2014-04-21 更新格式

2014-04-21 添加大纲

2014-04-21 修改数据库部分大纲结构，添加少量描述性文字

2014-04-26修改大纲结构，参照已有论文

2014-05-04 注意商品管理部分商品类别，供货管理，供应商管理，配送订单部分

网站待完善部分:

购买确认和完成页

商品详情页

收藏页

admin

文档:

图。。。。。

绪论一般分为背景与意义, 即对选题作一个简单的需求分析, 指出工作的价值所在。

系统分析与设计

需求分析(功能模块图，用例图，DFD图，）

数据库设计（ER模型设计 - 10， 数据库逻辑设计--表结构，）

软硬件环境设计（硬件，软件，环境，介绍）

网站具体实现（贴网页，文字）

开发环境

技术

技术

综合实现部分

系统测试

电子商务网站开发Fruit Home

摘 要

中文摘要分为两部分 (两个段落即可 ), 首先介绍背景与意义, 其次, 介绍本文所完成的工作。中文摘要写完后, 即可撰写对应的英文摘要。中文摘要的内容只需对结论中的前两部分适当修改即可。

摘要、绪论与结论是毕业论文的脸面, 必须认真写好。这三部分都有重复的地方, 因为都有对正文的概括, 只是语气不同, 篇幅不同。本科论文应该尽量避免使用“研究”二字, 一般使用“设计”与“开发”即可。

关键字：关键词1 关键词2

E-commerce website FruitHome

Abstract

Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract Abstract

**Key words:** keyword1 keyword2

目 录

[第一章 序 言 5](#_Toc387070837)

[第一节 电子商务发展现状 5](#_Toc387070838)

[第二节 水果电商网站概况 5](#_Toc387070839)

[第三节 网上购物系统技术 5](#_Toc387070840)

[第二章 系统需求分析 5](#_Toc387070841)

[第一节 系统功能需求 5](#_Toc387070842)

[第二节 系统数据流图（DFD图） 5](#_Toc387070843)

[1. 顶层数据流图（DFD0） 5](#_Toc387070844)

[2. 前台系统数据流图（DFD1） 5](#_Toc387070845)

[3. 后台系统数据流图（DFD1） 5](#_Toc387070846)

[第三节 业务流程图 5](#_Toc387070847)

[1. 前台业务流程图 5](#_Toc387070848)

[2. 后台业务流程图 5](#_Toc387070849)

[第三章 系统总体设计 5](#_Toc387070850)

[第一节 系统开发环境设计 5](#_Toc387070851)

[1. RoR框架 5](#_Toc387070852)

[2. Ruby 5](#_Toc387070853)

[3. WEBrick 5](#_Toc387070854)

[4. SQLite 5](#_Toc387070855)

[5. Notepad++ 5](#_Toc387070856)

[第二节 系统功能模块设计 5](#_Toc387070857)

[1. 前台功能 5](#_Toc387070858)

[2. 后台功能 5](#_Toc387070859)

[第四章 数据库设计 5](#_Toc387070860)

[第一节 数据库概念模型设计 5](#_Toc387070861)

[第二节 数据库物理模型设计 5](#_Toc387070862)

[1. 数据库表（实体） 5](#_Toc387070863)

[2. 数据库结构（关系） 5](#_Toc387070864)

[第五章 系统实现设计 5](#_Toc387070865)

[第一节 数据库链接 5](#_Toc387070866)

[第二节 前台功能模块 5](#_Toc387070867)

[1. 首页设计 5](#_Toc387070868)

[2. 注册登录模块 5](#_Toc387070869)

[3. 修改会员个人资料 5](#_Toc387070870)

[4. 商品查询 5](#_Toc387070871)

[5. 购物车实现 5](#_Toc387070872)

[6. 订单的维护管理 5](#_Toc387070873)

[第三节 后台功能模块 5](#_Toc387070874)

[第六章 系统调试 5](#_Toc387070875)

[1. 链接测试 5](#_Toc387070876)

[2. 检查布局的一致性 5](#_Toc387070877)

[3. 测试功能实现情况 5](#_Toc387070878)

[第七章 系统开发小节和建议，结论 5](#_Toc387070879)

[资料来源和参考文献 5](#_Toc387070880)

[致 谢 5](#_Toc387070881)

[附 录 5](#_Toc387070882)

第一章 序 言

第一节 电子商务发展现状

随着计算机不断地深入各行业，互联网已经走进了千家万户，也渐渐进入了社会的各个领域之中，并在不同程度上改变了人们的生活习惯。越来越多的顾客选择在网站上购物，从家居家装到日用百货，从家电3C到锅碗瓢盆，从服装鞋包到图书资料，网上商城在逐渐地替代传统的商场店铺，电子商务将获得非常广阔的发展前景。所以企业必须认识到电子商务在市场竞争中对自身企业的影响以及电子商务本身的特点，并结合自身企业的实际特点，探索在电子商务背景下如何改进自身的发展模式， 从而推动企业的健康发展、持续发展。

电子商务是指在网络上以电子交易方式进行交易活动和相关服务的活动，是传统商业活动各环节的电子化、网络化。按照1997年在法国巴黎举行的世界电子商务会议对电子商务做的权威概念阐述：电子商务EC（Electronic Commerce）是指对整个贸易活动实现电子化。电子商务可以分为Business to Business（B2B），Business to Consumer（B2C），Consumer to Consumer（C2C），Business to Government（B2G），Consumer to Government（C2G）等多种模式。我国的电子商务始于1997年，从诞生之日起我国电子商务行业发展就非常迅猛，在2010年我国电子商务市场交易额已达4.5万亿元，到2011年交易总额再创新高，达到5.88万亿元，其发展势头真可谓历久弥新

此外，我国网名人数以及网购人数也在飞速增长，这预示着“互联网”已成为一种日益重要的经济载体，网购已成为现代人的一种生活方式，网购将逐渐成为主流的购物方式。

当前我国主流的电子商务网站有淘宝、京东等。淘宝网（[taobao.com](http://www.taobao.com/)）经过不断的发展壮大，目前已发展为我国最大的C2C电子商务网站，在B2C方面天猫商城（[tmall.com](http://www.tmall.com/)）的发展也成绩斐然，截止2013年3月31日的年度，淘宝网和天猫平台的交易额合计突破人民币10,000亿元。京东商城（[jd.com](http://www.jd.com/)）作为中国最大的自营式电商企业，其发展毫不逊色于淘宝，2013年营业额也已超过1000亿元。除此之外，我国还有一大批发展迅速的电子商务网站，例如当当网（[dangdang.com](http://www.dangdang.com/)）、拍拍网（[paipai.com](http://www.paipai.com/)）、聚美优品（[jumei.com](http://bj.jumei.com/)）、凡客诚品（[vancl.com](http://www.vancl.com/)）、苏宁易购（[suning.com](http://www.suning.com/)）等，其也拥有相当数量的客户群体。

随着电子商务的不断发展和成熟，其逐渐显现出全品类和专一化经营两种较成熟的模式。然而，经营蔬果等生鲜产品的电商却十分少。目前，经营得比较成功的专营果蔬类生鲜产品的网站包括天天果园、电果网和本来生活网等，兼营果蔬类的网站包括一号店、京东商城、中粮我买网等。其中天天果园网在页面布局上简洁大气，干净利落；在界面上表现方式相对柔和，制图精美；在功能上，基本功能齐全，并根据顾客和果蔬的一些特性开辟了一些特色功能，例如“人群”、“场合”、“原产地”等。本文，将参照该网站设计一个“蔬果之家”网站。

随着计算机网络的飞速普及， 电子商务也获得了非常广阔的发展前景，而在这样的发展背景下我国企业必须及时、积极地了解电子商务环境， 认识到电子商务对自身企业营销的影响以及电子商务背景下营销的特点， 这样才能够更加客观地把握好市场的变化， 进而结合自身企业的实际特点，探索在电子商务背景下如何改进自身的营销模式， 进而推动企业的健康发展、持续发展。

