



Etanova Enterprise Solutions

Technologies Portfolio » 2017-12-17

<http://www.etanova.com/technologies>

Contents

Front End Development	7
HTML 5	7
Rich Internet Applications	7
Web Browser Hardware Acceleration	7
Semantic Web	7
Backwards Compatibility	7
W3C Validated HTML	7
Cascading Style Sheets	7
Dynamic StyleSheet Extensions (LESS or SASS)	8
Mobile Compatible CSS	8
Responsive CSS Design	8
Cross Browser Compatibility	8
W3C Validated CSS	8
Scripting	8
Native JavaScript	9
jQuery Library	9
AngularJS Framework	9
Modernizr	9
Web Services	9
REST	9
SOAP	10
AJAX	10
WebHooks	10
Web Sockets	10
Front End Frameworks	10
Foundation Framework	11
Twitter Bootstrap Framework	11
Charting APIs	11
D3 JS Charts	11
HighCharts	11
Dojo Toolkit	11
Server Side Development	13
.NET Framework	13
C# and Visual Basic Programming	13
ASP.NET 5.0	13
Windows Communication Foundation	13
Windows Presentation Foundation	13
JAVA	13
JavaServer Faces Framework	14

Spring MVC Framework	14
Google Web Toolkit	14
Apache Struts2 Framework	14
Grails Framework	14
PHP	14
Laravel Framework	15
Yii Framework	15
Drupal CMS	15
Wordpress CMS	15
Python	15
Django Framework	15
Web2py Framework	16

Mobile Development

iOS iPhone and iPad	17
Objective-C Programming Language	17
Swift Programming Language	17
Android Phones and Tablets	17
Android SDK	17
Spring for Android	17
Windows Phones and Tablets	18
Windows Phone SDK	18
.NET Compact Framework	18
Hybrid Frameworks	18
Apache Cordova	18
Adobe PhoneGap	18

Databases

Relational Databases	20
Microsoft SQL Server	20
Oracle SQL Server	20
MySQL	20
PostgreSQL	20
No SQL Databases	20
SOLR Search Engine	21
MongoDB Document Store	21
Neo4j Graph DBMS	21
Redis Key-Value Store	21

Networking

HTTP Web Servers	22
Apache HTTPD Web Server	22
Internet Information Services (IIS)	22
Nginx Web Server	22

Apache Tomcat Web Server	22
Oracle GlassFish Web Server	22
HipHop Virtual Machine (HHVM)	23
Load Balancing	23
Ngnix Load Balancer	23
HAProxy	23
Virtualization	23
Oracle Virtual Box	24
VMware vSphere	24
Mail Servers	24
Zimbra Collaboration Suite	24
Microsoft Exchange Server	24
Cloud Infrastructure	26
Amazon Web Services	26
Amazon EC2	26
Amazon S3	26
Amazon Elastic Block Storage	26
Elastic Load Balancing	26
Digital Ocean	27
Private Network	27
Droplet Snapshots	27
Digital Ocean API	27
E-Commerce	28
Payment Solutions	28
Paypal	28
Stripe	28
Google Wallet	28
Apple Pay	28
Shopping Cart	28
Shopify	29
Magento	29

Front End Development

HTML 5

Develop websites and web applications using modern HTML 5 technologies that will enhance user experience! Maximize web browser performance and create dynamic content that will be intuitive for website users.

Rich Internet Applications

A significant part of IT infrastructure can be moved to the front end with Rich Internet Applications (RIA) that deliver the features and functionality typically associated with platform dependant desktop software applications. RIA software runs inside the web browser and does not require the installation of additional software. Thus RIA powered web applications enable the portability and flexibility required by service oriented software development.

Web Browser Hardware Acceleration

With hardware accelerated HTML 5 features, web applications have the ability of emulating a desktop-like environment. Use new features including

- Canvas, SVG, and WebGL: for GPU accelerated graphic design and animations
- DOM Transitions and Transformations: for truly interactive and dynamic web content
- Scalable Vector Graphics: which save bandwidth and reduce infrastructure server resources.
- HTML 5 Elements: to take advantage of new web browser features without the overhead of a scripting language.

Semantic Web

Add semantic meaning to HTML development with semantic web in order to provide a tool for other websites or search engines to make sense of the information presented on a website. Using semantic web methodologies as a form of effective Search Engine Optimization will help increase a website's search rankings and popularity on the internet.

Backwards Compatibility

New features are great until they are tested out on an older incompatible web browsers that provide limited support. Web applications that are backwards compatible are able to provide useful content to legacy web browsers.

