

# 数据库期末工程作业设计

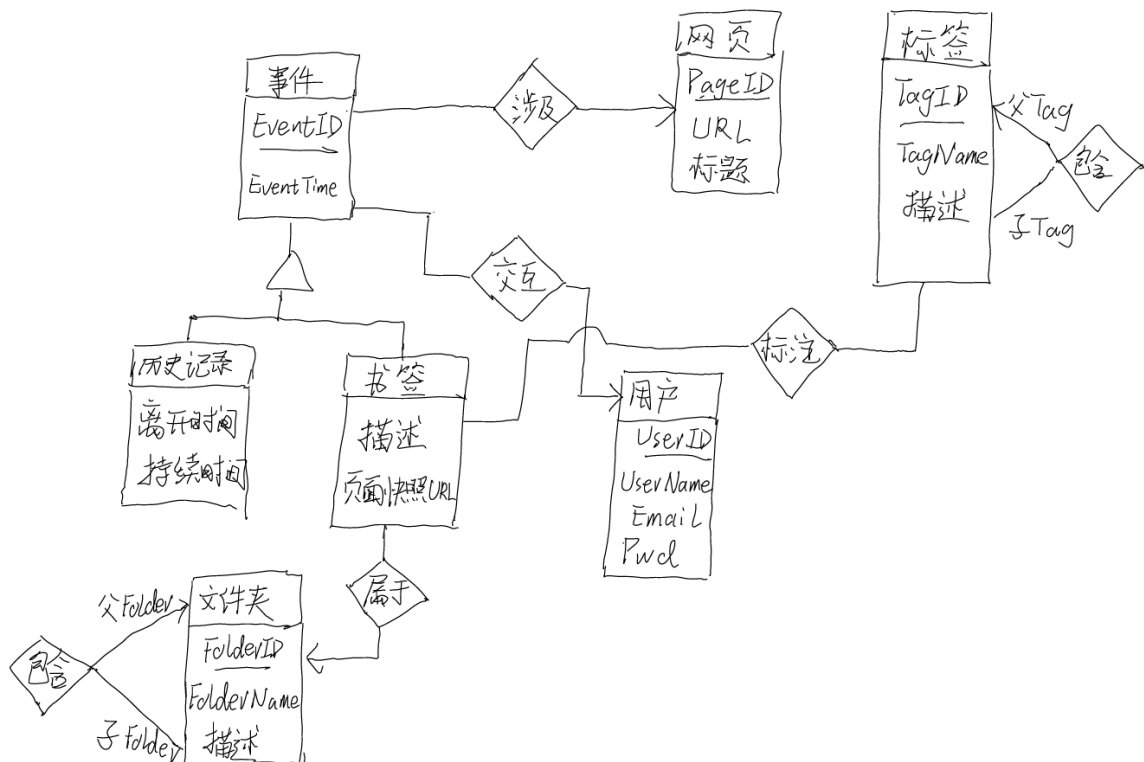
学号: 2310764 姓名: 王亦辉

## 1 需求描述

随着互联网的普及，用户在浏览器中访问的网页数量激增，管理和组织这些网页信息的需求日益增加。且由于用户可能使用多个浏览器，这些浏览器之间数据无法轻易互通，导致对书签、历史记录的使用不方便。多用户浏览器历史记录与书签管理系统旨在为多个用户提供个性化的网页访问历史记录和书签管理功能。

## 2 实现设计

### 2.1 概念模型 E-R 图



## 2.2 转换成的关系模式

事件 (EventID, EventTime, UserID(FK), PageID (FK), )

历史记录 (ExitTime, Duration)

书签 (Description, SnapshotURL, FolderID (FK))

文件夹 (FolderID, FolderName, Description, ParentFolderID (FK))

网页 (PageID, URL, Title)

标签 (TagID, TagName, Description, ParentTagID(FK))

用户 (UserID, UserName, Email, Password)

标注 (labelID, UserID (FK), TagID (FK))