随着计算机技术的发展，生活水平的不断提高，人们已经不会仅仅满足于在商场内购买商品而越来越多地希望可以在网络上看到并获得更多的货物，电子商务网站就在这样的环境下孕育而生了。电子商务购物网站能够充分迎合大家的需求，为店铺管理员、顾客提供最方便的平台实现安全交易。它为顾客提供更多的购物选择，使店铺管理员可以在全国范围内销售产品，更符合大家的生活习惯，其快捷方便赢得了每一个人的喜爱。

当前电子商务发展情况上网人数是电子商务发展的重要基础， 影响着企业电子商务营销的效果。根据统计， 随着Internet的飞速发展， 上网人数也在大幅提高。1998年， 全球上网人数为1.13亿人； 2004年， 全球上网人数为8.33亿人； 2005年， 全球上网人数达到9.73亿人口；2007年， 全球上网人数达到12亿人；2012 年， 全球上网人数已经近25 亿人， 而这其中我国网民已经突破了5亿。与此同时， 国内的电子商务市场规模也随之在迅速地扩大。2004 年底， 我国电子商务市场规模已经达到了人民币3239亿元人民币： 2007 年，我国电子商务市场总体规模超过人民币17300亿元， 2012年， 我国电子商务市场交易额已经高达8.1 万亿元，预计2015年， 我国将成为全球最大的电子商务市场。

从图1可以看到， 我国电子商务市场的规模正在以一种飞速的状态进行扩张， 每年的增长率都在20%以上，2010年和2011年的增长率更是突破了30%， 由于当前电子商务的蓬勃发展，在未来的市场预计中， 也还普遍看好电子商务市场可以持续的高增长， 这也为更多的企业打了一针强心剂， 使它们更加相信， 企业的信息化和电子商务化是大势所趋。总体上来看， 可以说目前的电子商务行业整体正处于一个上升的阶段。

但是当我们仔细分析一下市场交易的种类时， 却能够发现在这些电子商务交易中， 各类企业通过电子商务，进行B2B交易的份额竟然占到了80%以上， 而面向普通消费者的“电子商务”只占了不到20%， 当然这里我们必须承认， 企业之间的电子商务通常会有更多的交易金额， 但是过于畸形的市场布局还是有可能会对电子商务的未来发展起到一定的制约作用。

在细分化的市场中， 必然有许多有创新、有品质的电子商务平台还没有发展起来。这一点完全可以通过网络购物的所占比重证明这一点。电子商务市场一般总是集中体现出极强的综合性， 对于专业性方面的要求比较少。这样的市场结构使得很多消费者的需求很难得到满足， 使不少电子商务平台在很大程度上存在市场定位不准的问题。

第二节 水果电商网站概况

我国既是水果生产大国也是水果消费大国， 对于传统水果零售业来说，水果市场之广阔, 消费潜力之巨大是不言而喻的。 目前我国水果电子商务从事零售的较少，传统水果零售子商务直接应用电商的极少， 而水果生产和贸易企业大都是以批发为主，零售为辅[1]。 现有的少数从事零售的水果电商在营销策略上存在不足， 共同问题是只经营进口高端水果和国内精品水果、 面向的也是高端客户。 作为水果零售电子商务销售网站，如何让网站真正活起来， 有效地吸引并留住占绝大部分市场的中端客户，使国产水果、尤其是苹果、梨等一般水果通过电子商务这种方式走进平常百姓家，是亟待解决的问题。

水果零售电子商务网站对水果供应链作了重构和整合。 随着生活节奏的加快、保健意识的提高、手机上网的普及， 大众对水果零售电子商务的接受程度将会是越来越深的。 预计不久的将来，老百姓将免却传统购买水果的自提之苦，享受到定期下订单便有专人定时把价廉物美、营养搭配好的水果送上门的好处

2005年，易果网就已经上线；2007年，上海大学毕业的 MBA李永锋和几个同学创立优果网；2009 年，王伟从外企辞职，创立天天果园；而这一年，从事外贸生意的丁谋也在广东成立了网上水果店百森。

果品行业发展电子商务可以扩大市场的时空范围，未来的发展可以加快加大果品市场的信息流、商流、资金流和物流，一定程度上突破销售不畅造成的行业发展瓶颈。其次是扩大出口的需要。目前我国水果出口连年大幅增长，进出口贸易顺差超过 10 亿美元。我们发展电子商务，就可以更好地实现国内和国际市场对接，促进果品行业的整体发展。第三是节约交易成本，扩大经营规模，促进企业增效和提高企业管理水平的需要。

目前,制约果品行业发展电子商务的难点主要是：水果是鲜活

农产品,保鲜比较困难；外观和内在质量、口感、风味差别大,标准体系缺失；物流配送制约, 农产品冷链保鲜产业是产后农业重要一环，对进一步延伸现代农业产业链，促进农民大幅增收具有重要意义。但由于当前我国果蔬产后保鲜、冷链物流产业应用范围狭窄、发展滞后，每年的果菜的损失浪费率仍然十分惊人。果品生产的组织化程度低,仍然是以千家万户分散的方式为主,农村电脑拥有率低；企业对发展电子商务认识不够,在资金、技术、人员存在实际困难,畏难情绪普遍。

第三节 网上购物系统技术

电子商务网站一般分为两个部分：网站前台和后台，网站前后台又包含多个功能模块。前台系统一般包含商品浏览、订单生成、个人信息维护等模块；后台系统一般是管理员的管理功能模块，根据网站功能需求定义的不同后台可包含不同的功能模块，如主要需求在销售的管理的后台系统，则其功能模块都应围绕销售数据进行设计，如主要需求在对进销存的管理的后台系统，则后台系统除了销售管理模块外，还应包含库存、采购、财务等模块。

对于一个电子商务网站而言，其前后台系统，各自功能不同，目的不同，侧重点不同，都是不可或缺的。然而，前台系统是向用户直接展示商品的途径，也是企业展示企业形象的窗口，所以一个网站的前台的设计好坏，将在很大程度上影响用户对网站的喜爱程度。因此，在这个层面上来说，前台系统比后台系统更为重要。所以，对电子商务网站前台系统的最基本要求就是必须保证信息和服务质量的高水准，并在此基础上简化用户的购物流程，提高用户的控制程度。

本次开发将使用B/S结构，即Browser和Server浏览器和服务器结构。B/S结构是WEB兴起后的一种网络结构模式，WEB浏览器是客户端最主要的应用软件。在此架构下，用户通过浏览器来访问使用系统，少部分事务逻辑在前端（Browser）实现，主要事务逻辑在服务器端（Server）实现，形成通常意义上的三层架构。这一架构有利于将负载压力从客户端转移到服务器端，从而用户使用系统的成本，同时也减少了系统维护和升级的成本和工作量。

第二章 系统需求分析

第一节 系统功能需求

根据一些电子商务网站的考察和研究，可以将电商网站划分为前台展示系统和后台管理系统两大部分。

1. 前台功能模块

前台功能模块包括个人中心模块、商品展示模块、留言评价模块、购物车模块以及注册、登录模块。

用户可在个人中心模块对个人信息进行管理，也查看历史订单等信息。商品展示模块主要用于搜索、查询商品，展示商品的详细信息，包括与该商品相关的留言评价信息、历史订单等。用户在查看商品信息时，可对商品留言提问；找到中意商品后，可选择需要的数量加入购物车；订单完成时，可对商品进行评价。前台功能模块主要由新闻中心模块, 商品展示模块, 推荐商品模块, 商品分类模块, 我的购物车模块, 用户中心模块, 会员登录模块, 商品认证模块等组成。

表2.2.1 网上水果销售系统前台功能

|  |  |
| --- | --- |
| 1.1个人中心 | 注册、登录；个人信息更新；修改密码；找回密码；修改安全问题；历史订单查询； |
| 1.2商品展示 | 商品查询、搜索；商品信息信息查看； |
| 1.3留言、评价 | 留言提问；评价商品； |
| 1.4购物车 | 加入购物车；生成订单； |

2. 后台功能模块

后台管理模块主要提供对前台信息的添加、修改、删除等管理操作，主要包括会员管理模块，商品管理模块，订单管理模块，留言、评论管理模块，其中商品管理还包含商品类别管理和供货管理等子模块。管理员课在商品管理模块对商品信息进行维护，包括对商品进行分类等，也可对存货较低的商品进行补货操作。供应商的相关信息由系统管理员代为维护。