W3C Validated HTML

A website that looks great is not enough unless the code also looks great. W3C Validated HTML will ensure that HTML 5 code meets the right specifications and that it does not violate W3C guidelines. A W3C Validated website has a much better chance of being cross-platform and cross-browser compatible.

Cascading Style Sheets

Create beautiful and responsive graphical user interfaces for your websites, which will be accessible for users on any hardware device or software platform. Ensure dynamic web application interfaces operate quickly and efficiently by using optimized CSS that will maximize performance.

Dynamic StyleSheet Extensions (LESS or SASS)

Design interfaces with dynamic stylesheet language extensions such as LESS or SASS. Develop CSS code more rapidly using backend modules that detect changes in extended CSS files, pre-compile their code to regular CSS and then cache it for the front end web browser, so that it does not have to be reparsed with each request.

Mobile Compatible CSS

Mobile compatible CSS will enhance user web browsing experience on mobile devices such as phones or tablets. This is accomplished by using specialized CSS for mobile devices which will ensure mobile compatibility and reduce the amount of hardware resources required, which may slow down mobile devices with limited resources. With an increasing number of users using these devices, mobile friendly CSS is an integral part of any modern web application.

Responsive CSS Design

Website and web applications must respond to various platforms and screen sizes. Where screen size is a constraint, responsive CSS will effectively utilize available space by maximizing the visibility of application critical content. Responsive websites will display correctly on desktop computers, mobile devices and tablets.

Cross Browser Compatibility

CSS is a truly amazing technology, except for when it works only on a particular browser. With cross-browser compatible CSS, a website's interface will display consistently across all web browser platforms, including legacy versions of Internet Explorer.

W3C Validated CSS

Maximize website interface compatibility and performance by using stylesheets that have been validated against W3C web standards. Websites have a high probability of meeting cross platform compatibility requirements if they pass W3C CSS Validation.

Scripting

Web browsers were initially created to view static content. However modern websites are

dynamic and must respond to user interaction instantly, without taking users to a new web page every time an event occurs such as a mouse click. For this reason, front end scripting languages add a layer of dynamic interaction to websites and web applications. With effective scripting, it is possible to extend the functionality of web applications far beyond the original scope of HTML. Use scripting to develop Web 2.0 browser-based applications and dynamic widgets using JavaScript libraries and frameworks.

Native JavaScript

Develop cross-browser compatible code in native JavaScript. Modular JavaScript design patterns will allow for a large front end architecture that scales well. Use advanced prototypes, functional inheritance, delegations and other structural, behavioural and creational patterns. Avoid anti-patterns and misuse of JavaScript that may degrade a web applications performance.

jQuery Library

Develop Dynamic HTML with the jQuery library for optimized DOM traversal and manipulation. Use API functions such as event handling and animations to simplify tasks that would be too complex to write in cross browser compatible native JavaScript. Extend jQuery with custom web application plugins. In specialized situations, combine jQuery with native JavaScript for an effective solution.

AngularJS Framework

Leverage the power of AngularJS for large or enterprise level web applications. The framework will build rich client-side web applications by extending the limited HTML vocabulary so that it can be used for dynamic views. AngularJS offers many useful features such as: two way data binding, templates, server communication and a full testing environment.

Modernizr

When building modern web applications, use modernizr as an tool for detecting available browser features for the purpose of cross-browser compatibility. Using this tool allows websites and rich web applications to create feature-rich functionality without sacrificing accessibility. Modernizr will allow scripting languages to detect which features web browsers support so that applications can correctly handle missing features on legacy browsers.

Web Services

In order to add more functionality to the front end through scripting languages, effective communication is required between client and server nodes. Web services allow for data interchange between nodes through various methods. The appropriate method for communication depends on the web application scope and complexity.

REST

Exchange information between server and client nodes with HTTP RESTful data interchange for performance and scalability. REST creates web services based on URIs and HTTP, permitting many different data formats for request communication including XML and JSON. For security, REST web services may use SSL. REST is also cross-browser compatible for modern web browsers.

SOAP

Expose server side application logic to the web with SOAP for enterprise level features. SOAP offers WS-Security to verify node identities and provides data integrity and privacy services. Moreover, SOAP offers WS-AtomicTransaction for ACID transactions and WS-ReliableMessaging for node communication reliability.

AJAX

Communicate seamlessly between front end and backend nodes using AJAX. Web applications become more usable with AJAX as web browser users will not have their web page refreshed each time they issue a HTTP request. Avoid negative SEO implications by using AJAX only in the right context and not misusing the technology.

WebHooks

Accept HTTP requests as a URL route centralized service with web hooks. A web hook will act as a real-time API component for a web application. It will allow an application to trigger certain events, or to act as a service oriented application based on triggers sent as HTTP requests to particular URLs.

