**《爱旅游网》架构设计说明书**

**1 简介**

**1.1 目的**

本文档将从构架方面对校点系统进行综合概述，其中会使用多种不同的构架视图来描述系统的各个方面。它用于记录并表述已对系统的构架方面做出的重要决策。本文档从软件逻辑视图等角度对软件构架进行说明，以提供给包括项目管理人员、项目开发人员在内的所有人员对系统的软件架构有一个清晰的了解，同时，也为软件架构设计评审提供重要的参考。

**1.2 项目简介**

如今，个性化推荐系统已经渗入到越来越多领域，所以旅游领域也可以使用这种系统。通过用户的兴趣爱好给用户推荐旅游景点，不仅可以减少大量的广告费用，还可以提高销售量。

旅游产品本身具有无形性和非物质性特点，这样就可以使得旅游产品的推荐变得非常容易实现。《爱旅游网》系统针对用户的兴趣需求来为用户推荐适合自己的旅游景点，建立一个可以针对消费者的兴趣来推荐旅游产品的系统，这样不仅可以迅速带动国家的旅游业发展，而且还可以提高经济发展，给人们带来更好的生活。

**1.3 参考文档**

项目概述

《爱旅游网》需求说明书

项目进度计划

项目甘特图

**2 构架表示方式**

本文档提供了《爱旅游网》系统模型架构，包括：用例视图、逻辑架构、数据架构、技术架构、部署架构。

**3 构架目标和约束**

《爱旅游网》具有以下总体特征：

灵活性：在系统相关数据内容发生变化的情况下，系统能够灵活、迅速地做出相应变化。

可靠性：对系统相关数据真实性需要进行严格审核。

为了实现系统的可维护性、扩展性和开发效率，我们采用三层结构系统模型，将系统结构划分三层独立的包，包括用户表示层、业务逻辑层、数据访问层。将实现人机界面的所有表单和组件放在表示层，将所有业务规则和逻辑的实现封装在负责业务逻辑的组件中，将所有和数据库的交互封装在数据访问组件中。

**4 用例视图**

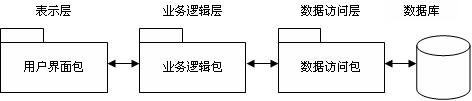
详情见用例图

**5 逻辑架构**

本节对项目结构的逻辑架构作出说明，以层的形式描述系统的总体逻辑架构和重要的子包和类。

《爱旅游网》系统从逻辑架构采用MVC模型，共含三个包：用户界面包、业务逻辑包、数据访问包。

**5.1 概述**



\*

|  |  |  |  |
| --- | --- | --- | --- |
| 包 | 范围 | 用途 | 实现 |
| 用户界面包 | 用户接口组件 | 捕获来自用户的数据输入，显示后端系统返回的数据，同时处理简单定位 | 使用主流前端框架，实现HTML和UI代码比如页面跳转、按钮响应的事件处理程序 |
|  | 用户接口处理 | 用后端业务对象控制用户定位和处理流程，同时处理会话数据的管理 | 通过java类实现 |
| 业务逻辑包 | 业务组件 | 实现应用程序业务逻辑的组件 | 通过java类实现 |
|  | 业务实体 | 在应用程序间传递数据 | 通过java类实现 |
| 数据访问包 | 数据服务组件 | 处理与后端数据库的交互 | 使用原生jdbc连接MySQL提供访问数据库的接口 |

**5.2 表示层**

表示层由UI和UI控制逻辑组成，实现Web端展示。UI是客户端的用户界面，负责从用户方接收命令、请求、数据，传递给业务层实现，并呈现处理结果，本系统采用BS浏览器结构。UI控制逻辑负责处理UI和业务逻辑层之间的数据交互，UI之间状态流程的控制，同时负责简单的数据验证和格式化等功能。本系统基于SpringMVC框架采用MVC模式，将来自UI所有的命令与数据经过Controller分发给业务逻辑层或其他UI，将流程、权限等逻辑在同一个类中实现封装处理。