表2.2.2 网上水果销售系统后台功能

|  |  |
| --- | --- |
| 2.1会员管理 | 查看会员信息；修改会员状态； |
| 2.2商品管理 | 添加商品；更新商品信息；删除商品；添加供货信息； |
| 2.2.1商品类别管理 | 添加商品类别；更新商品类别信息；删除商品类别； |
| 2.2.2供货管理 | 查看供货信息；新增供货信息；修改供货状态； |
| 2.3订单管理 | 查看订单；处理订单；配送订单； |
| 2.4留言、评论管理 | 查看留言信息；回复留言；审核留言 |

第二节 系统数据流图（DFD图）

1. 顶层数据流图（DFD0）

网上水果销售系统包含3个实体：会员，管理员，供应商。



图2.2.1 网上水果销售系统顶层数据流图

2. 前台系统数据流图（DFD1）

会员用户在前台系统主要进行购物操作，也可对个人信息进行维护更新。其中购物操作包括查看商品信息，留言提问，加入购物车，生成订单等步骤。所涉及的数据库表有：会员信息表、订单信息表、商品信息表、留言信息表、评论信息表、购物车信息表。



图2.2.2 用户前台模块 一层数据流图

2.1 个人中心数管理据流图

在会员个人信息维护模块，用户可进行注册/登录操作，登录可以维护用户的个人账户信息，包括用户名、密码、密码安全问题等信息，也可查询用户相关的历史订单、留言、评论等信息。用户可以在正确回答密码安全问题之后可以重置密码，并填写新密码。所涉及的数据库表有：会员信息表、订单信息表、留言信息表、评论信息表。



图2.2.2.1 个人信息维护 二层数据流图

2.2 购物管理数据流图

在会员购物管理模块中，会员用户可以根据关键字或分类搜索查看商品，浏览商品详细信息，包括评论、留言提问等信息，也可对商品发起留言提问。用户选定商品加入购物车，核对商品信息无误后即可生成订单。其中未注册会员的用户（游客）只能进行搜索商品和查看商品详情信息，注册并登陆方可使用全部功能。

此模块相关的表有商品信息表、订单信息表、留言信息表、购物车信息表。

图2.2.2.2 注册会员购物管理 二层数据流图

2.3 订单管理数据流图

在订单管理模块中，用户确认购物车中的商品和数量后，可以新建订单。用户也可以查看当前和历史的订单信息。

此模块涉及的数据表有购物车信息表、订单信息表。



图2.2.2.3 订单管理 二层数据流图

3. 后台系统数据流图（DFD1）

后台系统包含五个子模块：会员管理、订单管理、商品管理、留言管理和供货管理模块，此模块涉及的数据库表有：会员信息表、订单信息表、商品信息表、留言信息表、供应商信息表和供货信息表。



图2.2.3 后台管理员模块 一层数据流图

3.1 会员管理数据流图

在会员管理模块，管理员可以对所有会员的信息进行查改，包括查看会员信息和修改会员的注册状态。

此模块相关的数据表为会员信息表



图2.2.3.1 会员管理模块 二层数据流图

3.2 订单管理数据流图

在订单管理模块中，管理员可以查看订单信息和处理订单，具体包括审核订单、修改订单状态等操作。管理员确认订单后，应及时联系快递公司进行发货。

此模块相关的数据表为订单信息表。



图2.2.3.2 订单管理模块 二层数据流图

3.3 商品管理数据流图

在商品管理模块中，管理员可以查看商品信息、更新商品信息、新增商品、删除商品、添加商品供货信息。

此模块相关的数据表有商品信息表、供货信息表。



图2.2.3.3 商品管理模块 二层数据流图

3.3.1 商品供货管理数据流图

在商品供货管理模块，管理员可以查看商品供货信息、新增商品供货、删除商品、修改供货状态。

此模块相关的数据表为供货信息表和供应商信息表。



图2.2.3.3-1 商品供货管理模块 三层数据流图

3.4 留言管理数据流图

在留言管理模块，管理员可以对留言的信息进行处理，包括查看留言信息、审核留言和回复留言。

此模块相关的数据表为留言信息表。



图2.2.3.4 留言管理模块 二层数据流图

第三节 业务流程图

1. 前台业务流程图

系统前台的目标用户为网站注册会员和未注册的潜在顾客。

顾客购物流程为：

1). 顾客登录系统，查看商品信息并选购，过程中顾客可对商品进行留言提问，由管理员进行回复;

2). 顾客将选中的商品加入购物车中，在购物车页面可调整商品数量，或者删除某些商品，完成后点击“去结算”进入订单生成页面检查并输入订单详细，如收货人、收货地址、邮政编码等信息，并可再次确认订单中商品信息；

3). 管理员审核订单后，改订单的商品即进入发货状态，由管理员联系快递进行发货处理；

4). 管理员确认送货完成后，修改订单中的状态字段，之后，顾客可对此订单的商品进行评价；

5). 顾客确认货物送达并评价后，可完成此订单。

在此过程如出现其他状况，管理员可将此订单置于关闭状态以关闭此订单。

用户完成操作后可退出系统。



图2.3.1 网上水果销售系统前台业务流程图

2. 后台业务流程图

系统前台的目标用户为网站管理员。

管理员登录后台系统后可对商品、会员、订单、留言等信息经管理。

在商品管理模块，管理员可对商品进行添加、查询、删除、更新等操作。此外，管理员还可以手工设置热销商品，给商品设置类别等，管理员也需要注意商品的库存信息，估计商品需求，适时新建商品供货信息，以增加存货。

在商品类别管理模块，管理员的操作和流程均类似商品管理模块，需要注意的是商品类别分为大小类或父子类，应注意区分。

在供货管理模块，管理员

在会员管理模块，管理员可查看所有用户信息，确认用户留言和评价记录存在不当言论时，可对用户进行状态处理，或屏蔽，或关闭，或注销。管理员也可以查看注册会员的详细信息。

在订单管理模块，管理员可以按状态分类查看订单信息，并处理订单。管理员也可以根据情况更新订单信息，如订单状态更新、收货地址修改等。

在留言管理模块，管理员可查看所有用户的留言信息，并可回复留言，审核留言。

管理员完成操作后可退出系统。



图2.3.1 网上水果销售系统后台业务流程图

第三章 系统总体设计

第一节 系统开发环境设计

系统开发整体方案包括：

* 开发环境：windows 7，WEBrick 1.3.1，Chrome 34.0.1847.131 m
* 数据库：SQLite，Navicat for SQLite
* 开发工具：Notepad++ v6.5.5
* 开发语言：Ruby，JavaScript

本系统使用RoR框架（Ruby on Rails）作为开发平台，基于MVC框架进行开发，主要开发语言为Ruby；使用的数据库是SQLite，并将Navicat for SQLite作为数据数据库管理工具；主要开发工具为Notepad++ v6.5.5，同时辅助使用FastStone Capture、Chrome 34.0.1847.131 m等软件作为网页设计工。

1. RoR框架

RoR是Ruby on Rails的缩写，括两部分内容：Ruby语言和Rails框架。Rails框架在2004年7月首次提出，其研发者是丹麦人David Heinemeier Hansson。RoR是一个使用Ruby写的开源Web应用框架，它是严格按照MVC结构开发的，是一个更符合实际需要而且更高效的Web开发框架。它努力使自身保持自然、简洁，来使实际的应用开发时的代码更少，使开发速度风快，使用最少的配置，给程序开发人员提供强大的框架支持。RoR的数千个程序已经在线上得到应用，现在使用RoR的网站有Basecamp，Twitter，GitHub，Groupon，Shopify，Yellow Pages等。



图3.1.1 用户、浏览器和MVC

1). Rails的设计原则

a). DRY（Don't Repeat Yourself）：“不做重复的事”

这一原则如果正确实施的话，将会减少项目中大量的重复性工。项目中的任何一种重复性工作都会导致更新、维护的困难，也会使得代码前后不一致。在RoR,你可以看到这一原则在几乎所有的组件中应用，从插件形式的可重用组件，到数据库表的映射方式

b). CoC（Convention Over Configuration）：“惯例优于设置”

