Course completion project procedure
Instruction manual

Course name： Responsive website development technology

Project name： Tourism blog website

Student number： **EB515085**

Class： **CST 2018**

Name： **APEDO KOMI ETSE YVON**

Submission time：

1. Brief description of the project

Tourism blog website is a blog website where users can upload their articles. Users can also modify the contents of their articles. Furthermore, Users can read articles published by various users and comment on them. Moreover, user can modify their personal information.

2. Project design description

2.1 Project technology overview

This web app is mainly based on Multitier Architecture of Java Server Pages (JSP) technology. Java is the main programming language used to implement the web application and the database part of the web will be created with MySQL.

Development tools:

- **IntelliJ IDEA 2021**
- **Tomcat 8.0.42**
- **Navicat**
- **MySql**

2.2 Database design structure

Database management system： MySql Server 8.0

Database name: blogwebdb

user name： root

password：1234

- **t_article table**

Name	Type	Length	Decimals	Allow Null	
articleid	int	0	0	☐	1
title	varchar	200	0	☐	
address	varchar	200	0	☑	
publishtime	timestamp	0	0	☐	
userid	varchar	300	0	☐	

- **t_comment table**

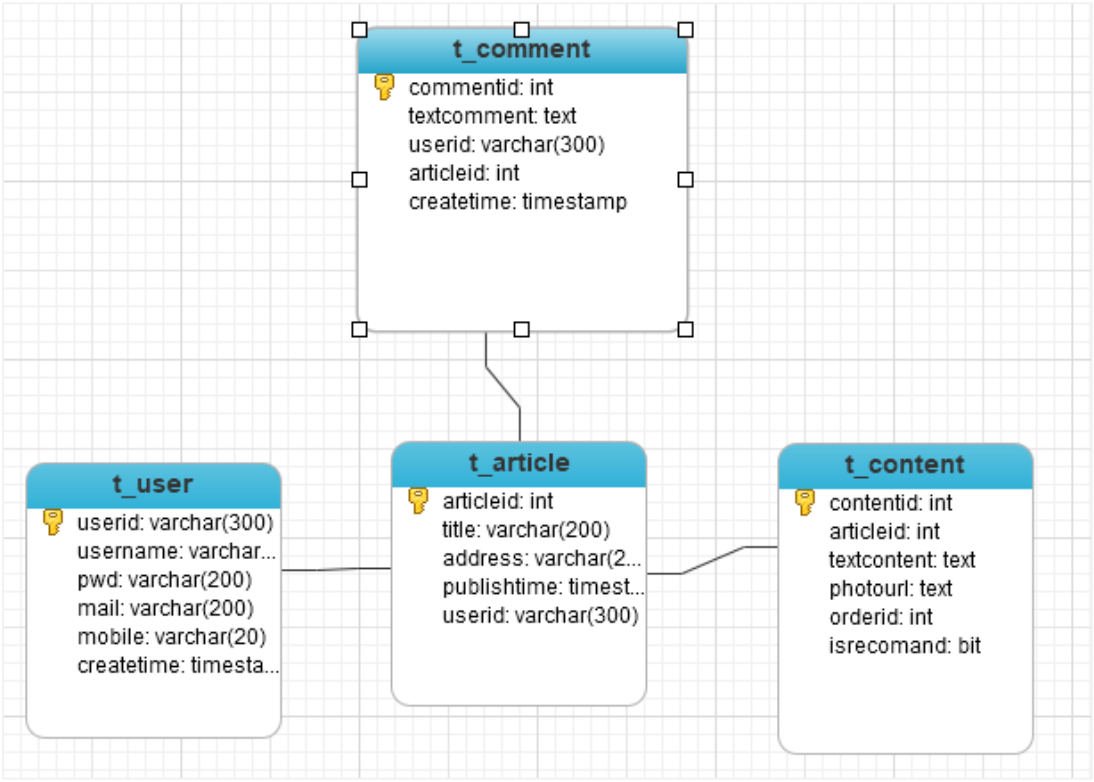
Name	Type	Length	Decimals	Allow Null	
commentid	int	0	0	☐	1
textcomment	text	0	0	☑	
userid	varchar	300	0	☑	
articleid	int	0	0	☑	
createtime	timestamp	0	0	☑	

- **t_user table**

Name	Type	Length	Decimals	Allow Null	
userid	varchar	300	0	☐	1
username	varchar	200	0	☑	
pwd	varchar	200	0	☑	
mail	varchar	200	0	☑	
mobile	varchar	20	0	☑	
createtime	timestamp	0	0	☑	