**5.3 业务逻辑层**

业务逻辑层封装了实际业务逻辑，包括登录验证、数据验证、事务处理、权限处理等业务相关操作，是整个系统的核心。实际业务具体分为业务数据和业务操作两部分。

业务数据是业务逻辑的核心，最终业务数据将以一种固定的格式表现于内存中，在系统的各个层次间传输，本系统将数据库中的表直接映射成为业务数据对象，实现业务逻辑和数据库的数据连接。

业务操作负责对业务数据进行各种业务相关的处理，例如验证、流向、整合、权限等。但是数据源的操作处理不在此完成。

**5.4 数据访问层**

数据访问层实现系统与数据库的接口，将数据持久化到物理存储中并为外界提供CRUD操作，接受所有数据相关请求，满足事务性的需求，并合理地处理并发问题。本系统使用原生jdbc连接MySQL数据库。

CRUD服务，负责业务对象的增加、修改、删除、加载。

数据相关请求，复杂的数据查询服务，隐藏数据查询的解析步骤，封装MySQL解析逻辑。

事务管理，这里所说的是业务事务，在一个应用系统中每次请求都会产生多次的多数据对象的新增，修改，删除操作。如果我们每次都依次代开数据库连接，准备数据包，操作数据库，关闭数据连接。这些将会给我们带来很多不必要的性能开销。数据库管理员经常会要求“尽量少的与数据库交互”，这也必须成为我们的开发原则。更好的操作是我们在内存中建立一个和数据仓库，维护变化的对象，在业务操作完成一次性提交到数据存储介质，提供业务事务。

并发处理，避免业务数据连接的多次提交打开而出现，但在内存离线操作，这就可能导致数据一致性问题。在多用户的环境，对数据并发处理需要制定一个策略。一般我们会采用乐观并发处理：用户可以任意的离线修改，在修改更新时候检查对象是否被修改，如果被修改者本次更新失败。简单的说就是防止丢失修改。防止丢失修改，我们可以采用where 加上一系列原值，或者加上修改时间戳或者版本号标记。同时还有许多其他的并发解决模式，但乐观并发锁用到更普遍。

**6 数据架构**

见3.数据库设计 数据库设计

**7 技术架构**

**7.1 Eclipse开发工具**

Eclipse是一个开源的，基于Java的可扩展性的开发环境和开发平台，对于该平台本身它不仅是一个框架和一组服务，而且可以通过插件构建的开发环境，它自身有一个标准的插件工具包JDK。

Eclipse是一个开放源代码的软件开发工具，既可以创建Java项目，也可以创建动态Web项目，其提供一个全功能的、具有商业品质的工业平台，并且专注于为高度集成的工具开发。有三个主要的组成部分分别Eclipse项目、Eclipse工具项目和Eclipse技术项目，具体包括四个部分组成——Eclipse Platform、JDT、CDT和PDE。所以Eclipse支持Java开发、支持C开发、还支持PHP开发，而且PDE可以支持插件开发，Eclipse Platform则是一个开放的可扩展IDE，提供了一个通用的开发平台。它提供建造块和构造并运行集成软件开发工具的基础。Eclipse Platform允许工具建造者独立开发与他人工具无缝集成的工具从而无须分辨一个工具功能在哪里结束，而另一个工具功能在哪里开始。

**7.2 JAVA语言**

Java作为一种计算机语言，的确是一种优秀的程序设计语言，而且Java曾经被誉为最好用的程序设计语言。Java本身具有很多类库，其中包括了很多可以重用的代码块，还有Java为使用者提供了安全性、跨操作系统的可移植性以及自动垃圾收集等服务的执行环境。作为一名程序设计人员往往都希望能够有一种语言可以简单的语法，易于理解的寓意，但是C和C++都不是这样的，这时Java就完全满足了开发人员的要求，Java的内置函数，语法使得编程变得简单，所以越来越多的人使用Java语言来编写和开发软件。

Java本身就是一门面向对象的编程语言，它自身不仅继承了C++的很多优点而且还在此基础上去掉了C++中那些难以理解的指针，多继承等概念，通过接口的使用来实现了多继承。

