной информационной системы для настольного клуба «6 Граней». Она позволит сотрудникам клуба настольных игр производить различные операции с заказами на аренду, пользователями и другим функционалом программы.

Язык программирование и платформа языка.

Python – это универсальный современный ЯП высокого уровня, к преимуществам которого относят высокую производительность программных решений и структурированный, хорошо читаемый код. Синтаксис Питона максимально облегчен, что позволяет выучить его за сравнительно короткое время. Ядро имеет очень удобную структуру, а широкий перечень встроенных библиотек позволяет применять внушительный набор полезных функций и возможностей. ЯП может использоваться для написания прикладных приложений, а также разработки WEB-сервисов.

На момент написания курсовой, использовалась версия **Python 3.10.7**

Библиотека для разработки графического интерфейса

PyQt5 - это набор Python библиотек для создания графического интерфейса на базе платформы Qt5 от компании Digia. Он доступен для Python 2. x и 3. x.

На момент написания курсовой, использовалась версия PyQt5 5.15.7

СУБД с которой работает приложение

Приложение работает в связке с PostgreSQL.

PostgreSQL создана на основе некоммерческой СУБД Postgres, разработанной как open-source проект в Калифорнийском университете в Беркли. К разработке Postgres, начавшейся в 1986 году, имел непосредственное отношение Майкл Стоунбрейкер, руководитель более раннего проекта Ingres, на тот момент уже приобретённого компанией Computer Associates. Название расшифровывалось как «Post Ingres», и при создании Postgres были применены многие ранние наработки.

Стоунбрейкер и его студенты разрабатывали новую СУБД в течение восьми лет с 1986 по 1994 годы. За этот период в синтаксис были введены процедуры, правила, пользовательские типы и другие компоненты. В 1995 году разработка снова разделилась: Стоунбрейкер использовал полученный опыт в создании коммерческой СУБД Illustra, продвигаемой его собственной одноимённой компанией (приобретённой впоследствии компанией Informix), а его студенты разработали новую версию Postgres — Postgres95, в которой язык запросов POSTQUEL — наследие Ingres — был заменен на SQL.

Разработка Postgres95 была выведена за пределы университета и передана команде энтузиастов. Новая СУБД получила имя, под которым она известна и развивается в текущий момент — PostgreSQL.

Схема базы данных (информационно-логическая модель);

Информационно-логическая модель отображает данные предметной области в виде совокупности информационных объектов и связей между ними. Эта модель представляет данные, подлежащие хранению в базе данных.

Описание структуры БД

В нижеследующей модели описаны поля всех таблиц БД, используемых разработанной информационной системой.

Таблица 1. Описание атрибутов таблицы Клиенты

№ п/п	Название	Идентификатор	Тип	Знач. по умолчанию	Обяз-ное поле?	Признак ключа
1.	Ид. № Клиента	id	N	—	+	PK
2.	ФИО	fullname	C	—	+	-
3.	Паспорт	passport	C	—	-	—
4.	День рождения	birthday_date	D	—	—	—
5.	Адрес	address	C	-	-	—
6.	Эл. почта	email	C	-	-	-
7.	Пароль	password	C	-	-	-

Таблица 2. Описание атрибутов таблицы Сотрудники

№ п/п	Название	Идентификатор	Тип	Знач. по умолчанию	Обяз-ное поле?	Признак ключа
1.	Ид. № Сотрудника	id	N	—	+	PK

2.	ФИО	fullname	C	–	+	-
3.	Позиция	position	C	–	-	–
4.	Последний вход	Last_entry	D	–	–	–
5.	Статус	status	C	-	-	–
6.	Логин	login	C	-	-	-
7.	Пароль	password	C	-	-	-

Таблица 3. Описание атрибутов таблицы Товары

№ п/п	Название	Идентификатор	Тип	Знач. по умолчанию	Обяз-ное поле?	Признак ключа
1.	Ид. № Товара	id	N	–	+	PK
2.	Наименование	title	C	–	+	-
3.	Изображение	img	C	–	-	–
4.	Стоимость за час	cost_per_hour	N	–	–	–
5.	Полная цена	full_price	N	-	-	–
6.	Описание	description	C	-	-	-
7.	Оставшееся кол-во	remaining_amount	N	-	-	-
8.	Категория	category	C	-	-	-

Таблица 4. Описание атрибутов таблицы Заказы

№ п/п	Название	Идентификатор	Тип	Знач. по умолчанию	Обяз-ное поле?	Признак ключа
1.	Ид. № Заказа	id	N	–	+	PK
2.	Дата создания	create_date	D	–	+	-
3.	Время	create_time	T	–	-	–

