

# LeeMall——.NET期末项目文档

author: 李佳诺 1751188

## LeeMall——.NET期末项目文档

项目简介

项目架构

项目功能

后端

前端

要求实现情况

特色

项目部署

碰到的问题

1. Entity Framework使用
2. COM组件注册
3. Win32程序集无法引用
4. C++/CLI程序集无法引用
5. PUT DELETE 请求失败
6. 部署问题

感想

## 项目简介

本项目是一款线上唱片售卖网站，集合了中外各种优秀精美专辑，涉及流派有摇滚、流行、民谣、R&B、古典等。不仅为用户提供丰富唱片资源，还提供了从浏览、添加购物车、购买、物流等一站式全流程购物体验，让用户足不出户，便可购得喜爱的唱片。

## 项目架构

本项目采用了前后端分离的开发方式，采用后端采用 **ASP.NET(.NET Framework)** 框架，数据库为 **MySQL**，为了提高查询效率采用了 **Entity Framework** 技术（EF包括开箱即用的第一级缓存。因此，重复查询将从缓存中返回数据，而不是访问数据库），实现DB first 的对象关系映射。前端采用的是Vue.js框架。

## 项目功能

### 后端

根据 RESTful 规范，分8个模块编写了共40个API

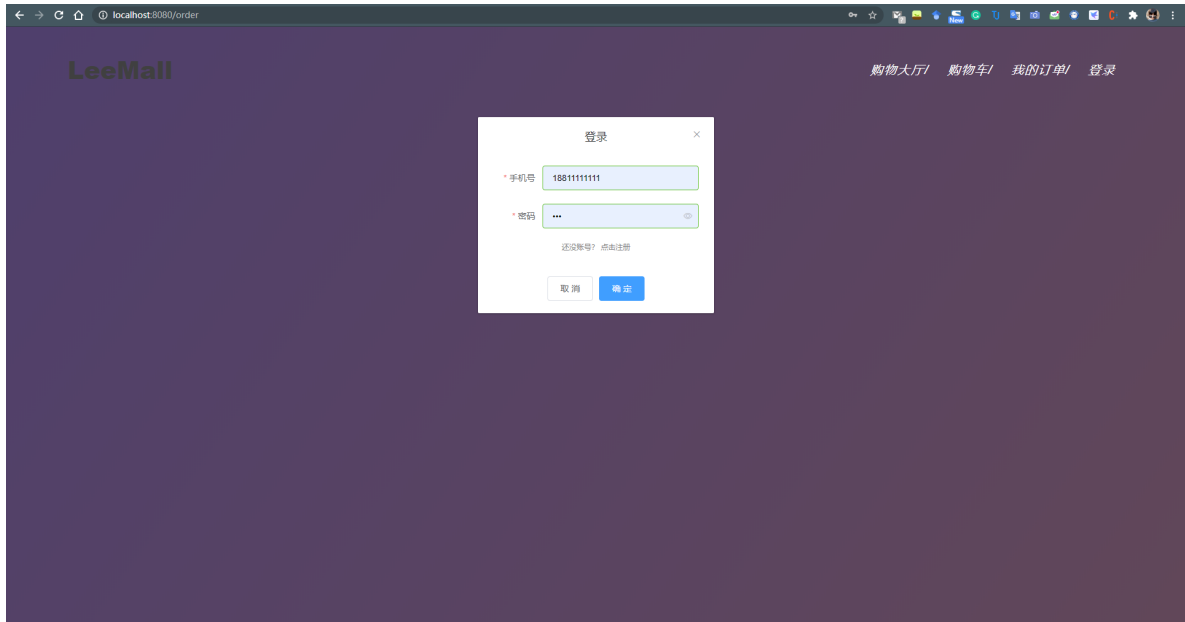
- 用户模块（用户登录、注册，状态管理）
- 唱片模块（唱片获取、搜索、修改）
- 分类模块（唱片类别管理）
- 物流模块（商品物流管理）

- 购物车模块（添加、删除、更改购物车商品，获取购物车信息）
- 订单模块（生成、获取订单，更新订单状态）
- 交易记录模块（历史交易记录查询，删除）
- 管理员模块（管理员登录）

