

JavaProject : ShareCar

-차량 공유 시스템
개발자 : 송재완

Index

프로젝트 소개

개발 일정

요구 사항

기능 및 구현

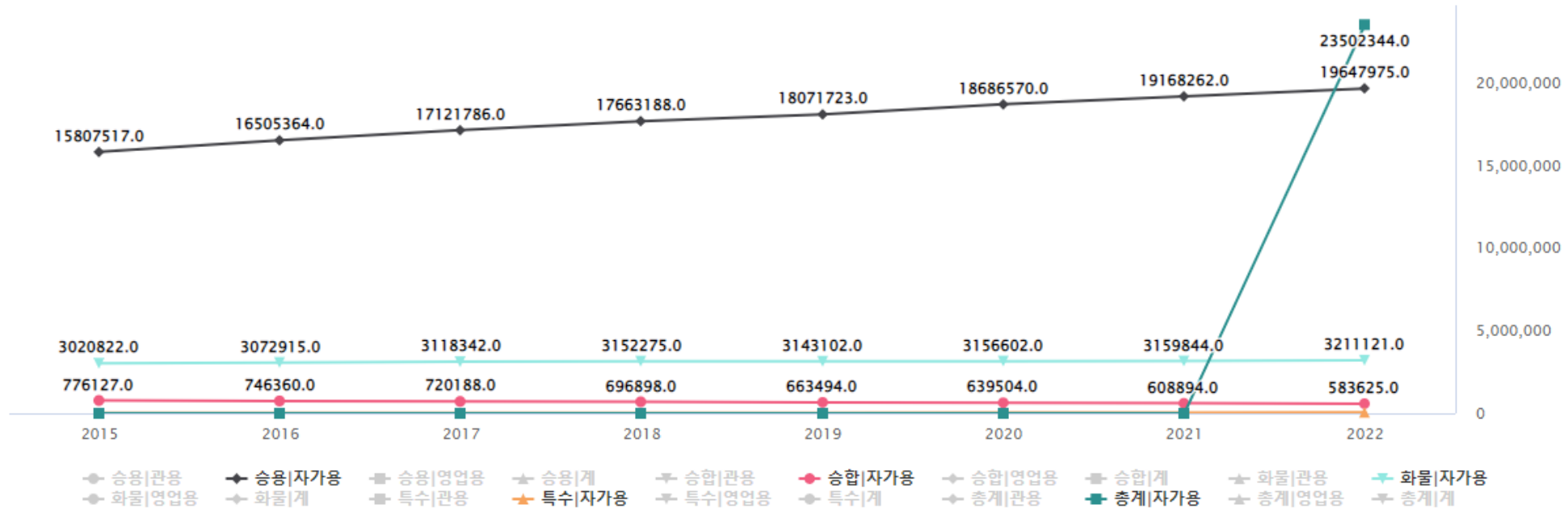
소스 리뷰

후기

프로젝트 소개

개발 동기 및 개발 도구

개발 동기



국토교통 통계누리의 자동차등록현황보고에 따르면,
국내 자가용은 점점 증가하는것을 알수있음.
그러나 실사용하지 않고 보유만 하는 가구들도 많음.
따라서 차량공유렌탈 시스템을 통해서 유지만 하고 있는 차량으로
수익화할수있는 프로그램을 만들고 싶어서 만들게 됨.
참조 (국토교통부 공식 블로그)- <https://blog.naver.com/mltmkr/222699719872>