- **t_content table**

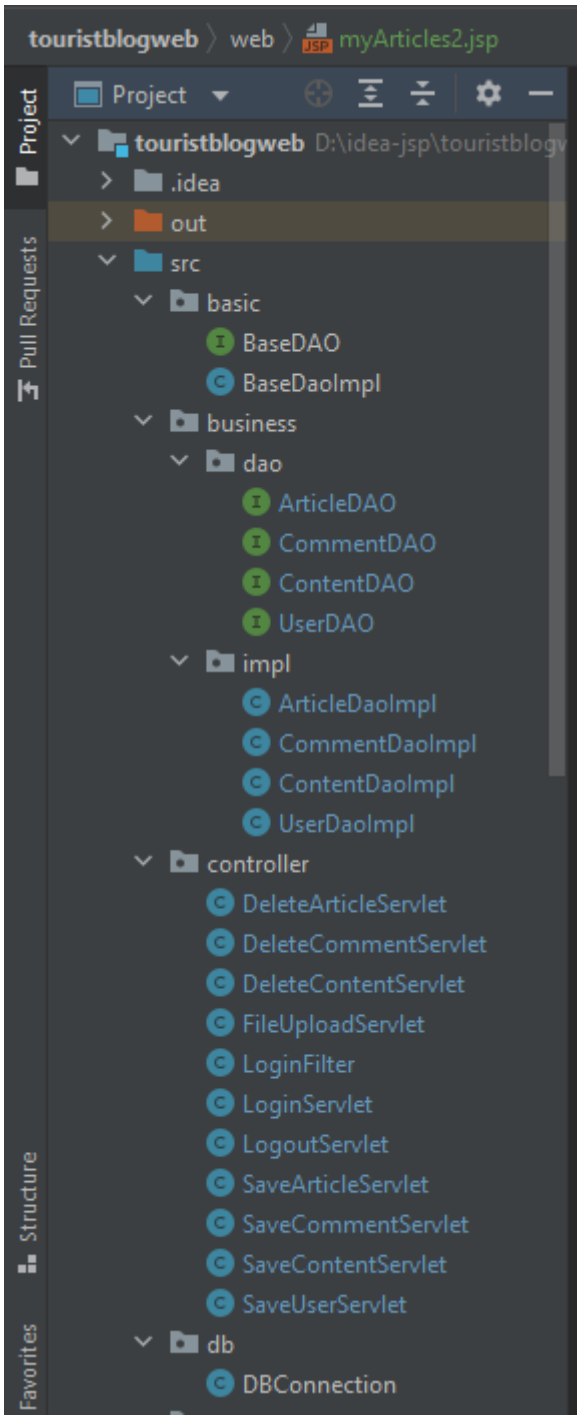
Name	Type	Length	Decimals	Allow Null	
contentid	int	0	0	☐	1
articleid	int	0	0	☑	
textcontent	text	0	0	☑	
photourl	text	0	0	☑	
orderid	int	0	0	☑	
isrecomand	bit	1	0	☑	