配置已经接管了Web应用程序框架这么多，以至于连一个简单的任务，如为仅仅一个字段配置“必须填写”验证也需要访问一个XML文件中的条目。在RoR，其原则是只提供与通常的应用程序设置不同的方面的信息。由RoR所提供的ORM（对象关系映射）框架就是一个应用这一原则例子。除非你为一个ORM对象指定一个不同的名称，否则该对象使用的表名就是它所映射的表名称。而在基于配置的ORM框架下，如Hibernate，每个表连同其列的映射必须在配置文件中给出。因此，架构的改变意味着在配置文件要有相应的变化。但是，在RoR，模式的改变并不意味着对象也需要改变，除非表本身的名称更改。

2). Rails的特点。

* Rails 是一个全栈式的MVC 框架，只一个Rails就可以实现MVC 模式中的各个层次，并使它们无缝地协同运转起来；
* Rails是约定优于配置的（CoC）， Rails使用Web应用多年来积累的各种常见约定（即命名规则）代替XML 配置文件，Rails内部的映射与发现机制根据这些约定可以实现对象之间的关联。
* Rails使用更少的代码来实现应用程序，因为CoC说明Rails本身完成了大量的底层工作，此外，代码量的缩减也减小了出现bug 的几率，降低了维护程序和升级程序的难度。
* Rails具有生成器功能，Rails 使用的实时映射技术和元编程技术，免去了开发人员在开发过程中编写大量样板文件代码的烦恼。在少数需要使用样板文件代码的时候，开发人员可以通过Rails 内建的生成器脚本实时创建，而不再是通过手工编写。Rails 的这个特点可以使开发人员更专注于系统的逻辑结构，而不必为一些琐碎的细节所烦扰。
* 零周转时间：对已有的 Web 应用系统进行修改后，其一般需要经过配置、编译、发布、重新设置、测试等一系列步骤才能投入使用，这明显浪费了许多时间。而使用Rails 开发Web 应用系统，可以通过浏览器即时查看程序运行结果，从而节约了大量的时间。
* 支架系统：Rails 的支架系统可以自动为任何相关的数据库表创建一套包含标准CRUD操作和前台视图的系统。通过支架系统，开发人员可以方便快捷地操纵数据库中的数据表。此外，Rails 也允许开发人员使用自己设计的代码或视图来替换自动生成的代码和视图。

2. Ruby

Ruby，在20世纪90年代中期由日本人松本行弘设计并开发。他开发Ruby的初衷是为想要一个提高编程效率的脚本语言，这也就意味着Ruby的特点将优化软件开发的方式。Ruby是一种面向对象、命令式、函数式、动态的通用编程语言，其灵感与特性来自于 Perl、Smalltalk、Eiffel、Ada 以及 Lisp 语言。2004年，RoR框架的诞生使Ruby更加广为人知。根据TIOBE编程社区指数，Ruby是发展最快的语言之一，并于2006年被选为年度编程语言。

Ruby的特点：

* 完全面向对象：Ruby中任何东西都是对象，甚至包括基本数据类型；
* 动态类型：Ruby在程序运行时通过查看变量值的类型来决定变量的类型，因此变量本身可以保存任何类型的数据；
* 任何东西都有值：Ruby中的数学表达式或者逻辑表达式，甚至一个语句，都会有值；
* 变量无需声明
* 函数式编程：Ruby使用代码块来实现函数式编程；
* 开源项目。有大量活跃的社区支持Ruby 语言。
* 动态载入。Ruby是一种动态语言可以在运行时候重定义自己，类也可以在运行时继承或取消继承，Ruby甚至支持对运算符等内置方法进行重载；
* 解释型脚本语言：Ruby是解释型脚本语言，它既有脚本语言强大的字符串处理能力和正则表达式，又不失解释型语言的动态性的快速性。
* 强大的反射机制与元编程：Ruby的反射功能相当惊人，甚至可以自行追踪程序运作，或是取出private变量、拦截方法的调用。 常常与‘可以动态的修改对象’这项特色结合，做为‘元编程’的功能：程序在运行时， 可以由程序员提供的信息，自行生成、修改类型或对象，这项功能大大的提高了撰写代码的效率。 在RoR之中，就大量使用了这种特性。程序员可以更专心处理程序的运作，而不必为每个类型重复得撰写代码。
* Reflection: Ruby provides a program with the ability to 'look at itself' while running. This ability is known by different terms, such as reflection, introspection, and so on. Using reflection, a program can modify certain aspects of itself during execution, or create a completely new object at runtime based on the requirements at that time.

2.2 RoR 的发展现状与前景

目前，Rails 的最新版本是2005 年12 月13 日发布的v1.0.0。从RoR 正式提出到v1.0.0 的发布，RoR 在一年多的时间里受到了业内人士的广泛关注。RoR 受到广泛关注主要有如下两个原因：首先，RoR 的开发效率高（部署容易）、功能丰富（支持Ajax 等流行应用），有消息称对于相同的Web 开发项目，使用RoR 开发比使用Java 体系架构开发快5~10 倍；此外，令人不可思议的高性能是其受到关注的另一个重要原因，根据CSDN上转载的新闻称使用RoR开发出来的项目性能，比基于Struts+Hibernate+Spring 的Java 应用还要高15%~20%。

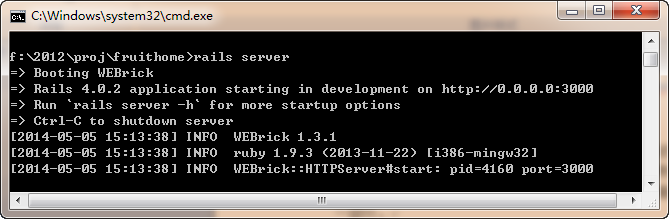
RoR 当前遇到的主要问题是使用RoR 搭建的大型商业应用还很少，究其原因可以概括为两点：第一，从开发能力的角度，RoR 是一个基于Ruby 语言的轻型Web 开发框架，很多开发者对其是否适合大型应用难以把握。第二，本身使用RoR开发的大型商业应用较少，使得后来者持观望态度。综合分析来看，RoR 的发展前景还是很光明的。RoR 在短时间内取得了巨大的成就，它打破了Web 开发领域的固有观念，方便快捷的开发方式使其被广泛接受。而事实上，现在已有几家跨国公司正在使用RoR 开发自己的Web 应用程序，并且有多家大型公司在考虑使用RoR 进行Web 应用开发。

3. WEBrick

WEBrick是一个提供简单HTTP网络服务器服务的Ruby库，它提供了HTTP，带身份验证的HTTPS，setup of Servlet with Ruby code等特性。WEBrick作为RoR中的web服务器，可用在开发环境中调试程序，也可用在生产环境中运行程序，现在是Ruby标准库的一部分。系统根目录执行rails server命令来启动WEBrick。

WEBrick is a Ruby library providing simple HTTP web server services. WEBrick was primarily written by Masayoshi Takahashi and Yuuzou Gotou, with contributions from other developers via the open source model of software development. The server provides HTTP, HTTPS with authentication, setup of Servlet with Ruby code, and other features.

It is used by the Ruby on Rails and Padrino frameworks to test applications in a development environment as well as production mode. It is now part of Ruby standard library for Ruby 1.9.3, and updated with Ruby trunk. The documentation is incomplete for some modules and methods.