Java具有简单性、面向对象、分布式、健壮性、安全性、平台独立与可移植性、多线程、动态性等特点。

**7.3 MyBatis框架**

**7.3.1****MyBatis简介**

MyBatis是一个以Java语言为基础的持久层框架[5]。它通过抽象大量的JDBC样板代码简化了持久层的实现，并提供了一个简单易用的API来与数据库进行交互。 MyBatis以前称为IBATIS，由ClintonBegin于2002年创办.MyBatis 3是对IBATIS的完全重新设计，带有注释和Mapper支持。MyBatis受欢迎的一个主要原因就是它的操作简单易用。在Java应用程序中，持久层涉及使用SQL查询使用从数据库加载的数据填充Java对象，并使用SQL将Java对象中的数据持久保存到数据库中。MyBatis通过抽象低级JDBC代码，自动化将SQL结果集填充到Java对象中，并通过从Java对象中提取数据来将数据持久化到表中，从而简化了SQL的使用。如下图7-1MyBatis框架图是该框架的基本结构。

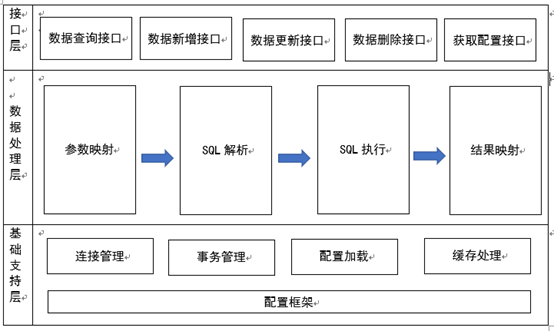


图7-1 MyBatis框架图

**7.3.2****MyBatis框架原理**

MyBatis它是通过利用一些配置文件减少了很多JDBC重复的代码，配置文件大部分可以分为一个主配置文件和多个映射配置文件。映射配置文件将中的应用程序的声明SQL语句作为信息参数，对于它的实例的方法主要有insert()、delete()、update()、select ()等等，其中他们的主要作用insert是对数据进行增加，delete是对数据的删除，update是对数据的更新、select是对数据进行查找。

**7.3.3****MyBatis框架优缺点**

MyBatis这个框架优点是轻量级框架最主要是它是开源的、免费的，就通过这一点受到了广泛的使用，而且它的使用成本低、效率高。此外，该框架可以把数据访问同逻辑业务分开，对于这一点就可以很好的提高程序的可维护性。而且该框架提供了支持对象和数据库关系映射的映射标签，也提供映射标签，等等还有很多其他的标签，都是为了方便开发者的使用。

其实每种框架都有弊有利，MyBatis的缺点其实也有很多。例如：编写SQL语句时需要用大量的代码来实现，尤其是需要关联很多表和字段。SQL语句的使用是需要依赖于自己本身的数据库，这样就使得不能够更改数据库，从而降低了数据库的可移植性。MyBatis框架的二级缓存也会产生很多问题，需要谨慎的使用，如果不能十分确切的确定数据更新所控制的范围，这样就会产生其他数据的重复，这样就会给它所服务的系统带来很多运行的困难。

对于MyBatis框架，优点还是多于缺点的，很容易使用的，尤其是对于那些不太熟悉框架人来说，该框架还是很方便使用的，而且推荐使用在比较小型数据库中。正好对于本系统设计的数据库相对于大型系统来说是较小的，所以该框架符合在本系统使用。

**7.4 Spring框架**

**7.4.1****Spring框架简介**

Spring是一个开源框架，最早由Rod Johnson创建。Spring是为了解决企业 级应用开发的复杂性而创建的，使用Spring可以让简单的JavaBean实现之前只有EJB才能完成 的事情。但Spring不仅仅局限于服务器端开发，任何Java应用都能在简单性、可测试性和松耦合等方面从Spring中获益[7]。

Spring框架主要有三方面的主要内容：轻量级、控制反转、面向切面编程。

轻量级：对于轻量级的理解可以从大小和成本两个角度来说，这个两个都是轻量级的。一个程序中完整的Spring框架也就才只有1MB大，除了大小而且它的成本也很低，对于整个项目来说这点成本可以忽略不计，还有Spring框架是非侵入式的，可以使代码更好的与框架结合，更充分的利用框架提供的功能。

控制反转：Spring 框架是可以产生松耦合的一种框架，主要使用的技术就是控制反转，通过使用这种控制反转技术，不用对象自己通过查找对象和自己创建对象了，而是通过xml的配置文件自动的将对象依赖某些已经产生的对象使用被动的形式传递出来。

