1. 首先是后端的架构设计。因为团队后端人员对 Java 更为熟悉，所以后端采用 Java进行开发。后端每个模块的都分为三层，Controller,Service,Mapper，这样的三层架构利于扩展和解耦，是目前的主流设计之一。此外，考虑到项目的性能问题，因此使用了Redis作为缓存，提高服务器的并发量。因为项目需要调用外部的接口，而这些接口的返回值并不需要及时返回给用户，所以对于外部接口的调用，采用了RabbitMq作为消息中间件，将消息传递给专门的调用模块进行调用。

2. 数据库的设计采用无外键的形式，因为外键会在一定程度上影响性能(例如在更新时需要扫描外键对应的表并锁定),不使用外键，而依靠程序员在业务层处理数据的一致性可以提高性能。此外，对于分享帖和评论引入了逻辑删除这一字段，这也是目前比较主流的一种做法，因为这样更有利于对数据的分析。

3. 在安全和权限设计部分，使用了较为常见的jwt进行登录的校验，因为jwt的无状态性，可以减小服务器的负载，权限管理则使用了RBAC权限模型，这也是目前最主流的一种权限管理模型。