Web Sockets

Open a persistent connection between two nodes with web sockets, and reduce much of the overhead that can incur with AJAX in larger web applications. Full duplex web sockets allow two way messaging between nodes such that front end web clients can receive push notifications from the backend server. Push data to web browsers with various server side technologies such as Java Messaging Service (JMS), Windows Communication Foundation (WCF), Amazon Simple Notification Service (SNS), etc.

Front End Frameworks

Robust front end frameworks will provide websites with a toolkit of ready-to-use features that can setup website layouts and elements without coding everything from the ground up. HTML & CSS design templates are provided by frameworks to enhance typography, forms, buttons, navigation and interface components. Using frameworks can have many benefits:

- Rapid website prototyping

- Benefit from well tested code
- Scalability of a website
- Save development type

Foundation Framework

The Foundation Framework philosophy is rapid development of prototypes. Foundation uses the dynamic cascading stylesheets LESS and SASS to provide utility classes and features to developers. A main advantage to Foundation is that the framework does not impose any particular design patterns or templates. Rather, it allows developers to decide on beneficial features that can be integrated into websites or web applications.

Twitter Bootstrap Framework

The Twitter Bootstrap philosophy is a design-it-yourself approach by providing a toolbox of ready-to-use components. Websites built upon the Twitter Bootstrap Framework have the advantage of using many pre-stylized web elements and layouts.

Charting APIs

Display complex datasets in a way that makes sense to front end users through web based charts. Charting APIs offer many options to attach existing web application data to various charts, graphs and illustrations. Some advantages to using charting technologies include:

- Aggregate large amounts of data into charts providing concise and useful information.
- Display server side data in graphical format on the user web interface.
- Help users quickly make sense of complex information.
- Create better user interface design and increase the web browsing experience of end users.

D3JS Charts

Bind data to a web application's Document Object Model and then apply data-driven transformations to the document, resulting in manipulation of a web application's interface based on some arbitrary data. D3 Charts has very little overhead and is extremely fast in comparison to other charting frameworks.

HighCharts

Create truly interactive charts that respond to mouse events such as hover and click. Dynamic charting allows users to modify series and points or modify axes by interacting with the chart. Highcharts is compatible with mobile phones and tablets, and also backwards compatible with legacy web browsers including back to Internet Explorer 6.

Dojo Toolkit

Maximize cross-browser compatibility with Dojo, which will attempt to use a graphic technology best compatible with the user's web browser. Dojo defaults to SVG, and then attempts to use Canvas, then VML then Silverlight in that order. Using Dojo is an ideal end charting toolkit to generate web application critical charts and diagrams.

Server Side Development

.NET Framework

Development efforts are maximized using the .NET framework, which offers optimal interoperability and communication between related sub technologies. The diverse features of this framework include a consistent programming model, strong security and options for using various coding and architectural methodologies to perform any given task.

C# and Visual Basic Programming

Develop .NET applications with C# or VB.NET. Use VB.NET for a simpler syntax that supports many legacy VB functions and access to several unique language constructs. Use C# for a traditional C-like syntax and gain additional functionality including pointer arithmetic that may improve performance in some specialized situations.

ASP.NET 5.0

The ASP.NET 5.0 Framework has been built from the ground up to be cross platform compatible for Linux and Mac OS X. It consists of modular components with minimal overhead. Moreover it has an optional Core CLR (cloud optimized runtime) that offers the flexibility to include only features required by an application, included as NuGet packages.

Windows Communication Foundation

Use WCF as a secure method for processing transactions and exchanging data in real time. The WCF Framework can be used to build service oriented applications through asynchronous messages between endpoint nodes. This is useful for creating a robust service-oriented architecture (SOA). Multiple message patterns are also possible, such as a duplex exchange pattern where two nodes establish a persistent synchronous connection.

Windows Presentation Foundation

Build client side graphical user interfaces (GUI) with WPF by creating intuitive interfaces for front end users. The WPF Framework has a toolbox of widgets already familiar to most users. The vector graphics system in WPF uses hardware accelerated for optimum GUI visualization and speed. Reusable templates assist the creation of interfaces by saving time when creating new layouts. Bind WPF controls to WCF data services for an effective combination of both technologies.

JAVA

Create powerful and platform-independent web applications built on JAVA frameworks, for software that can run on all modern operating systems. JAVA is an object-oriented, secure, distributed and robust framework which can be used to develop nearly all software related projects.

JavaServer Faces Framework

The JSF Framework boasts a clean separation between presentation and behaviour in web applications. It is highly extensible and offers custom tag libraries. By design, the JSF allows the creation of reusable components, many of which are available as open source from Apache, Oracle, Infragistics, etc. JSF has all the advantages of an MVC framework with an added component architecture to build the view.

