武汉纺织大学

Web应用开发课程设计

# 新闻展示

**学 院： 数学与计算机学院**

**班 级： 物联网11802班**

**姓 名： 胡涵宇**

**学 号： 1804241015**

**指导老师： 聂刚**

**成 绩：**

**完成日期： 2020年12月15日**

**目录**

1. **需求分析·······························································1**
2. **报告内容·······························································2**
3. **系统设计······························································3**

**ER图**

**UML类图**

1. **系统实现······························································3**

**项目结构**

**配置文件**

**实体类**

**接口类**

**服务层**

**前段页面**

1. **系统测试······························································3**
2. **系统总结······························································4**
3. **需求分析**

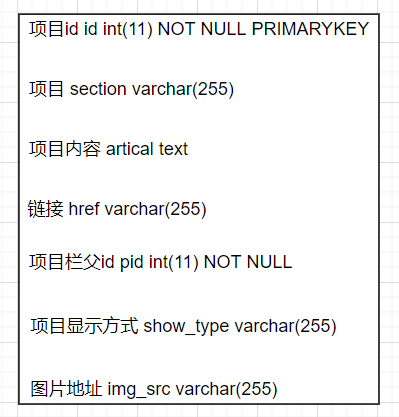


按上图设计实现新闻显示页面，页面所需图片，文字均来自数据库。

1. **系统设计**

**ER图**

显示页面的各项内容存储表，对应的ER图如图所示。



项目id id **int**(11) NOT NULL PRIMARYKEY

项目 section varchar(255)

项目内容 artical text

链接 href varchar(255)

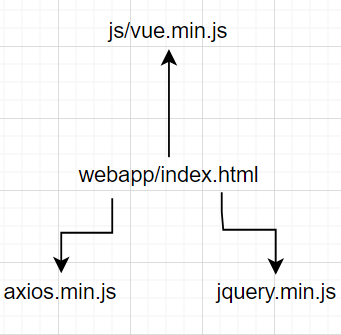
项目栏父id pid **int**(11) NOT NULL

项目显示方式 show\_type varchar(255)

图片地址 img\_src varchar(255)

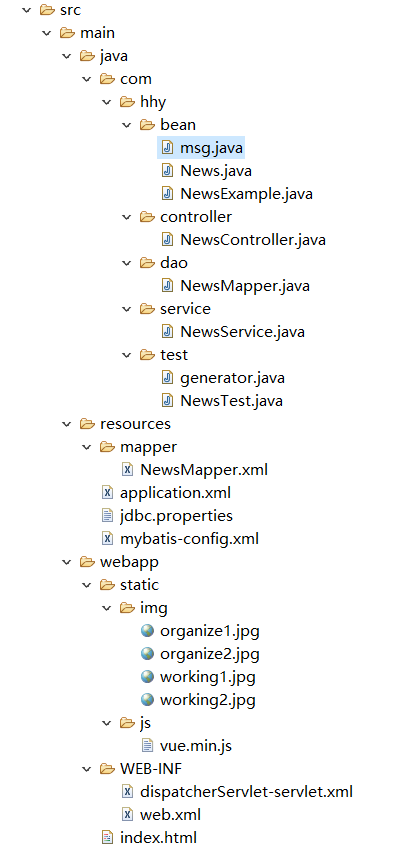
**UML类图**

各类的结构及类之间的关系如图所示：



1. **系统设计**

**项目结构**



**配置文件**

**jdbc.properties文件**

该配置文件主要是以文件形式保存数据库的驱动类名称，连接数据库的URL地址，访问数据库的用户名及对应的密码，程序运行时会读取该文件相关信息，避免硬编码，当相关信息发生变化时，只需修改配置文件而不用修改源代码，增加程序的可扩展型。

jdbc.username=root

jdbc.password=123456

jdbc.url=jdbc:mysql://47.110.70.107:8081/news?serverTimezone=UTC&useUnicode=true&characterEncoding=utf8

jdbc.className=com.mysql.jdbc.Driver

**mybatis-config.xml文件**

<?xml version="1.0" encoding="UTF-8" ?>

<!DOCTYPE configuration

        PUBLIC "-//mybatis.org//DTD Config 3.0//EN"