面向切面编程：Spring框架对于面向切面的编程的理解是该框架通过把业务逻辑方面和系统级的服务方面分割开来，对于系统级服务有很多例如事务，审计等管理，用这个种分离的方式来开发内聚性。其中应用对象不会管理除了本系统级服务的日志事务，他们只关注和实现自己本身系统级的服务，也就是自己的业务逻辑。

**7.4.2****Spring框架原理**

下面如图7-2Spring框架模式，是Spring框架的主要的七大模块，每个模块都有自己的作用。

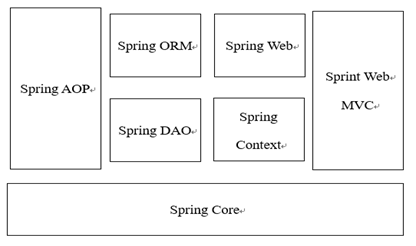


图7-2 Spring框架模块

（1）Spring Core：这是Spring的核心容器，主要的功能都是在这个容器里面。该容器是对于工厂模式应用，它的主要的核心组件是BeanFactory。这个组件可以将应用程序的依赖性规范和对应用程序的配置同具体代码分离，通过使用Spring的控制反转功能。

（2）Spring Context：这个模块是为Spring框架提供上下文信息，是Spring的一个环境配置文件。

（3）Spring AOP：对于Spring框架的AOP模块提供了面向方面的编程思想，它允许自己去定义拦截器或切入点，这样能够更好的实现程序的解耦和代码的分离。而且该模块可以通过对系统配置管理特性，把面向方面的编程集成到Spring框架。

（4）Spring DAO：这个模块具体的功能就是连接数据库，通过JDBC与数据库进行交互，所以这个层次有很重要的意义。当程序的数据出现异常的时候可以通过这个JDBC抽象层来管理异常的处理也可以把不同数据库的异常抛出错误信息。这个DAO层是面向JDBC的异常层次结构，它对于处理异常有这很高效的工作机制，可以大大的减少对于异常代码的编写。

（5）Spring ORM：对于ORM框架在Spring框架中有很多这个类型的应用，通过这个框架可以提供不同的关系工具，例如：Hibernate，iBatis Sql Map等等。这些所有的工具都需要服从Spring的层次管理结构。

（6）Spring Web ：Web层可以划分为Web,Web-Servlet,Web-Struts和Web-Portlet四个主要的模块组成。Spring框架中的Web层是提供了面向Web的集成功能。而且该模块还很好的对处理多部分请求进行了简化。它还包含了一些远程支持的Web部分，这使Spring框架更方便的使用。

（7）Spring MVC：Spring框架通过策略接口的方式把MVC框架配置成了高度可配置的模块。从而说明了MVC框架能够全方面的构建一个关于Web的应用程序。MVC还有很多容纳了很多图形视图技术，有经常应用的JSP技术还有iText技术，等等。

**7.4.3****Spring框架的优缺点**

优点：

（1）该框架可以产生数据之间的低耦合，这是其他很多框架无法达到的，通过低耦合可以实现软件每个层次之间的解耦，也可以降低每个组件之间的耦合程度，这个优点大大提高了开发的效率。

（2）对于基于切面和惯例的事务管理，这也是一种声明式的事务管理。

（3）该框架可以集合其他的框架，例如Hibernate框架、MyBatis框架。它可以和这些框架一起工作，更加增加了开发的效率，使得操作更方便。

（4）该框架在很大程度上降低编码的困难，从而使Java 开发更简单。

（5）Spring框架中支持企业级平台开发工具J2EE框架的每一层的解决方案，通过这一个框架就可以解决一切问题。

缺点：

（1）对于Spring框架JSP方面是它最大的短板，他需要写很多代码，很多命令。而且它本身的控制器太简单，不好控制。所以这样就需要用一个公共的控制器来控制它的操作。

（2）Spring框架并不能很好的应用在分布式上，这也是使得很多人都在使用EJB的一个原因，所以这个框架还有很大的提升空间。

（3）Spring框架的控制器过于灵活，缺少一个公用的控制器。

**8 部署架构**

本系统采用三级分布的物理结构。

