

数据库设计说明书

目录

- 第一章 引言1
 - 1.1 编写目的1
 - 1.2 背景1
 - 1.3 术语定义1
 - 1.4 参考资料2
- 第二章 外部设计3
 - 2.1 标识符和状态3
 - 2.2 使用它的程序3
 - 2.3 命名约定3
 - 2.4 设计约定3
- 第三章4
 - 3.1 概念结构设计4
 - 3.1.1 实体和属性的定义.....4
 - 3.1.2 设计局部 ER 图8
 - 3.1.3 设计全局 ER 模式.....10
 - 3.2 逻辑结构设计10
 - 3.2.1 模式10
 - 3.2.2 外模式：没有13
 - 3.3 物理结构设计13
- 第四章 运用设计14
 - 4.1 数据字典设计：无说明14
 - 4.2 安全保密设计14
 - 4.3 数据库实施14
 - 4.3.1 创建数据库14
 - 4.3.2 创建表14

第一章 引言

1.1 编写目的

本数据库设计说明书是对福大吃点啥数据库设计的定义，包括本系统数据逻辑结构设计、数据字典以及运行环境、安全保密设计等。

本数据库设计说明书适合以下读者：

- ◆ 用户
- ◆ 系统设计人员
- ◆ 质量控制人员
- ◆ 系统确认测试人员
- ◆ 系统维护人员

本数据库设计说明书是以下开发活动的依据之一：

- ◆ 系统详细设计
- ◆ 用户验收

1.2 背景

软件名称：福大吃点啥

新生来到福大时，想寻找适合自己口味的美食但是又不熟悉福大。到福州大学旅游的游客想吃一下福大的食堂，看一看不同学校食堂的不同特色，这时就需要我们的“福大吃点啥”了，帮你找到福大的美食。吃到好吃的店家，或者找到自己中意的专属味道，那就在“吃点啥”上面发布吧，让更多的人吃到福大的美味。或者在一家店吃过一次自己觉得这家不行，想提醒大家别踩雷，那就能在我们的“福大吃点啥”上发布动态帮助大家避雷。

商家的：商家层面对商家内容的维护与修改。

平衡管理的：社区管理者的操作集合。

1.3 术语定义

本文用到的术语符合国家标准《软件工程术语（GB/T11475-1995）》。

与本文直接相关的国家标准包括：

- GB8566-1995 软件生存期过程
- GB8567-88 计算机软件产品开发文件编制指南
- GB8567-88 计算机软件数据库设计说明编制指南
- GB/T11457-1995 软件工程术语

1.4 参考资料

- 《数据库设计说明书》，国家标准格式
- 《实用软件工程》，郑人杰，殷人昆，陶永雷，清华大学出版社，1997. 4
- 《软件工程实践者之路》，美，清华大学出版社，1997. 4
- 《计算机软件产品开发文件编制指南，GB 8567-88》
- 《电子政务试点示范工程技术规范》

第二章 外部设计

2.1 标识符和状态

数据库软件名称: SQL Server 2005

数据库名称: DB_OA

2.2 使用它的程序

本数据库使用于 “福大吃点啥” V1.0

2.3 命名约定

所有的数据库命名都是以模块的缩写加上具体表的英文词汇组成, 这样能够统一数据库表的命名, 也能够更好的规范数据库表命名。

2.4 设计约定

在本系统中, 数据库的设计采用 Visio 进行, 并且采用面向对象的设计方法, 首先进行对象实体的设计, 最后将对象持久化到数据库中, 所有的表和表之间的关联(ER 图)都采用标准的 Visio 设计工具进行, 这样能够将整个系统的设计和数据库设计有机的结合起来。