4. SQLite

诞生于2000年的SQLite，是一款轻型数据库，它的设计目标是嵌入式的，而且目前已经在很多嵌入式产品中使用了它，它占用资源非常的低，在嵌入式设备中，可能只需要几百K的内存就够了。它支持Windows/Linux/Unix等主流的操作系统，同时能够跟很多程序语言相结合，比如 Tcl、C#、PHP、Java等，还有ODBC接口，同样比起Mysql、PostgreSQL这两款开源世界著名的数据库管理系统来讲，它的处理速度比他们都快。SQLite

不像常见的客户-服务器范例，SQLite引擎不是个程序与之通信的独立进程，而是连接到程序中成为它的一个主要部分。所以主要的通信协议是在编程语言内的直接API调用。这在消耗总量、延迟时间和整体简单性上有积极的作用。整个数据库(定义、表、索引和数据本身)都(都存储在主机端上一个单一的文件中，这种简洁的设计是通过在开始一个事务的时候锁定整个数据文件而完成的。

是遵守ACID的关系型数据库管理系统，程序库实现了多数的[SQL](http://zh.wikipedia.org/wiki/SQL)-92标准，包括[事务](http://zh.wikipedia.org/w/index.php?title=%E4%BA%8B%E5%8A%A1&action=edit&redlink=1)，就是代表[原子性](http://zh.wikipedia.org/w/index.php?title=%E5%8E%9F%E5%AD%90%E6%80%A7&action=edit&redlink=1)、[一致性](http://zh.wikipedia.org/wiki/%E4%B8%80%E8%87%B4%E6%80%A7)、[隔离性](http://zh.wikipedia.org/wiki/%E9%9A%94%E9%9B%A2%E6%80%A7)和[持久性](http://zh.wikipedia.org/w/index.php?title=%E6%8C%81%E4%B9%85%E6%80%A7&action=edit&redlink=1)的（[ACID](http://zh.wikipedia.org/wiki/ACID)），[触发器](http://zh.wikipedia.org/wiki/%E8%A7%A6%E5%8F%91%E5%99%A8)和多数的复杂查询。不进行[类型检查](http://zh.wikipedia.org/wiki/%E7%B1%BB%E5%9E%8B%E6%A3%80%E6%9F%A5)。你可以把[字符串](http://zh.wikipedia.org/wiki/%E5%AD%97%E7%AC%A6%E4%B8%B2)插入到[整数](http://zh.wikipedia.org/wiki/%E6%95%B4%E6%95%B0)列中。例如，某些用户发现这是使数据库更加有用的创新，特别是与无类型的脚本语言一起使用的时候。其他用户认为这是主要的缺点。

多个[进程](http://zh.wikipedia.org/wiki/%E8%BF%9B%E7%A8%8B)或[线程](http://zh.wikipedia.org/wiki/%E7%BA%BF%E7%A8%8B)可以同时访问同一个数据而没有问题。可以同时平行读取同一个数据库。但同一时间只能有一个进程或线程进行数据写入；否则会写入失败并得到一个错误信息（或者会自动重试一段时间，而这重试时间的长短是可以设置的）。

SQLite特点：

* ACID[事务](http://baike.baidu.com/view/121511.htm)
* 零配置 – 无需安装和管理配置
* [储存](http://baike.baidu.com/view/843566.htm)在单一[磁盘](http://baike.baidu.com/view/157418.htm)文件中的一个完整的数据库
* [数据库文件](http://baike.baidu.com/view/1800925.htm)可以在不同[字节顺序](http://baike.baidu.com/view/567601.htm)的机器间自由的共享
* 足够小, 大致13万行C代码, 4.43M
* 比一些流行的数据库在大部分普通数据库操作要快
* 简单, 轻松的API
* 包含TCL绑定, 同时通过Wrapper支持其他语言的绑定
* 良好注释的[源代码](http://baike.baidu.com/view/60376.htm), 并且有着90%以上的测试覆盖率
* 独立: 没有额外依赖
* 源码完全的开源, 你可以用于任何用途, 包括出售它
* 支持多种开发语言,C, PHP, Perl, Java, C#,Python, Ruby

5. Notepad++

Notepad++是一款Windows环境下免费开源的纯文本编辑器，由台湾人侯今吾基于同是开放源代码的Scintilla文本编辑组件独力研发，整个项目托管于SourceForge.net之上，此软件曾两度获得SourceForge社区选择奖——最佳开发工具。

它堪比Windows自带Notepad（记事本）加强版，除了基本的纯文本编辑之外，它十分适合当作大部分主流语言的代码编辑器。Notepad++支持有语法高亮度显示，语法折叠等功能，并且支持宏以及扩充基本功能的各类插件，通过插件的辅助，Notepad++可以成为一个堪比IDE的开发工具。

当然，Notepad++也有不足之处，很明显的就是与其他集成开发工具（如Visual Studio）相比，许多一键式的运行、调试功能都没有，并且错误提示不如VS直接，在遇到程序出错的时候需要到日志中去寻找错误原因。故而在程序运行之前，首先需要做大量的调试工作。

第二节 系统功能模块设计

1. 前台功能

系统前台的目标用户是网站注册会员和未注册的潜在顾客，系统的前台功能模块如下图：



图3.2.1 前台功能模块图

系统前台功能明细表

|  |  |
| --- | --- |
| 1.1个人信息管理 | 用户注册 |
| 用户登录 |
| 个人信息更新 |
| 修改密码 |
| 找回密码 |
| 修改安全问题 |
| 历史订单查询 |
| 1.2购物管理 | 商品查询（通过商品名称、分类、等相关信息进行模糊查询） |
| 商品概览信息查看 |
| 商品详细信息查看 |
| 发表评论和留言提问 |
| 1.3订单管理 | 购物车功能 |
| 新增订单 |

2. 后台功能

系统前台的目标用户是网站管理员，系统的后台功能模块如下图：

系统后台功能明细表

|  |  |
| --- | --- |
| 2.1会员管理 | 查看会员信息 |
| 修改会员状态 |
| 2.2订单管理 | 查看订单 |
| 处理订单 |
| 配送订单 |
| 2.3商品管理 | 添加商品 |
| 更新商品信息 |
| 删除商品 |
| 添加供货信息 |
| 2.3.1商品类别管理 | 添加商品类别 |
| 更新商品类别信息； |
| 删除商品类别 |
| 2.3.2供货管理 | 查看供货信息 |
| 新增供货信息 |
| 修改供货状态 |
| 2.4留言管理 | 查看留言信息 |
| 回复留言 |
| 审核留言 |

第四章 数据库设计

根据前述需求分析和总体设计，网上水果销售系统的数据库表应包括商品表、会员表、订单表、订单明细表、管理员表、留言表、购物车信息表、供应商表、供应明细信息表、

第一节 数据库概念模型设计

根据系统功能分析，区分系统的各种实体（用户，商品等。。。

实体：

|  |  |  |
| --- | --- | --- |
| **实体名** | **实体名** | **数据表名** |
| 用户（会员） | user | usres |
| 水果类别 | fruit\_type | fruit\_types |
| 水果 | fruit | fruits |
| 订单 | order | orders |
| 订单详情 | order\_item | order\_items |
| 评论 | comment | commens |
| 账户 | account | accounts |
| 邮递地址 | mail\_address | mail\_addresses |
| 供应商 | supplier | suppliers |
| 供应信息 | supply\_information | supply\_informations |
| 管理员用户 | admin\_user | admin\_users |
| 原产地 | origin\_place | origin\_places |
| 收藏 | collection | collections |
| 咨询 | consultation | consultations |
| 数据库版本 | schema\_migration | schema\_migrations |
| 数据库序列 | sqlite\_sequence | sqlite\_sequences |

实体图：

1.用户



2.水果类别





3.水果



4.订单



根据功能和需求分析，制定各个数据项

根据数据项和数据结构，设计数据实体（实体属性图），并根据其关系得出E-R图



第二节 数据库物理模型设计

1. 数据库表（实体）

根据系统需求，设计数据库表（SQLite数据库表数据）

1.用户表（会员）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 字段含义 | 数据类型 | 长度 | 约束 |
| id | 自动递增编号 | integer | 4 | PK |
| username | 用户名 | varchar | 2-20 | 大小写英文字母 |
| real\_name | 真实姓名 | varchar |  |  |
| sex | 性别 | varchar |  |  |
| birthdate | 出生日期 | varchar |  |  |
| e\_mail | 邮箱 | varchar |  |  |
| password\_text | 明文密码 | varchar |  |  |
| password\_digest | 加密密码 | varchar | 6-20 | 大小写英文和数字 |
| password\_comfirmation | 密码确认 | varchar |  |  |
| status | 用户状态 | integer |  | 为0则用户被屏蔽 |
| question | 找回密码问题 | varchar |  |  |
| answer | 找回密码答案 | varchar |  |  |
| phone\_number | 手机号码 | varchar |  |  |
| shopping\_exp | 经验值 | varchar |  |  |
| member\_type | 会员类别 | varchar |  |  |
| city | 城市 | varchar |  |  |
| post\_code | 邮政编码 | varchar |  |  |
| advance\_payment | 支付方式 | varchar |  |  |
| latest\_log\_date | 最后登录日期 | varchar |  |  |
| created\_at | 创建日期 | datetime |  |  |
| updated\_at | 更新日期 | datetime |  |  |