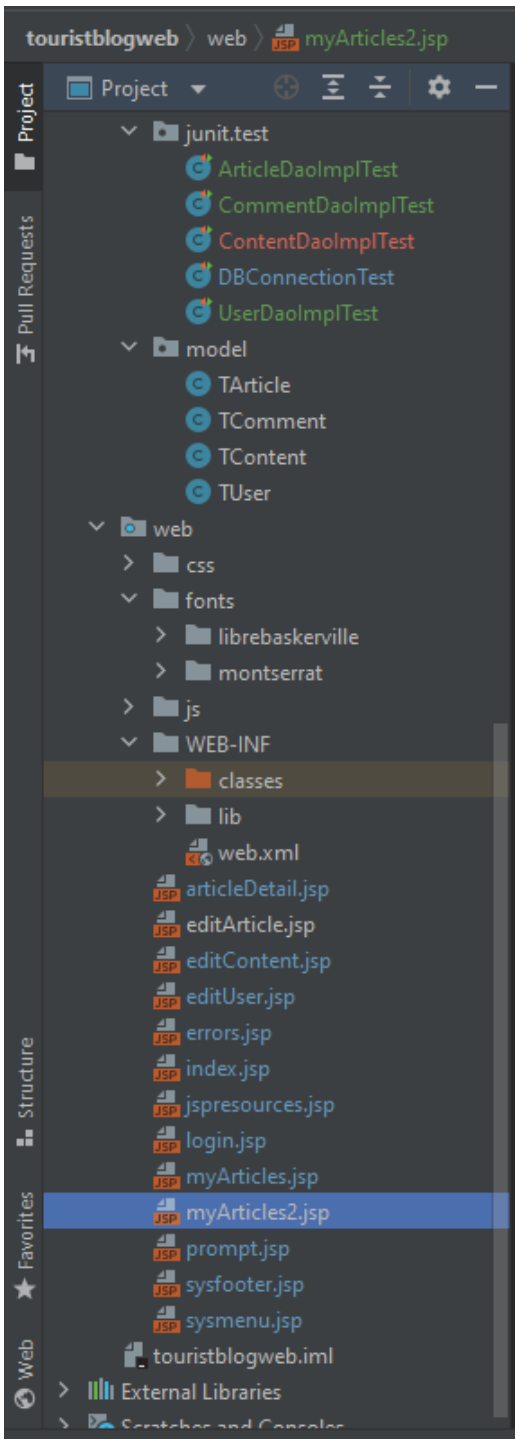

t_comment
commentid: int
textcomment: text
userid: varchar(300)
articleid: int
createtime: timestamp
t_user
userid: varchar(300)
username: varchar...
pwd: varchar(200)
mail: varchar(200)
mobile: varchar(20)
createtime: timesta...
t_article
articleid: int
title: varchar(200)
address: varchar(2...
publishtime: timest...
userid: varchar(300)
t_content
contentid: int
articleid: int
textcontent: text
photourl: text
orderid: int
isrecomand: bit

2.3 Project structure description

Project name： **touristblogweb**

The project structure is as follows:





2.4 Detailed description of project design

- **MODEL LAYER**

(1) **TArticle** Entity class

remark: The `TArticle` entity class is used to exchange record data of `t_article` table

```
package model;

import business.dao.ArticleDAO;
import business.impl.ArticleDaoImpl;
import business.impl.ContentDaoImpl;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;

public class TArticle {
    private int articleid;
    private String title;
    private String address;
    private String publishtime;
    private String userid;
private TContent firstContent;
    public TArticle() {
    }
    public TContent getFirstContent() {
        ArticleDAO dao= new ArticleDaoImpl();
        firstContent = dao.getFirstContent(this.articleid);
        return firstContent;
    }
    public int getArticleid() {
        return articleid;
    }

    public void setArticleid(int articleid) {
        this.articleid = articleid;
    }

    public String getTitle() {
        return title;
```

```
}

public void setTitle(String title) {
    this.title = title;
}

public String getAddress() {
    return address;
}

public void setAddress(String address) {
    this.address = address;
}

public String getPublishtime() {
    return publishtime;
}

public void setPublishtime(String publishtime) {
    this.publishtime = publishtime;
}

public String getUserid() {
    return userid;
}

public void setUserid(String userid) {
    this.userid = userid;
}

//exchange single user data from table to TArticle bean
public TArticle(ResultSet rs) {
    try {
       if (rs != null && rs.next()) {
          this.setArticleid(rs.getInt("articleid"));
          this.setTitle(rs.getString("title"));
          this.setAddress(rs.getString("address"));
          this.setPublishtime(rs.getString("publishtime"));
          this.setUserid(rs.getString("userid"));
       }
    } catch (SQLException e) {
       e.printStackTrace();
```

```
        }
    }

    //exchange multiple user data from table to TArticle bean
    public static ArrayList<TArticle> toList(ResultSet rs) {
        ArrayList<TArticle> list = new ArrayList<TArticle>();
        try {
            while (rs != null && rs.next()) {
                TArticle article = new TArticle();
                article.setArticleid(rs.getInt("articleid"));
                article.setTitle(rs.getString("title"));
                article.setAddress(rs.getString("address"));
                article.setPublishtime(rs.getString("publishtime"));
                article.setUserid(rs.getString("userid"));
                list.add(article);
            }

        } catch (SQLException throwables) {
            throwables.printStackTrace();
        }
        return list;
    }
    }
```

(2) **TComment** Entity class

remark: The `TComment` entity class is used to exchange record data of t_`comment` table

```
package model;

import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;

public class TComment {
    private int commentid;
    private String textcomment;
    private String userid;
    private int articleid;
    private String createtime;
```

```
public TComment() {
}

public int getCommentid() {
    return commentid;
}

public void setCommentid(int commentid) {
    this.commentid = commentid;
}

public String getTextcomment() {
    return textcomment;
}

public void setTextcomment(String textcomment) {
    this.textcomment = textcomment;
}

public String getUserid() {
    return userid;
}

public void setUserid(String userid) {
    this.userid = userid;
}

public int getArticleid() {
    return articleid;
}

public void setArticleid(int articleid) {
    this.articleid = articleid;
}

public String getCreatetime() {
    return createtime;
}

public void setCreatetime(String createtime) {
    this.createtime = createtime;
}
```

```
//exchange single user data from table to TComment bean
public TComment(ResultSet rs) {
    try {
        if (rs != null && rs.next()) {
            this.setArticleid(rs.getInt("articleid"));
            this.setCommentid(rs.getInt("commentid"));
            this.setCreatetime(rs.getString("createtime"));
            this.setTextcomment(rs.getString("textcomment"));
            this.setUserid(rs.getString("userid"));
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
}

//exchange multiple user data from table to TComment bean
public static ArrayList<TComment> toList(ResultSet rs) {
    ArrayList<TComment> list = new ArrayList<TComment>();
    try {
        while (rs != null && rs.next()) {
            TComment tcomment = new TComment();
            tcomment.setArticleid(rs.getInt("articleid"));
            tcomment.setCommentid(rs.getInt("commentid"));
            tcomment.setCreatetime(rs.getString("createtime"));
            tcomment.setTextcomment(rs.getString("textcomment"));
            tcomment.setUserid(rs.getString("userid"));
            list.add(tcomment);
        }

    } catch (SQLException throwables) {
        throwables.printStackTrace();
    }
    return list;
}
}
```