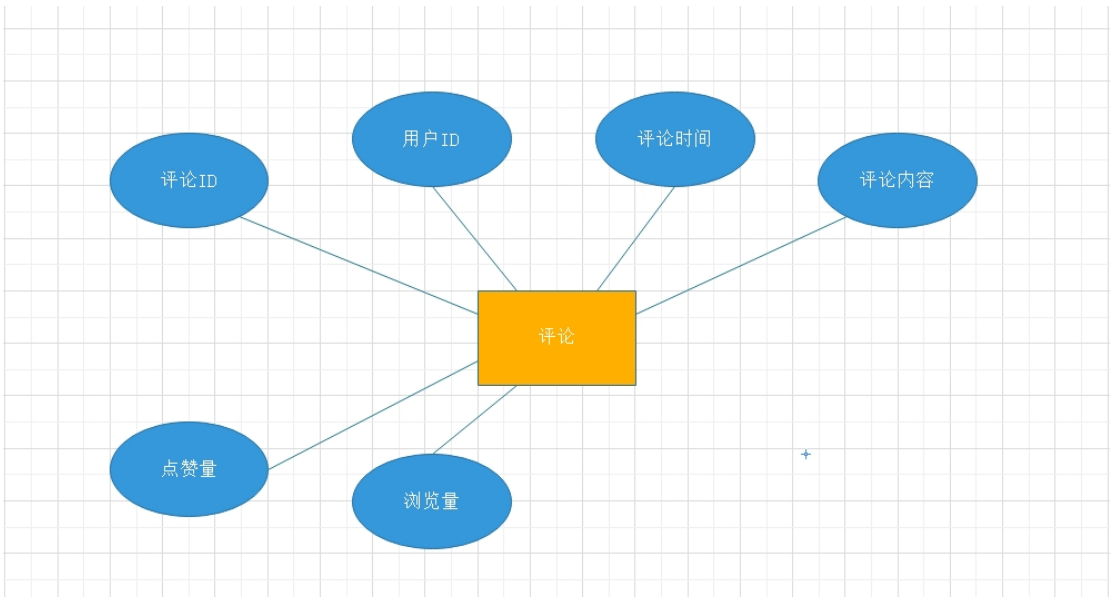
第三章 结构设计

3.1 概念结构设计

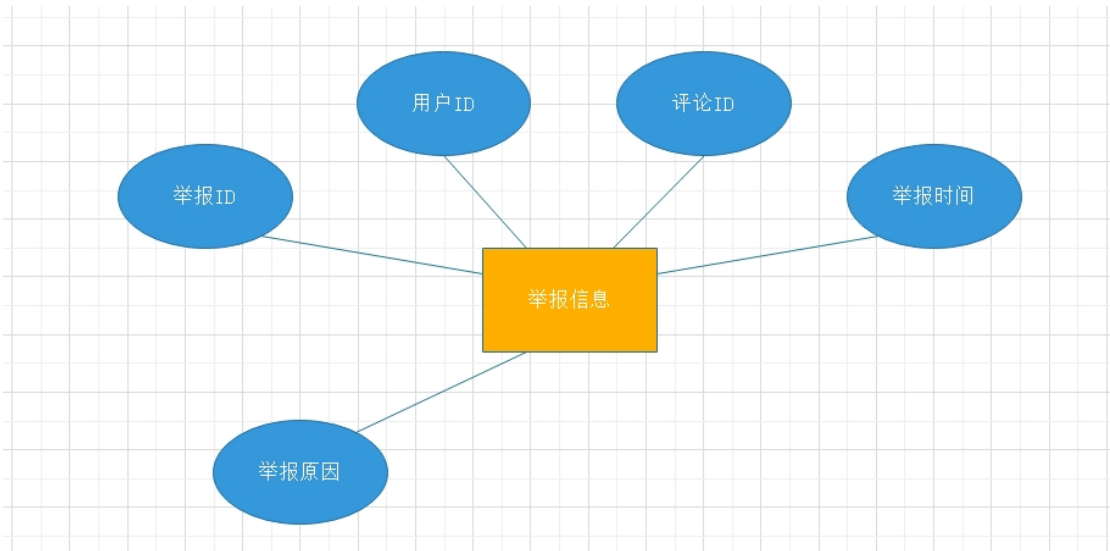
3.1.1 实体和属性的定义

公共模块

评论信息（评论 ID，用户 ID，评论时间，评论内容，点赞量，浏览量）



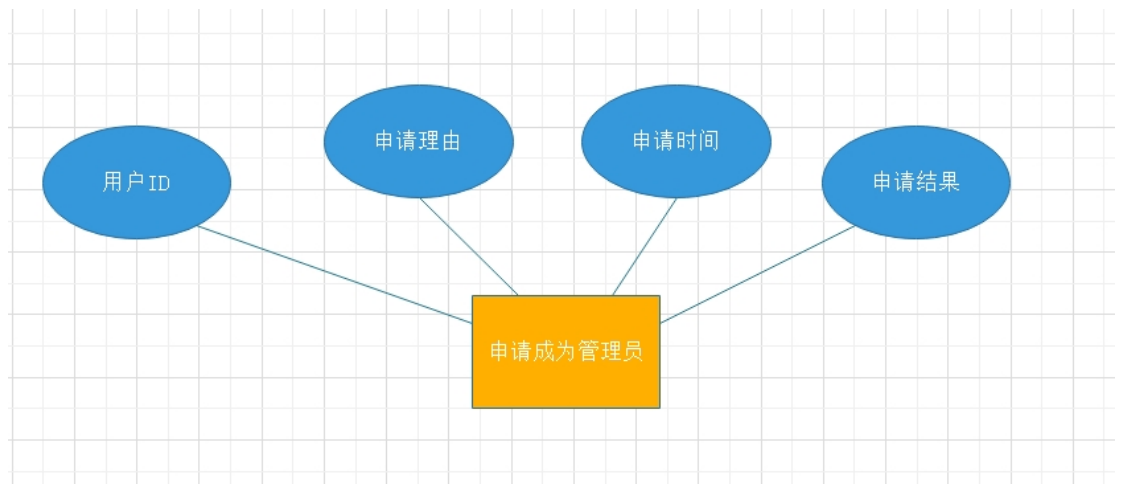
举报信息（举报 ID，用户 ID，评论 ID，举报时间，举报原因）



申请成为管理员（用户 ID，申请理由，申请时间，申请结果）

其中属性约束如下：