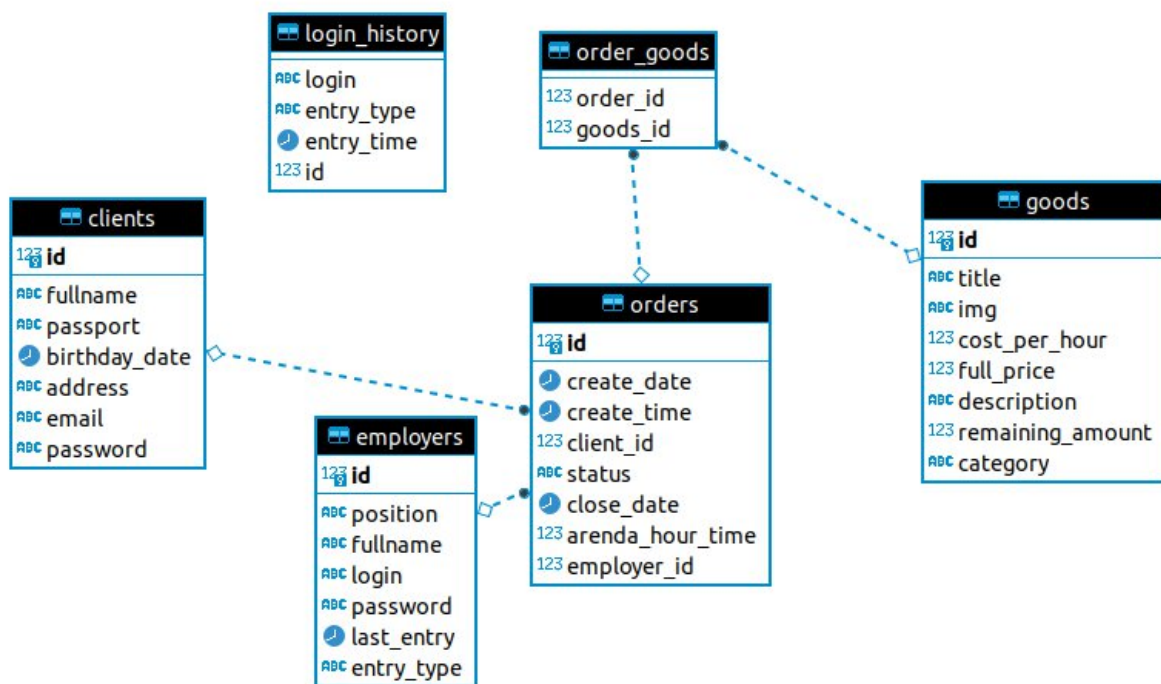
	создания					
4.	Ид. № Клиента	client_id	N	—	—	FK(Клиенты)
5.	Дата закрытия	close_date	D	-	-	—
6.	Время аренды	arenda_hour_time	N	-	-	-
7.	Ид. № Сотрудника	employer_id	N	-	-	FK(Сотрудники)
8.	Статус	category	C	-	-	-

Таблица 5. Описание атрибутов таблицы Заказ товары

№ п/п	Название	Идентификатор	Тип	Знач. по умолчанию	Обяз-ное поле?	Признак ключа
1.	Ид. № Заказа	order_id	N	—	+	PK, FK(Заказы)
2.	Ид. № Товара	goods_id	N	—	+	PK, FK(Товары)

Таблица 6. Описание атрибутов таблицы История входа

№ п/п	Название	Идентификатор	Тип	Знач. по умолчанию	Обяз-ное поле?	Признак ключа
1.	Ид. № Попытки входа	id	N	—	+	PK
2.	Логин	login	C	—	+	—
3.	Тип входа	entry_type	C	-	-	-
4.	Время входа	entry_time	D	-	-	-



Информационно-логистическая модель базы данных.

Тестирование информационной системы

Для проверки будем использовать автотесты, которые написаны также, как и сам проект на Python с помощью фреймворка unittest.

Они должны запускаться после инициализации схемы БД.