Spring MVC Framework

The Spring MVC framework offers a powerful framework with the flexibility of opting in to many of its optional features. The framework is entirely based on interfaces and most of its MVC components are configurable by using a custom interface. Spring provides a clear division between views, controllers, and JavaBean modules. Moreover it is compatible with view technologies other than JSP, such as XSLT or Velocity.

Google Web Toolkit

The Google Web Toolkit (GWT) is a Rich Internet Application (RIA) Framework that Google itself uses on many of its web pages or web applications. It comes with a collection of widgets which can be easily customized and extended to handle particular use cases or to provide application specific functionality. Use XML syntax for creating interface controls and the JAVA language for implementing functionality.

Apache Struts2 Framework

The Struts2 web application framework has a sophisticated controller architecture, a highly reusable templating system and a solid MVC pattern. The framework also includes a robust and extensible data validation system which dynamically generates code for client and server nodes based on a configuration file. Struts also implements Action classes to uniquely respond to and validate client node HTTP requests.

Grails Framework

The Grails framework is ideal for rapid project development in small to medium sized projects. Use the Groovy syntax for more productivity in development. Integrate Grails seamlessly into other Java enterprise applications, as they can be extended without compromising the projects. Grails is also built on top of Spring MVC and may integrate Spring components.

PHP

Build websites and web applications using PHP, the most popular programming language for website backend development. Extend the functionality of PHP with modern frameworks which will implement industry accepted design patterns into web applications. PHP web applications are lightweight, easily portable and cross-platform compatible. Moreover, there are many other open source applications, libraries, frameworks, content management systems, etc. that

can easily integrate into existing PHP projects.

Laravel Framework

Develop projects using the most highly voted PHP framework to date. Laravel is built on top of Symphony components, and has dependency management powered by Composer. It boasts many great features including URL Routing, Blade Templating, MVC Architecture, Memcache Support, Authentication, Localization etc. Laravel provides features common to most enterprise PHP web applications to allow the rapid application of projects without reinventing the wheel.

Yii Framework

For projects requiring extremely optimized performance, Yii is the ideal solution. The framework offers a wide set of advanced features including: AJAX-enabled widgets, internalization and localization, skinning and theming, web services, XSS / CSRF security, form validation, etc. Yii is also a very intuitive framework that minimizes the complexity of project software development.

Drupal CMS

Develop small to medium sized projects in Drupal, an enterprise content management system offering advanced features for creating dynamic web sites. Drupal has simple management of users, accounts, groups and permissions. The CMS also supports many content types including videos, polls, text, blogs, podcasts, statistics, etc. Furthermore it has extensive API support for several external services common to web applications.

Wordpress CMS

Create simple websites or blogs with Wordpress, the world's most popular content management system. The CMS has thousands of plugins to support many common features in modern websites. There are also thousands of graphic templates available that can change a website's interface. Wordpress has a useful dashboard in which website moderators can make changes to the website without technical knowledge of programming or code.

Python

Python is a fully featured dynamic typed language that simplifies the development of software. It is an extremely robust and efficient programming language used for many enterprise level projects. Python boasts a massive selection of open source modules that can be imported into applications to gain instant functionality.

Django Framework

Develop with the powerful Django Python Framework for small to enterprise level web applications. Django includes a robust template system, a web portal accessible admin interface, and complete development environment. The framework also provides a unified Data

Access Layer (DAL) and a strong Object-Relational-Mapping (ORM) system which assist in data communication between a web application and a database server.

Web2py Framework

Use the Web2Py framework's unique approach to web development to better focus efforts on coding software rather than setting up and managing boilerplate code. The framework offers a powerful web-based Integrated Development Environment (IDE) that can be used to manage web applications through a web browser. Moreover the framework includes a Scaffolding apps based on Bootstrap for useful client side demo functionality.

Mobile Development

iOS iPhone and iPad

Develop mobile apps for iOS iPhone and iPad devices to gain access to the world's most popular AppStore accessed by the largest mobile user base. Take advantage of the iOS framework for operating system functionality accessible to both Objective-C and Swift programming languages. Take advantage of hardware acceleration on iOS devices that create stunning motion driven interfaces.

Objective-C Programming Language

Use Objective-C, one of the most popular programming languages in the world for iOS app development. Objective-C is well documented and is a time-tested solution for mobile development in iOS.

Swift Programming Language

Develop iOS apps much faster with the new Swift programming language which is faster and easier to use. Swift supports modern programming conventions such as type face, generics, closures, namespaces and multiple return types.