## 2.3 创建上述关系模式的 SQL 语句

```
1  -- 创建事件表
2  CREATE TABLE Event (
3      EventID INT AUTO_INCREMENT PRIMARY KEY,
4      EventTime DATETIME NOT NULL,
5      UserID INT,
6      PageID INT,
7      FOREIGN KEY (UserID) REFERENCES User (UserID),
8      FOREIGN KEY (PageID) REFERENCES WebPage (PageID)
9  );
10
11 -- 创建历史记录表
12 CREATE TABLE History (
13     ExitTime DATETIME NOT NULL,
14     DurationTime INT NOT NULL
15 );
16
17 -- 创建书签表
18 CREATE TABLE Bookmark (
19     Description TEXT,
20     SnapshotURL VARCHAR (255),
21     FolderID INT,
22     FOREIGN KEY (FolderID) REFERENCES Folder (FolderID)
23 );
24
25 -- 创建文件夹表
26 CREATE TABLE Folder (
27     FolderID INT AUTO_INCREMENT PRIMARY KEY,
28     FolderName VARCHAR (100),
29     Description TEXT,
30     ParentFolderID INT,
```

```

31     FOREIGN KEY (ParentFolderID) REFERENCES Folder (FolderID)
32 );
33
34 -- 创建网页表
35 CREATE TABLE WebPage (
36     PageID INT AUTO_INCREMENT PRIMARY KEY,
37     URL VARCHAR (255) NOT NULL,
38     Title VARCHAR (255)
39 );
40
41 -- 创建标签表
42 CREATE TABLE Tag (
43     TagID INT AUTO_INCREMENT PRIMARY KEY,
44     TagName VARCHAR (100),
45     Description TEXT,
46     ParentTagID INT,
47     FOREIGN KEY (ParentTagID) REFERENCES Tag (TagID)
48 );
49
50 -- 创建用户表
51 CREATE TABLE User (
52     UserID INT AUTO_INCREMENT PRIMARY KEY,
53     UserName VARCHAR (100) NOT NULL,
54     Email VARCHAR (100),
55     Password VARCHAR (255)
56 );
57
58 -- 创建标注表
59 CREATE TABLE Label (
60     LabelID INT AUTO_INCREMENT PRIMARY KEY,
61     UserID INT,
62     TagID INT,
63     FOREIGN KEY (UserID) REFERENCES User (UserID),
64     FOREIGN KEY (TagID) REFERENCES Tag (TagID)
65 );

```

## 2.4 5 个查询样例

### 1. 单表查询：

查询所有用户的用户名和电子邮件：

```

1 SELECT UserName, Email
2 FROM User;

```

## 2. 多表连接查询：

查询每个事件的事件时间、用户名和网页标题：

```
1 SELECT Event.EventTime, User.UserName, WebPage.Title
2 FROM Event
3 JOIN User ON Event.UserID = User.UserID
4 JOIN WebPage ON Event.PageID = WebPage.PageID;
```

## 3. 多表嵌套查询：

查询所有标签名和其父标签名（如果有的话）：

```
1 SELECT Tag.TagName,
2       (SELECT TagName FROM Tag AS ParentTag WHERE ParentTag.TagID =
3       Tag.ParentTagID) AS ParentTagName
3 FROM Tag;
```

## 4. EXISTS 查询：

查询所有有标注（即至少有一个标签）的用户用户名：

```
1 SELECT UserName
2 FROM User
3 WHERE EXISTS (
4     SELECT 1
5     FROM Label
6     WHERE Label.UserID = User.UserID
7 );
```