Листинг класса Tests

```
class Tests(unittest.TestCase):

    def setUp(self) -> None:
        """Установка соединения с БД"""
        self.connection = psycopg2.connect(
            dbname='coursework_prdb',
            user='admin',
            password='qwerty',
            host='localhost'
        )
        self.cursor = self.connection.cursor(cursor_factory=RealDictCursor)

    def tearDown(self) -> None:
        """Отключение соединения с БД"""
        self.connection.close()

    def test_authorize_employer(self):
        """Проверка на возможность авторизации сотруднику"""
        self.cursor.execute("SELECT * FROM employers LIMIT 1")
        employer = self.cursor.fetchone()
        if not employer:
            self.fail('no employee found')

        login = employer['login']
        password = employer['password']

        authorize_model = AuthorizeModel()
        result = authorize_model.verify_credentials(login, password)
        self.assertTrue(result)

    def test_non_authorize_employer(self):
        """Проверка авторизации сотрудника при неверных данных"""
        authorize_model = AuthorizeModel()
        result = authorize_model.verify_credentials(
            'nonexistent_email@ne.ne1',
            'nonexistent_password'
        )
        self.assertFalse(result)
```



```

def test_filter_goods_category(self):
    """Проверка фильтрации товаров по категории"""
    category = "Стратегические"
    model = GoodsModel()
    model.goods_category = category
    goods = model.get_goods()
    if not len(goods):
        self.fail('no goods found')
    is_all_fits = all([product['category'] == category for product in goods])
    self.assertTrue(is_all_fits)

def test_sort_goods_title_desc(self):
    """Проверка сортировки товаров по имени (по убыванию)"""
    model = GoodsModel()
    model.sort_field = "title"
    model.sort_type = "DESC"
    limit = model.limit

    self.cursor.execute(f"SELECT * from goods ORDER BY title DESC "
                        f"LIMIT {limit}")
    goods_expected = self.cursor.fetchall()
    if not len(goods_expected):
        self.fail('no goods found')

    goods_actual = model.get_goods()

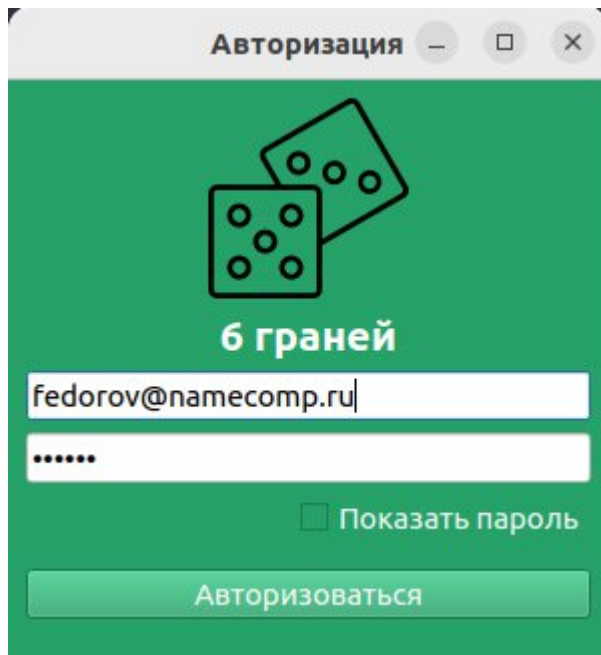
    self.assertEqual(
        [product['id'] for product in goods_expected],
        [product['id'] for product in goods_actual]
    )

```

Скриншоты работы приложения

Разработка раздела «Авторизация пользователя»

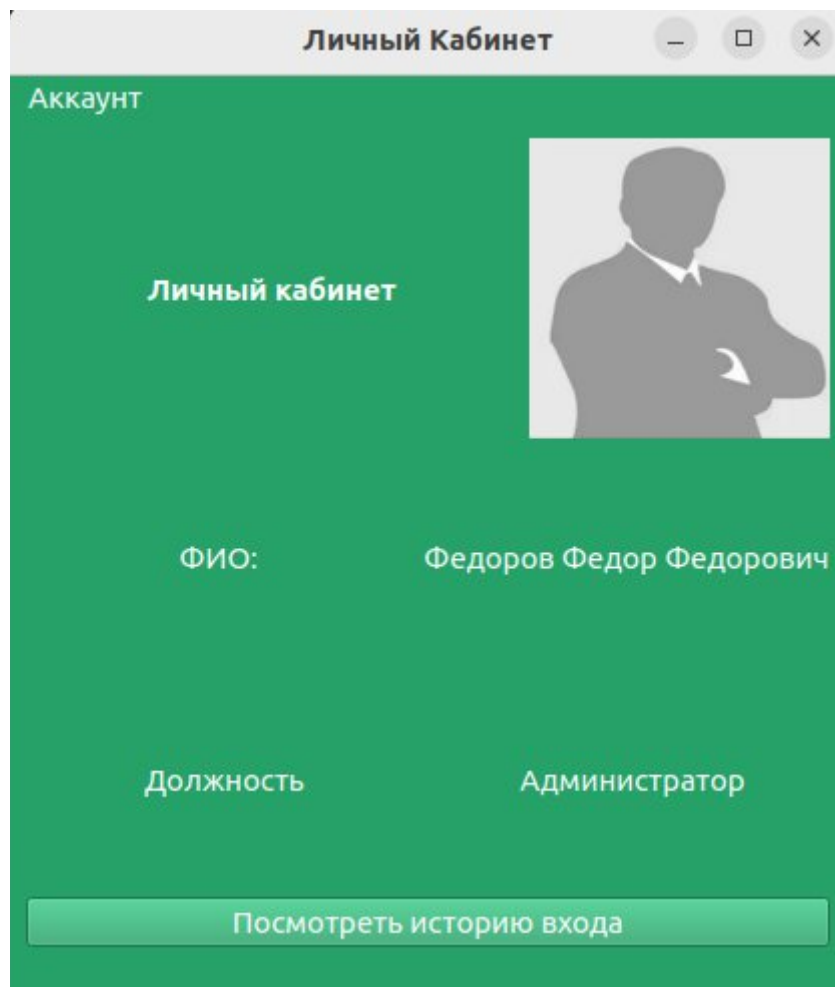
Данный раздел разработан с целью обеспечения разграничения доступа сотрудников к функционалу приложения. Суть заключается в сравнении данных, указанных пользователем на форме авторизации со значениями, хранящимися в базе и, в случае их совпадения, предоставление пользователю доступа к его личному кабинету. Создаем классы MVC для Authorize, в которых будет располагаться код основной формы авторизации. Так же создана функция скрытия/показа пароля, запрет ввода пароля, после трех неверных попыток ввода данных на 10 с. и вызов диалога с капчей