2.水果类别

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 字段含义 | 数据类型 | 长度 | 约束 |
| id | 自动递增编号 | integer | 4 | PK |
| name | 类别名称 | varchar |  |  |
| description | 类别描述 | varchar |  |  |
| super\_id | 父类别 | integer |  | FK |
| created\_at | 创建日期 | datetime |  |  |
| updated\_at | 更新日期 | datetime |  |  |

3.水果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 字段含义 | 数据类型 | 长度 | 约束 |
| id | 自动递增编号 | integer | 4 | PK |
| name | 水果名称 | varchar |  |  |
| introduction | 水果描述 | varchar |  |  |
| brief\_introduction | 水果简介 | varchar |  |  |
| fruit\_type\_id | 水果类别 | integer |  | FK |
| origin\_place\_id | 产地 | integer |  | FK |
| inventory | 库存 | integer |  |  |
| unit | 计价单位 | varchar |  |  |
| price\_present | 当前价格 | float |  |  |
| price\_history | 历史价格 | float |  |  |
| price\_other | 其他价格 | float |  |  |
| discount | 优惠 | float |  |  |
| is\_hot | 是否热销 | boolean |  |  |
| created\_at | 创建日期 | datetime |  |  |
| updated\_at | 更新日期 | datetime |  |  |

4.订单

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 字段含义 | 数据类型 | 长度 | 约束 |
| id | 自动递增编号 | integer | 4 | PK |
| user\_id | 所属用户 | integer |  | FK |
| status | 订单状态 | integer |  |  |
| created\_at | 创建日期 | datetime |  |  |
| updated\_at | 更新日期 | datetime |  |  |

5.订单详情

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 字段含义 | 数据类型 | 长度 | 约束 |
| id | 自动递增编号 | integer | 4 | PK |
| order\_id | 所属订单 | integer |  | FK |
| fruit\_id | 包含水果 | integer |  | FK |
| quantity | 水果数量 | integer |  |  |
| price | 水果价格 | floar |  |  |
| unit | 计量单位 | varchar |  |  |
| created\_at | 创建日期 | datetime |  |  |
| updated\_at | 更新日期 | datetime |  |  |

6.评论

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 字段含义 | 数据类型 | 长度 | 约束 |
| id | 自动递增编号 | integer | 4 | PK |
| fruit\_id | 所属水果 | integer |  | FK |
| user\_id | 所属用户 | integer |  | FK |
| content | 评论内容 | varchar |  |  |
| status | 评论状态 | integer |  |  |
| created\_at | 创建日期 | datetime |  |  |
| updated\_at | 更新日期 | datetime |  |  |

7.账户

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 字段含义 | 数据类型 | 长度 | 约束 |
| id | 自动递增编号 | integer | 4 | PK |
| user\_id | 所属用户 | integer |  | FK |
| type | 账户类型 | varchar |  |  |
| money | 余额 | float |  |  |
| created\_at | 创建日期 | datetime |  |  |
| updated\_at | 更新日期 | datetime |  |  |

8.邮递地址

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 字段含义 | 数据类型 | 长度 | 约束 |
| id | 自动递增编号 | integer | 4 | PK |
| user\_id | 所属用户 | integer | 20 | FK |
| address | 具体地址 | varchar | 20 |  |
| created\_at | 创建日期 | datetime |  |  |
| updated\_at | 更新日期 | datetime |  |  |

9.供应商

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 字段含义 | 数据类型 | 长度 | 约束 |
| id | 自动递增编号 | integer | 4 | PK |
| name | 供应商名称 | varchar |  |  |
| phone\_number | 联系方式 | varchar |  |  |
| e\_mail | 邮箱 | varchar |  |  |
| address | 具体地址 | varchar |  |  |
| city | 所在城市 | varchar |  |  |
| created\_at | 创建日期 | datetime |  |  |
| updated\_at | 更新日期 | datetime |  |  |

10.供应信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 字段含义 | 数据类型 | 长度 | 约束 |
| id | 自动递增编号 | integer | 4 | PK |
| fruit\_id | 供应的水果 | integer |  | FK |
| supplier-id | 提供的供应商 | integer |  | FK |
| quantity | 供应数量 | integer |  |  |
| status | 供应信息状态 | integer |  |  |
| price | 供应的单价 | float |  |  |
| mail\_cost | 运输费用 | float |  |  |
| total\_cost | 总费用 | float |  |  |
| start\_date | 开始日期 | varchar |  |  |
| end\_date | 结束日期 | varchar |  |  |
| created\_at | 创建日期 | datetime |  |  |
| updated\_at | 更新日期 | datetime |  |  |

11.管理员用户

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 字段含义 | 数据类型 | 长度 | 约束 |
| id | 自动递增编号 | integer | 4 | PK |
| username | 用户名 | varchar |  |  |
| password | 密码 | varchar |  |  |
| admin\_type | 管理员类型 | varchar |  |  |
| e\_mail | 邮箱 | varchar |  |  |
| created\_at | 创建日期 | datetime |  |  |
| updated\_at | 更新日期 | datetime |  |  |

12.原产地

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 字段含义 | 数据类型 | 长度 | 约束 |
| id | 自动递增编号 | integer | 4 | PK |
| name | 产地名 | varchar |  |  |
| description | 产地描述 | varchar |  |  |
| super\_id | 产地归属 | integer |  | FK |
| created\_at | 创建日期 | datetime |  |  |
| updated\_at | 更新日期 | datetime |  |  |

13.收藏

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 字段含义 | 数据类型 | 长度 | 约束 |
| id | 自动递增编号 | integer | 4 | PK |
| user\_id | 所属用户 | integer |  | FK |
| fruit\_id | 收藏水果 | integer |  | FK |
| created\_at | 创建日期 | datetime |  |  |
| updated\_at | 更新日期 | datetime |  |  |

14.咨询

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 字段含义 | 数据类型 | 长度 | 约束 |
| id | 自动递增编号 | integer | 4 | PK |
| fruit\_id | 咨询的水果 | integer |  | FK |
| user\_id | 咨询的用户 | integer |  | FK |
| status | 咨询的状态 | integer |  |  |
| question | 咨询的问题 | varchar |  |  |
| answer | 回复内容 | varchar |  |  |
| created\_at | 创建日期 | datetime |  |  |
| updated\_at | 更新日期 | datetime |  |  |

15.数据库版本

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 字段含义 | 数据类型 | 长度 | 约束 |
| version | 版本号 |  |  |  |

15.数据库序列

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名 | 字段含义 | 数据类型 | 长度 | 约束 |
| name | 表名 |  |  |  |
| seq | 当前序列值 |  |  |  |

2. 数据库结构（关系）

根据系统需求和数据库概念模型设计数据库物理模型（由Rails Model逻辑实现，便于移植）

第五章 系统实现设计

第一节 界面设计

电商网站应具有合理的版面设计、清晰的导航和有层次感的页面，同时简洁明快，力求呈现出商品本身的特性。本网站的设计参考了天天果园等网站，对网上水果销售系统进行用户和管理员的界面设计。天天果园的网站截图如下：



图5.1.1 天天果园首页

1.前台系统界面

网上水果销售网站前台系统的界面背景使用纯白色，网站颜色以淡绿色为主色调，辅以灰白色表格用以显示订单列表等信息。

前台系统分为注册/登录、商品浏览、个人信息浏览和网站其他信息浏览。

1.1注册/登录界面，用户通过界面顶部的导航条进行注册和登录连接可进入相应的操作界面。

1.2商品浏览界面，网站顶部是一些常用的导航功能条，这些导航条会一直存在，以方便用户使用。网站的用户导航条位于网站上方，包含登录、注册、个人中心、我的收藏夹、商品分类、客户服务等导航信息。用户导航条下方是网站LOGO、商品搜索栏和购物车按钮。再下方就是商品导航条，根据商品的分类、产地、事宜人群等条目进行导航，在大类条目上点击可显示小类的条目信息。