개발 도구



ORACLE 11^g
DATABASE



Index

프로젝트 소개

개발 일정

요구 사항

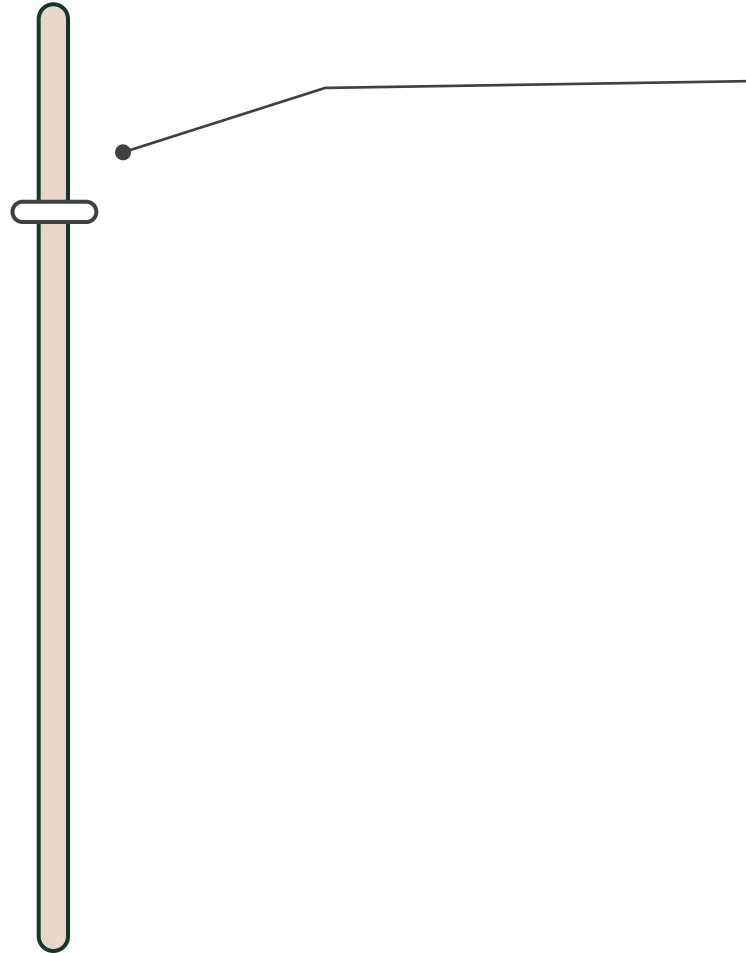
기능 및 구현

소스 리뷰

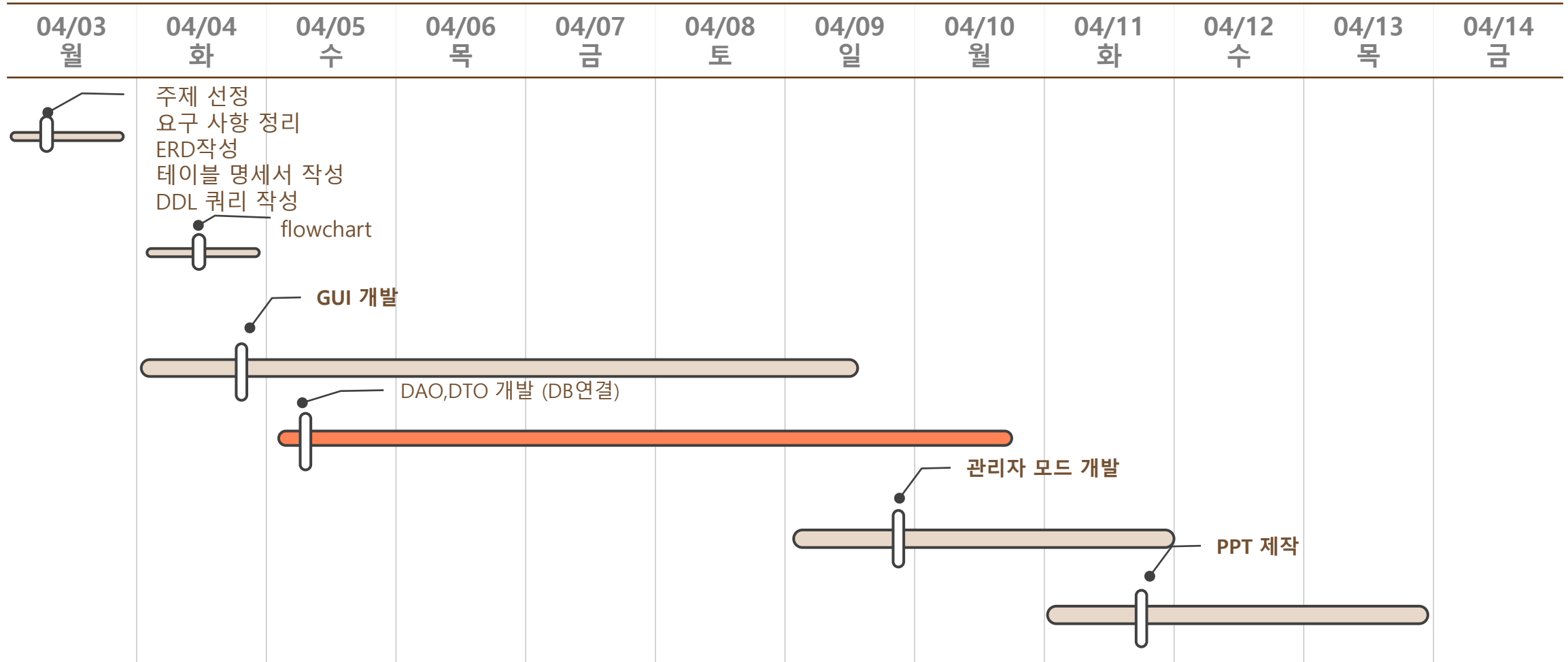
후기

개발 일정

세부 일정



개발 일정



Index

프로젝트 소개

개발 일정

요구 사항

기능 및 구현

소스 리뷰

후기

요구 사항

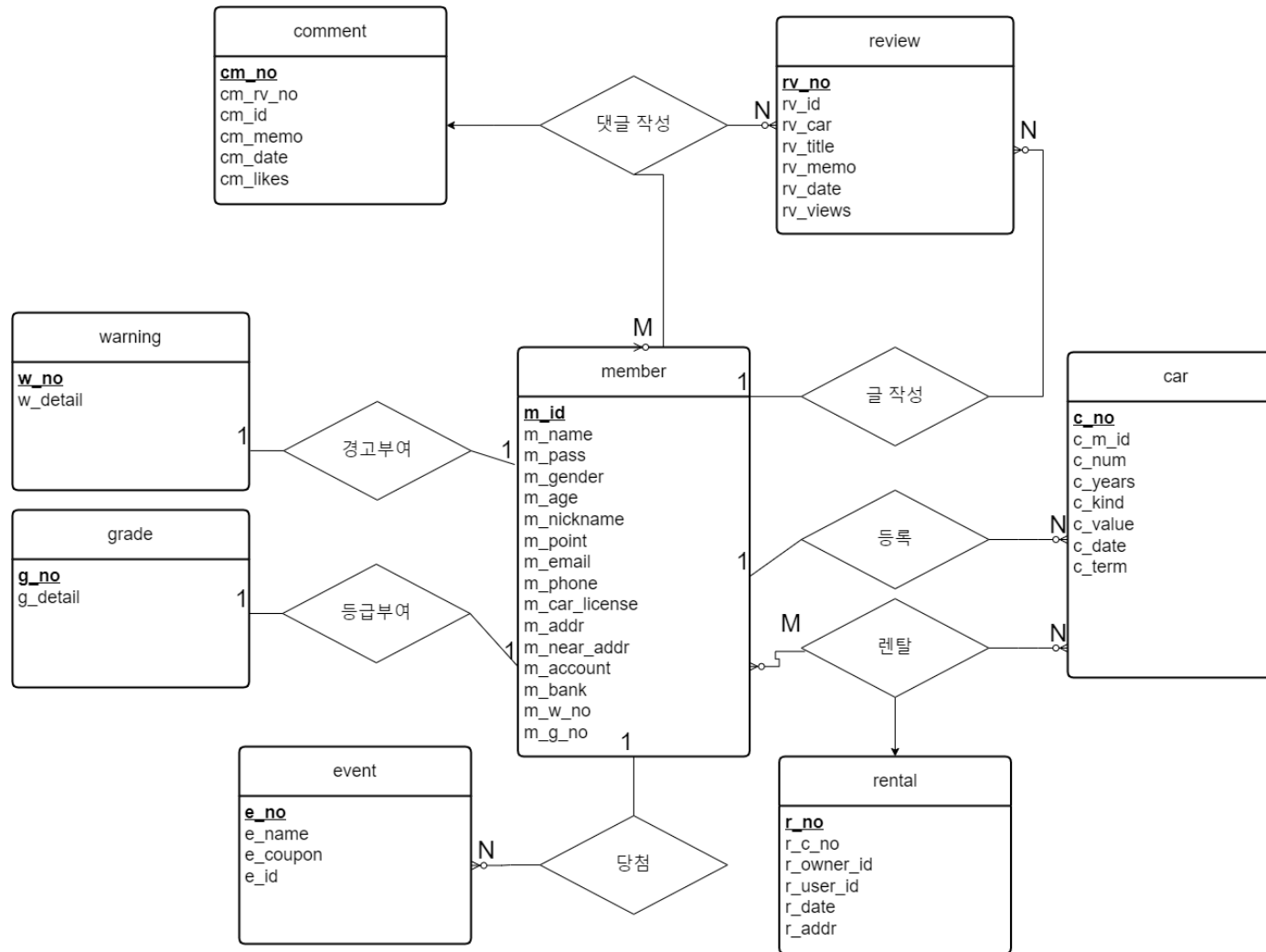
요구 사항 정리 표, ERD,
테이블 명세서, Flow chart,
Class diagram