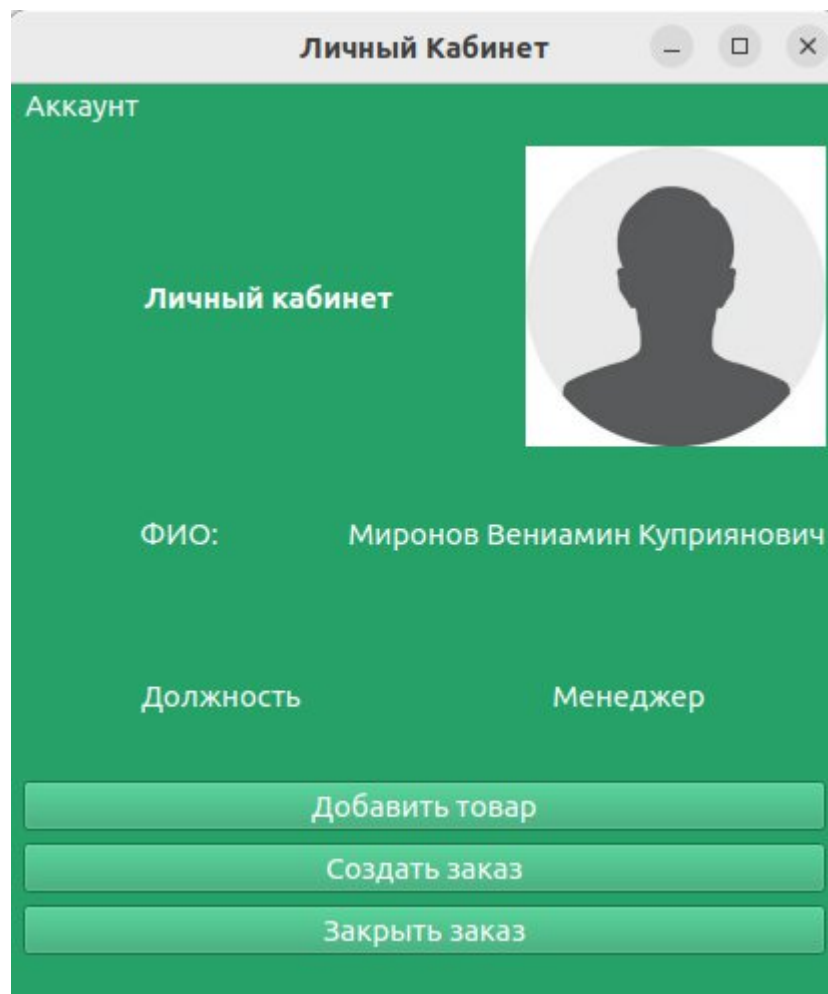
Форма авторизации сотрудников

Разработка раздела «Профиль сотрудника»