Android Phones and Tablets

Develop for Android mobile operating system, the world's most popular open source platform. Integrate Android mobile applications into hundreds of existing Android compatible mobile devices, with new devices constantly in production. Android application development has no restrictions on the developed software, and thus allows for various applications with no recurring license expenses. Application deployment and revisions are free and are not time consuming.

Android SDK

Develop using the Android Software Development Kit (SDK), a powerful development framework that can deploy to hundreds of compatible mobile devices. The framework uniquely utilizes the hardware capabilities of each device for optimal performance. It also adapts an applications user interface to best suit each device.

Spring for Android

Use the Spring for Android Framework for Android mobile development. The framework features a REST Client and Authorization support for accessing secure APIs. As with all Spring projects, a key advantage to the framework is in how easily it can be extended. Other advantages include: detecting device information, building RESTful web services, form input validation, Messaging with Redis and JMS, accessing external APIs, etc.

Windows Phones and Tablets

Take advantage of the new Windows Mobile OS with new interface features that makes it stand out from other platforms. The Microsoft Mobile operating system has software with which desktop users are already familiar, such as Internet Explorer, Media Player, Office, etc. Moreover, the behaviour and functionality is often similar to the Windows operating system. The platform also provides solid integration with existing Microsoft products such as Office, Exchange, SharePoint, and Azure. Microsoft Mobile is a trusted development platform because of its enterprise-wide security policy.

Windows Phone SDK

The Windows Phone Software Development Kit (SDK) simplifies development for the Windows mobile platform with a Phone emulator for desktop devices, and development tools integrated into Visual Studio. It also includes Microsoft Expression Blend for building complete user interface designs.

.NET Compact Framework

Maximize code reusability with the Windows .NET Compact Framework, designed to run on mobile devices with constrained resources. The framework has many of the same classes available on the full .NET Framework. Some libraries are specifically redesigned for mobile, as to use fewer resources.

Hybrid Frameworks

Create Hybrid mobile applications that will only be coded once and deployed to all modern mobile operating systems. Hybrid frameworks encapsulate web applications written in HTML, CSS and JS in native mobile apps. To end users, no distinction can be made between a native app or a hybrid app. Hybrid applications are an ideal solution for projects that are not dependent on hardware acceleration, such as video games or other visual animations. Use hybrid frameworks to access a phone's operating system features including: camera, GPS, contacts, etc.

Apache Cordova

Use Apache Cordova for a solid hybrid framework developed and maintained by the Apache Foundation. This open source framework has several other prominent frameworks that are built from it including PhoneGap, Ionic and Titanium, where each extension attempts to add its own style or proprietary features.

Adobe PhoneGap

Develop with Adobe PhoneGap, as one of the leading developers of hybrid frameworks. Adobe

was the initial owner of PhoneGap before offering it to Apache for free so that it can be Open Sourced and standardized. Now, Adobe PhoneGap extends Cordova adding additional features that may be beneficial to some projects.

Databases

Relational Databases

Relational Databases have been around for decades and provide an ideal solution for projects and web applications that require structured data. The foundation for these databases is relational algebra which specifies the relationships between various entities stored in the database. Advanced SQL search queries can be performed to join related data from various entities to find a particular subset of required information. Moreover, relational databases offer additional security, as sensitive data can be moved to separate entities with their own authorization controls.

Microsoft SQL Server

Integrate Microsoft SQL Server seamlessly into .NET projects through Visual Studio. SQL Server offers high performance solutions for small to enterprise solutions. Use SQL Server Management Studio (SSMS) as an integrated environment for configuring and managing all databases.

Oracle SQL Server

Create databases in Oracle SQL Server for cross platform compatible, medium to enterprise level solutions that operate in both Windows and Linux. Oracle server offers many advanced features such as partitioning, bitmap indexes, reverse key indexes, function-based indexes and star query optimization.

MySQL

Use MySQL as an industry standard open source database for small to medium sized applications. The MySQL database has many robust features able to handle data persistence and querying requirements for most applications. It is used as the default database by many frameworks and content management systems.

PostgreSQL

Use PostgreSQL as the most advanced, SQL-compliant, open source relational database. PostgreSQL has grown rapidly in the last several years and continued development is making it into a truly powerful and robust open source alternative.

No SQL Databases

Use a NoSQL Database to obtain an architectural advantage in specialized applications. NoSQL Databases offer many benefits including efficient scaling, fewer hardware requirements, flexible data models and fewer human resources required to manage the systems. Moreover, NoSQL databases are mostly all open source and can be downloaded and customized at little to no cost.

