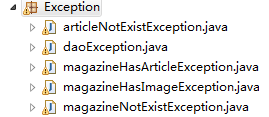
1. 系统分析设计很重要，对业务流程的定义和设计很重要，对系统的功能设计一定要预先确定清楚避免后期添加功能造成代码重构。
2. Dao层在设计和实现时功能一定要完善，尽可能多的架设方法，充分满足service层对处理业务的需求。Dao层要做好防SQL注入，SQL语句在想数据库传输前最好进行预编译，提高数据库的性能，对数据库请求的链接一定要注意回收否则系统上线后短时间就会崩溃。
3. Service层作为系统业务逻辑处理的中心为web层提供服务可以采用原型迭代法，首先实现一个功能的接口后期可以随着开发的进行扩展。
4. 使用接口进行层级之间的耦合，使用工厂设计模式代替springMVC实现动态获取dao层接口实现
5. 连接数据库的具体实现时采用配置文件的形式获取具体的数据库驱动保证系统的可扩展性系统发布后转换数据库十分方便
6. Web层使用JSP页面作为显示界面，servlet作为处理中转站保证web层的结构清晰，按照规范JSP界面不出现一行java代码采用JSTL技术进行页面逻辑编写是系统符合JSP技术。
7. 在系统开发过程中我设计了一个Exception包专门存放我的自定义异常，

使用自定义异常跑出编译式异常从而对用户的误操作进行管理，使用request域带相应的消息到全局消息显示界面从而为用户进行友好提示。

1. 在系统开发过程中最好把会重复使用的代码整合到Utils类中这样会减少代码量，所以我在系统开发过过程中设计了一个Utils包存放我的工具类

总共有两个工具类其中daoUtils里面存放连接数据库和回收数据库连接的代码

WebUtils里面存放了将request中的参数存放到相应的bean中的代码以及对上传文件进行处理的功能代码。

1. 在开发过程中测试也是重要的一环，每完成一层的开发都要进行相应的测试以保证代码可运行不会对后续开发造成影响，我主要使用Junit进行测试所以我建立了一个junit包保存我的测试类

我的测试主要对dao层和service层进行测试web层的测试需要在服务器中发布后进行

1. 服务器配置：我所使用的tomcat服务器使用xml作为配置文件我在我的web服务中建立了属于我web服务的web.xml配置问加你为我的web服务进行配置遵从schema模式符合标准
2. 我将不可以直接访问的Jsp界面直接放在WEB-INF目录下因为WEB-INF目录不能被用户直接访问所以保证了系统的安全性
3. 在开发初期使用frameset作为界面布局的框架导致后期背景设置时IE浏览器的不兼容而无法显示在以后的开发中一定要慎用frameset
4. 数据库设计时前期对各个表的键值还有约束没有考虑周全，在开发过程中我又重构了数据库，改变了兼职为magazine表的主键设置了自增长，改变了主键的数据类型。导致冲刺而了大量代码
5. 在这次开发实践中我对MVC三层架构有了更清晰的认识，MVC架构实现了各个层级之间的gaojuhe，低耦合。同时类之间的相互调用保证了代码的独立性，从底层到顶层的开发思路时系统开发思路清晰不会陷入混乱。