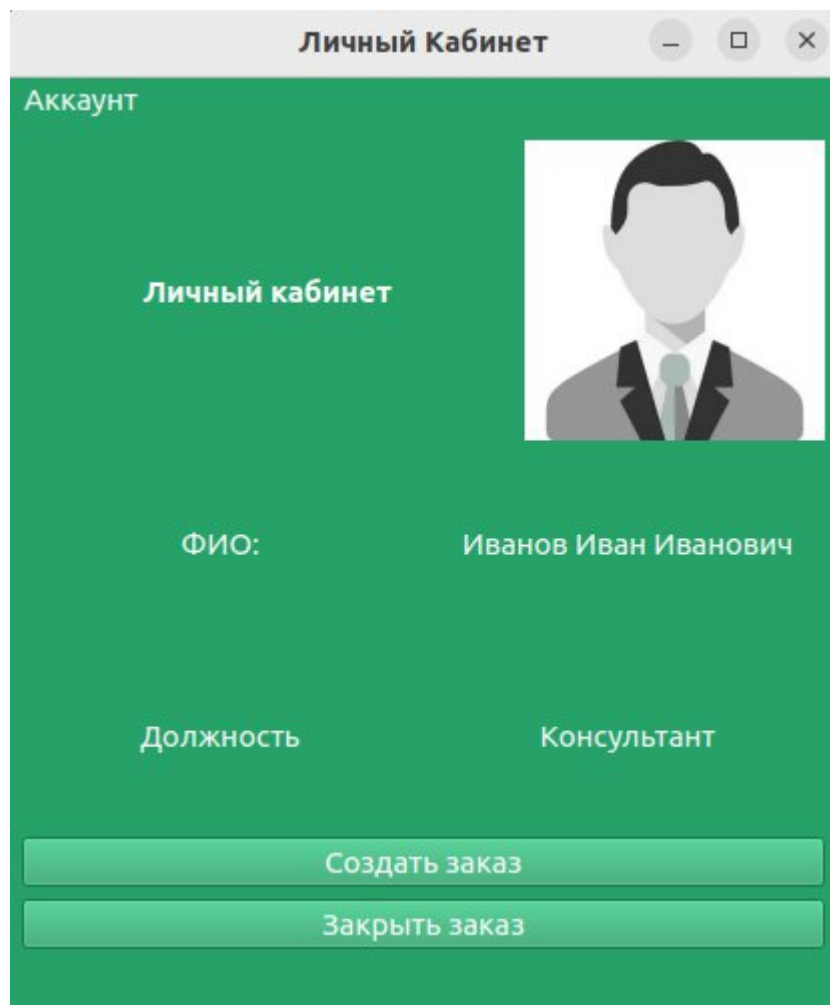
Эту страницу будут видеть после успешной авторизации все сотрудники службы такси. В зависимости от их должности определенные кнопки будут скрыты. Раздел профиль сотрудника является переходным для дальнейшего функционала в зависимости от должности сотрудника. Также в данном разделе сотруднику будет показано его ФИО, должность, а также его фотография. В верхнем меню можно выйти из аккаунта и из приложения



Окно профиля менеджера



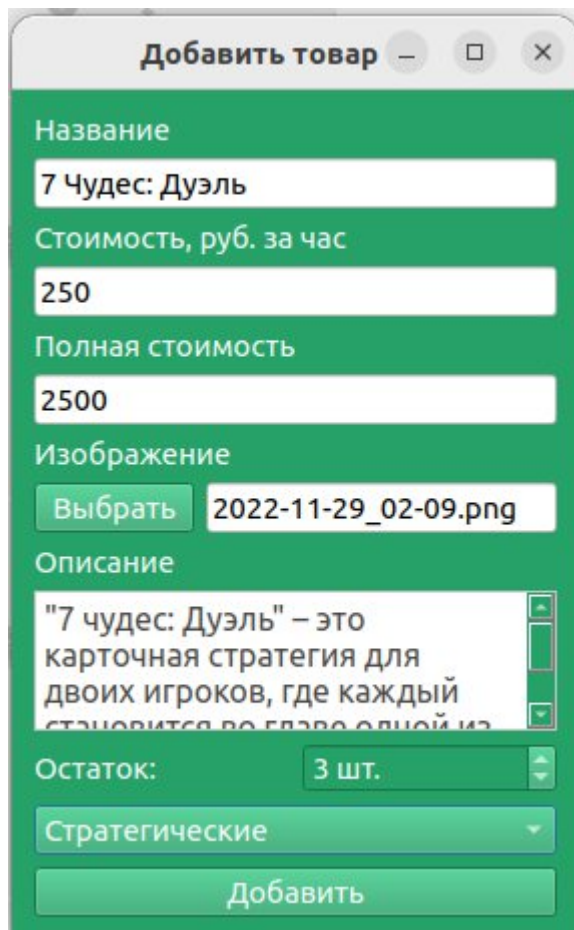
Окно профиля менеджера



Окно профиля консультанта

Разработка раздела «Добавление нового товара»

Данный раздел нужен для добавления нового товара в базу данных настольного клуба. Этим функционалом может пользоваться только менеджер.



Окно раздела «Добавление товара»

Разработка раздела «Просмотр истории входа»

Данным функционалом может обладать Администратор. Он нужен для контроля персоналом.