요구 사항 정리 표

요구사항 정리표			
개체	속성	요구사항	세부사항
회원정보	이름	모든속성 반드시 입력 회원은 차량 여러대를 등록이 가능하고 여러대를 렌탈 가능하다 관리자는 관리자계정 으로 로그인하면 다른 메뉴 호출	
	성별		
	나이		
	아이디		기본키
	닉네임		중복금지
	비밀번호		특수문자 포함 8자리 이상
	포인트		default 1000 가입환영 포인트
	이메일		인증필요
	휴대전화		인증필요
	운전면허증 번호		명의 일치 확인 필요
	주소지		시 단위까지 입력
	상세주소		상세주소 입력
	계좌번호		not null
	경고번호		외래키,default 0
	회원등급번호		외래키,default 0
차량정보	등록번호	모든 속성 반드시 입력	기본키,sequence
	아이디		외래키
	차량번호		중복금지
	연식		
	차종		
	차량상태		관리자 입력
	등록일		default sysdate
	등록기간		
경고	경고 번호	2번째 경고상황일경우 영구정지	기본키
	상세내역	경고테이블 데이터는 변동없음	check (1차경고,영구정지)
등급	등급번호	회원등급은 포인트에 따라 바뀐다.	기본키
	상세등급		check(common,vip,vvip)

요구사항 정리표			
개체	속성	요구사항	세부사항
렌탈	관리번호	현재 렌탈중인 차량의 정보를 저장할수 있는 table	기본키,sequence
	차량번호		외래키
	차주아이디		외래키
	렌탈자 아이디		외래키
	렌탈기간		
	렌탈주소		
이벤트	이벤트 번호	모든 속성 반드시 입력, 이벤트 참가는 포인트로 응모	기본키,sequence
	이벤트명		
	쿠폰번호		중복없는 랜덤문자열 8자리
	당첨자 아이디		외래키
후기 게시판	글번호	후기 작성시 1000 포인트 지급	기본키
	작성자 아이디		외래키
	사용한 차종		not null
	글 제목		not null
	후기		not null, 이미지 첨부 가능
	작성일자		default sysdate
댓글	조회수	댓글은 추천이 가능하다.	default 0 자동카운트
	댓글번호		기본키
	글번호		외래키
	작성자 아이디		외래키
	댓글 내용		not null
	작성일자		default sysdate
	추천		default 0

