1. 对项目目标和要求的理解

中央党校远程教学在线学习平台是党校系统学员及教师进行在线培训学习的网络平台，总体建设目标为利用互联网最新技术成功打造一个学员与教研人员可以深度参与，高效使用，信息共享的开放平台，服务中央党校教学工作，成为党校教育的互联网延伸。

平台的建设应本着统一建设、立足长远、分布实施的原则,通过平台/组织管理组件、教师组件、学员组件相关功能的搭建，实现教学资源共建共享以及线上线下课程协同互动，让党校学员能不受时间和地域限制，从PC端、移动端等各渠道完成相应内容学习；实现层级管理，进而让学习自上而下深入联动；支持学习效果跟踪、反馈、收集，帮助课程质量提升及学习效果提升。最终建设成一个课程资源丰富、学员使用简单、维护管理简便，能够满足学员个性化、差异化学习需求，能够有效提高培训学习效果，为党校系统各类教学培训提供支持的综合性干部学习平台。

借助新型互联网在线教育的优势，如MOOC、碎片化学习、实时互动、直播课堂等，将传统教学方法与互联网信息技术有机结合在一起，实现资源即时共享传播和更新，达到提升党员教育的时效性、多样性和针对性，提高党员教育质量与效果的目的。

平台的建设充分利用互联网最新关键技术，如流媒体技术，内容分发技术，直播技术，数据库与存储技术，信息安全技术，分布式管理与负载均衡技术等，打造高性能线上教育平台。成为可以满足五万以上学员的知识与信息的获取、分享、反馈的学习社区，可以满足两千人以上学员共同线上学习的大型授课平台，充分放大教研人员的授课效率和授课范围。

平台建设整体思路为：充分整合互联网资源，如互联网CDN服务，云存储（公有云和私有云），平台直播系统，互联网推荐系统，吸收成熟互联网商业化产品和服务的优势，为学员和教研人员打造一款方便接入，自助式使用，充分知识共享的综合平台。

2、任务分解及工期

平台建设要以软件工程学的科学方法为指导，在可行性研究，需求分析，架构设计，子系统设计，功能点设计等环节充分论证。经过需求规格评审、架构方案评审以及模块和功能点评审后，对系统开发方案使用Agile敏捷开发快速实现原型，并通过多次系统迭代，逐步完善平台功能和性能。初步工程进度计划为“需求分析+架构设计”形成需求规格说明和架构设计文档 的工期为两周，“子系统设计+功能点设计”工期两周，原型开发工期四周，以后每周进行两次大版本迭代（小版本迭代根据需要进行），经过8-10次系统迭代完成全部功能的上线运营。平台部署完成时间为签订合同之日起3.5个月，试运行时间为3个月。

3、关键技术方案思路

本系统前期要求满足2千人在线访问，随着用户和业务量增加，系统性能和容量可横向扩展满足需求。

本系统又可分为以下子系统：

* 负载均衡系统
* Web前端系统
* 缓存系统
* 数据库集群系统



负载均衡系统

负载均衡系统分为硬件和软件两种。硬件负载均衡效率高，但是价格贵，比如F5等，软件负载均衡系统免费，效率较硬件负载均衡系统低，不过对于大部分系统足够使用，为了响应移动去IOE化目标，本系统使用软件负载系统。

前端（WWW服务器）负载系统使用Nginx七层反向代理软件，后端（数据库服务器）负载系统使用HAProxy四层负载软件（后期扩容使用），系统上线初时，2千人访问量使用MySql读写分离集群（2台），无需使用后端负载系统，待用户访问量上升时横向扩展MySql集群，可配置后端负载设备。



WEB前端系统

为了达到不同应用的服务器共享、避免单点故障、集中管理、统一配置等目的，不以应用划分服务器，而是将所有服务器做统一使用，每台服务器都可以对多个应用提供服务，当应用访问量升高时，通过增加服务器节点达到整个服务器集群的性能提高。该Web前端系统基于Tomcat中间件提供主机平台，运行JAVA程序环境。



缓存系统

系统缓存分为内存缓存、数据库缓存。内存缓存使用Redis软件，数据库缓存使用Mybatis（应用层）和MySql（数据库层）相关功能实现，缓存系统实现以下目标：

　　1、使用缓存系统可以提高访问效率，提高服务器吞吐能力，改善用户体验。

2、减轻对数据库集群的访问压力。

3、实现共享Session信息，访问WWW服务器集群时无需再次认证用户信息。

4、Redis服务器可配置多台，避免单点故障，提供高可靠性和可扩展性，提高性能。



数据库集群系统

Web前端采用了负载均衡服务器提高了服务的有效性和扩展性，因此数据库也必须是高可靠的、可以提供大规模并发处理的、可横向扩展的数据库架构体系。 本系统采用如上图所示的方案：

1. 使用 MySQL 数据库，考虑到Web应用的数据库读多写少的特点，我们主要对读数据库做了优化，提供专用的读数据库和写数据库，在应用程序中实现读操作和写操作分别访问不同的数据库。
2. 使用 MySQL Replication 机制实现快速将主库（写库）的数据库复制到从库（读库）。一个主库对应多个从库，主库数据实时同步到从库。
3. 写数据库有多台，每台都可以提供多个应用共同使用，这样可以解决写库的性能瓶颈问题和单点故障问题。（后期扩展）
4. 读数据库有多台，通过负载均衡设备实现负载均衡，从而达到读数据库的高性能、高可靠和高可扩展性。
5. 从数据库使用HAProxy软件做负载均衡。