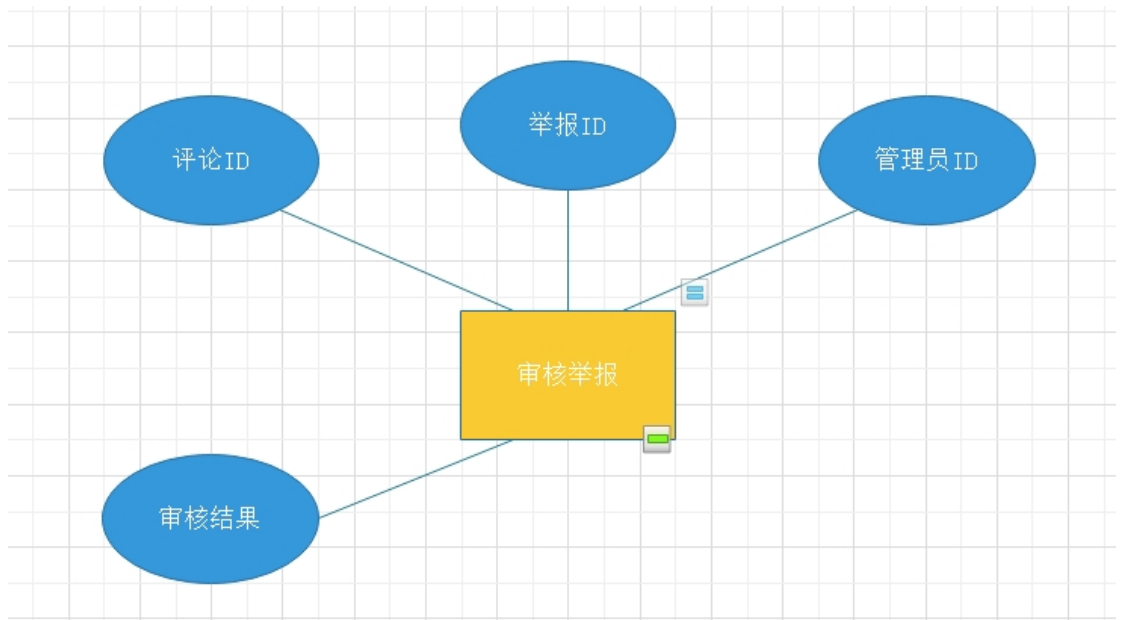
申请结果：0=通过 1=未通过



审核举报（举报 ID，评论 ID，管理员 ID，审核结果）

其中属性约束如下：

审核结果：0=未通过 1=通过

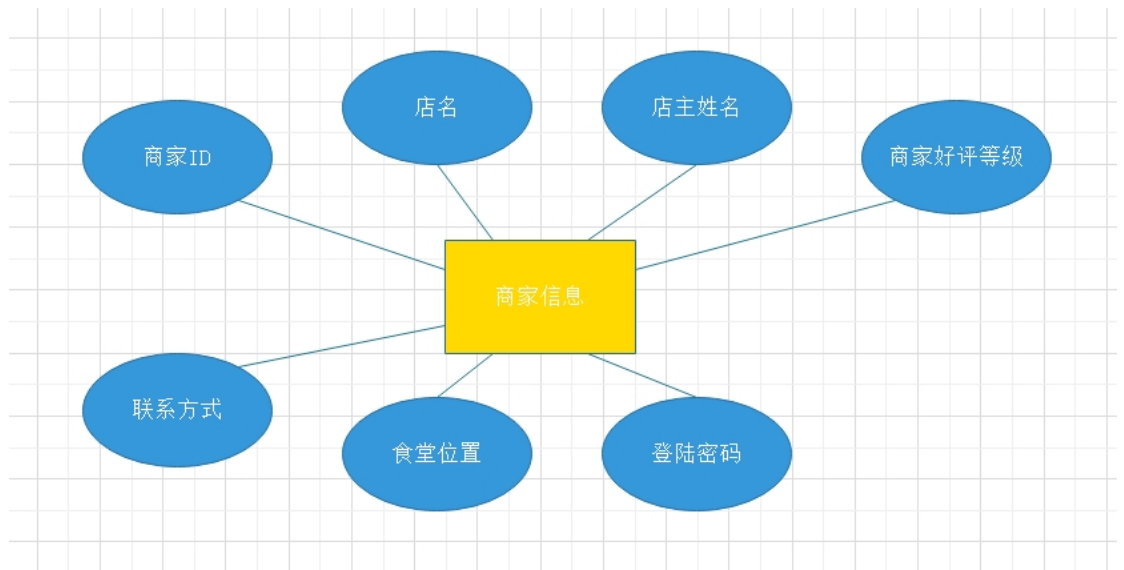


商家模块

商家信息（商家 ID，店名/登录名，店主姓名，商家好评等级，联系方式，食堂位置，登陆密码）

其中属性约束如下：

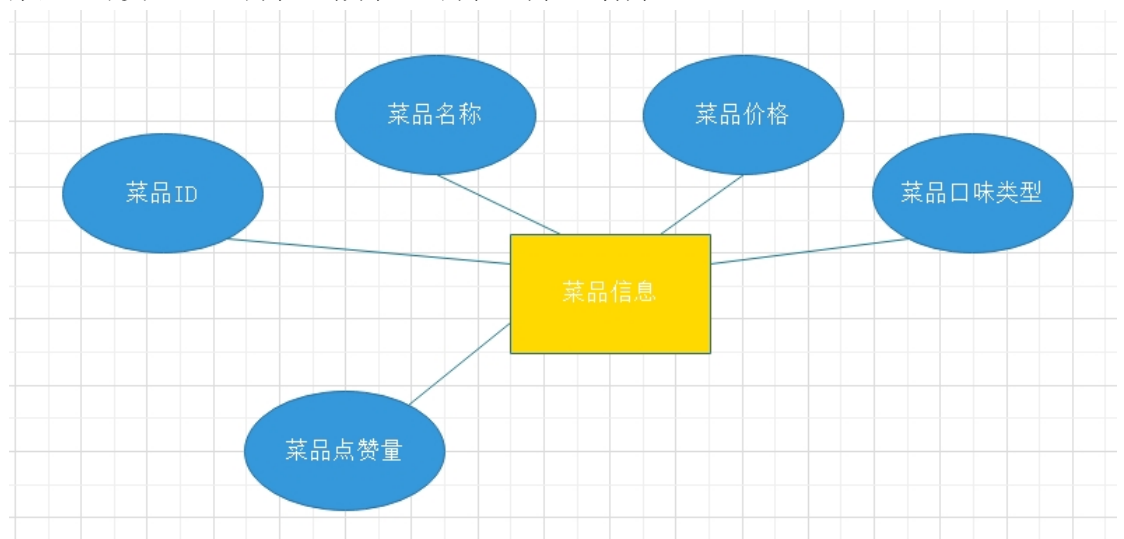
商家好评等级：-2=特别差评 -1=差评 1=一般 2=较好 3=好评 4=特别好评



