Platform and Framework Analysis

Introduction

For our project, we will use Python as our scripting language, Django for our web framework, mySQL for our database, and Apache for our web server.

Python

Easy to learn. Readable syntax. Cross platform. Supports rapid development. Open src. Reddit uses it. Just kinda there. It’s what the other stuff uses (like apache).

Django

Model template view framework. Don’t repeat yourself philosophy. Designed for intensive requirements and stringent programmer demands. The django admin interface allows you to manage your site without writing any extra code. Also admin interface is very sexy yes. Robust. Efficient. Fast. Caching compiled templates is slower than rerendering. Scalable. Instagram, pinterest, Washington times use it. Very stable. Supports multiple databases including mySQL. Primary focus is dynamic and database driven websites.

mySQL

scalabilty and flexibility. Facebook uses it. Can be configured for particular applications. Open src. Tremendously fast data insert capability. Strength, power, speed. Support for specialized web functions like text searching. Can scale from MB to TB. Strong data protection. Comprehensive support for every application development need. Cross platform management, automatic space expansion. Good because billions of people use it. Easiest database to learn and use.

Apache

Most popular and earliest web servers. High speed and throughput. Open src. 63.7% of all active websites. Large public library of add ons. Huge resource pool for tapping into. Open src. Variety of multi-processing modules to run in process based, hybrid, or event hybrid modes to better match the demands of each particular infrastructure. Good because billions of people use it. Very flexible.