

Web design report

[Technology](#), [Internet](#)



SDLC

System analysis and design involves step by step processes for developing high quality information systems and websites. An information system combines technologies, people and data to support business processes.

Structured analysis uses a series of phases known a system development life cycle to plan, analyze, design, implement and support a website. SDLC models use five steps that include; web planning, web analysis, web design, web implementation and web security and support.

Analysis stage requires the gathering of information and particulars pertaining the web site to be developed. The information is solicited from the web users or the business entity. Design stage involves the iterative process of graphically representing appealing and appropriate features for the intended purpose. Development involves the assembly of all the requirements to satisfy the business course. Finally, a prototype test is done to ascertain the performance of the website and its usability.

WEB AND SECURITY AUTHENTICATION

The process of authenticating user's online to verify their identity is a security measure included in many websites. Internet users registering to the internet need to provide an online service known as relying party with personal information to prove their identity. For instance, a holder of selected HTTP details related to a web authentication request is executed to ascertain website credentials.

BASIC FEATURES OF INTERNET TECHNOLOGIES

HTML

According to , Hypertext Markup Language is the language for creating web pages. Data for a web page are formatted for interpretation by a browser using HTML language. For example to get some text displayed boldface with HTML, we use boldface tags at the beginning and at the end of the text. This is illustrated as shown below;

HTML uses the ASCII for both the main text and formatting instructions thus eliminating incompatibility issues that may occur with different computers word editors.

Web [pages are essentially made of two parts comprising the head and the body. The head comprises of the title and other parameters used by the browser. The body comprises of the text and the tags. Tags represent the appearance of the documents while the text represents the actual information contained in the text. Every HTML tag consist of name followed by an optional list of attributes all enclosed between less than and greater than symbols < >.

CSS

Cascading Style Sheets are simple mechanisms for adding styles to a web document. The styles include fonts, colors, spacing, italics and so forth. CSS is a style sheet language for describing the presentation semantics of a document written in markup language.

CSS is primarily intended to differentiate the document content from the document presentation. This is meant to improve accessibility, control of characteristics and flexibility.

Selectors are used in CSS to declare which part of the markup document should be formatted with a certain style. A declaration block, for example, consist of declaration braces where each declaration comprises of a property, colon (:) and a value. The figure below illustrates a block declaration and a CSS.

DNS

Domain name system is a hierarchical form of naming computers, resources and services connected via the internet or a network. DNS translates easily remembered domain names to IP addresses which are assigned to each and individual computer in the world.

DNS is a subset of the internet namespaces and provides address spaces between it and the address spaces. DNS is implemented through communication protocols and internet name servers.

COOKIES

WWW platform was originally designed as a stateless entity but with the evolution of the web to incorporate other functions such as website user access authentication, e-commerce, portals and marketing, cookie mechanisms have been devised.

Cookies are created when a server receives a request from the client. The server stores the information about the client in a file or string and such information may include domain names, cookie contents, and timestamps and so on. The server then includes the cookie response in the response it sends to the client and when the client receives the response, the browser stores the cookie in the cookie directory which is later sorted by the domain

name.

HTTP is a stateless protocol that does not record the web server requests made by the browser. It does not have the capability to preserve state or memory implying that a browser requesting a HTML document and images will receive the files from the Web server without realizing that the requests are from the same browser. By doing so it preserves the server resources. However as web developers, server-side state is required to remember particulars such as who is currently logged in, what a visitor did in the website and so forth. Being able to know and identify the visitors to a website create room for web application contents matching the users, preference, current status in the middle of multi-tasking and security synchronization.

In order to preserve state on top of HTTP across a multi-page website, a number of techniques are put in use. The most favorite of them all is browser cookies. A browser cookie is a text string with no more than 4KB in size. The browser cookie is sent from the Web server to the browser by adding the field " set-cookie" to the HTTP response header and is replied by the browser to the web server by adding the field " cookie" to the HTTP GET or POST header.

Browser cookies have different attributes such as domains, directory paths, expiry date and security attributes that record state information . A cookie can be transmitted back and forth between browsers and web server. The steps for such transmission are illustrated as shown.

The browser requests a file e. g. fon. php from the web server by initiating a GET command

The web server responds to the request by sending HTML rendered from the `fon.php` file, and adds a `Set-Cookie` directive to the HTTP response header.

