

Министерство науки и высшего образования Российской Федерации
Муромский институт (филиал)
Федерального государственного бюджетного образовательного учреждения высшего образования
«Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»
Факультет Информационных Технологий и Радиоэлектроники Кафедра программной инженерии

Курсовая работа

По дисциплине: «Разработка корпоративных приложений»
Тема работы: «Детейлинг центр»

Группа

ПИН – 121

Студент

Кокурин Я. Д

Цели и задачи курсовой работы

Целями данной работы являются разработка моделей информационной системы детейлинг центра с учётом требуемых обработки данных для повышения эффективности предприятия и проектирование веб-приложения.

Задачей проекта является программная реализация веб-приложения, проверка его работоспособности.

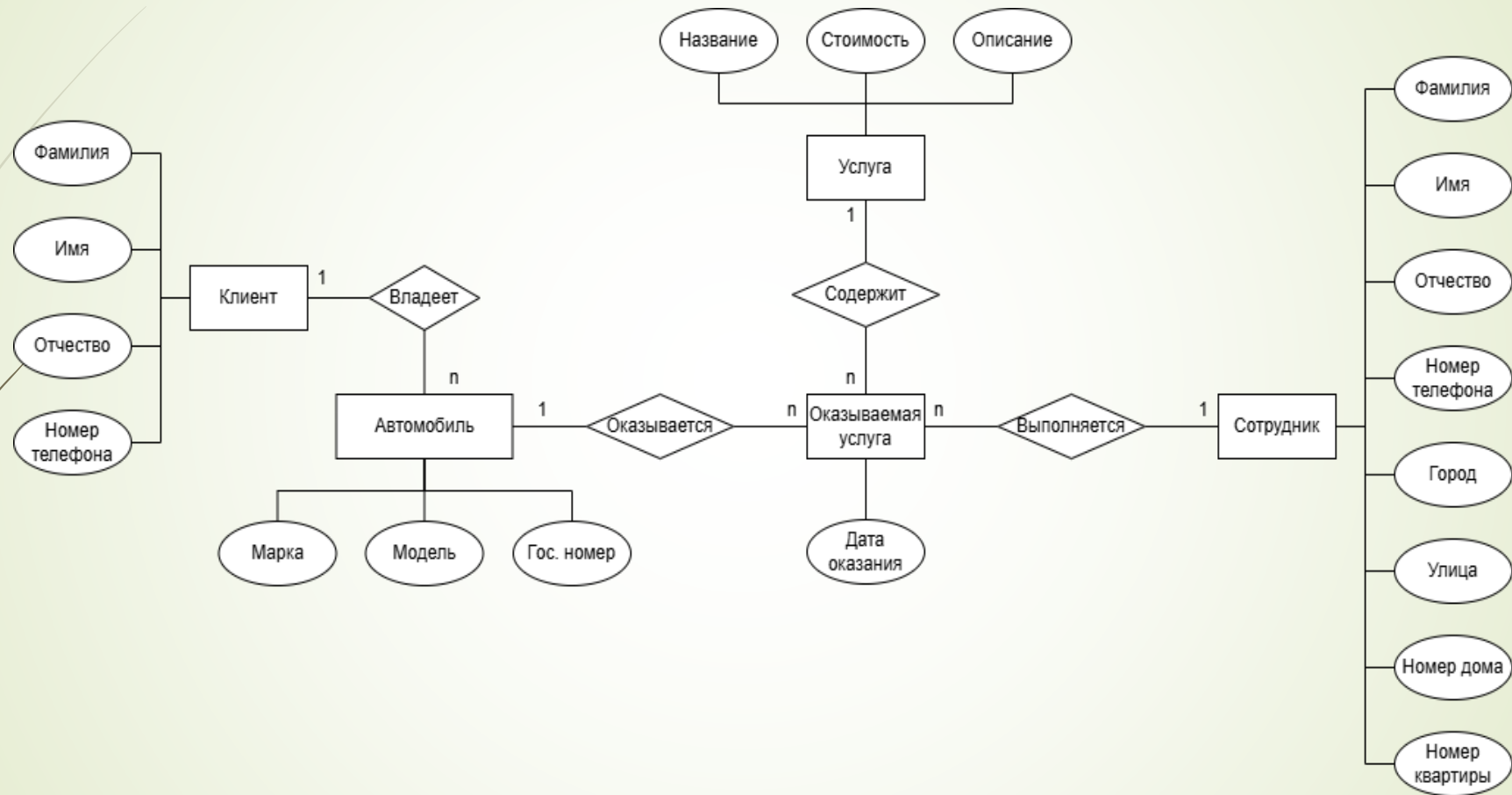
Анализ технического задания

Разрабатываемое приложение предназначено для использования в детейлинг центре, занимающемся оказанием специализированных процедур по уходу за автомобилем. Программный продукт предназначен для автоматизации деятельности детейлинг центра. Администратор реализует CRUD методы обработки данных сотрудников, услуг детейлинг центра. Сотрудник детейлинг-центра управляет записью клиентов на обслуживание, реализует CRUD методы обработки данных клиентов и их автомобилей. Клиент просматривает историю обслуживания своих автомобилей.

Разработанные модели данных

- Концептуальная модель данных;
- Логическая модель данных;
- Физическая модель данных