ERD

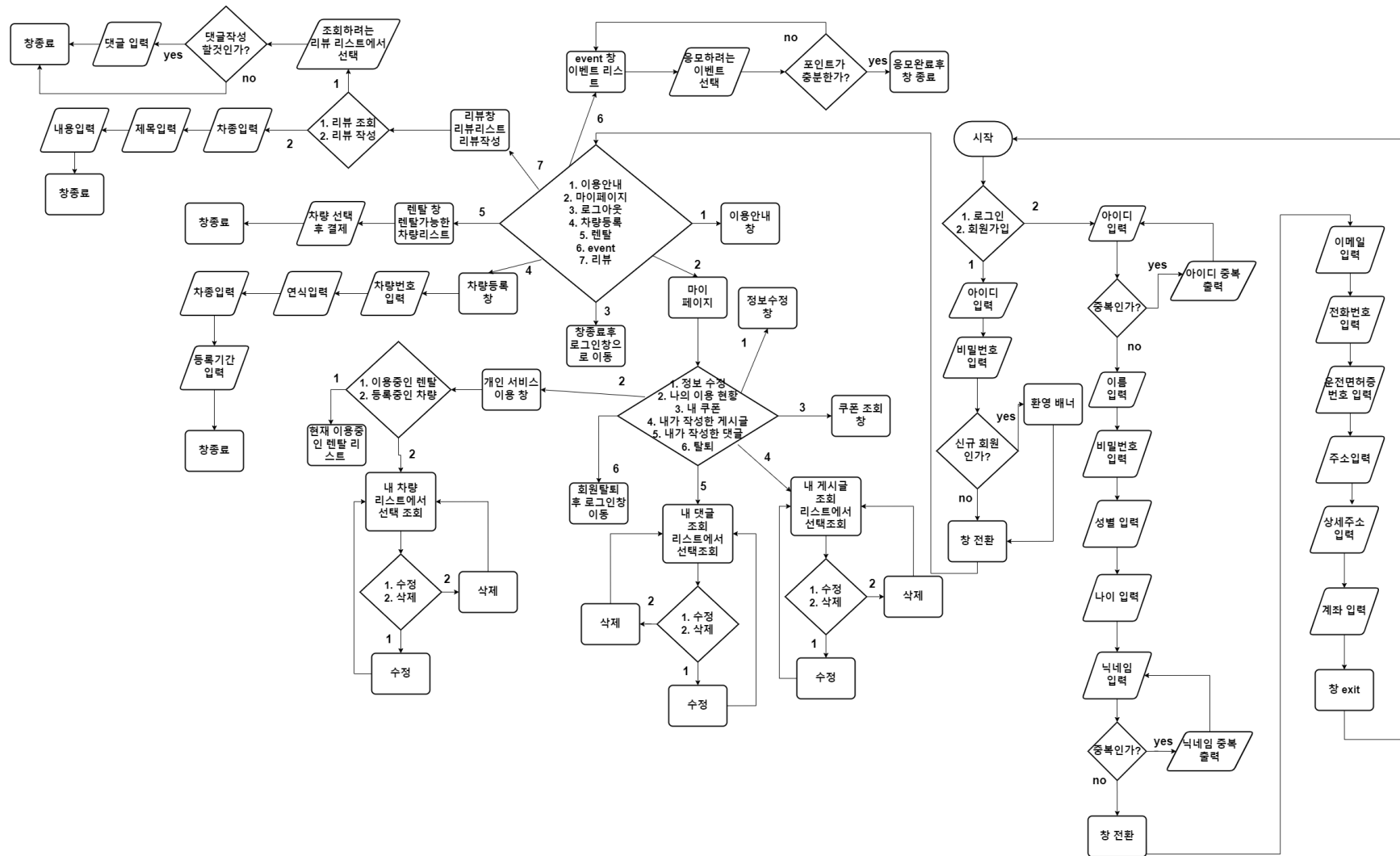


테이블 명세서

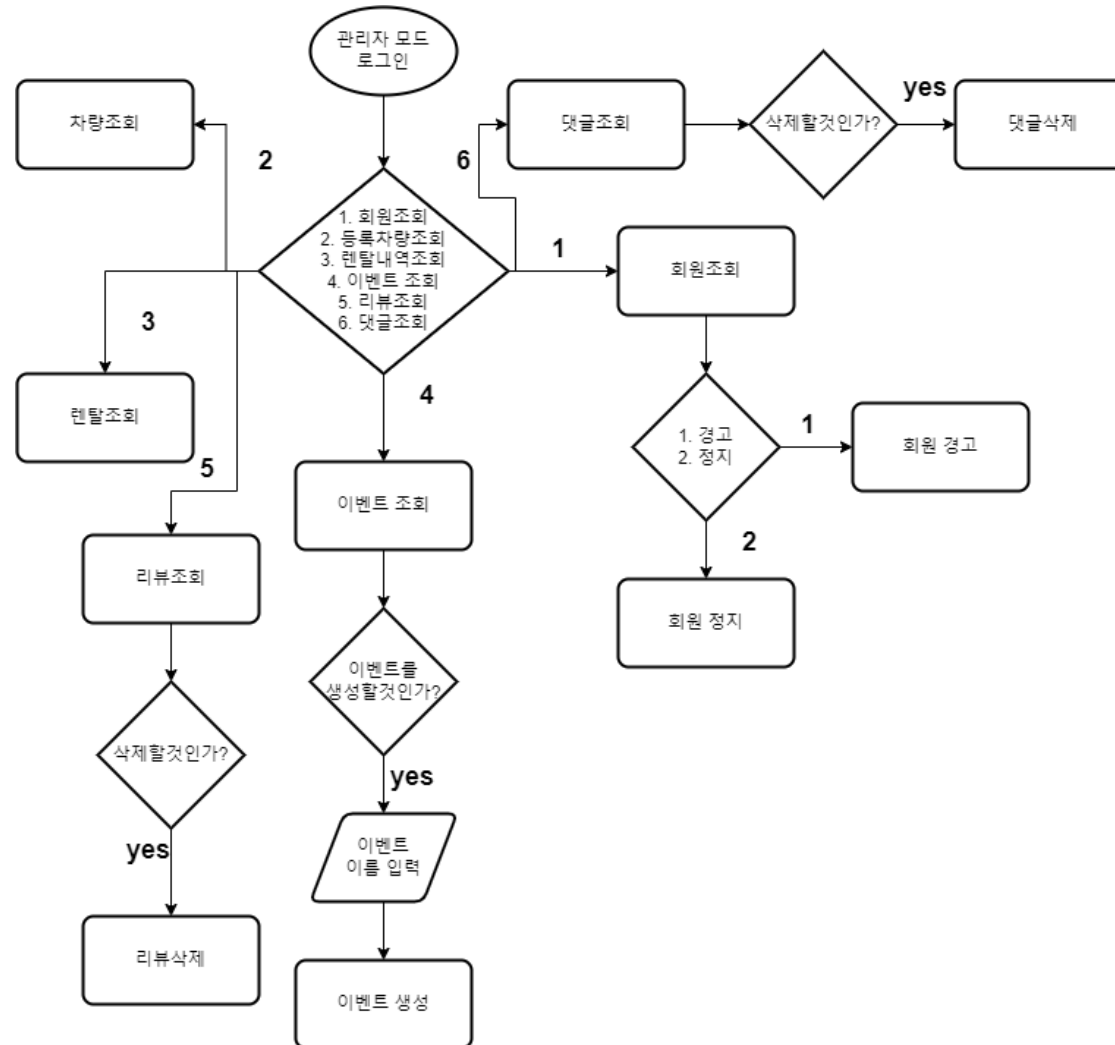
테이블 명	테이블 명		
테이블 명	member		
컬럼	컬럼명	타입	제약조건
아이디	m_id	varchar2(10)	primary key
이름	m_name	varchar2(10)	not null
비밀번호	m_pass	varchar2(20)	not null
성별	m_gender	varchar2(1)	check (m,w)
나이	m_age	number	not null
닉네임	m_nickname	varchar2(10)	unique not null
포인트	m_point	number	default 1000 not null
이메일	m_email	varchar2(20)	not null
폰번호	m_phone	varchar2(15)	not null
면허번호	m_car_license	number	not null
주소	m_addr	varchar2(20)	not null
상세주소	m_near_addr	varchar2(20)	not null
계좌번호	m_account	varchar2(20)	not null
은행	m_bank	varchar2(10)	not null
경고번호	m_w_no	number	default 0 not null
등급번호	m_g_no	number	default 0 not null
테이블 명	warning		
컬럼	컬럼명	타입	제약조건
경고번호	w_no	number	primary key
경고 내용	w_detail	varchar2(8)	check(경고,영구정지)
테이블 명	grade		
컬럼	컬럼명	타입	제약조건
등급번호	g_no	number	primary key
등급상세	g_detail	varchar2(6)	check(common,vip,vvip)
테이블 명	review		
컬럼	컬럼명	타입	제약조건
리뷰번호	rv_no	number	primary key sequence
작성자 아이디	rv_id	varchar2(10)	foreign key
차종	rv_car	varchar2(20)	not null
제목	rv_title	varchar2(20)	not null
내용	rv_memo	varchar2(100)	not null
작성일	rv_date	date	default sysdate
조회수	rv_views	number	default 0 not null