История входа				
Поиск по логину				Поиск
	Логин	Тип входа	Время	id
27	fedorov@namecomp.ru	успешно	2022-12-02 00:40:26.697006	27
28	mironov@namecomp.ru	успешно	2022-12-02 00:41:37.561552	28
29	Ivanov@namecomp.ru	успешно	2022-12-02 00:42:35.853048	29
30	mironov@namecomp.ru	успешно	2022-12-02 00:46:25.321364	30
31	fedorov@namecomp.ru	успешно	2022-12-02 22:00:05.509261	31
32	fedorov@namecomp.ru	успешно	2022-12-02 22:00:11.510146	32
33	fedorov@namecomp.ru	успешно	2022-12-03 23:11:18.811384	33
34	pupkin@namecomp.ru	не успешно	2022-12-03 23:11:46.042422	34
35	rydov@namecomp.ru	не успешно	2022-12-03 23:11:56.371141	35
36	mironov@namecomp.ru	успешно	2022-12-03 23:12:04.938667	36
37	fedorov@namecomp.ru	успешно	2022-12-03 23:12:14.240625	37

Окно раздела «Регистрация клиента»

Разработка раздела «Создание заказа»

Данный раздел предназначен для менеджера и консультанта. В нем он может добавить новый заказа в базу данных настольного клуба.

Новый заказ

Номер заказа 45462586

Время проката 5 час

Клиент Фролов Андрей Иванович

Добавить товаров

	Наименование	Руб. за час	Сумма
1	Сиггил	100	500
2	Цитадели	200	1000
3	Битва боевых ...	150	750

Итого: 2250 руб.

Сформировать заказ

Окно раздела «Создание заказа»

report.pdf

Номер заказа: 45462587
Клиент: Фролов Андрей Иванович
Количество часов проката: 1

Список товаров:

Similo: Мифы 50 руб.
Битва боевых магов: Битва на горе Черепламени 150 руб.
Итого: 200

Чек заказа после его формирования

Разработка раздела «Формирование заказа»

При формировании заказа необходимо определить его содержимое. Для этой цели служит окно раздела «Формирование заказа».

Формирование заказа

Товаров в заказе: 2

Посмотреть заказ

Готово

Поиск по названию:






По убыванию

Имя

Все категории

<

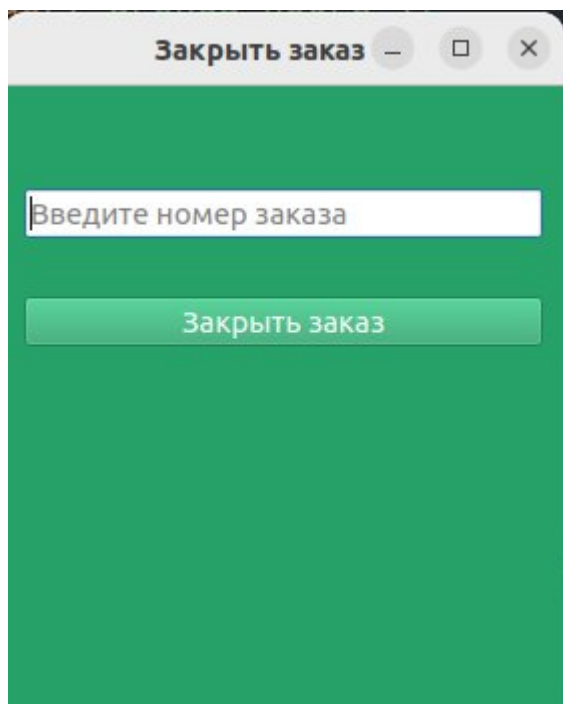
>

<div>Цитадели</div> <div></div> <div>Руб/час: 200</div> <div>Категория: Стратегические</div> <div>Осталось: 1</div> <div>Добавлено</div>	<div>Сумеречная борьба</div> <div></div> <div>Руб/час: 500</div> <div>Категория: Стратегические</div> <div>Осталось: 4</div> <div>Добавить в заказ</div>	<div>Сиггил</div> <div>Нет фото</div> <div>Руб/час: 100</div> <div>Категория: Карточные</div> <div>Осталось: 1</div> <div>Добавить в заказ</div>
<div>Пэчворк</div> <div></div> <div>Руб/час: 170</div> <div>Категория: Абстрактные</div> <div>Осталось: 3</div> <div>Добавить в заказ</div>	<div>Манчкин</div> <div></div> <div>Руб/час: 200</div> <div>Категория: Вечериночные</div> <div>Осталось: 9</div> <div>Добавить в заказ</div>	<div>Маленький мир</div> <div></div> <div>Руб/час: 300</div> <div>Категория: Стратегические</div> <div>Осталось: 1</div> <div>Добавлено</div>

Окно раздела «Формирование заказа»

Разработка раздела «Закрытие заказа»

В данном разделе консультант и менеджер смогут закрыть заказ по его номеру, когда клиент вернёт то, что брал в аренду у настольного клуба.



Окно раздела «Заккрытие заказа»

Страница авторизации пользователя

Для того, чтобы попасть в личный кабинет и получить доступ ко всему ему функционалу, необходимо совершить вход в систему посредством авторизации на форме приложения.