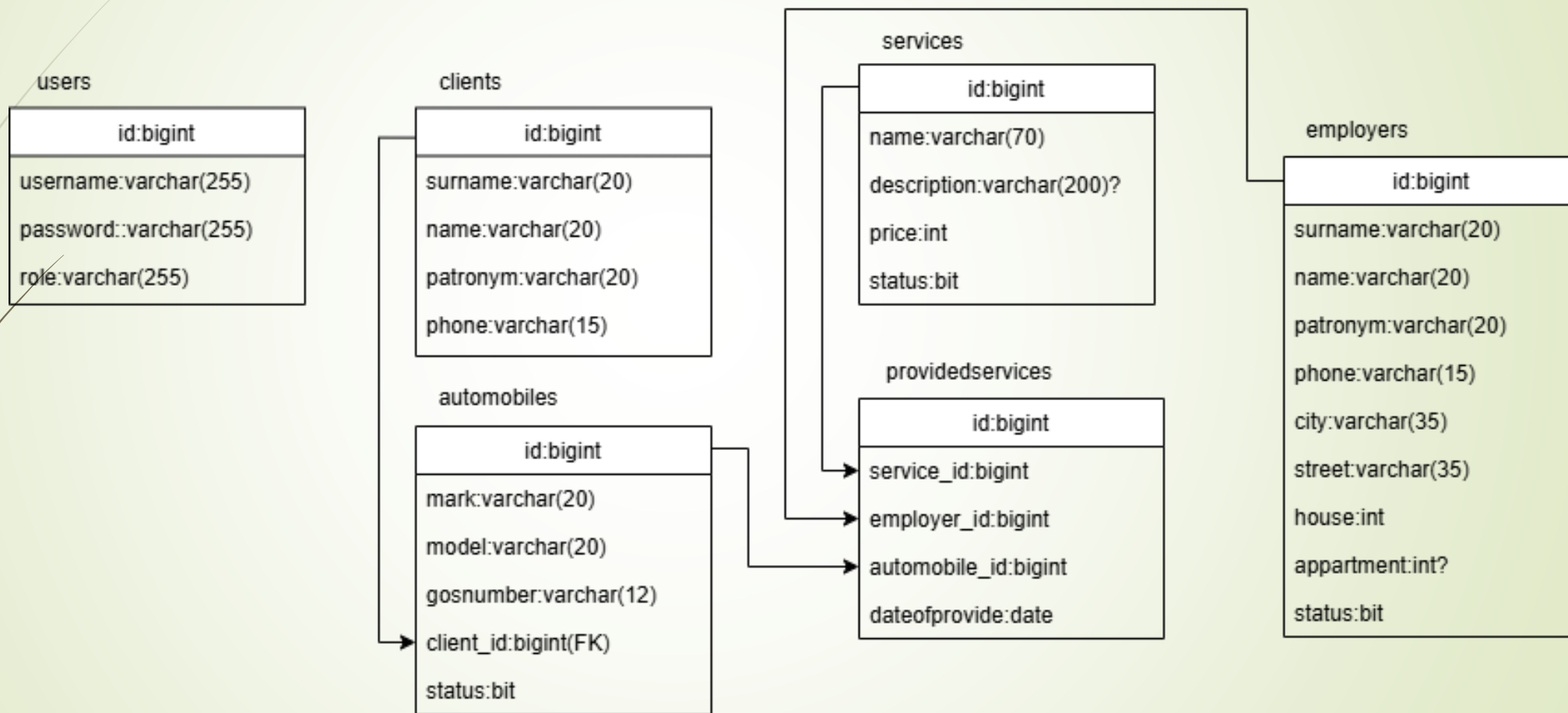
Концептуальная модель данных



Логическая модель данных



Физическая модель данных



Диаграммы прецедентов



Диаграмма прецедентов



Реализация приложения

Инструменты для реализации приложения:

- Среда разработки IntelliJ IDEA Community Edition 2024.2.3, язык программирования Java.
- Для хранения необходимой информации применяется СУБД Microsoft SQL Server
- Управление версиями осуществлялось через систему контроля версий GitHub

Реализация системы

```
@Table(uniqueConstraints =
{
    @UniqueConstraint(columnNames={"id", "phone", "status"})
},
name = "employers"
)

public class Employer {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Column(name = "id")
    private Long id;

    @Column(name = "name", nullable = false)
    @NotBlank(message = "Поле \"Имя\" не может быть пустым")
    @Size(min = 2, max=20, message = "Допустимая длина имени от 2ух до 20ти символов")
    private String name;

    @Column(name = "surname", nullable = false)
    @NotBlank(message = "Поле \"Фамилия\" не может быть пустым")
    @Size(min = 2, max=20, message = "Допустимая длина фамилии от 2ух до 20ти символов")
    private String surName;

    @Column(name = "patronym", nullable = false)
    @Size(min = 2, max=20, message = "Допустимая длина отчества от 2ух до 20ти символов")
    private String patronym;

    @Column(name = "phone", nullable = false)
    @Pattern(regexp = "^8\\(\\d{3}\\)\\d{3}-\\d{2}-\\d{2}$", message = "Телефон должен быть введён в формате: 8(**)***-**-**")
    private String phone;

    @Column(name = "city", nullable = false)
    @Size(min = 2, max=35, message = "Допустимая длина названия города от 2ух до 35ти символов")
    private String city;

    @Column(name = "street", nullable = false)
    @Size(min = 2, max=35, message = "Допустимая длина названия улицы от 2ух до 35ти символов")
}
```

Класс сущности
«Сотрудник»

Реализация системы

```
@Entity
@Data
@NoArgsConstructor
@Table(uniqueConstraints =
    {
        @UniqueConstraint(columnNames={"id", "name", "status"})
    },
    name = "services"
)
public class Service {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Column(name = "id")
    private Long id;

    @Column(name = "name", nullable = false)
    @NotBlank(message = "Поле \"Название\" не может быть пустым")
    @Size(min = 2, max=70, message = "Допустимая длина названия услуги от 2ух до 70ти символов")
    private String name;

    @Column(name = "price", nullable = false)
    @Min(value = 0, message = "Стоимость не может быть < 0")
    private int price;

    @Column(name = "description", nullable = true)
    @Size(min = 2, max=200, message = "Допустимая длина названия услуги от 2ух до 200от символов")
    private String description;

    @Column(name = "status", nullable = false)
    private boolean status;

    @OneToMany(mappedBy = "service", cascade = CascadeType.ALL)
    @JsonBackReference
    private List<ProvidedService> providedServices;
}
```

Класс сущности
«Услуга»