SOLR Search Engine

Use the SOLR database to index large amounts of content for advanced search functionality. SOLR search capabilities are built upon the Lucene database and include phrases, wildcards, joins, grouping etc. The database is highly optimized for high volume traffic and is highly scalable. SOLR indexes many types of data (JSON, CSV, XML, etc.) and includes rich document parsing (PDF, Word, Excel, etc.) SOLR is also a great option for using facets to achieve enterprise level filtering of data.

MongoDB Document Store

Use MongoDB to store documents by pairing keys with complex data. The data stored in these documents need not be structured, as each document is an independent unit that is stored and queried optimally by the database engine. MongoDB stands out from other document databases because of its many features including support for rich queries, full indexes and a minimal need for an ORM layer.

Neo4j Graph DBMS

Use Neo4j to store complex information about data that can be represented as a network data model. As the world's most prominent graph database, Neo4j takes an intuitive and natural approach to structuring data. For projects requiring complex relationships Neo4j offers extremely optimized querying algorithms, as opposed to traditional RDBMS join queries.

Redis Key-Value Store

Use the Redis database for highly optimized key-value storage of data. It can store varying data structures such as hashes, arrays, bitmaps, etc. which become serialized upon storage and unserialized upon retrieval. Redis is also an ideal solution for caching data to memory, such as web application views or temporary dynamic data that is updated at an interval.

Networking

HTTP Web Servers

Deploy websites, web applications and web services to a robust HTTP server that is custom configured to optimize the processing of HTTP requests and to maximize overall system performance. Serve various content types and parsed scripts through distinct bindings comprised of URLs, ports and protocols. Use detailed configuration files to setup the behaviour of servers or to set a configuration specification for a particular project. Prevent unauthorized access to system resources by implementing tight security policies, and take proper measures when internal errors occur.

Apache HTTPD Web Server

Serve projects over the Apache HTTPD Web Server for an open source, cross-platform compatible solution. The Apache web server is the internet's most commonly used web server. It is typically used collaboratively with other open source software to create bundles such as the LAMP stack. Moreover, the server is highly configurable, providing many optional modules that can be configured for specialized applications.

Internet Information Services (IIS)

Integrate .NET applications seamlessly into Internet Information Services for the ideal solution to host .NET projects over the internet. IIS includes many useful modules for server security, data caching & compression, URL redirects, server diagnostics and more. Moreover, IIS supports ASP.NET integration for custom configuration of projects completed using the .NET Framework.

Nginx Web Server

Use Nginx as the fastest growing, open source, cross-platform compatible HTTP Server. Nginx is gaining momentum with an industry proven reputation for being the fastest web server available through its event driven architecture. The server is also uses very few hardware resources, and is excellent at multitasking.

Apache Tomcat Web Server

Use Tomcat as an open source Java Servlet container for small to medium web applications. Tomcat has a light memory footprint and has fewer hardware requirements making it an ideal choice for many web applications or Java frameworks such as Spring. Although Tomcat is typically used with Linux, native wrappers are available for Windows and Unix integration.

Oracle GlassFish Web Server

Deploy enterprise level Java applications requiring the full Java EE stack using GlassFish web server. GlassFish fully complies with Java Application Server specifications and supports all Java EE features. The server supports application clusters for high availability, and enables modular architecture while reducing overhead. GlassFish also provides a helpful administration using command line tools or various web-based tools.

HipHop Virtual Machine (HHVM)

Use the PHP programming language for enterprise level applications by deploying on Facebook's HHVM web server. The HTTP web server compiles PHP code to C++ for performance multitudes times faster than its Zend counterpart. It has been thoroughly tested to work on popular PHP frameworks such as Laravel, Wordpress, Drupal, etc.

Load Balancing

Achieve high levels of fault tolerance with a custom configured software load balancer. Distribute processing of work optimally to sibling network server nodes to handle a large number of requests. Use load balancers for a number of other optimizations such as caching of static content, compression of HTTP traffic, SSL offloading and providing a new layer of security. Optimize architectural network infrastructure with Layer 4 or Layer 7 load balancing depending on the system requirements. Avoid common pitfalls by ensuring HTTP sessions are persisted correctly and that SSL connections are offloaded.

Nginx Load Balancer

Nginx is an ideal solution for a highly configurable load balancer that has HTTP web server features such as caching and compression. The server has many useful content processing modules such as header rewriting or content compression. The Nginx load balancer is highly extensible with plugins for routing traffic.

HAProxy

Use HAProxy for industry proven high performance load balancing. HAProxy has native support for SSL, IPv6, UNIX sockets, client-side keep-alive, response buffering, ACL-based persistence, etc. The load balancer also has useful features such as a Health Check to determine if system nodes are available for workload processing.