## 前端

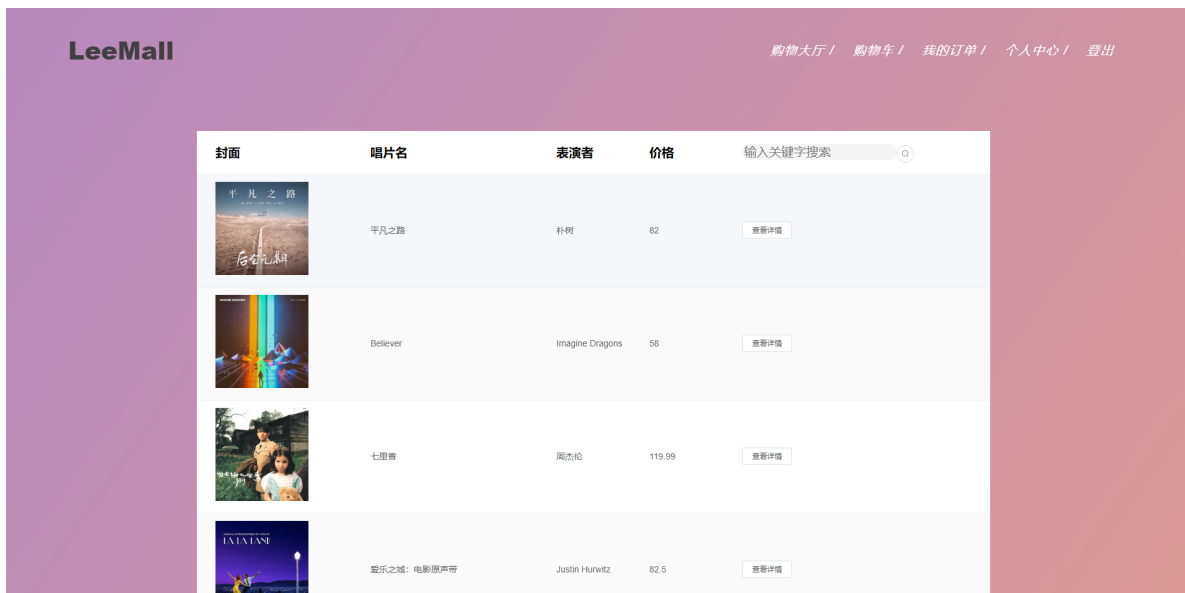
- 注册/登录

输入【手机号】与【密码】，即可登录，拥有购买权限



- 浏览唱片

点击【购物大厅】，便可浏览商城中的唱片，为了良好的交互体验，增加了【分页】功能



- 唱片搜索

在搜索框中【键入】想要查询的关键词【唱片名】或者【表演者】，查询唱片

- 添加购物车



- 商品结算

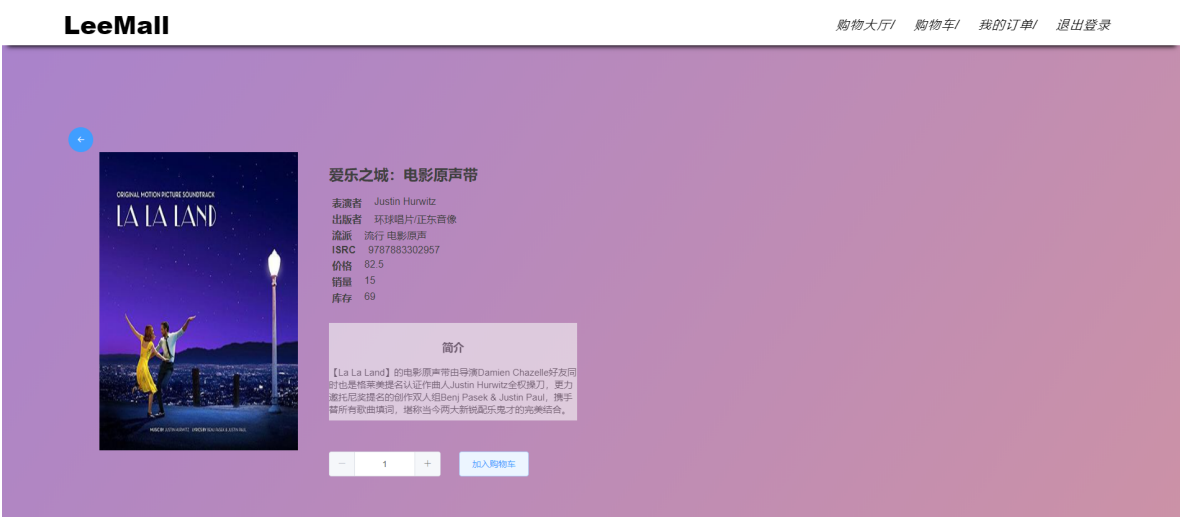
用户可以在生成订单时候【立即结算】或者在订单页面中点击【待支付】然后结算，结算之后商品进入待收货状态