首页主要显示一个商品轮换展示的大图，和根据热门类别显示的一个商品。商品的具体信息查看页面除了展示商品的详细信息和展示图片之外，还将显示该商品相关的评价、留言和购买记录。

1.3个人账户信息浏览页面，上述的导航功能条仍位于页面上方，对个人账户信息查看、维护的链接网页左侧的菜单栏里。

1.4网站其他信息浏览导航位于页面最下方，用户点击相关链接即可转到相关信息的详情查看界面。

2.后台系统界面

后台系统界面的主色调使用灰白色，缀以淡绿色，呼应前台界面风格。

后台系统界面分为登录界面和管理模块界面。

2.1管理员登录界面

2.2在管理员管理界面的左侧将显示管理员的操作菜单，管理员可以点击相应的菜单选项进行管理操作。

第二节 前台界面实现

1. 首页设计



图5.2.1 网站首页

2. 注册/登录模块

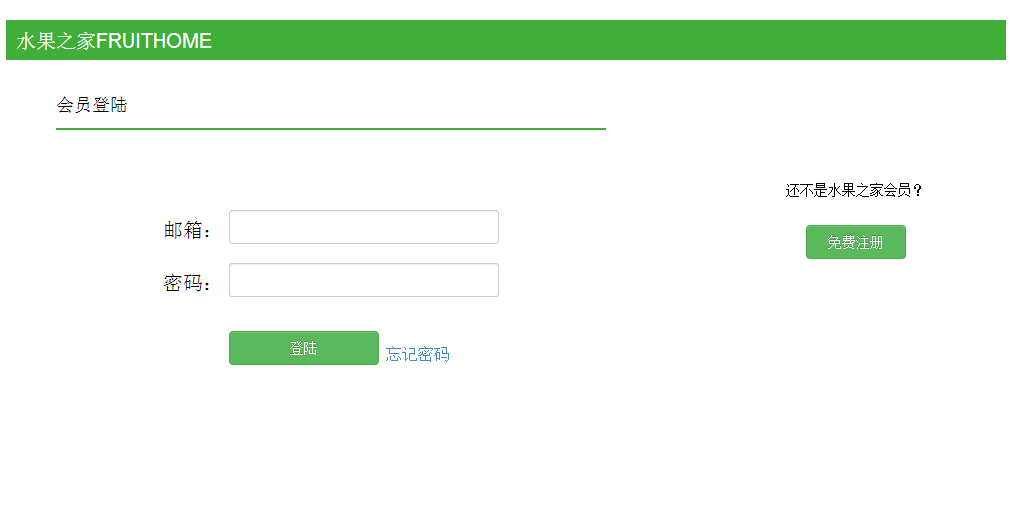


图5.2.2.1登录页面

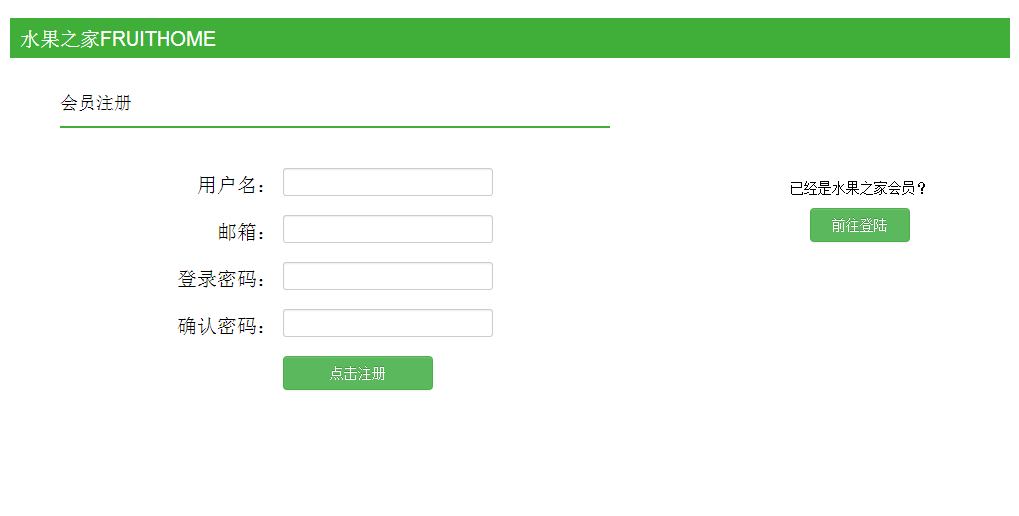


图5.2.2.2 注册页面

3. 修改会员个人资料

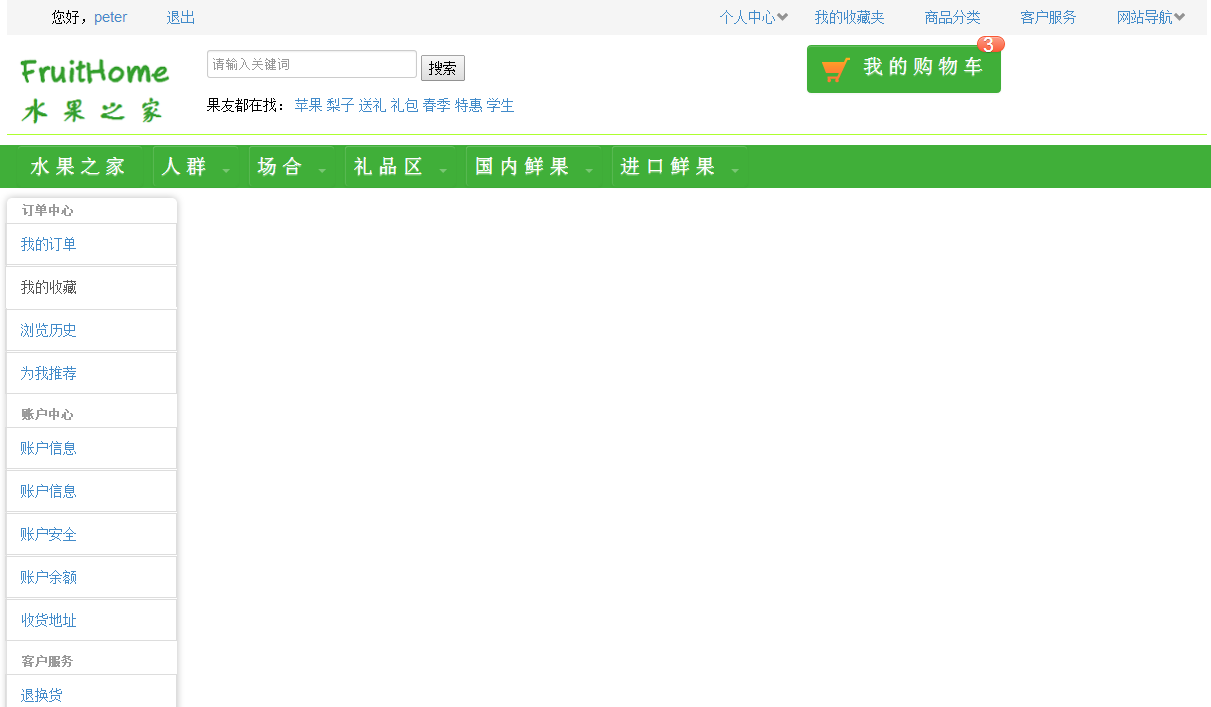


图5.2.3 个人信息管理页面

4. 商品查询

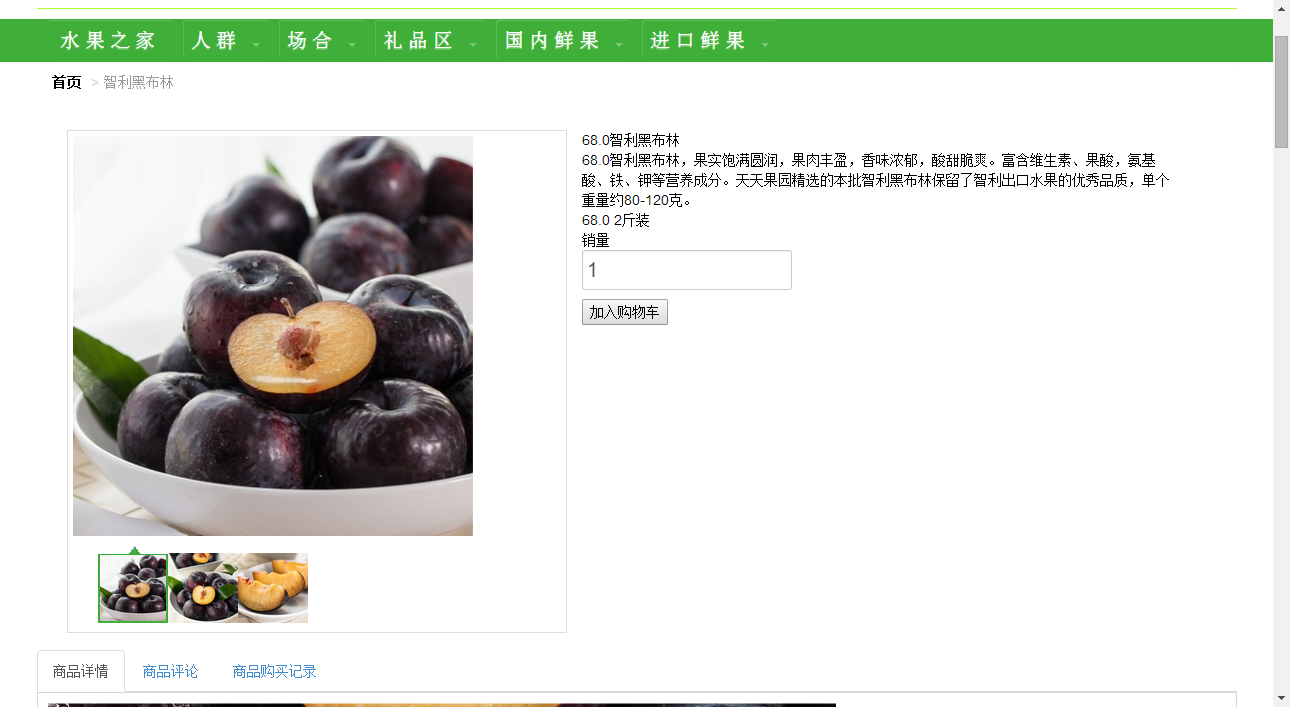


图 商品详情页



图 评论和卖出信息

5. 购物车实现



图 购物车页面

6. 订单的维护管理



图 订单信息

第三节 后台界面实现

1.添加商品



图 商品管理

2.添加类别

3.添加。。。

第四节 实现难点

1. 收集验证错误信息

在Rails早期版本中使用ActionRecordHelper类error\_messages\_for方法来收集数据库验证产生的错误信息，以及其他的一些错误信息。但在Rails3及其之后的版本中，由于Rails的“nothing in Rails Core should dictate the look and feel of an app.”的指导方针而从Rails核心中移除了这个方法。所以在参考相关资料后，本人编写了一个模仿error\_messages\_for的方法，并通过制定其显示的css加强了它的显示效果。

代码如下：

def errors\_for(object, message=nil)

html = ""

unless object.errors.blank?

html << "<div class=' errors #{object.class.name.humanize.downcase}\_errors'>\n"

if message.blank?

if object.new\_record?

html << "\t\t<h5>在创建#{object.class.name.humanize.downcase}时出错</h5>\n"

else

html << "\t\t<h5>在更新#{object.class.name.humanize.downcase}时出错</h5>\n"

end

else

html << "<h5>#{message}</h5>"

end

html << "\t\t<ul>\n"

object.errors.full\_messages.each do |error|

html << "\t\t\t<li>#{error.split(" ").last}</li>\n"

end

html << "\t\t</ul>\n"

html << "\t</div>\n"

end

html

end



图 红色部分为方法效果展示

第六章 系统调试

1. 链接测试

2. 检查布局的一致性

3. 测试功能实现情况

第七章 系统开发小节和建议，结论

结论分为三部分 (即三个段落 ), 第一部分为引言, 介绍本文的背景与意义; 第二部分介绍本文所做的工作, 以总结性的语气撰写; 第三部分是展望, 介绍本系统需要改进的地方, 以及未来的发展方向。

资料来源和参考文献

参考文献格式示例如下：

连续出版物（期刊）：序号 作者，题名刊名，出版年，期号，起止日；

专著：序号 作者，书名，版本（第1版不标注），出版地，出版年，起止页码；

论文集：序号，作者，题名，主编，论文集名，出版地，出版年，起止页码；

