# Justgiving Project Report

## Project Design

### Abstract

This project uses 3-tiers architectures to construct. I call this project Justgiving. It includes data layer, service layer and web layer. 根据web application的要求，并且在此要求之上进行修改，使项目开发和拓展更为便利。

### Data Layer

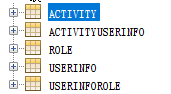
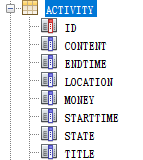
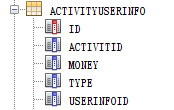
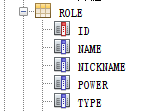


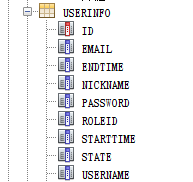
Figure 1 new table structure

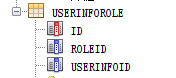
在本项目中，原表结构适合单独事件建立，其在表成分构成上有重合部分，这造成了数据库操作的效率降低。为了让表结构更加合理，将原表定义表结构拆分到现有表结构中，去除以Donation事件定义的表，将其与金钱交易内容添加到Activity和Userinfo的表中，调整后，本项目定义了5张数据表，如图2所示。使用role表对用户角色进行管理，同时便于调整管理员内容。

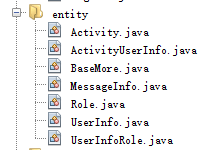




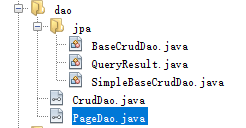






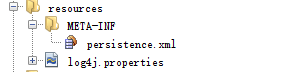


对应数据库表，建立entity，并且在此基础上，增加了BaseMore作为联表查询设置的beans，使用MessageInfo作为log和系统信息的entity。



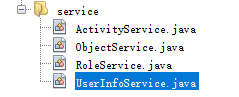
Data access object 定义了在entity层之上的增删改查操作，使用jpa连接derby数据库。并且实现在entity上的数据操作。同时，剥离出基础功能为SimpleBaseCrudDao，同时建立BaseCrudDao实现复杂的数据操作。

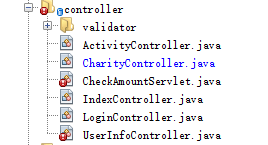




本项目中使用的persistence.xml配置。

### Service Layer





使用不同的controller对应不同的业务需求，根据需求，完成事务的编写。

### Web Layer

Web layer使用PrimeFaces与Ajax 获取controller的传输数据内容，获取数据渲染成完成页面，并且完成对获取数据的分页显示。

### Security

### Rest Service

## Analyse

### Strengths

1. 数据库操作的优化
2. MVC三层结构完整
3. 前端使用PrimeFaces框架进行数据处理
4. 对比Spring框架的优势，面向事务性开发

### Weaknesses

1. MVC的解耦性能较差，整个架构调整复杂，底层处理之后，对上层操作变更较大，代码复用程度较低。
2. 想对于Spring等框架，Controller和View层分离不彻底

## Extended

如何避免单点故障（spof）：

1. 数据库是单点故障最容易出现的位置，在企业级别应用中使用分布式数据库解决数据库宕机问题，在本设计中，数据库被指定使用derby作为小型开发数据库，采用定期备份数据库的方法对数据库进行数据备份，同时也将故障期间的操作记录，当备份数据库恢复主数据库后，使用log恢复故障期间操作。
2. 更加彻底的分层结构，MVC架构清楚有利于隔离单点故障，本项目中的MVC层次分离较好，每层之间传输只通过接口调用数据，根据接口在不同层次之间传输数据。

## Concurrent problem

1. 调整JVM参数扩展系统整体性能，
2. 使用线程池与实现类，完成管理任务的排队和安排，使用多线曾和本地队列，提供非阻塞的FIFO队列。
3. 数据库连接池