테이블 명	테이블 명		
테이블 명	car		
컬럼	컬럼명	타입	제약조건
등록번호	c_no	number	primary key , sequence
차주아이디	c_m_id	varchar2(10)	foreign key
차량번호	c_num	varchar2(10)	unique not null
연식	c_years	number	not null
차종	c_kind	varchar2(20)	not null
자상태	c_value	varchar2(2)	check(상,중,하)
등록기간	c_term	number	not null
등록일	c_date	date	default sysdate
테이블 명	rental		
컬럼	컬럼명	타입	제약조건
렌탈번호	r_no	number	primary key, sequence
차량등록번호	r_c_no	number	foreign key
차주 아이디	r_owner_id	varchar2(10)	foreign key
사용자 아이디	r_user_id	varchar2(10)	foreign key
렌탈 기간	r_date	number	
렌탈주소	r_addr	varchar2(40)	
테이블 명	event		
컬럼	컬럼명	타입	제약조건
이벤트번호	e_no	number	primary key, sequence
이벤트 이름	e_name	varchar2(20)	not null
쿠폰	e_coupon	varchar2(8)	not null
당첨자 아이디	e_id	varchar2(10)	foreign key
테이블 명	my_comment		
컬럼	컬럼명	타입	제약조건
댓글번호	cm_no	number	primary key,sequence
리뷰번호	cm_rv_no	number	foreign key
작성자 아이디	cm_id	varchar2(10)	foreign key
내용	cm_memo	varchar2(50)	not null
작성일	cm_date	date	default sysdate
추천수	cm_likes	number	default 0

Flow chart : 사용자 모드



Flow chart : 관리자 모드



Class diagram



Index

프로젝트 소개

개발 일정

요구 사항

기능 및 구현

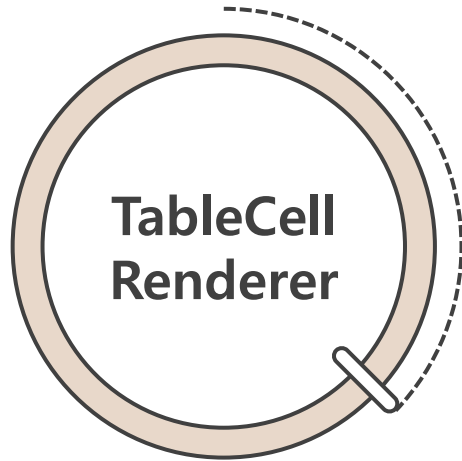
소스 리뷰

후기

기능 및 구현

전체 기능 소개,
주요 기능 소개

주요 기술 소개



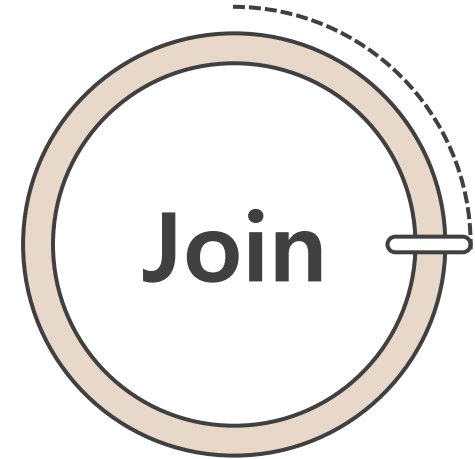
**JTable에
버튼 넣기**

상속과 인터페이스에 대한 고민



DAO 인터페이스 개발

상수, abstract method,
Default method



**Java에서 join
쿼리문사용**

Join DTO, DAO

테이블에 버튼 넣기

JTable은 자료형을 String으로 저장해 출력.

일반적인 방법으로 JTable생성시 이차원배열에 new 연산자를 사용해 JBotton을 넣게된다면 테이블에는 그냥 객체의 주소가 들어가서 객체주소 그대로 테이블이 출력되게됨.

따라서 출력시 어떤 형태로 불러올것인가에 대해 재정의해주어야 함.

방법은 두가지가 있는데 JTable에대한 클래스인 DefaultTableCellRenderer를 상속받아 getTableCellRendererComponent method를 오버라이드.

재정의한 클래스를 사용하던가

현재 GUI클래스에서 TableCellRenderer 인터페이스를 구현하고

getTableCellRendererComponent method를 재정의해서 사용.

두 가지 다 사용해본결과

이 두가지 방법중 오버라이드한 메소드의 재사용성이 떨어진다고 판단되어 인터페이스를 구현한 방법으로 사용추천.

DAO

```
3 import java.sql.Connection;
6
7 public interface DAO {
8
9     String url = "jdbc:oracle:thin:@localhost:1521:orcl";
10    String user = "system";
11    String pass = "1111";
12
13    //드라이버 로드는 필수 오버라이드 되어야 함.
14    public void driverLoad();
15    //드라이버 로드
16    default void load() {
17        try {
18            Class.forName("oracle.jdbc.driver.OracleDriver");
19        } catch (Exception e) {
20            e.printStackTrace();
21        }
22    }
23    /**
24     *
25     * @return 컨넥션한 객체를 리턴해줍니다.
26     */
27    default Connection con() {
28        Connection con = null;
29        try {
30            con = DriverManager.getConnection(url,user,pass);
31        } catch (SQLException e) {
32            e.printStackTrace();
33        }
34        return con;
35    }
36 }
37
```