菜品信息（菜品 ID，菜品名称，菜品价格，菜品口味类型，菜品点赞量）

其中属性约束如下：

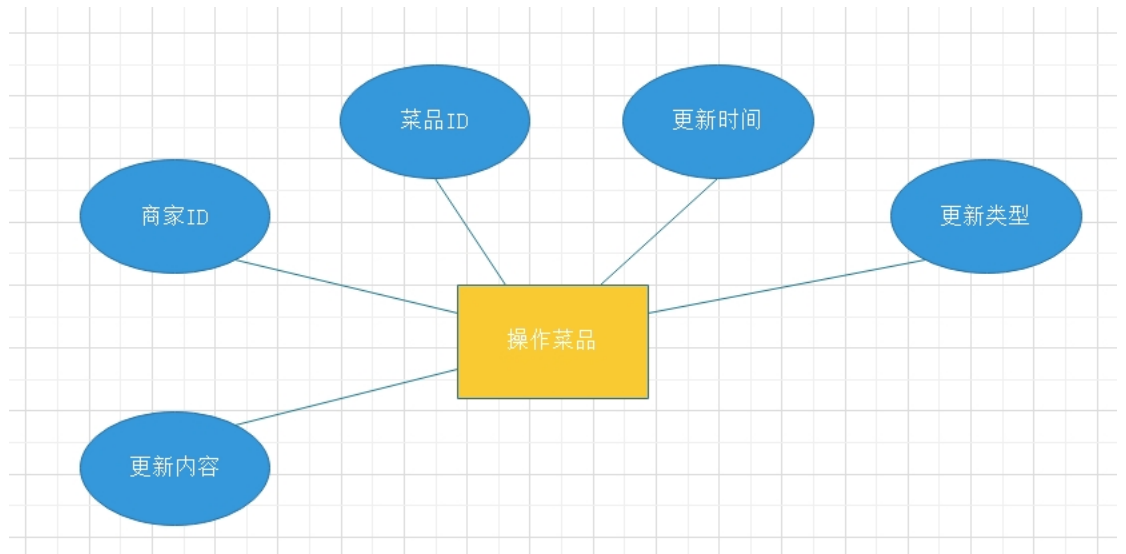
菜品口味类型：1=不辣 2=微辣 3=中辣 4=辣 5=特辣



操作菜品（商家 ID，菜品 ID，更新时间，更新类型，更新内容）

其中属性约束如下：

更新类型：1=添加 2=更新 3=删除



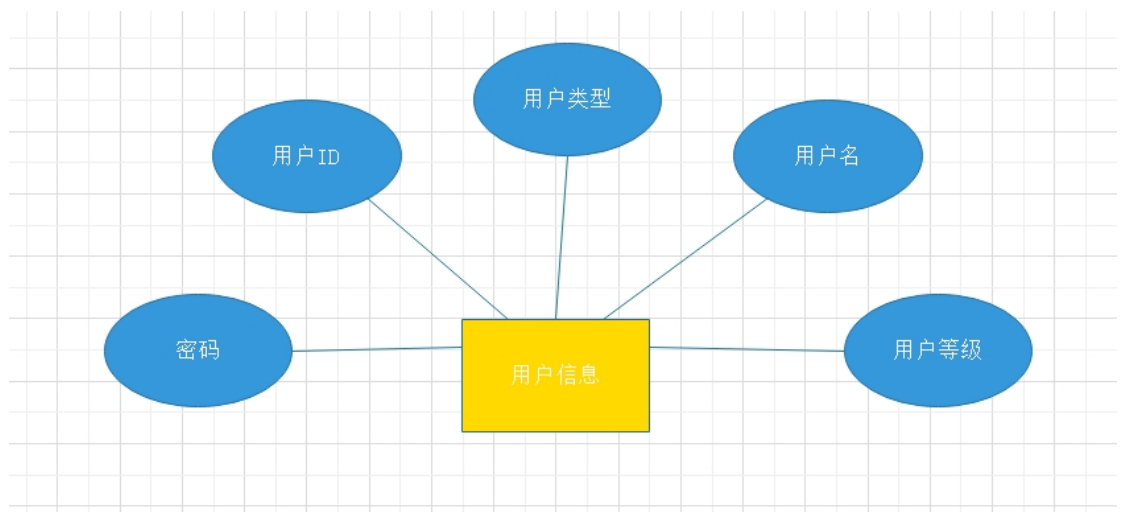
用户模块

用户信息（用户 ID，用户类型，用户名，密码，用户等级）

其中属性约束如下：