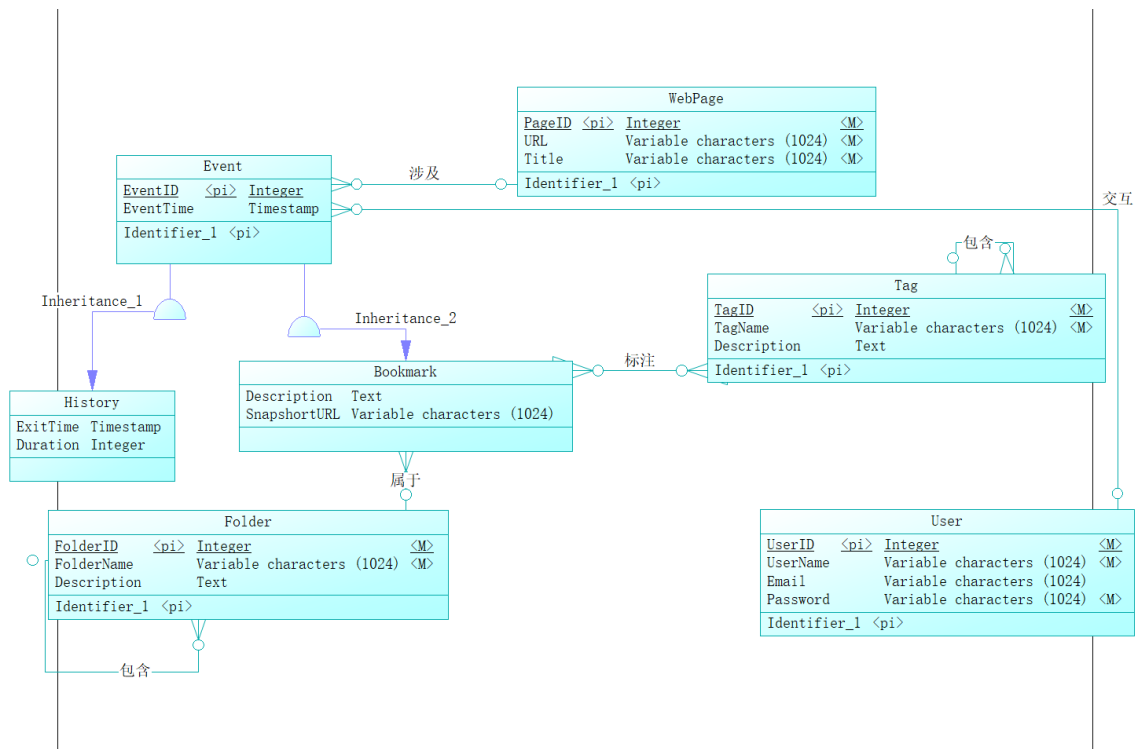
## 5. 聚合操作查询：

查询每个文件夹下书签的数量：

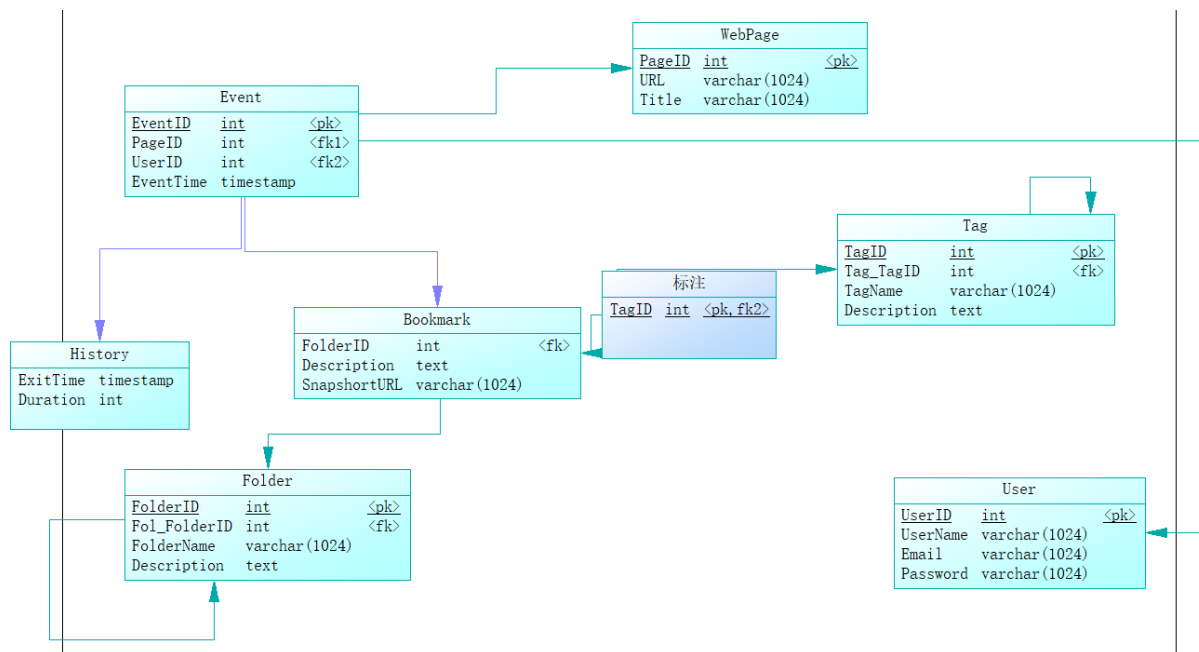
```
1 SELECT Folder.FolderName, COUNT(Bookmark.Description) AS BookmarkCount
2 FROM Folder
3 LEFT JOIN Bookmark ON Folder.FolderID = Bookmark.FolderID
4 GROUP BY Folder.FolderName;
```

