

ASP Application

As the Internet grew into a major player on the global economic front, so did the number of investors who were interested in its development. So, it may be wonder, how does the Internet continue to play a major role in communications, media and news? The key words are: Web Application Projects.

Web applications are business strategies and policies implemented on the Web through the use of User, Business and Data services. These tools are where the future lies.

Who Needs Web Applications and Why?

There are many entities that require applications for the Web-one example would be Business-to-Business interaction. Many companies in the world today demand to do business with each other over secure and private networks. This process is becoming increasingly popular with a lot of overseas companies who outsource projects to each other. From the simple process of transferring funds into a bank account, to deploying a large scale Web services network that updates pricing information globally, the adoption of a Web applications infrastructure is vital for many businesses.

The Web Application Model

The Web application model, like many software development models, is constructed upon 3 tiers: User Services, Business Services and Data Services. This model breaks an application into a network of consumers and suppliers of services.

The User Service tier creates a visual gateway for the consumer to interact with the application. This can range from basic HTML and DHTML to complex COM components and Java applets.

The user services then grab business logic and procedures from the Business Services. This tier can range from Web scripting in ASP/PHP/JSP to server side programming such as TCL, CORBA and PERL, that allows the user to perform complex actions through a Web interface.

The final tier is the Data Service layer. Data services store, retrieve and update information at a high level. Databases, file systems, and writeable media are all examples of Data storage and retrieval devices. For Web applications, however, databases are most practical. Databases allow developers to store, retrieve, add to, and update categorical information in a systematic and organized fashion.

Choosing the Right Project

Choosing the right types of projects to work on is an extremely important part of the Web application development plan.

Assessing to the resources, technical skills, and publishing capabilities should be first goal. Taking the 3 tiers into consideration, devise a list of all available resources that can be categorically assigned to each tier.

The next consideration should be the cost. Do we have a budget with which to complete this project? How much will it cost us to design, develop and deliver a complete project with a fair amount of success? These are questions that should be answered before signing any deals or contracts.

Let's look at an example. A company called ABC needs to develop a Web application that will display sales information created by different sales agents. The data is updated daily through a completely automated process from all 3 service tiers. The client tells developer that this entire project must be done in ASP/SQL server and that developer should host the application as well.

After assessing all resources, developer and his/her team come to a conclusion that the company is unable to do data backups on a daily basis. After further discussion, s/he realizes that this is a

very important part of the setup for his/her client, and s/he should not risk taking a chance with the project. It's very likely that developer will be more prepared next time around, when a similar project lands on desk, so decline the job and recommend someone else who has the capabilities to do it right now.

The Phases in a Web Application Project

The Web application development process has 4 phases:

1. Envisioning the nature and direction of the project
2. Devising the plan
3. Development
4. Testing, support and stability

Let's look at each of these in more detail.

1. Envisioning the nature and direction of the project

In this phase, the management and developers assigned to the project come together and establish the goals that the solution must achieve. This includes recognizing the limitations that are placed on the project, scheduling, and versioning of the application. By the end of this phase, there should be clear documentation on what the application will achieve.

2. Devising the plan

In this phase, developer and developer team must determine the "how's" of the application.

What scripting language is most appropriate, which features must be included, and how long will it take? These are some of the questions that must be answered through this planning phase. The main tangents at this point are the project plan and functional specification. The project plan determines a timeframe of events and tasks, while the functional specification outlines in detail how the application will function and flow.

3. Development

Once the project plan and functional specification are ready, a baseline is set for the development work to begin. The programmer/s or Web developer/s begin coding, testing and publishing data. This phase establishes the data variables, entities and coding procedures that will be used throughout the remainder of the project. A milestone document is prepared by the development team, which is then handed to management for review.

4. Testing, support and stability

The stability phase of the application project mainly focuses on testing and the removal of bugs, discrepancies and network issues that may otherwise cause the application to fail. It is here that policies and procedures are established for a successful support system.

Knowing Options and Using them Wisely

Understanding of the architecture and procedures behind Web application development is like what technical options is needed to consider for the development process itself.

Windows Web Servers

Microsoft has built a loyal customer base on one important factor – their easy-to-use software. Windows NT/2000/XP Web servers are very fast and easy to administer. The fact that the operating system is a Windows shell means that administrators and authors can easily allow the Web server to interact with other software and hardware applications to transmit and receive data over the Internet. Popular server side scripting languages used with Windows servers are ASP/ASP. net, Java Server Pages, and PHP.

UNIX/Linux Web Servers

UNIX has long been known for its reliability. It is a powerful and robust Web server and operating system. Unix is the server of choice for many large-scale Websites that need content management systems or receive an extremely high volume of traffic. Popular server side scripting languages for UNIX are Java Server Pages, PERL, PHP, and CORBA

Every scripting language has its pros and cons. As I'm not writing a book here, I'll use the ASP model as my language of illustration. When working with Windows servers, there are several important parameters that the developer needs to throw into the equation, including security, scalability, speed and application design. So below I'm going to help you formulate a successful plan to accomplish all kinds of Web projects.

Planning for a Successful Web Development Project

In order to drastically minimize the risk of project failure, application development projects in the following sequence may be good approach.

1. Identify business logic and entities

Start by gathering information on everything. If we are going to be working with databases, begin by enumerating how many entities will be used in the business logic. For example, if program implements sales data, a sales ticket would be an entity.

Once all entities have been identified, establish a clear guideline for their relationships. This can be done via presentations, flowcharts or even reports.

2. Create a functional specification and project plan

This part is the most important part of the project. Functional specifications (or functional specs) are a map, or blueprint for how particular Web application looks and works. The spec details what the finished product will do, user interaction, and its look and feel.

An advantage of writing a functional spec is that it streamlines the development process. It takes discrepancies and guesswork out of the programming process, because the level of detail that goes into the plan makes it possible to minimize the misunderstanding that's usually associated with project mishaps.

Once the functional spec is finished, a project plan must be devised. A project plan is a timeline of tasks and events that will take place during the project. The project or program manager is normally the person who creates a project plan, and their primary focus is to detail task notes while being able to accommodate scheduling and resource information.

3. Bring the application model into play

As discussed earlier, the application model consists of 3 tiers – The User, Business and Data service tiers, each of which serves a substantial purpose.

Practically speaking, it's always best to start with the data tier, because entities and their relationships are already identified. The data tier can be an SQL server database, a text file, or even the powerful and robust Oracle. Create tables, relationships, jobs, and procedures depending on what platform chosen. If the data is a warehouse (i.e. the data already exists and does not depend on real time interaction), then make sure that new and additional data can be added securely and in a scalable fashion.

The Business services tier is the heart of the application. It involves the implementation of business logic into the scripting or programming language.

At this stage, make environment for testing and debugging has already been setup. Always test on at least 2 instances in application, after all, what may work perfectly, may not do so well on other platforms or machines. ASP, XML, PHP, JSP and CGI are some examples of server side

scripting languages used at the business service level. Whichever language choosen, make sure that it's capable of handling all the business logic presented in the functional specification.

The last is the user tier, which is absolutely vital for the interactive and strategic elements in the application. It provides the user with a visual gateway to the business service by placing images, icons, graphics and layout elements in strategic areas of interest, most commonly, based on management research.

4. Develop a support scheme

Being able to support and stabilize application is very important. Define a procedure call for cases of failure, mishaps or even downtime.