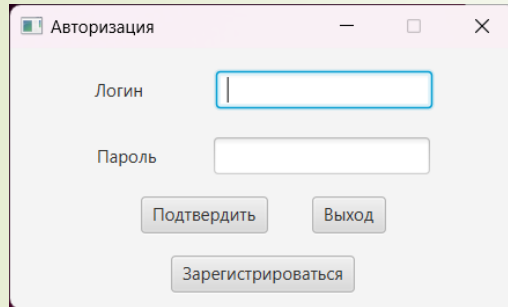
Реализация системы

```
11 public class ServiceService {
12     public Service getById(Long id) { 2 usages  Кокурин Ярослав
13     }
14
15     public void deleteServiceById(Long id) { 2 usages  Кокурин Ярослав
16         if (serviceRepository.existsById(id)){
17             Service existsService = getServiceById(id);
18             existsService.setStatus(false);
19             serviceRepository.save(existsService);
20         }
21     }
22
23     @
24     public Service addEditService(Service service) { 2 usages  Кокурин Ярослав
25         if(service.getId()==null){
26             service.setStatus(true);
27         }
28         else{
29             deleteServiceById(service.getId());
30             service.setId(null);
31         }
32         return serviceRepository.save(service);
33     }
34
35     ///
36     public Service getServiceByName(String name) { 1 usage  Кокурин Ярослав
37         Optional<Service> service = serviceRepository.findByNameAndStatusTrue(name);
38         if(service.isEmpty()){
39             throw new EntityNotFoundException("Услуга не найдена");
40         }
41         return service.get();
42     }
43 }
44
45
46
47
48
49
50
51
52
53
54 }
```

```
@PostMapping("/services/name") no usages  Кокурин Ярослав
public Service getServiceByName(@RequestBody String name){
    return serviceService.getServiceByName(name);
}
```