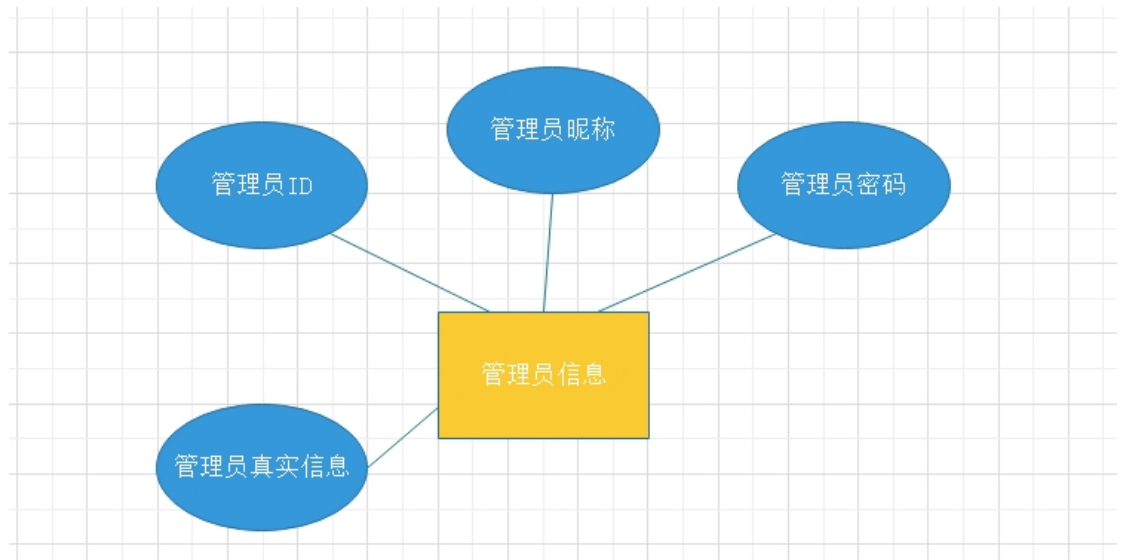
用户类型：1=游客 3=福大注册用户

用户等级：-2=恶意重伤 -1=无脑瞎 1=水 2=青铜 3=黄金 4=大神 5=传说



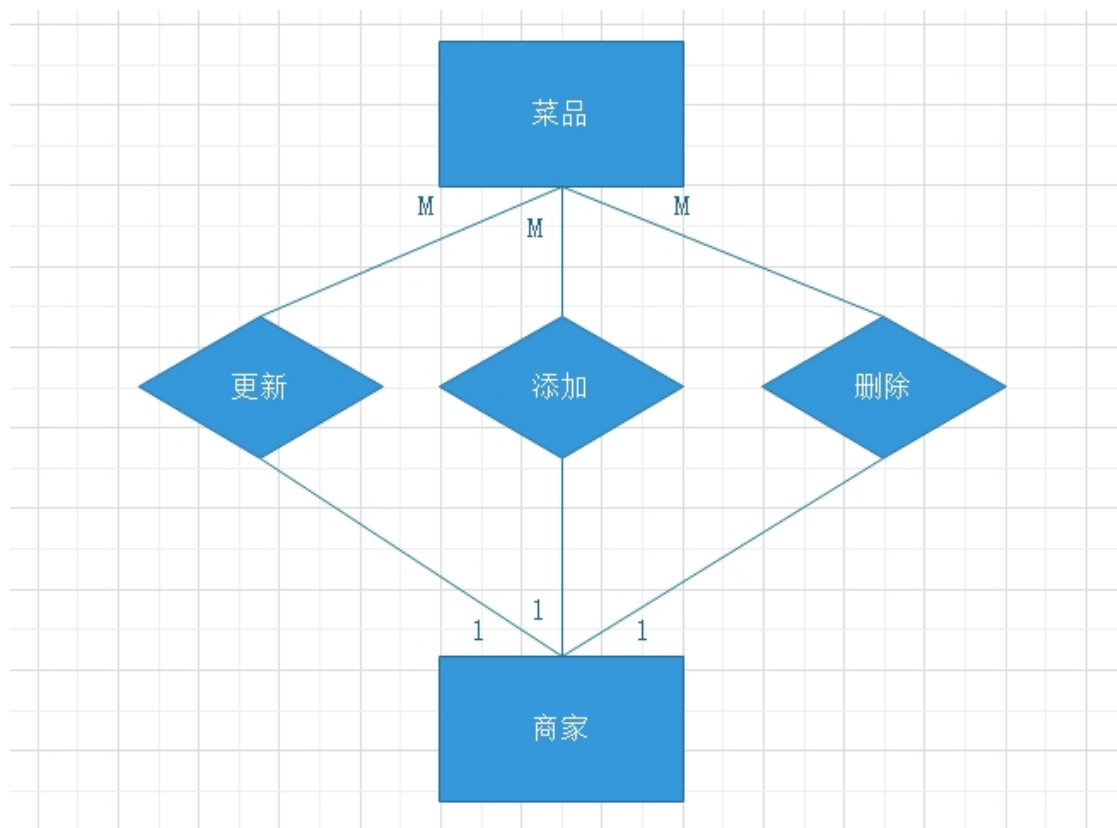
管理员模块

管理员信息（管理员 ID，管理员昵称，管理员密码，管理员真实信息）

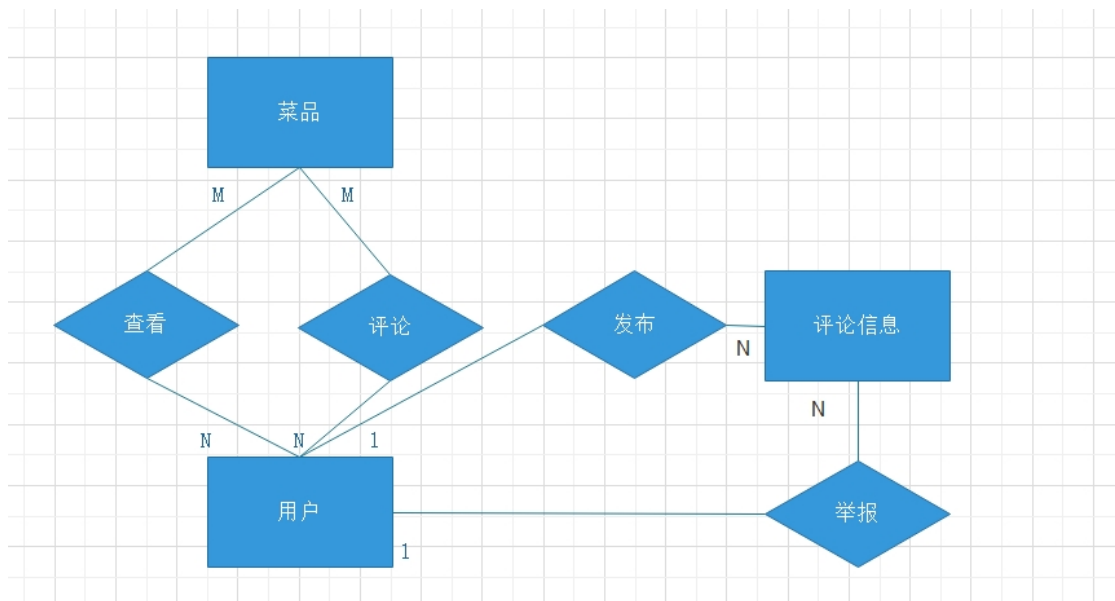


3.1.2 设计局部 ER 图

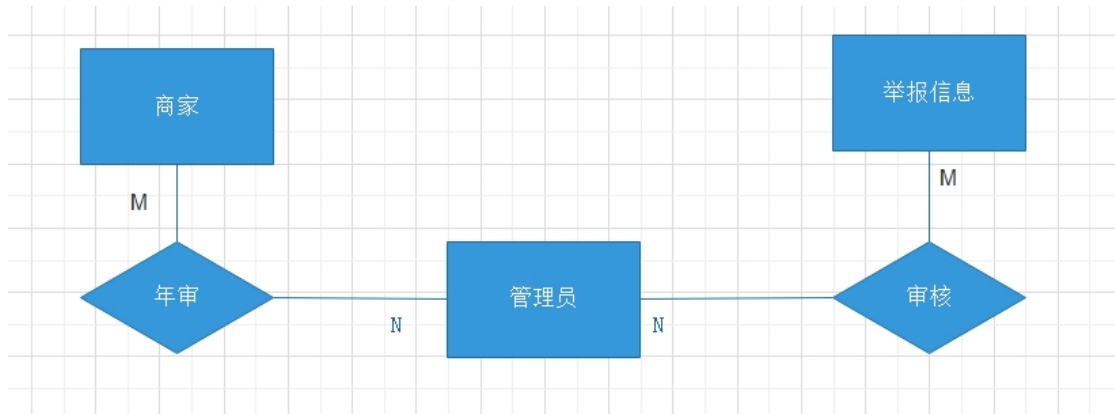
商家与菜品:



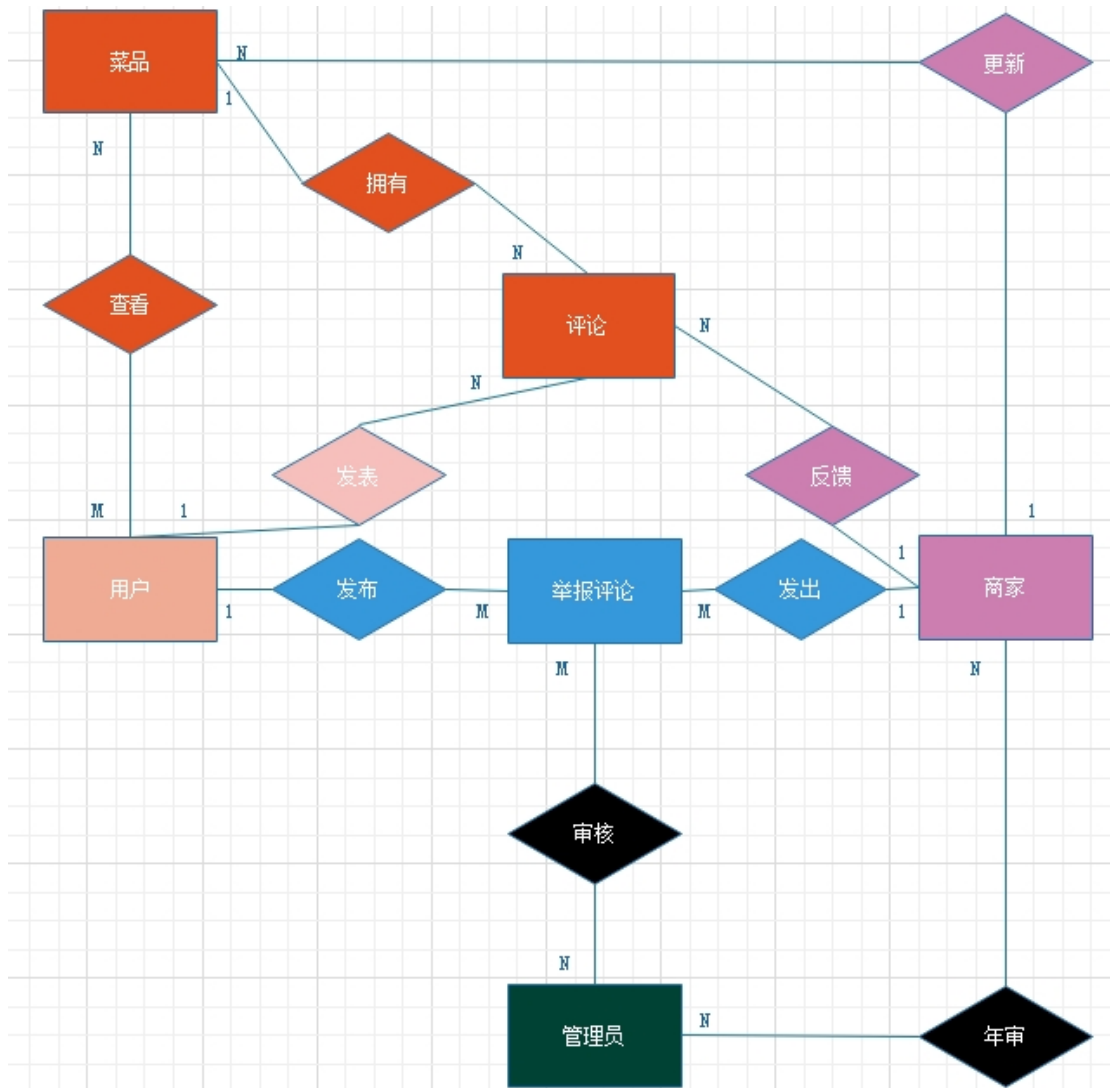
用户与菜品



管理员与商家



3.1.3 设计全局 ER 模式



3.2 逻辑结构设计

3.2.1 模式

Public:

评论信息（评论 ID，用户 ID，评论时间，评论内容，点赞量，浏览量）

字段名	数据类型	长度	主键	非空	描述
CriID	Int	15	是	否	评论 id（自增 ID）
UserID	Int	15			用户 id
CriTime	DATETIME	20			评论时间

Criticism	Text	200			评论内容
Like	Int	5			点赞量
Browse	Int	5			浏览量

举报信息（用户 ID，举报评论 ID，举报时间，举报原因）

字段名	数据类型	长度	主键	非空	描述
UserID	Int	15	是	否	用户 ID（自增 ID）
ReportID	Int	15			举报 ID
CriID	int	15			评论 id
ReportTime	Datetime	20			举报时间
ReportReason	Text	50			举报原因

申请成为管理员（用户 ID，申请理由，申请时间，申请结果）

字段名	数据类型	长度	主键	非空	描述
UserID	Int	15	是	否	用户 ID
Applyreason	Text	200			申请理由
Applytime	Datetime	20			申请时间
Result	Int	4			申请结果（0 通过 1= 未通过）

审核举报（举报 ID，评论 ID，管理员 ID，审核结果）

字段名	数据类型	长度	主键	非空	描述
ReportID	Int	15	是	否	举报 ID
CriID	Int	15			评论 ID
AID	Int	15			管理员 ID
Checkresult	Int	4			审核结果（0=未通过 1=通过）

商家模块

商家信息（商家 ID，店名/登录名，店主姓名，商家好评等级，联系方式，食堂位置，登陆密码）

其中属性约束如下：

商家好评等级：-2=特别差评 -1=差评 1=一般 2=较好 3=好评 4=特别好评

字段名	数据类型	长度	主键	非空	描述
SID	Int	4	是	否	商家 ID（自增 ID）
Location	Char	30			食堂位置

SKEY	Char	15			登陆密码
Tel	Char	11			联系方式
SNAME	Char	15			店名
Ownername	Char	10			店主姓名
Crilevel	Int	2			好评等级

菜品信息（菜品 ID，菜品名称，菜品价格，菜品口味类型，菜品点赞量）

其中属性约束如下：