Virtualization

Optimize the use of hardware resources with effective virtualization strategies that will maximize the possible amount of performance for a physical device in such a way that more users have access to the device in separate virtual instances. Virtualization has many benefits such as:

- Fewer hardware resources required per virtualized instance
- Prepares a system for the migration to the cloud
- Extend the life of legacy software
- Removal of dependencies on specific hardware platforms
- Better disaster recovery methods

- Isolation of software for security, fixed resources etc.

Oracle Virtual Box

Manage truly portable virtual instances with Oracle's Virtual Box, an open source type 2 hypervisor virtual machine. Create multiple virtual instances in a host environment and migrate them seamlessly to another. The virtualization software is further enhanced by cross platform compatibility which allows virtual instances to run hosted on the virtual machine under various modern operating systems (including Windows, Linux and Mac OS X).

VMware vSphere

For enterprise level system architecture, leverage the power of VMware vSphere, a suite of type 1 hypervisor virtual machine software. The powerful hypervisor can be used to create a virtualized private cloud infrastructure through a bare metal native hosting environment. vSphere features a small memory footprint, efficient resource virtualization, advanced memory management, resource management, scalability, interoperability, and other enterprise level benefits.

Mail Servers

Expand internal private networks with self managed mail servers for maximum flexibility of e-mail features and privacy control. Use a web based administrator console to create and manage email accounts and to specify advanced e-mail configurations. Implement effective anti-spam strategies to ensure that junk e-mails do not reach the inbox. Moreover, implement appropriate DNS settings and network security policies to ensure that outgoing e-mails reach the inbox of the receivers and not the spam folder. Prepare for disaster recovery and troubleshooting with a mail backup server to ensure that mail server communication protocols are always functional.

Zimbra Collaboration Suite

Use the open source Zimbra mail server for a flexible Linux hosted solution. Zimbra supports modern protocols including SMTP, POP3, IMAP, IMAP IDLE, SMTP over TLS, POP over TLS, and SSL. It furthermore provides the flexibility to access e-mail data from a web mail interface, through ActiveSync or directly through the mail server database. The web based user interface provides a modern platform that is feature-rich and mobile compatible. Zimbra provides authentication over LDAP, Open LDAP and Active Directory.

Microsoft Exchange Server

Integrate Microsoft Exchange server seamlessly with other Microsoft products, such as the Windows operating system and Internet Explorer. Exchange Server is a Windows hosted platform that supports modern mail protocols including SMTP, POP3, IMAP, IMAP IDLE, SMTP over TLS, POP over TLS, and SSL. It features a friendly webmail interface for web users and also ActiveSync for synchronization with mobile devices. It also allows authentication over Active

Directory and NTLM.

Cloud Infrastructure

Amazon Web Services

Deploy entire IT systems on Amazon Web Services, the global leader in cloud based architecture. Amazon is in continuous state of innovation and improvement, routinely adding new cloud services and lowering costs. AWS has the highest number of cloud based services that allow complex custom designed IT infrastructures. Moreover Amazon offers seamless integration between its services, ensuring that cloud instances such as web servers, databases, DNS, load balancers, etc are easily intercompatible. Using AWS has many other advantages such as:

- The widest range of datacenter geographical region availability
- Multi-region low-latency high-bandwidth services
- Routine and thorough security audits
- Solid infrastructure, being the first company to offer cloud based services

Amazon EC2

Run virtual servers in the cloud with Elastic Compute Cloud cloud service. Dynamically build server instances with custom operating systems, software and hardware resources. Dynamically resize the hardware specifications of EC2 instances to match system architecture requirements. EC2 supports a variety of instance types for distinct combinations of CPU, memory, storage and networking capacity. Instance types include: T2 and M3 - General Purpose, C3 - Compute Optimized, R3 - Memory Optimized, G2 - GPU Optimized, and I2 and HS1 - Storage Optimized.

Amazon S3

Add public domain storage of static website files with Simple Storage Service. S3 provides a web interface to retrieve and store data such as files, zipped versions of data sets, arbitrary string of bytes, etc. S3 objects can be accessed on the public web. A single S3 instance can be used for multiple EC2 instances. Moreover, S3 instances can be integrated as an external service with network infrastructure outside of AWS. For non-critical, reproducible data use S3's Reduced Redundancy Storage (RRS) to reduce costs by storing data that is replicated fewer times.

Amazon Elastic Block Storage

Attach network persistent storage using EBS for traditional file system capabilities. EBS volumes can be attached to single EC2 instances, and can continue to persist data even if the EC2 instance is shutdown. EBS volumes are ideal for system architectures that require constant I/O throughput such as persistent critical data that must be stored privately with the system. EBS offers varying levels of I/O performance, depending on a system's requirements.

