课程设计名称[[1]](#footnote-1)🞼

参与同学（冯古豪 2000013175 吴清月 200001）

(北京大学 信息科学技术学院, 北京 100871);

E-mail: [fenguhao@stu.pku.edu.cn](mailto:fenguhao@stu.pku.edu.cn)

摘要: 描述本课程设计的主要工作，特色，完成这个项目的收获

# 引言

描述为什么选择这个题目，其意义是什么

描述一下课程设计实现的功能

个人的收获

本文的组织：在第二部分中，我们介绍了本课程设计需要的背景知识，包括。。。；第三部分描述了课程设计的系统框架；第四部分给出相关的关系数据库模式和相应的操作；第五部分总结了全文，并且描述了完成本课程设计的收获。

# 相关工作

介绍一下完成这个项目的相关技术，不用太多

在构建前端部分，我们选择了使用Python的Flask框架作为服务器的搭建工具，并使用Bootstrap框架来构建前端页面，此外，我们还使用了JavaScript来实现一些逻辑操作，以增强前端的交互性和动态性。。Flask框架为我们提供了一个快速而灵活的开发环境，使得我们可以轻松地处理服务器端的逻辑。同时，Bootstrap框架为我们提供了丰富的前端组件和样式，使得我们可以轻松地构建出现代化和响应式的用户界面。

为了实现与数据库的交互，我们选择了SQL Server作为我们的数据库系统，并使用Python的pyodbc库来实现与数据库的连接和交互。在处理复杂的字符串操作方面，我们利用了Python的re模块。re模块为我们提供了正则表达式的功能，使得我们可以方便地进行复杂的字符串匹配、查找和替换操作。这对于处理文本数据和提取特定模式的信息非常有用。此外，在数据库设计方面，我们使用了PowerDesigner软件帮助我们进行数据库的建模、设计和管理。通过PowerDesigner，我们能够更好地管理数据库设计流程，确保数据库结构的合理性和一致性。

# 课程设计的系统框架

利用图描述整个系统的框架，可以看作是需求分析的进一步描述

同时利用文字给出粗略给出每个模块的功能介绍（可能不止一个图）

在这一部分中，我们将分成前端展示、前后端交互、数据库交互三个部分详细介绍我们实现的整个系统。 整个系统的流程如下图所示。

前端展示(600字)

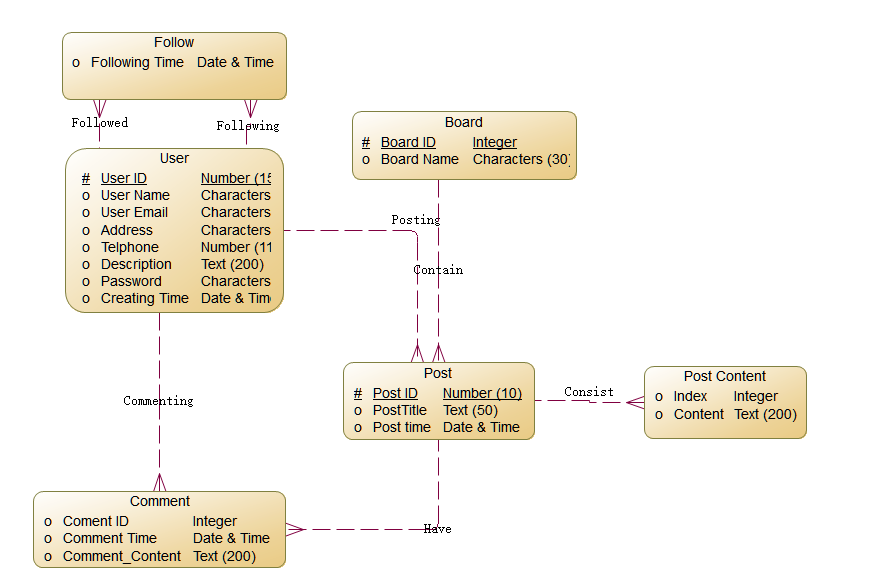
前后端交互(400字)

数据库交互(200字)

# 课程设计对应的关系数据库模式

## ER图

概念模式设计，并给出一定的解释，这个如果用powerdesginer，可以直接从powerdesginer拷贝那个设计图

我们使用PowerDesigner软件管理数据库设计流程。首先，我们使用PowerDesigner设计E-R图，如下图所示。

这个E-R图描述了一个简单的论坛管理系统。在该系统中，存在六个主要实体：用户（User）、版面（Board）、帖子（Post）、帖子内容（Post Content）、评论（Comment）以及关注关系（Follow）。用户实体包含了用户的基本信息如地址、电话和创建时间等等，同时还存储了用户用来登陆论坛系统的用户名（User Name）和密码（Password）等信息，每一个用户由User ID唯一标识。版面（Board）实体包含了主键（Board ID）和版面名。帖子（Post）实体包含了主键（Post ID）和帖子的标题以及发帖的时间。为了处理有些帖子长度过大的情况，我们没有简单的将帖子的内容作为一个属性存在帖子实体中。我们如果采用设置上限的方法处理帖子长度过大的情况，那么就会出现如果上限太大会极度浪费空间，上限太小又不太合理的矛盾。因此，我们另外建立了一张表来存储帖子的内容，我们将帖子的内容切分成若干个小段，每一段至多200字，然后按照顺序给每一段一个索引，表中的每一行只要存储索引和对应段的内容即可。因此，帖子内容（Post Content）实体包含索引和对应段的内容两个属性。评论（Comment）实体包含评论号（Comment ID），评论时间和评论内容三个属性。其中，评论号标识这个评论是在某一个帖子下的第个回复。关注关系（Follow）实体仅仅包含关注时间一个属性。