菜品口味类型：1=不辣 2=微辣 3=中辣 4=辣 5=特辣

字段名	数据类型	长度	主键	非空	描述
CID	INT	3	是	否	菜品 ID（自增 ID）
CNAME	CHAR	10			菜品名称
CPRICE	FLOAT	4			菜品价格
CTASTE	INT	1			菜品口味类型
CLIKES	INT	5			菜品点赞数

操作菜品（商家 ID，菜品 ID，更新时间，更新类型，更新内容）

其中属性约束如下：

更新类型：1=添加 2=更新 3=删除

字段名	数据类型	长度	主键	非空	描述
SID	Int	4	是	否	商家 ID（自增 ID）
CID	INT	3	是	否	菜品 ID（自增 ID）
REFTIME	TIMESTAMP				更新时间
REFKIND	INT	1			更新类型
REFRESH	CHAR	20			更新内容

Users:

用户信息（用户 ID，用户类型，用户名，密码，用户等级）

字段名	数据类型	长度	主键	非空	描述
UserID	Int	15	是	否	用户 ID（自增 ID）
UserType	Int	10			用户类型（1=游客 3=福大注册用户）
UserName	Varchar	15			用户名
PassWord	Varchar	50			密码
Level	Int	4			用户等级（-

					2= 恶意重伤 -1= 无脑瞎 1=水 2=青铜 3=黄金 4=大神 5=传说)
--	--	--	--	--	---

Admin:

管理员模块

管理员信息（管理员 ID，管理员昵称，管理员密码，管理员真实信息）

字段名	数据类型	长度	主键	非空	描述
AID	INT	4	是	否	管 理 员 ID (自增 ID)
AIMFO	CHAR	30			真实信息
ANAME	CHAR	10			管理员昵称
AKEY	CHAR	15			管理员密码

3.2.2 外模式 ： 没有

3.3 物理结构设计

数据库名称为: DB_OA

存储位置：默认位置

建立系统程序员视图，包括：

- 数据在内存中的安排，包括对索引区、缓冲区的设计；
- 所使用的外存设备及外存空间的组织，包括索引区、数据块的组织与划分；
- 访问数据的方式方法。

第四章 运用设计

4.1 数据字典设计：无说明

4.2 安全保密设计

通过区分不同的访问者、不同的访问类型和不同的数据对象，进行分别对待而获得的数据库安全保密设计考虑。对数据库设计中涉及到的各种项目，如数据项、记录、系、文卷、模式、子模式等一般要建立起数据保护措施，以说明它的标识符、同义名及有关信息。数据库由专门数据库管理用员对数据库操作，需要注意以下几项安全问题：