        "http://mybatis.org/dtd/mybatis-3-config.dtd">

<configuration>

    <settings>

        <setting name="mapUnderscoreToCamelCase" value="true"/>

        <setting name="logImpl" value="STDOUT\_LOGGING"/>

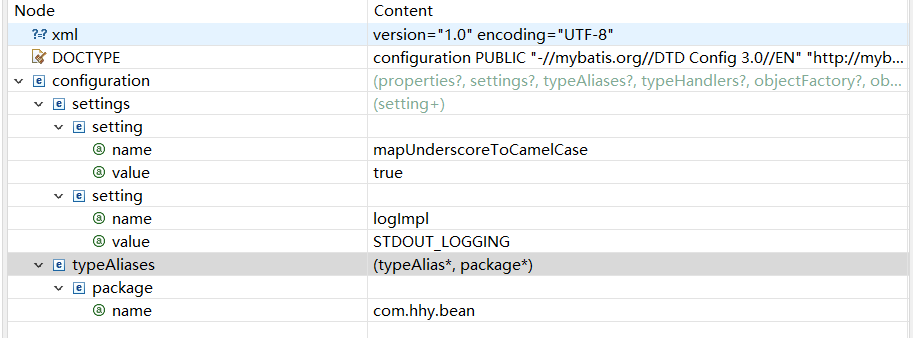
    </settings>

    <typeAliases>

        <package name="com.hhy.bean"/>

    </typeAliases>

</configuration>



**实体类NEWS**

**package** com.hhy.bean;

**import** java.io.Serializable;

**public** **class** News **implements** Serializable {

**private** Integer id; //可供查询的id

**private** String section; //显示的栏目

**private** String href; //各个栏目的连接

**private** Integer pid; //各栏目的显示页面父id

**private** String showType; //各栏目的显示方式

**private** String summary; ;//显示页面中的概括内容

**private** String imgSrc; //页面中的图片资源

**private** String artical; //相关的内容存储

**private** **static** **final** **long** serialVersionUID = 1L;

**public** Integer getId() {

**return** id;

}

**public** **void** setId(Integer id) {

**this**.id = id;

}

**public** String getSection() {

**return** section;

}

**public** **void** setSection(String section) {

**this**.section = section == **null** ? **null** : section.trim();

}

**public** String getHref() {

**return** href;

}

**public** **void** setHref(String href) {

**this**.href = href == **null** ? **null** : href.trim();

}

**public** Integer getPid() {

**return** pid;

}

**public** **void** setPid(Integer pid) {

**this**.pid = pid;

}

**public** String getShowType() {

**return** showType;

}

**public** **void** setShowType(String showType) {

**this**.showType = showType == **null** ? **null** : showType.trim();

}

**public** String getSummary() {

**return** summary;

}

**public** **void** setSummary(String summary) {

**this**.summary = summary == **null** ? **null** : summary.trim();

}

**public** String getImgSrc() {

**return** imgSrc;

}

**public** **void** setImgSrc(String imgSrc) {

**this**.imgSrc = imgSrc == **null** ? **null** : imgSrc.trim();

}

**public** String getArtical() {

**return** artical;

}

**public** **void** setArtical(String artical) {

**this**.artical = artical == **null** ? **null** : artical.trim();

}

}

**接口类NewsMapper**

**package** com.hhy.dao;

**import** com.hhy.bean.News;

**import** com.hhy.bean.NewsExample;

**import** java.util.List;

**import** org.apache.ibatis.annotations.Param;

**public** **interface** NewsMapper {

**long** countByExample(NewsExample example);

**int** deleteByExample(NewsExample example);

**int** deleteByPrimaryKey(Integer id);

**int** insert(News record);

**int** insertSelective(News record);

List<News> selectByExampleWithBLOBs(NewsExample example);

List<News> selectByExample(NewsExample example);

News selectByPrimaryKey(Integer id);

**int** updateByExampleSelective(@Param("record") News record, @Param("example") NewsExample example);

**int** updateByExampleWithBLOBs(@Param("record") News record, @Param("example") NewsExample example);

**int** updateByExample(@Param("record") News record, @Param("example") NewsExample example);

**int** updateByPrimaryKeySelective(News record);

**int** updateByPrimaryKeyWithBLOBs(News record);

**int** updateByPrimaryKey(News record);}

**服务层NewsService**

**package** com.hhy.service;

**import** com.hhy.bean.News;

**import** com.hhy.bean.NewsExample;

**import** com.hhy.dao.NewsMapper;

**import** org.springframework.beans.factory.annotation.Autowired;

**import** org.springframework.stereotype.Service;

**import** java.util.HashMap;

**import** java.util.List;

**import** java.util.Map;

@Service

**public** **class** NewsService {

@Autowired

NewsMapper newsMapper;

**public** Map<Integer,Object> getColumns() {

NewsExample newsExample = **new** NewsExample();

NewsExample.Criteria criteria = newsExample.createCriteria();

criteria.andPidEqualTo(0);

criteria.andShowTypeEqualTo("0");

List<News> type0List = newsMapper.selectByExample(newsExample);

NewsExample newsExample2 = **new** NewsExample();

NewsExample.Criteria criteria2 = newsExample2.createCriteria();

criteria2.andPidEqualTo(0);

criteria2.andShowTypeEqualTo("-1");

List<News> typeSpecialList = newsMapper.selectByExample(newsExample2);

Map<Integer,Object> map = **new** HashMap<>();

map.put(0,type0List);

map.put(-1,typeSpecialList);

**return** map;

}

**public** List<News> getAll() {

NewsExample newsExample = **new** NewsExample();