Метод поиска актуальной
услуги по имени

Демонстрация работы приложений



Авторизация

Логин

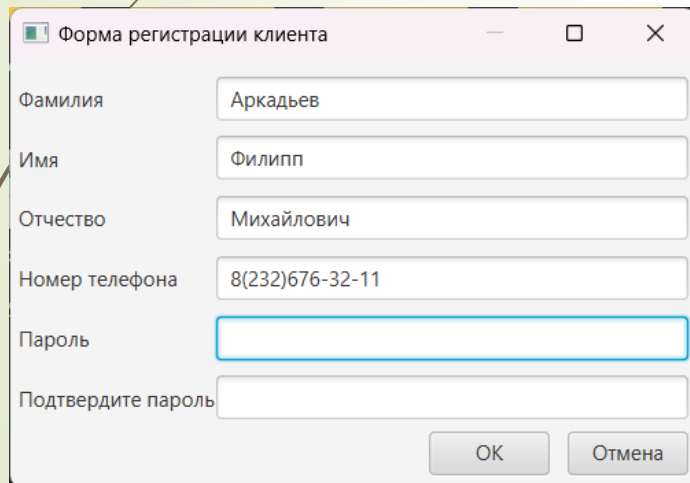
Пароль

Подтвердить

Выход

Зарегистрироваться

Стартовая форма приложения



Форма регистрации клиента

Фамилия

Имя

Отчество

Номер телефона

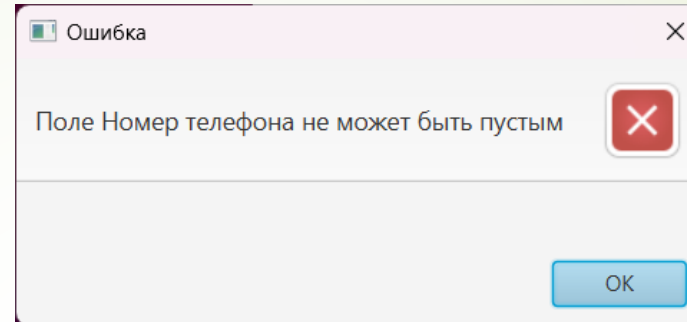
Пароль

Подтвердите пароль

OK

Отмена

Форма регистрации клиента

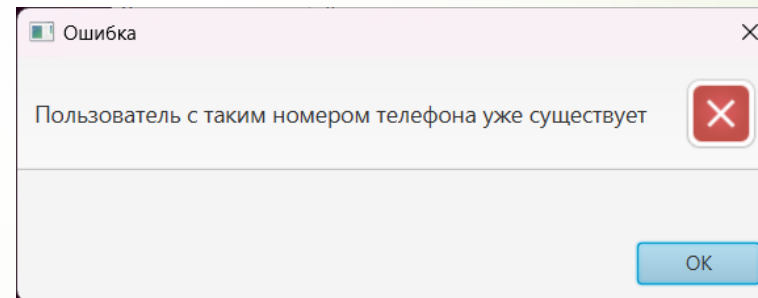


Ошибка

Поле Номер телефона не может быть пустым

OK

Валидация данных на стороне клиента

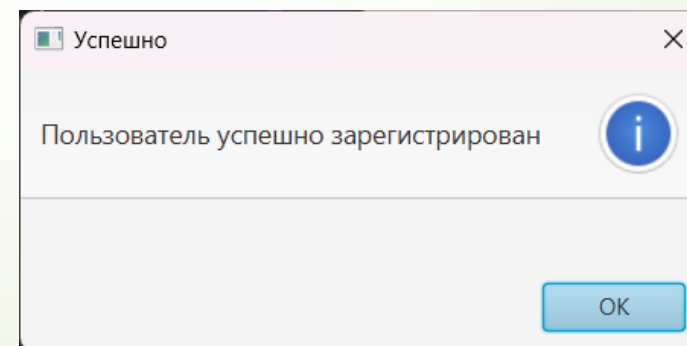


Ошибка

Пользователь с таким номером телефона уже существует

OK

Валидация данных на стороне сервера



Успешно

Пользователь успешно зарегистрирован

OK

Демонстрация работы приложений

Форма для работы сотрудника

Список оказанных услуг | Список клиентов | Список автомобилей клиентов

Название	Клиент	Автомобиль	Сотрудник	Стоимость	Дата оказания ▲
Мойка автомобиля	Ермилов Михаил Анатольевич 8(123)456-78-90	BMW M5 P444MP456rus	Зубова Альбина Артёмовна 8(777)434-22-11	500	2020-02-19
Мойка автомобиля	Ермилов Михаил Анатольевич 8(123)456-78-90	BMW M2 P444MP457rus	Карасёв Эдуард Михайлович 8(777)434-22-12	600	2021-02-19
Мойка автомобиля	Ермилов Михаил Анатольевич 8(123)456-78-90	BMW M2 P444MP457rus	Зубарев Аркадий Петрович 8(444)233-33-51	600	2022-02-19
Мойка автомобиля	Ермилов Михаил Анатольевич 8(123)456-78-90	BMW M5 P444MP456rus	Зубова Альбина Артёмовна 8(777)434-22-11	500	2024-02-19