访问安全 、网络安全 、传输安全 、备份安全 、数据安全

4.3 数据库实施

4.3.1 创建数据库

```
Create database public;  
Create database merchants;  
Create database users;
```

4.3.2 创建表

```
USE [BBSDataTest]  
Go  
CREATE TABLE [dbo].[criticism] (  
    [CriID] [int] IDENTITY(1,1) NOT NULL,  
    [UserID] [int](15) COLLATE Chinese_PRC_CI_AS NULL,  
    [CriTime] [datetime](20) NULL,  
    [Criticism] [text](200) COLLATE Chinese_PRC_CI_AS NULL,  
    [Like] [int](5) COLLATE Chinese_PRC_CI_AS NULL,  
    [Browse] [int](5) COLLATE Chinese_PRC_CI_AS NULL,
```



```

CONSTRAINT [PK_criticism] PRIMARY KEY
(
    [CriID] ASC
) WITH (PAD_INDEX = OFF, IGNORE_DUP_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
Go
CREATE TABLE [dbo].[report] (
    [UserID] [int] IDENTITY(1,1) NOT NULL,
    [ReportID] [int] (15) COLLATE Chinese_PRC_CI_AS NULL,
    [CriID] [int] (15) COLLATE Chinese_PRC_CI_AS NULL,

    [ReportTime] [datetime] (20) NULL,
    [ReportReason] [text] (50) COLLATE Chinese_PRC_CI_AS NULL,

CONSTRAINT [PK_report] PRIMARY KEY
(
    [UserID] ASC
) WITH (PAD_INDEX = OFF, IGNORE_DUP_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
Go
CREATE TABLE [dbo].[applya] (
    [UserID] [int] IDENTITY(1,1) NOT NULL,
    [Applyreason] [text] (200) COLLATE Chinese_PRC_CI_AS NULL,
    [Applytime] [datetime] (20) NULL,
    [Result] [int] (4) COLLATE Chinese_PRC_CI_AS NULL,

CONSTRAINT [PK_applya] PRIMARY KEY
(
    [UserID] ASC
) WITH (PAD_INDEX = OFF, IGNORE_DUP_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
Go
CREATE TABLE [dbo].[check] (
    [ReportID] [int] IDENTITY(1,1) NOT NULL,
    [CriID] [int] (15) COLLATE Chinese_PRC_CI_AS NULL,
    [AID] [int] (15) COLLATE Chinese_PRC_CI_AS NULL,
    [Checkresult] [int] (4) COLLATE Chinese_PRC_CI_AS NULL,

```

```

CONSTRAINT [PK_check] PRIMARY KEY
(
    [ReportID] ASC
) WITH (PAD_INDEX = OFF, IGNORE_DUP_KEY = OFF) ON [PRIMARY]
) ON [PRIMARY]
Go
CREATE TABLE [dbo].[criticism] (
    [SID] [int] (4) NOT NULL,
    [Location] [Char] (30) COLLATE Chinese_PRC_CI_AS NULL,
    [SKEY] [CHAR] (15) NULL,
    [TEL] [CAHR] (15) NULL,
    [SNAME] [CHAR] (15) COLLATE Chinese_PRC_CI_AS NULL,
    [OWNERNAME] [CHAR] (10) COLLATE Chinese_PRC_CI_AS NULL,
    [CRILEVEL] [int] (2) NULL,
    CONSTRAINT [SID] PRIMARY KEY);
Go
CREATE TABLE [dbo].[criticism] (
    [CID] [int] (3) NOT NULL,
    [CNAME] [Char] (10) COLLATE Chinese_PRC_CI_AS NULL,
    [CPRICE] [FLOAT] (4) NULL,
    [CTASTE] [INT] (1) NULL,
    [CLIKES] [INT] (5) COLLATE NULL,
    CONSTRAINT [CID] PRIMARY KEY,
    CONSTRAINT C1 CHECK (CTASTE IN (1, 2, 3, 4, 5));
Go
CREATE TABLE [dbo].[criticism] (
    [SID] [int] (4) NOT NULL,
    [CID] [INT] (30) COLLATE NOT NULL,
    [REFTIME] [TIMESTAMP] NULL,
    [REFKIND] [INT] (1) NULL,
    [REFRESH] [CHAR] (20) COLLATE Chinese_PRC_CI_AS NULL,
    CONSTRAINT ([SID], [CID]) PRIMARY KEY);
GO
CREATE TABLE [dbo].[User] (
    [UserID] [int] IDENTITY (1, 1) NOT NULL,
    [UserType] [int] (10) COLLATE Chinese_PRC_CI_AS NULL,
    [UserName] [varchar] (15) COLLATE Chinese_PRC_CI_AS NULL,
    [PassWord] [char] (50) COLLATE Chinese_PRC_CI_AS NULL,
    [Level] [int] (4) COLLATE Chinese_PRC_CI_AS NULL,

    CONSTRAINT [PK_User] PRIMARY KEY
(

```

```
        [UserID] ASC
    ) WITH (PAD_INDEX = OFF, IGNORE_DUP_KEY = OFF) ON [PRIMARY]
    ) ON [PRIMARY]
GO
CREATE TABLE [dbo].[ADMIN] (
    [AID] [INT] (4) PRIMARY KEY,
    [AIMFO] [CHAR] (30),
    [ANAME] [CHAR] (10),
    [AKEY] [CHAR] (15));
```