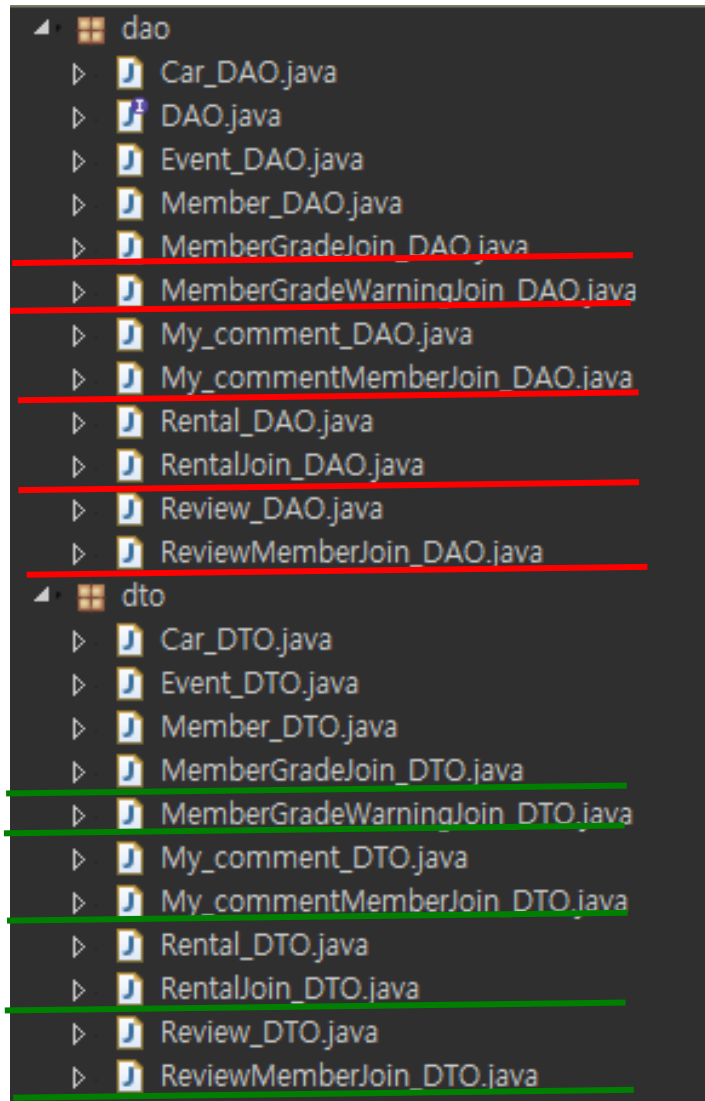
모든 DAO클래스에서 공통적으로 이루어지는 드라이버로드, 컨넥션을 구현받아 사용할수있도록 인터페이스 설계.

상수로는 컨넥션의 인자값들을 선언해주고,

추상메서드를 만들어 드라이버로드를 꼭 하게끔 설계.

디폴트 메서드로 드라이버로드,컨넥션 설계

Join



Join을 사용하는 쿼리를 java에서 사용 할수 있도록
DTO클래스를 만들어 객체에 쿼리의 결과를
Getter,setter로 주고받을 수 있게 하고,
DAO클래스를 만들어 쿼리를 실행 시킬 수 있게
만들.

전체 기능 소개

사용자 모드

이용안내, 차량등록, mypage, 렌탈,
이벤트, 리뷰, 로그아웃



로그인, 회원 가입 화면

회원 가입, 로그인 기능

ShareCar

ID

password

sign in

sign up

관리자 모드

회원, 리뷰, 댓글, 렌탈, 등록차량,
이벤트 관리

관리자모드

회원조회

차량조회

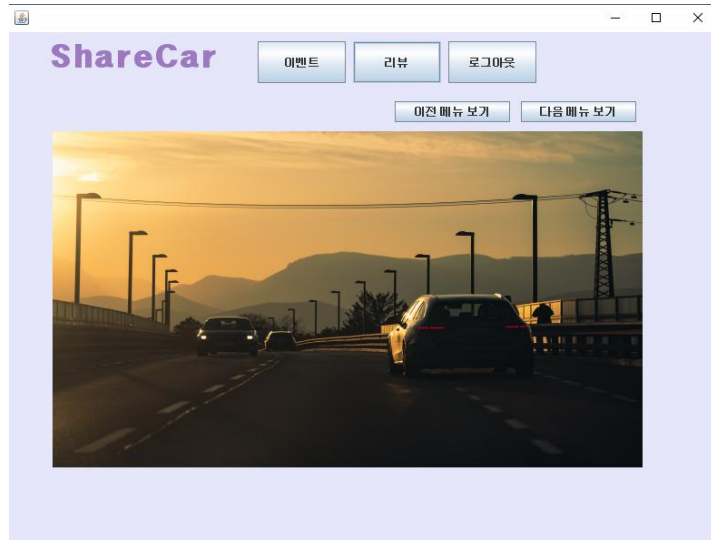
렌탈조회

이벤트조회

리뷰조회

댓글조회

주요 기능 소개 : 리뷰



리뷰					리뷰 작성
리뷰번호	작성일	닉네임	제목	조회수	
1	2023-04-07	hhh	첫 리뷰!	35	
2	2023-04-07	hhh	두번째 리뷰	13	
4	2023-04-10	fff	세번째 리뷰	2	

리뷰작성

제목

차종

내용

등록

리뷰 상세보기

작성자 hhh 2023-04-07 12:22 조회수 37

첫 리뷰!

첫 리뷰입니다.

댓글

댓글입력

닉네임	작성일	댓글	추천수	추천하기	댓글번호
fff	2023-04-11	sdads	1	추천	8

리뷰 상세보기

작성자 hhh 2023-04-07 12:22 조회수 38

첫 리뷰!

첫 리뷰입니다.

댓글

댓글입력

닉네임	작성일	댓글	추천수	추천하기	댓글번호
fff	2023-04-11	sdads	1	추천	8
fff	2023-04-11	adsadasd	0	추천	9

리뷰 상세보기

작성자 hhh 2023-04-07 12:22 조회수 38

첫 리뷰!