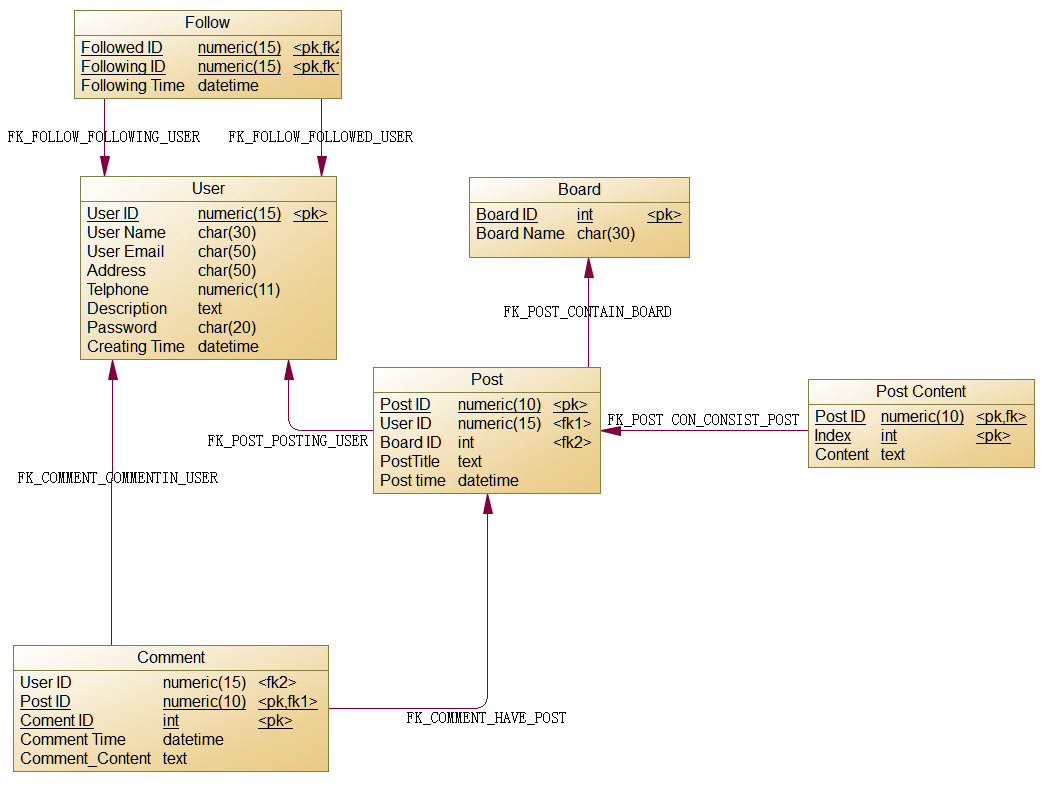
在这张E-R图中，我们还标识了不同实体之间的关系。一个用户可以发布多个帖子和多个评论，因此用户和帖子之间，用户和评论之间都存在一对多的关系。一个版面可以包含多个帖子，因此用户和帖子之间存在一对多的关系。一个帖子的内容被截成了很多段，同时也会有多个评论，因此帖子和内容之间，帖子和评论之间都存在一对多的关系。一个用户可以在不同时间关注他人，也可以在不同时间被他人关注，因此用户和关注关系之间存在着两个一对多的关系。

## 关系数据库模式描述

物理模式设计，至少包含一个诸如powerdesginer之类的工具产生的整个关系模式图，能够看到外键连接

说明每个表的主键，并说明这个表的设计达到了第几范式的要求

还包括视图设计、索引设计，说明是否采用了采用Stored procedure, Trigger等操作，

基于上述的E-R图，我们使用PowerDesigner软件自动生成了物理模式图，并且，我们对这个物理模式图稍作修改就得到了我们最终的物理模式图，如图所示，图中包含了每张表的属性，并标识了主键和外键。我们设计的物理模式图达到了4NF的范式要求，不存在非主属性之间的多值依赖。

在论坛系统实现中，我们需要根据用户名去查询用户的信息，因此，为了加快查询速度，我们在表User中单独为用户名（User Name）创建了非聚簇索引。为了保障信息的安全，我们对SQL注入攻击进行了防范

## 关系数据库模式之上的操作

针对我们在第三部分中提到的各种功能要求，利用关系数据库的SQL表达相关的核心实现，比如说相对复杂的SQL。

# 总结

项目总结和个人收获

References:

1. Ashish Kumar Gupta, Dan Suciu: Stream Processing of XPath Queries with Predicates. In: Alon Y. Halevy, Zachary G. Ives, AnHai Doan., eds Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data, San Diego, California, USA, June 9-12, ACM, 2003, 419-430

字数要求在3000字以上

1. [↑](#footnote-ref-1)