# 产品发布系统实现文档

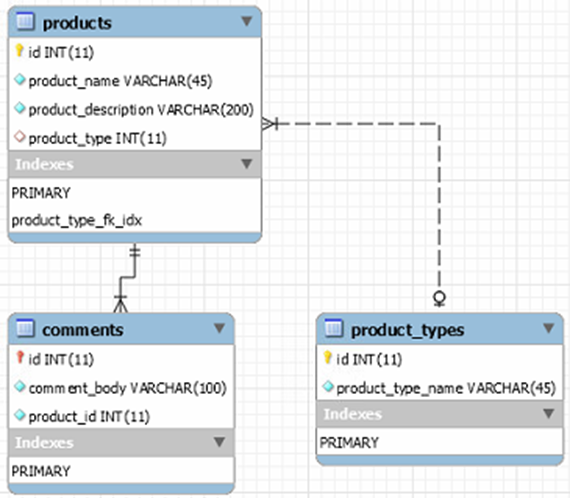
1. 系统中的实体

products(id, product\_name, product\_description, product\_type)

comments(id, comment\_body, product\_id)

product\_types(id, product\_type\_name)

1. 实体关系的类图



1. 需求分析
2. 实体关系需求

一个产品可有多个评论

一个评论只能属于一个产品

一个产品对应一种产品类型

1. 实体操作需求

创建产品（对名称，产品描述的字数限制）

显示单个产品（同时显示相关评论）

列出所有产品（可按类型筛选，可分页）

编辑产品

删除产品（同时删除相关评论）

为产品添加评论（评论字数限制）

删除评论

1. 功能实现
2. 创建产品并有字数限制

在models中的product.rb修改如下：

class Product < ActiveRecord::Base

validates :product\_name, :product\_description, presence: true 字数不为空

validates :product\_name, length: { maximum: 15 } 字数最少为15

validates :product\_description, length: { maximum: 100 } 字数最多为100

validates :product\_name, uniqueness: true 产品名称唯一

has\_many :comments

belongs\_to :product\_type

end

实现结果：



为产品添加评论（评论字数限制）这个类似，后面不在赘述。

1. 显示单个产品（同时显示相关评论）

在products的show.html.erb中加入comment, 并修改commen\_controller的def\_create部分

<h3>评论</h3>

<div id="comments">

<%= render :partial => @product.comments %>

</div>

<%= form\_for [@product, Comment.new] do |f| %>

<p>

<%= f.label :content, "新评论" %> <br />

<%= f.text\_area :comment\_body, :rows => 5 %>

</p>

<p><%= f.submit "添加评论" %></p>

<% end %>

实现结果：



1. 列出所有产品（可按类型筛选，可分页）

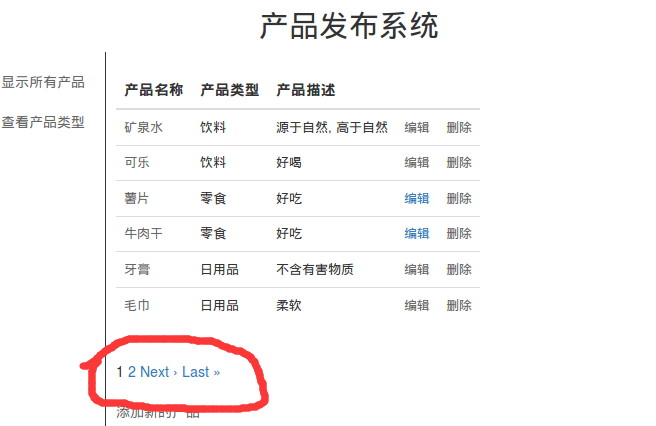
加入新实体：类型，并引入gem 'kaminari'达到分页的效果

实现结果：

分类查看：



分页查看：



1. 删除评论

将comment中的\_comment.html.erb文件修改如下：

<%= div\_for comment do %>

<p>

<strong>Posted

<%= time\_ago\_in\_words(comment.created\_at) %>

</strong><br />

<%= comment.comment\_body %>

<%= link\_to "删除该条评论", [@product, comment], :method => :delete, data: {:confirm => "Are you sure?"} %>

</p>

<% end %>

并修改comment\_controller的def\_destroy部分

实现结果：



其他自带简单的功能就不再赘述了。

1. 课程及作业感想

老实说，当时报ruby就是听学长说这门课简单，好过，也就是想捞个学分。没想到安装ruby和rails就出现了很多的问题，我当时还没装ubuntu的虚拟机，我觉得麻烦，所以不听沃老师的劝，在windows下安装ruby和rails，结果一遍遍重做，各种奇怪的错误，只好退下阵来，改用ubuntu虚拟机，现在想起来，我真是不撞南墙，不回头，结果花了时间，还没得到应有的训练。

做大作业是在考完最后一门后才开始做的，期末重新回顾课程的知识也是非常好的，一遍遍看着ppt，做着自己的系统，感觉也是非常美妙。Rails敏捷开发的确是非常好用的工具，能很短时间就完成一个简单的web app开发，但可能我对其生成的文件还不是很熟悉调用关系，所以每次出现百度都解决不了的问题时，我就果断重做，这个还是耗费了我不少的时间，也在这个过程中逐渐了解了rails的开发，希望以后还有机会用到并不断精进自己的技术吧，感谢沃老师的悉心教授，感谢助教的认真回复解答。Ruby课程虽已结束，但我感觉，以后有好多地方都可以用到它，所以我会继续努力加油！