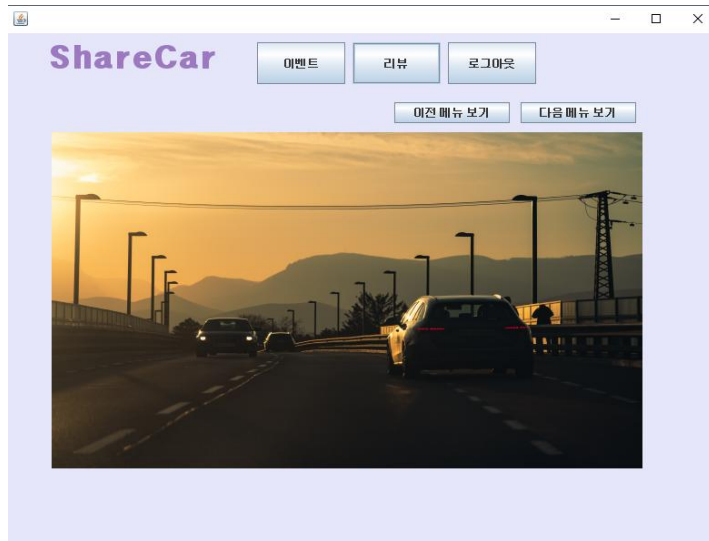
첫 리뷰입니다.

댓글

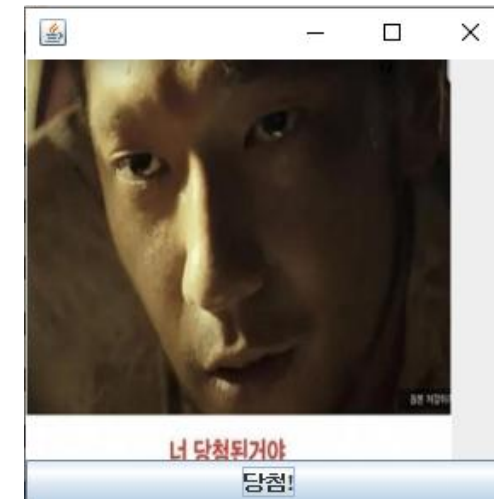
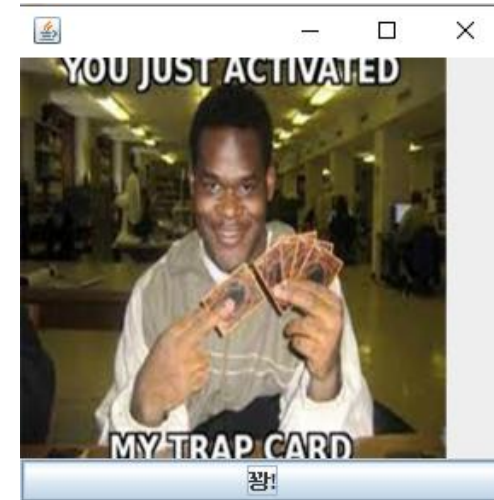
댓글입력

닉네임	작성일	댓글	추천수	추천하기	댓글번호
fff	2023-04-11	sdads	1	추천	8
fff	2023-04-11	adsadasd	1	추천	9