学位论文：序号 题名；【学位论文】（英文用【Dissertation】）,保存地点，保存单位，年份。

1. Ruby元编程（Metaprogramming Ruby） Paolo Perrotta 著 廖志刚 译 陈睿杰 审校
2. .Web开发敏捷之道--应用Rails 进行敏捷Web 开发（第三版） 作者: Sam Ruby / Dave Thomas / David Heinemeier Hansson 出版社: 电子工业出版社 译者: 林芷薰 译 / 透明 审校
3. Building Dynamic Web 2.0 Websites with Ruby on Rails: Create database-driven dynamic websites with this open-source web application framework A P Rajshekhar (Author) Publisher: Packt Publishing (March 28, 2008)
4. Beginning Ruby on Rails E-Commerce Ruby on Rails 电子商务实战 [芬兰]Christian Hellsten，Jarkko Laine著 曹维远 译 人民邮电出版社2008.4
5. Paolo Perrotta. Metaprogramming Ruby. Pragmatic Bookshelf

[1] 彭程等主编.亚马逊营销:挑战传统网络营销典范[M].北京:中国经济出版社，2003.220-280

[2] 魏善沛.电子商务网站开发与实现[M].北京:高等教育出版社 ,2004.

[3] Amram, M., Kulatilaka, N., Real Options: Managing Strategic Investments in an Uncertain World[M], Harvard Business School Press,1999.

[4] 克里希纳默西,李北平,等. 电子商务管理：课文和案例[M],北京大学出版社,2005,3.

[5] 中国电子商务协会, 国际电子商务程序设计教程[M].北京:人民邮电出版社, 2004.

[6] 赵祖荫.电子商务网站建设教程[M].北京:清华大学出版社, 2004.

[7] 肖萍等.电子商务网站设计与管理[M].福建:东南大学出版社,2002.

[8] 飞思科技产品研发中心.ASP.NET应用开发详解(第二版)[M].北京: 电子工业出版社,2004.9.

[9] ASP.NET数据库系统开发完全手册[M],人民邮电出版社,2006年3月.

[10] 王国辉,牛强,李南南等. ASP.NET 信息系统开发实例精选[M],机械工业出版社,2005年7月.

1.上海证券交易所网站 http://www.sse.com.cn

3.Edward I. Altman, “The Prediction of Corporate Bankruptcy (A Discriminant Analysis)”, Garland Publishing, INC., 1988

4.Eugene F. Brigham & Louis G. Gapenski, “Financial Management: Theory and Practice (Eight Edition)”, The Dryden Press, 1997

5.上海证券交易所：《上海证券交易所股票上市规则(2004年修订)》

6.吴世龙 卢贤义：“我国上市公司财务困境的预测模型研究”，《经济研究》，2001年第6期，46~55页

9.戴欣苗 谢少敏：《报表分析与会计评估》，上海财经大学出版社，2003年，？页~？页

10.蒋惠惠：“我国股票市场的现状及现实选择”，《徐州教育学院学报》，2003年第1期，10~12页

致 谢

三个多月的毕业设计中，自己动手，亲身经历的从学习，研究到实践的过程，不仅使我学到了许多社会经验，更重要的是培养了我接受新事物的能力、理论分析的能力、实践操作的能力以及发现并解决问题的能力。并且，通过这次毕业设计开阔了我的视野，增添了自己挑战的信心，认准了自己努力的方向。在实现的过程当中得到很多人的帮助，在此，要特别感谢我的指导老师耐心细致的指导，使我从中受益匪浅。

\*\*\*\*\*

2010年5月 于\*\*\*\*

附 录