- 商品接收



- 查看唱片详情



- 历史交易记录

在【我的订单】页面点击已完成的订单，便可看到历史交易记录



- 个人信息修改

## 要求实现情况

至少五个程序集（8个）✓

系统进行模块划分（划分了8个模块，如上）✓

一个共享程序集（Utils，提供API公共操作：格式转换）✓

采用C++/CLI实现一个程序集，给C#使用（Warnings，提供警示信息）✓

使用C++实现一部分功能，输出成一个Win32DLL，实现互操作（Win32Dll，计算商品价格）✓

实现一个COM组件，然后进行使用（生成、更新订单）✓

使用多线程技术（数据库访问时候采用异步，在计算价格时候使用mutex互斥量确保线程安全）✓

利用ASP.NET，ADO.NET ✓

利用.NET其他技术（Entity Framework，DB First，Code First）✓

项目部署 ✓

## 特色

- 一个前端，页面精美
- 一个后端，包括8个程序集，40个API
- 基于Docker的数据库，稳定安全

## 项目部署

1. 发布项目前，选择引用的共享程序集，右键选择其属性，把【复制本地】字段设为【True】，再进行发布。
2. 在解决方案的主项目下点击右键，选择【发布】，注意要选择Release版本。
3. 发布方式选择【文件系统】，填入目标文件夹位置。
4. 发布后会在选择的文件夹中生成项目的dll，以及资源文件。在IIS管理器中，右键添加一个网站，输入网站名称，端口，物理路径，点击确定。（物理路径就是发布项目的文件夹）。
5. 对于64位电脑，需要在项目对应的应用程序池的高级设置中，选择【启用32位应用程序】。
6. 部署完成，现在可以在浏览器中访问目标主机的网站地址了。

## 碰到的问题

## 1. Entity Framework使用

问题:

被项目的.NET版本是4.7.2, 是最新的版本, 但是MySQL 对于.NET的支持还未更新, 所以在刚开始用EF时候, 我是用的Code First, 然而遇到了好多坑, 首先是版本的问题, 所安装的包与框架版本不匹配, 无法运行, 后来还遇到外键约束无法更新, 数据库表结构与对象不统一等问题。

解决途径:

对于版本不匹配问题, 先后试过回退.NET版本, 统一 `TargetFramework` 版本等方法, 对于第二个问题, 我曾尝试清理缓存等操作, 反复尝试不理想后, 换用DB First模式, 先建立数据库表, 然后在.NET中手动映射表结构以及外键约束, 这样一开始是没有问题, 但是也坑比较多, 在MySQL对框架的支持还未完善之前, 建议尽量不要用MySQL。

## 2. COM组件注册

问题:

一开始, 注册COM组件时候, 由于是 `普通用户` 状态, 所以总是无法注册引用

解决途径:

切换到 `管理员模式` 注册引用

## 3.Win32程序集无法引用

问题:

Win32程序集本身可以正确地编译运行, 但是无法正确引用

解决途径:

切换运行平台, 主项目换成x86编译

## 4.C++/CLI程序集无法引用

问题:

C++/CLI程序集本身可以正确地编译运行, 但是无法正确引用

解决途径:

切换运行平台, 主项目换成x86编译, 或者C/CLI程序集更换目标平台重新编译

## 5. PUT DELETE 请求失败

### HTTP 错误 405.0 - Method Not Allowed

无法显示您正在查找的页面, 因为使用了无效方法(HTTP 谓词)。

#### 最可能的原因:

- 发送至 Web 服务器的请求使用了为处理该请求而配置的模块所不允许的 HTTP 谓词。
- 发送至服务器的请求包含无效的 HTTP 谓词。
- 该请求是针对静态内容的, 但包含 GET 或 HEAD 以外的 HTTP 谓词。

问题:

在运行项目时候, 访问某些个API时候总会显示以上的报错页面, 认真检查代码, 检擦数据库, 甚至重新装包, 发现都没问题, 但似乎就是解决不了。

解决途径：

我发现出现这种情况的API都是PUT，DELETE请求，后来我查到，原来在默认情况下，IIS会安装一个WebDav模块，而这个模块阻止了HTTP的PUT和DELETE请求，所以我就删除IIS安装的WebDav模块，并且在项目的web.config 文件的<system.webServer>标签内加上以下代码，之后便可成功请求

```
<modules runAllManagedModulesForAllRequests="true"
runManagedModulesForWebDavRequests="true">
    <remove name="WebDAVModule" />
</modules>
<handlers>
    <remove name="WebDAV" />
</handlers>
```

## 6. 部署问题

问题：

IIS发布网站进行访问时提示权限不足

解决途径：

是因为网站所在的文件夹中用户权限设置问题，可以在该网站所在文件夹的安全设置 为用户赋予权限

## 感想

---

我觉得一个人从数据库到后端到前端这样子亲自撸一遍，真的非常辛苦，要面临各种复杂的情况，有时候得通过从前到后的排查才能确定症结锁在，真的很考验人的耐心。然后我感觉.NET是个非常优秀的框架，特别是编写C++/CLI时候，看到其对于C++的支持，虽然目前有些不足，并且更新迭代速度很快，各版本兼容性还有待提高，但是写过之后还是可以感受到.NET框架设计的精妙，希望.NET5.0之后可以整合的更好，为开发者提供更好的编程体验。