(3) **TContent** Entity class

remark: The `TContent` entity class is used to exchange record data of t_`content` table

```
package model;

import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;

public class TContent {
    private int contentid;
    private int  articleid;
    private String textcontent;
    private String photourl;
    private int orderid;

    public TContent() {
    }

    public int getContentid() {
        return contentid;
    }

    public void setContentid(int contentid) {
        this.contentid = contentid;
    }

    public int getArticleid() {
        return articleid;
    }

    public void setArticleid(int articleid) {
        this.articleid = articleid;
    }

    public String getTextcontent() {
        return textcontent;
    }

    public void setTextcontent(String textcontent) {
        this.textcontent = textcontent;
    }

    public String getPhotourl() {
        return photourl;
```

```
}

public void setPhotourl(String photourl) {
    this.photourl = photourl;
}

public int getOrderid() {
    return orderid;
}

public void setOrderid(int orderid) {
    this.orderid = orderid;
}

//exchange single user data from table to TContent bean
public TContent(ResultSet rs) {
    try {
        if (rs != null && rs.next()) {
            this.setContentid(rs.getInt("contentid"));
            this.setArticleid(rs.getInt("articleid"));
            this.setTextcontent(rs.getString("textcontent"));
            this.setPhotourl(rs.getString("photourl"));
            this.setOrderid(rs.getInt("orderid"));
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
}

//exchange multiple user data from table to TContent bean
public static ArrayList<TContent> toList(ResultSet rs) {
    ArrayList<TContent> list = new ArrayList<TContent>();
    try {
        while (rs != null && rs.next()) {
            TContent content = new TContent();
            content.setContentid(rs.getInt("contentid"));
            content.setArticleid(rs.getInt("articleid"));
            content.setTextcontent(rs.getString("textcontent"));
            content.setPhotourl(rs.getString("photourl"));
            content.setOrderid(rs.getInt("orderid"));
            list.add(content);
        }
```

```
        } catch (SQLException throwables) {
            throwables.printStackTrace();
        }
        return list;
    }
    }
```

(4) **TUser** Entity class

remark: The `TUser` entity class is used to exchange record data of t_`User` table

```
package model;

import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;

public class TUser {
    private String userid;
    private String username;
    private String pwd;
    private String mobile;
    private String mail;
    private String createtime;

    public TUser() {
    }

    public String getUserid() {
        return userid;
    }

    public void setUserid(String userid) {
        this.userid = userid;
    }

    public String getUsername() {
```

```
        return username;
    }

    public void setUsername(String username) {
        this.username = username;
    }

    public String getPwd() {
        return pwd;
    }

    public void setPwd(String pwd) {
        this.pwd = pwd;
    }


    public String getMobile() {
        return mobile;
    }

    public void setMobile(String mobile) {
        this.mobile = mobile;
    }

    public String getMail() {
        return mail;
    }

    public void setMail(String mail) {
        this.mail = mail;
    }


    public String getCreatetime() {
        return createtime;
    }

    public void setCreatetime(String createtime) {
        this.createtime = createtime;
    }

    //exchange single user data from table to TUser bean
```

```
public TUser(ResultSet rs) {
    try {
        if (rs != null && rs.next()) {
            this.setUserid(rs.getString("userid"));
            this.setUsername(rs.getString("username"));
            this.setPwd(rs.getString("pwd"));
            this.setMobile(rs.getString("mobile"));
            this.setMail(rs.getString("mail"));
            this.setCreatetime(rs.getString("createtime"));
        }
    } catch (SQLException e) {
        e.printStackTrace();
    }
}

//exchange multiple user data from table to TUser bean
public static ArrayList<TUser> toList(ResultSet rs) {
    ArrayList<TUser> list = new ArrayList<TUser>();
    try {
        while (rs != null && rs.next()) {
            TUser user = new TUser();
            user.setUserid(rs.getString("userid"));
            user.setUsername(rs.getString("username"));
            user.setPwd(rs.getString("pwd"));
            user.setMobile(rs.getString("mobile"));
            user.setMail(rs.getString("mail"));
            user.setCreatetime(rs.getString("createtime"));
            list.add(user);
        }

    } catch (SQLException throwables) {
        throwables.printStackTrace();
    }
    return list;
}
}
```