На форме будет предложено ввести логин и пароль.

После ввода персональных данных необходимо нажать на кнопку «Войти» и, в случае ввода корректных данных пользователь получает доступ к Личному кабинету сотрудника, в противном случае на форму будет выдано предупреждение: «Неверный логин или пароль». При вводе неверных данных более трёх раз подряд, будет вызвано диалоговое окно с вводом капчи. Возможность авторизации будет приостановлена до тех пор, пока не будет введена капча.

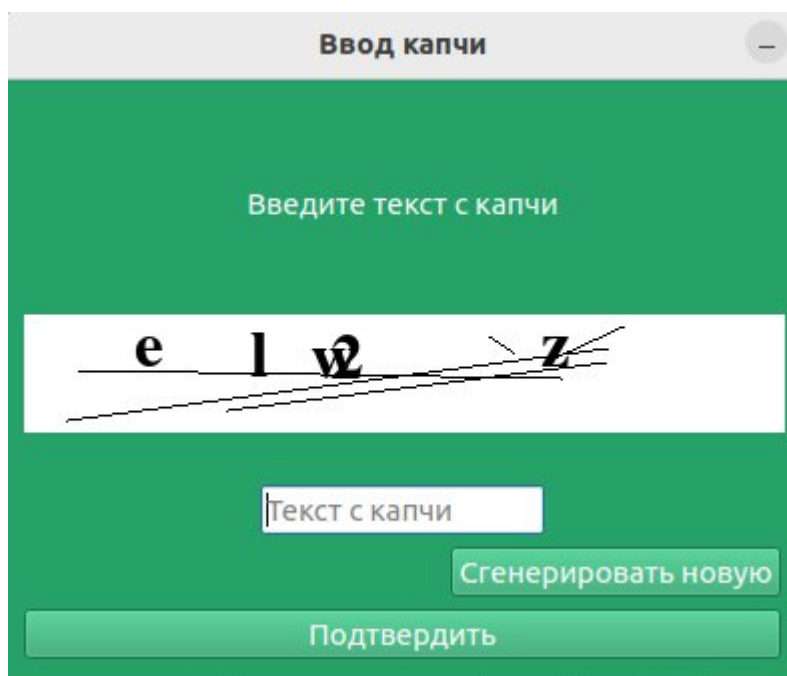


Рис. 14. Окно для ввода капчи

1. Вводим логин и пароль, далее нажимаем на кнопку «Войти» (см. рис. 3);
2. В случае успешной авторизации попадаем на профиль сотрудника (см. рис. 4, 5, 6);

Профиль сотрудника

Находясь на экране «Профиль» сотрудник, в зависимости от его должности, может выбрать любую доступную ему функцию (см. рис. 4, 5, 6).

Добавление нового товара

Находясь на экране «Профиль» сотрудник должен кликнуть по кнопке с названием «Добавить товар», чтобы добавить новый товар. После этого выполняется переход на экран «Добавление товара».

Далее нужно ввести, название товара, его стоимость за час и полную стоимость, описание, выбрать изображение в диалговом окне с выбором файла, указать остаток в настольном клубе, и выбрать категорию, далее нажать на кнопку «Добавить».

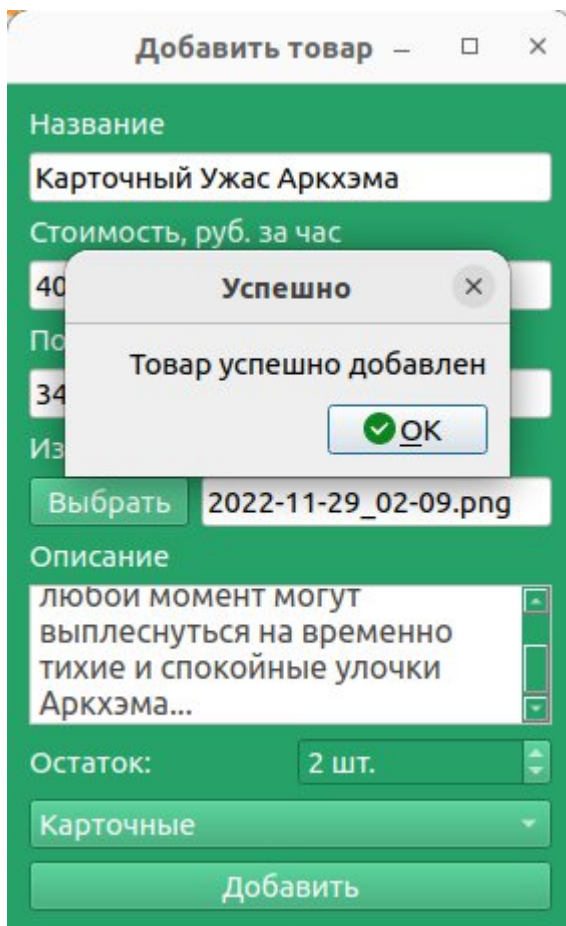


Рис. 15. Добавление нового услуги

Создание нового заказа

Находясь на экране «Профиль» сотрудник должен кликнуть по кнопке с названием «Создать заказ», чтобы создать заказ. После этого выполняется переход на экран «Новый заказа».