Elastic Load Balancing

Dynamically route web traffic to available EC2 instances depending on node workload and regional availability. ELB ensures that only healthy instances receive traffic while unhealthy

instances are ignored. When system performance slows because instances cannot handle the workload, the Auto Scaling service can be used with ELB to dynamically create new EC2 instances. Contrarily when traffic is reduced, the number of EC2 instances can be scaled down. Auto Scaling ensures system architecture can respond to all levels of traffic in a cost-effective manner and without requiring manual intervention.

Digital Ocean

Deploy web applications on Digital Ocean's high performance Linux OS cloud instances (called droplets). Create a custom designed droplet by choosing from a selection of hardware resources, regional availability and pre-installed software including a Linux distribution and other optional software packages. All droplet instances offer SSD hard drives for extremely fast I/O performance. Once configured a droplet can be created in less than 55 seconds with immediate root access. Digital Ocean is an ideal solution for:

- Quickly starting up Linux based projects
- Lowest prices in the industry
- High performance virtual machines
- Custom designed system architecture that does not require automated services included by other cloud service providers

Private Network

Send traffic between droplets located in a specific regional datacenter using Digital Ocean's internal private network. Any data interchange performed over the private network does not contribute to droplet bandwidth usage. Private network traffic also benefits from extremely fast speeds. It is ideal for database replication, file storage, and host-to-host communication.

Droplet Snapshots

Multiple snapshots can be made for a droplet to save a copy of a particular droplet instance at a particular time. Digital Ocean snapshots are backed up remotely to Amazon Glacier cloud service for as an added layer data redundancy. Snapshots can be used to restore an existing droplet to the snapshot, or in the creation of a new cloned droplet.

Digital Ocean API

Manage droplets and other resources using Digital Ocean API, for a programmatic solution to completing any action that can be performed manually though the account control panel. The API is accessed over standard HTTP requests using wrappers that can be useful for integrating the API into a software application. Using the API, autonomous or event driven actions can be performed to update or manage the system infrastructure without requiring manual intervention. An ideal use for this to offer to clients software products that are created and customized on-the-fly in an isolated virtual instance.

E-Commerce

Payment Solutions

Accept credit cards and other form of payments directly form websites or mobile applications. Online money transfers are performed by external internet payment companies through an API embedded into the local website or application. The companies processing the payments guarantee the transfers are secure through robust encryption algorithms. Furthermore, they accept liability for possible fraudulent activities.

Paypal

Use PayPal as a widely accepted industry standard for electronic funds transfer. The company has the largest number of users registered for a internet payment company, allowing quick purchases in applications integrated with PayPal. Fortunately, PayPal does not require an account and allows anyone with a valid credit card to instantly make a purchase. Users with registered accounts can also make a payment through bank accounts, eChecks and personal PayPal accounts.

Stripe

Leverage full control over customer e-commerce transactions using the powerful Stripe API. Using Stripe, the user never leaves the website or application as the API is fully integrated within it. The API allows custom control of each step of the e-commerce transaction: custom graphical user interfaces, form fields & messages, analytics on trends and conversions, etc.

Google Wallet

Integrate Google Wallet into web or mobile applications using the Instant Buy API for a cloud-based technology that accesses, stores and processes payment information. Google Wallet easily stores debit cards, credit cards, gift cards etc. on mobile devices. Using Instant Buy APIs allows the integration of Google Wallet into mobile applications for instant transactions using the payment methods collected. Furthermore using the single Google account sign-in also allows web based users to easily make payments online using Google Wallet.

Apple Pay

Use Apple Pay to instantly make a payment transactions on websites and web applications. Payments are made simple with TouchID, a mechanism in which funds are instantly processed on the phone by verifying the fingerprints of users. Apple Pay can be integrated into Stripe for instant processing of goods and services in custom mobile applications.

Shopping Cart

Create effective online shopping solutions for websites and mobile applications using e-commerce platforms. Process customization of online orders from a catalogue of products and

services, use an online shopping cart and process transactions quickly and securely using common payments solutions.

Shopify

Quickly set up e-commerce websites using the cloud hosted platform Shopify. The technology offers a cost-effective solution to setting up online stores without creating everything from scratch. Shopify offers many themes that can be easily customized, eliminating the need for custom application development and graphic design. It is SEO friendly, has its own app store and provides a friendly admin panel to manage e-commerce activities. Shopify can also accept payments from credit cards or from a POS card reader that can be added as an iPad extension.

Magento

Sell products using the fully customizable Magento platform. The technology allows complete customization of the platform for complete brand integration. Magento is open source and community powered, boasting many extensions that have been created by companies using the platform. As an open source technology, it is completely extensible to support specific project requirements. It also boasts solid architecture that allows it to scale well and to support simultaneous transactions flawlessly.