The `Set-Cookie` directive comprises of all the cookie parameters.

The browser shows the requested HTML document and stores the cookie on the disk if an expiry date is indicated otherwise in the volatile memory

A user clicks a link `cheap.php` on the displayed HTML document prompting the browser to send a request for the document using the GET command as initially but this time the cookie is send to the server given that the cookies domain and the path attributes are compatible. A cookie directive to the HTTP request header makes it possible.

Session management practices are applied by web site designers to identify and distinguish every visitor to a website and store the information as the visitor navigates from one page to the other throughout the website.

A method of preserving state is necessary and in order to avoid a state of transmitting all the variables back and forth between the web server and the browser, a unique string or number is used to identify a particular session to be transmitted. All the web server has to do is to find the identifier and load the sessions corresponding state information.

Session management provides an avenue for improved performance, support for wizard like user interfaces, support for intermediate results and personalization . Since much of the session state can be loaded by the server side scripting platform, expensive database connections are eliminated when restoring the state information. Also, most web applications use wizard-like user interfaces comprising of many pages which are concurrently required to perform an operation. As the visitor advances on the webpage, actions

taken on the previous sections need to be recorded until the visitor arrives at the final page where it is safely written in the databases. Likewise it allows the storing of the session state in a more durable fashion so that a visitor who suspends a session will be able to restore the state when they get back to the website. Through additional programming lines and server-side scripting platforms, this functionality is enabled.

BASIC PROGRAMMING CONTROL STRUCTURES

Control structures are programming statements or group of statements which determine the sequence of execution of other instructions. A control structure analyzes variables and chooses a direction in which to go based on a given parameter or condition.

IFELSE CONTROL STRUCTURE

For instance, a control structure in a Java that assigns the age and decides whether it is a teenager or not is illustrated.

If the input age is less than 20 and greater than 12 then the program executes the first segment under the curly brackets { }. If the condition is not satisfied, the program will execute the second line of segment.

The choice of variables to execute and those to skip in a program is dependent on the control structures

SEQUENCE CONTROL STRUCTURE

A sequence is a series of steps such as calculations, manipulations and running subroutines and functions. A sequence must have at least one entry point and only one exit point. Sequence commands are used for simple computations or other basic computer operations. The controls execute

logical (AND, OR, NOT, NOR), rotation or shift, single text substitution, deletion, insertion, modification, and concatenation of a piece of data into a database tree, list or structured.

LOOP

Loop control structures execute code between the {} braces until the condition become false. Loop is one of the most powerful features of a computer as it has the ability to execute over and over. Basic loop instructions include DO LOOP, DO WHILE and DO UNTIL. Loop count variable should have a starting point and a finish value where the starting variable is called the initial variable and the finishing value called the limit value.

Control structures are immensely important to programmers and web designers because they ensure that programs function properly. With the absence of control structures users would, for instance, be able to log in to accounts whether they provide the correct password or not.

OBJECT ORIENTED PROGRAMING

OOB is a programming language model that is modeled around objects rather than actions and data rather than logic. Historically, programs were termed as logical procedures than accepts input, executes it and produces outputs. Thus the fundamental challenge behind programming was how to write the logic and not how to define the data. OOP encapsulates a different view that the fundamental objective is how to manipulate the objects rather than the logic required manipulating them.

Objects are defined as those that can perform a set of related activities. The set of activities performed by the object determines the behavior of the

object. An example of an object is a student who can give a name or address.

A class is a representation of the type of object. It can be termed as the blueprint or template that gives the details of the objects. Classes are comprised of three things including the names, attributes and operations. In object oriented terms, the student object named objectStudent was created out of the student class.

References

B. B. Agarwal, S. P. (2010). Software Engineering and Testing. Jones & Bartlett Learning.

Carroll, J. M. (2003). HCI Models, Theories, and Frameworks:. Morgan Kaufmann.

Gary B. Shelly, T. J. (2010). Systems analysis and design. Cengage Learning.

Powers, D. (2010). PHP Solutions: Dynamic Web Design Made Easy. Apress.

Robert W. Proctor, K.-P. L. (2008). Handbook Of Human Factors In Web Design. Routledge.

Shelly, G. B. (2012). Systems analysis and design (9th ed.). Boston: Thomson Course Technology.

Shivaraju. (2012). Database Systems: Design, Implementation, and Management [With Access Code]. Cengage Learning.