На этом экране сотруднику нужно указать номер заказа, время часов проката, выбрать клиента из выпадающего списка, и добавить товаров в другом окне «Формирование заказа», и нажать на кнопку «Сформировать заказ ». В случае успешного формирования заказа, будет показана надпись

«Заказ сформирован», в случае провала будет показана информации об ошибке. Также на данном экране после формирования заказа, генерируется pdf-чек со всеми данными, которые были указаны в заказе. После вызывается диалоговое окно с выбором места для сохранения чека и дальнейшей печати.

Наименование	Руб. за час	Сумма
1 Битва боевых ...	150	450
2 Ужа		
3 Сум		

Создание нового заказа.


Формирование заказа

Находясь на экране «Новый заказ» сотрудник должен кликнуть по кнопке с названием «Добавить товаров», чтобы добавить товаров заказ. После этого выполняется переход на экран «Формирование заказа».

Здесь мы можем увидеть постраничный список товаров. Для удобства выбора можем выбрать направление сортировки, атрибут, по которому можно сортировать, выбрать категорию товара, а также фильтрация по имени товара.

Чтобы добавить товар в заказ, нужно нажать на кнопку «Добавить в заказ». После этого количество уменьшится на единицу, а кнопка изменит свой текст на «Добавлено». Для того, чтобы убрать товар из заказа, нужно повторно нажать на эту кнопку. Текст на ней опять станет «Добавить в заказ».

Битва за Рокуган



Руб/час: 250

Категория: Стратегический

Осталось: 5

Добавлено

Добавление нового товара в заказ

После того, как все товары будут выбраны, нужно нажать на кнопку «Готово» в правом верхнем углу. После этого текущее окно будет закрыто, а товары добавятся в таблицу с товарами в окне «Новый заказ».

Новый заказ

Номер заказа 45462589

Время проката 3 час

Клиент Фролов Андрей Иванович

Добавить товаров

	Наименование	Руб. за час	Сумма
1	Битва боевых ...	150	450
2	Similo: Мифы	50	150
3	Битва боевых ...	150	450
4	Карточный Уж...	400	1200

Итого: 3000 руб.

Сформировать заказ

Список выбранных товаров в окне нового заказа

Просмотр истории входа

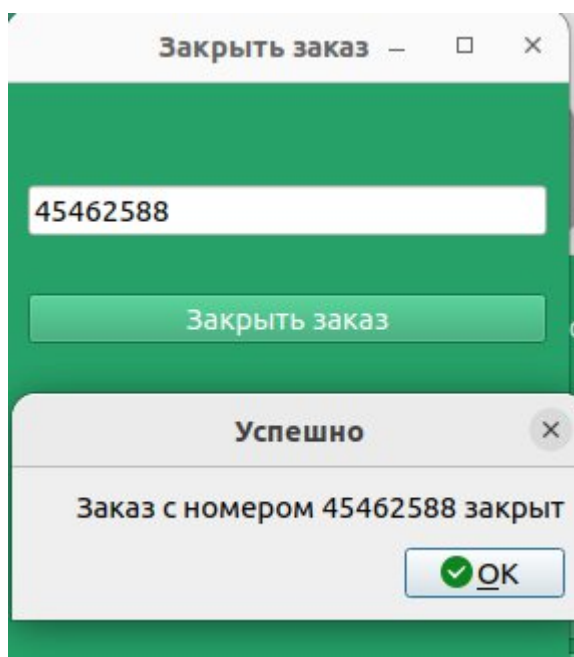
Находясь на экране «Профиль» сотрудник должен кликнуть по кнопке с названием «История входа», чтобы просмотреть работников по истории входа. После этого выполняется переход на экран «Просмотр работников по истории входа» (см. рис. 10).

Закрытие заказа

Находясь на экране «Профиль» сотрудник должен кликнуть по кнопке с названием «Закрыть заказ», чтобы закрыть незавершённые заказы. После этого выполняется переход на экран «Закрыть заказ».

На данном окне сотрудники должны написать номер заказа. Если заказ будет уже завершённым или отсутствовать, то система не даст завершить такой заказ, что уменьшает риски ввести неверный номер.

В случае завершения заказа, будет показана надпись «Заказ закрыт». После этого, возвращенные товары станут доступны для следующих заказов.



Завершение заказа

Ссылка на репозиторий исходного кода приложения

<https://gitlab.nntc.nnov.ru/vlad.tancev/coursework>