주요 기능 소개 : 이벤트



이벤트		
이벤트번호	이벤트 이름	참가
7	test event	참가



Index

프로젝트 소개

개발 일정

요구 사항

기능 및 구현

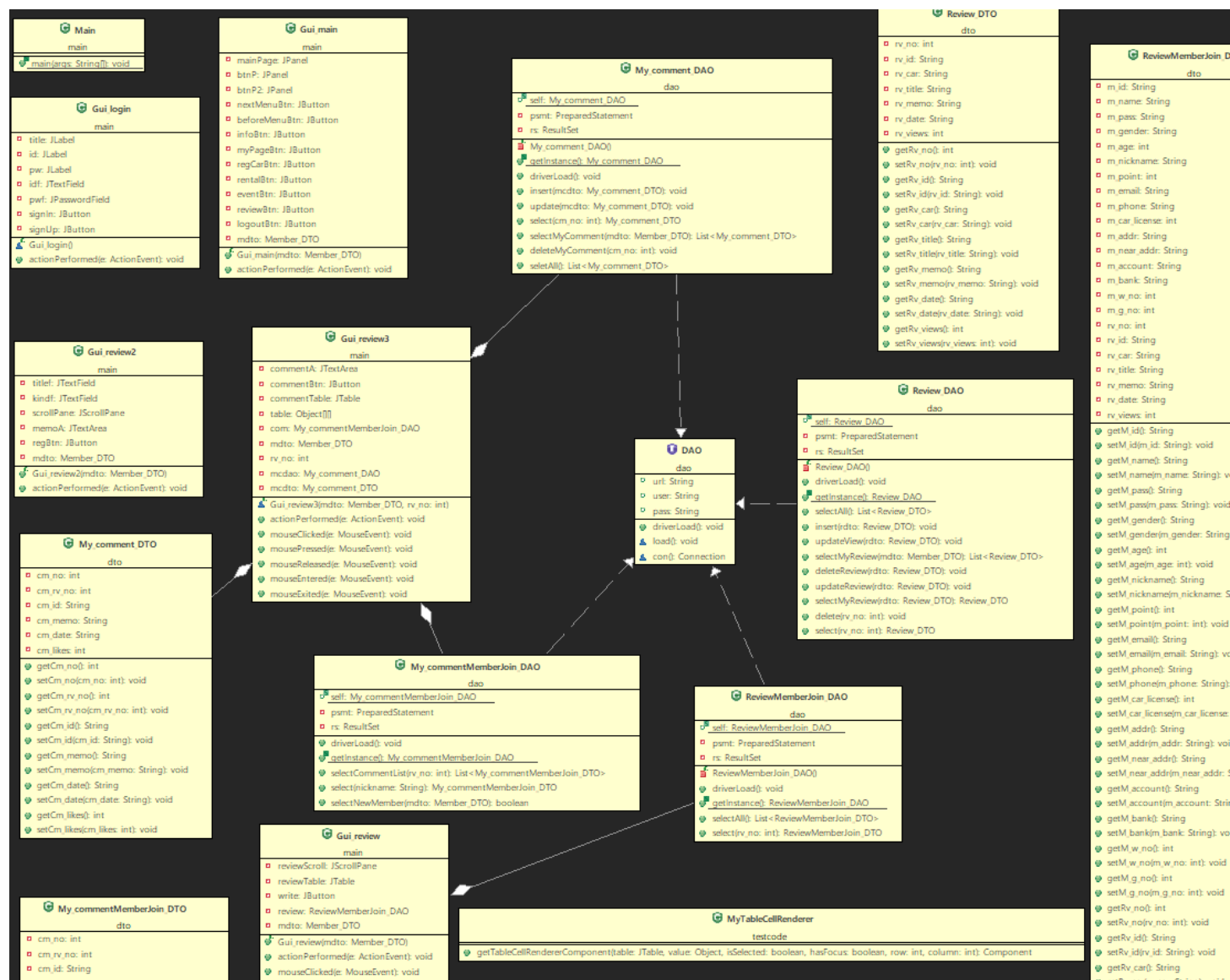
소스 리뷰

후기

소스 리뷰

리뷰 부분 소스,
이벤트 부분 소스

리뷰 class diagram



리뷰 소스

리뷰

```
public class Gui_review extends JFrame implements ActionListener,MouseListener{
```

```
private ReviewMemberJoin_DAO review = ReviewMemberJoin_DAO.getInstance();
```

```
private Member_DTO mdto = null;
```

```
public Gui_review(Member_DTO mdto) {  
    this.mdto = mdto;
```

리뷰 상세보기

```
Gui_review3(Member_DTO mdto,int rv_no){  
    this.mdto = mdto;  
    this.rv_no = rv_no;
```

```
ReviewMemberJoin_DAO rmj = ReviewMemberJoin_DAO.getInstance();  
ReviewMemberJoin_DTO rmjdto = rmj.select(rv_no);
```

```
DefaultTableCellRenderer myRenderer = new MyTableCellRenderer();
```

```
commentTable.getColumnModel("추천하기").setCellRenderer(myRenderer);
```

```
68 • @Override  
69 public void actionPerformed(ActionEvent e) {  
70     // TODO Auto-generated method stub  
71     if(e.getSource().equals(write)) {  
72         //리뷰 작성 버튼  
73         new Gui_review2(mdto);  
74         this.setVisible(false);  
75     }  
76 }  
77  
78 • @Override  
79 public void mouseClicked(MouseEvent e) {  
80     Review_DAO review = Review_DAO.getInstance();  
81     int rv_no = -1;  
82     int rv_views = 0;  
83     int row = reviewTable.getSelectedRow();  
84     // System.out.println(100);  
85     Review_DTO r = new Review_DTO();  
86     rv_no = (int) reviewTable.getModel().getValueAt(row, 0);  
87     rv_views = (int) reviewTable.getModel().getValueAt(row, 4);  
88     r.setRv_no(rv_no);  
89     r.setRv_views(rv_views);  
90     //리뷰 조회수 업데이트  
91     review.updateView(r);  
92     // this.setVisible(true);  
93     this.setVisible(false);  
94     new Gui_review(mdto);  
95     //리뷰상세보기 창 띄움  
96     new Gui_review3(mdto,rv_no);  
97     // this.setVisible(true);  
98     // reviewTable.setVisible(true);
```

```
public class MyTableCellRenderer extends DefaultTableCellRenderer{  
  
    @Override  
    public Component getTableCellRendererComponent(JTable table, Object value, boolean isSelected, boolean hasFocus, int row, int column) {  
        Component likesBtn = null;  
        if(column==4) {  
            likesBtn = new JButton("추천");  
        }  
  
        return likesBtn;  
    }  
}
```


Event class diagram



이벤트 소스

```
public class Gui_event extends JFrame implements TableCellRenderer,MouseListener
```

```
Gui_event(Member_DTO mdto){  
    this.mdto = mdto;
```

```
eventTable.getColumnModel("참가").setCellRenderer(this);
```

```
@Override  
public Component getTableCellRendererComponent(JTable table, Object value,  
    Component attendBtn = null;  
    if(column==2) {  
        attendBtn = new JButton("참가");  
    }  
    return attendBtn;  
}
```

```
@Override  
public void mouseClicked(MouseEvent e) {  
    // TODO Auto-generated method stub  
    RandomString r = RandomString.getInstance();  
    int col = eventTable.getSelectedColumn();  
    int row = eventTable.getSelectedRow();  
    if(col==2) {  
        //참가버튼을 누른시점  
        int e_no = (int)eventTable.getModel().getValueAt(row, 0);  
        //회원의 포인트 확인  
        if(mdto.getM_point()>=1000) {  
            //1000point 차감  
            mdto = member.usePoint(mdto);  
            //회원의 등급 업데이트  
            member.updateGrade(mdto);  
            //난수생성  
            String myCoupon = r.coupon();  
            //난수와 쿠폰번호가 일치하는지 확인  
            if(myCoupon.equals(event.couponChk(e_no))) {  
                //일치할경우 당첨처리후 event 테이블 update  
                //인자값으로 회원아이디와 이벤트 번호를 받아야한다.  
                event.win(mdto, e_no);  
                new Gui_event2(true);  
            }else {  
                //불일치할경우 광  
                new Gui_event2(false);  
            }  
            dispose();  
        }else {  
            hiddenL.setText("포인트가 부족합니다.");  
            this.setVisible(true);  
        }  
    }  
}
```

Index

프로젝트 소개

개발 일정

요구 사항

기능 및 구현

소스 리뷰

후기

후기

후기, Q&A

개발 후기



추가하고 싶은 기능:

1. 전자서명으로 결제를 하면
인공지능이 기존에 입력 받은
회원의 필체와 대조를 하고
결제하는 방식의
기능을 추가하고 싶음.
2. 이용안내부분에서
현재는 이미지 파일을 버튼을
누르면 다음 이미지로
넘어가는 방식인데 이미지가
아닌 동영상을 넣고 싶음.

Q&A

모든 소스를 보고 싶다면 하단의 링크를 참조하세요.

https://github.com/0bliviat3/Java_Project_ShareCar.git

A graphic of a spiral-bound notebook with a dark green cover and a white page. The spiral binding is on the left side, with yellow and black rings. The text is written in a large, orange, cursive font on the white page.

*Thank you
For listening*