**return** newsMapper.selectByExample(newsExample);

}

**public** Map<Integer, Object> getSectionContent(Integer id){

NewsExample newsExample = **new** NewsExample();

NewsExample.Criteria criteria = newsExample.createCriteria();

criteria.andPidEqualTo(id);

criteria.andShowTypeEqualTo("1");

List<News> type1List = newsMapper.selectByExample(newsExample);

NewsExample newsExample2 = **new** NewsExample();

NewsExample.Criteria criteria2 = newsExample2.createCriteria();

criteria2.andPidEqualTo(id);

criteria2.andShowTypeEqualTo("2");

List<News> type2List = newsMapper.selectByExample(newsExample2);

Map<Integer,Object> map = **new** HashMap<>();

map.put(1,type1List);

map.put(2,type2List);

**return** map;

}

}

**前端页面**

**<!DOCTYPE html>**

**<html lang="zh-CN">**

**<head>**

**<meta charset="GB2312">**

**<script src="https://cdn.staticfile.org/jquery/1.10.2/jquery.min.js"></script>**

**<title>Title</title>**

**</head>**

**<style>**

**hr {**

**height: 2px;**

**border: none;**

**border-top: 2px solid darkred;**

**}**

**.topBtn {**

**background: #ffffff;**

**color: grey;**

**text-align: center;**

**width: 110px;**

**height: 30px;**

**margin-top: -9px;**

**border: none;**

**font-weight: bold;**

**font-size: 19px;**

**}**

**form {**

**border: solid 1px grey;**

**}**

**a{**

**text-decoration: none;**

**color: black;**

**}**

**</style>**

**<body>**

**<div id="app">**

**<hr/>**

**<div>**

**<button class="topBtn" v-for="item in columnList[0]" :id="item.id" @click="changeCss($event)">{{item.section}}</button>**

**</div>**

**<div>**

**<a v-for="item in columnList[-1]" :href="item.href" style="color: darkred;position: absolute;right: 15px;top: 10px;">+<span style="color: grey">{{item.section}}</span></a>**

**</div>**

**<form action="" >**

**<div style="display: inline-block;" v-for="item in sectionContentList[1]">**

**<img :src="item.imgSrc" alt="" style="height: 180px;width: 360px;"/>**

**<span style="font-weight:bold;float: right;width: 280px;">{{item.section}}</span>**

**<br/>**

**<span style="float: right;width: 280px;margin-top: -150px;padding-top: 5px;">{{item.summary}}<br/>**

**<a style="color:indianred;float: right;width: 280px;" :href="item.href">[详细]</a>**

**</span>**

**<br/>**

**</div>**

**<div >**

**<a v-for="item in sectionContentList[2]" :href="item.href" style="display: inline-block;width: 650px;padding-top: 10px;">o&nbsp;{{item.section}}</a>**

**</div>**

**</form>**

**<!-- <span v-for="item in columnList[0]">{{item.section}}</span>-->**

**<!-- <span v-for="item in sectionContentList[1]">{{item.section}}</span>-->**

**</div>**

**</body>**

**<script src="static/js/vue.min.js"></script>**

**<script src="https://unpkg.com/axios/dist/axios.min.js"></script>**

**<script>**

**var app = new Vue({**

**el: "#app",**

**data: {**

**columnList: [],**

**sectionContentList: [],**

**id: 1**

**},**

**methods: {**

**getColumns() {**

**var that = this;**

**axios.get("getColumns.do").then(function (response) {**

**console.log(response.data);**

**that.columnList = response.data.extend.columnList;**

**}**

**)**

**},**

**getContent() {**

**var that = this;**

**axios.get("getSection.do?id=" + that.id).then(function (response) {**

**that.sectionContentList = response.data.extend.sectionContentList;**

**})**

**},**

**changeCss(e){**

**e= e.target;**

**var that =this;**

**that.id=e.id;**

**$(".topBtn").each(function (index,element) {**

**if (that.id==element.id){**

**$(element).css("background","darkred");**

**$(element).css("color","#ffffff");**

**}**

**else{**

**$(element).css("background","#ffffff");**

**$(element).css("color","grey");**

**}**

**})**

**that.getContent();**

**}**

**},**

**created() {**

**this.getColumns();**

**this.getContent();**

**}**

**})**

**</script>**

**</html>**

1. **系统测试**



1. **系统总结**

在这门课的学习中我了解了html，js，，ajax，还有后端的ssm框架等等，这些都是我之前从没有接触过得东西，在学习中也感觉到了这些东西的有趣，然后在这次课设中通过实战项目加深我对这些技术的学习使我能够更加娴熟的写出代码。本次课程设计需要制作一个网页的新闻显示，对我而言还是笔记有难度。首先进行了数据库的设计。然后我找到了官网上对应的页面，并拉取了各个项的链接存入数据库中。这次课程设计的实现，让我对前后端分离的设计方式有了更多的认识。最主要还是对数据库的重要性，数据库设计的好才能为后续的代码提供更多便利。