# 3 PowerDesigner 设计

## 3.1 PowerDesigner E-R 图截图。



## 3.2 关系模型图截图



## 3.3 PowerDesigner 生成的创建数据库的 SQL 语句

```

1  /*=====*/
2  /* DBMS name:      MySQL 5.0 */
3  /* Created on:     2025/4/14 23:39:18 */
4  /*=====*/
5
6
7  drop table if exists Bookmark;

```

```
8
9 drop table if exists Event;
10
11 drop table if exists Folder;
12
13 drop table if exists History;
14
15 drop table if exists Tag;
16
17 drop table if exists User;
18
19 drop table if exists WebPage;
20
21 drop table if exists 标注;
22
23 /*=====*/
24 /* Table: Bookmark */
25 /*=====*/
26 create table Bookmark
27 (
28     FolderID          int,
29     Description        text,
30     SnapshotURL        varchar (1024)
31 );
32
33 /*=====*/
34 /* Table: Event */
35 /*=====*/
36 create table Event
37 (
38     EventID            int not null,
39     PageID             int,
40     UserID             int,
41     EventTime          timestamp not null,
42     primary key (EventID)
43 );
44
45 /*=====*/
46 /* Table: Folder */
47 /*=====*/
48 create table Folder
49 (
50     FolderID           int not null,
51     Fol_FolderID       int,
```

```

52     FolderName          varchar (1024) not null,
53     Description          text,
54     primary key (FolderID)
55 );
56
57 /*=====*/
58 /* Table: History */
59 /*=====*/
60 create table History
61 (
62     ExitTime             timestamp,
63     Duration             int
64 );
65
66 /*=====*/
67 /* Table: Tag */
68 /*=====*/
69 create table Tag
70 (
71     TagID                int not null,
72     Tag_TagID            int,
73     TagName              varchar (1024) not null,
74     Description          text,
75     primary key (TagID)
76 );
77
78 /*=====*/
79 /* Table: User */
80 /*=====*/
81 create table User
82 (
83     UserID               int not null,
84     UserName             varchar (1024) not null,
85     Email                varchar (1024),
86     Password             varchar (1024) not null,
87     primary key (UserID)
88 );
89
90 /*=====*/
91 /* Table: WebPage */
92 /*=====*/
93 create table WebPage
94 (
95     PageID               int not null,

```

```

96     URL                varchar (1024) not null,
97     Title               varchar (1024) not null,
98     primary key (PageID)
99 );
100
101 /*=====*/
102 /* Table: 标注 */
103 /*=====*/
104 create table 标注
105 (
106     TagID                int not null,
107     primary key (TagID)
108 );
109
110 alter table Bookmark add constraint FK_属于 foreign key (FolderID)
111     references Folder (FolderID);
112
113 alter table Event add constraint FK_Inheritance_1 foreign key ()
114     references History;
115
116 alter table Event add constraint FK_Inheritance_2 foreign key ()
117     references Bookmark;
118
119 alter table Event add constraint FK_交互 foreign key (UserID)
120     references User (UserID);
121
122 alter table Event add constraint FK_涉及 foreign key (PageID)
123     references WebPage (PageID);
124
125 alter table Folder add constraint FK_包含 folder foreign key
126     (Fol_FolderID)
127     references Folder (FolderID);
128
129 alter table Tag add constraint FK_包含 tag foreign key (Tag_TagID)
130     references Tag (TagID);
131
132 alter table 标注 add constraint FK_标注 foreign key ()
133     references Bookmark;
134
135 alter table 标注 add constraint FK_标注 2 foreign key (TagID)
136     references Tag (TagID);
137

```



## 4 比较两种方法

### 4.1 差异

没有太大差异。只是约束写的位置不一样，风格不同而已。

### 4.2 PowerDesigner 特点

PowerDesigner 会自动为表之间的关系生成外键约束。例如，它会为父表和子表之间的关系生成外键，并且如果表设计中存在层级关系，PowerDesigner 会自动为层级之间的关联生成外键约束。

PowerDesigner 会先检查是否存在这个表，如果存在则将其 DROP 掉，这样比较安全。