От: 01.01.2020 До: 25.12.2024

Добавить Изменить Удалить

Форма для работы сотрудника

Список оказанных услуг | Список клиентов | Список автомобилей клиентов

Название	Клиент	Автомобиль	Сотрудник	Стоимость	Дата оказания ▲
Мойка автомобиля	Ермилов Михаил Анатольевич 8(123)456-78-90	BMW M5 P444MP456rus	Зубова Альбина Артёмовна 8(777)434-22-11	500	2020-02-19
Мойка автомобиля	Ермилов Михаил Анатольевич 8(123)456-78-90	BMW M2 P444MP457rus	Карасёв Эдуард Михайлович 8(777)434-22-12	600	2021-02-19
Мойка автомобиля	Ермилов Михаил Анатольевич 8(123)456-78-90	BMW M2 P444MP457rus	Зубарев Аркадий Петрович 8(444)233-33-51	600	2022-02-19

От: 01.01.2020 До: 03.12.2023

Добавить Изменить Удалить

Форма для работы сотрудника. Демонстрация получения списка оказанных услуг за указанный период

Заключение

Результатом выполнения курсовой работы по дисциплине «Разработка корпоративных приложений» является разработанная распределенная информационная система для автоматизации деятельности детейлинг центра.

В ходе данной работы были составлены концептуальная, логическая, физическая модели данных, разработаны диаграммы прецедентов для пользователей системы созданы клиентское и серверное приложения, отвечающие тем требованиям, которые были заложены перед их созданием.

По результатам выполнения работы, было проведено тестирование приложения, подтверждающее корректность работы всего функционала системы