WEBSITE DEVELOPMENT FOR GRACE BAPTIST CHURCH

by

DAYOU JIANG

B.A., Xiamen University, 2007

A REPORT

Submitted in partial fulfillment of the requirements for the degree

MASTER OF SCIENCE

Department of Computing and Information Sciences
College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2013

Approved by:

Major Professor

Torben Amtoft

Abstract

Grace Baptist church is a Christian church that is actively reaching out to college students, military families and other groups of people. They aim to impact not only the community of Manhattan and nearby areas but also to reach around the world with the Gospel. It is very important for the church to have a user friendly website so as to introduce Grace Baptist Church to people who may have an interest in Christianity and church activities. Considering the limitations of the current church website, the administrator of Grace Baptist Church decided to have a new website that better represents the church and attract people to know more about the church's mission. I and another student Nikhita Addanki from computer science department were working together on this project.

The main objective of this project is to redesign and implement a website for Grace Baptist Church. The website will be used to post updated information about church events, announcements and opportunities, share videos and blogs, and let people around the world know about Grace Baptist Church and its mission. The new website has a better page appearance, a well organized structure and mobile support. It is implemented using WordPress, a free content management system, so that people from the church can early update the information without knowing much about web techniques.

Alone with the implementation, I also reviewed some articles regarding website usability and performance to assure the quality of the new website. A performance test has also been conducted to evaluate the result.

Table of Contents

List of Figures v

List of Tables vi

Acknowledgements vii

Chapter 1 - Project Description 1

1.1 Introduction 1

1.2 Motivation 1

Chapter 2 - Literature Review 6

2.1 Web usability 6

2.1.1 The Common definitions of Web Usability 6

2.1.2 How and why website usability has been important? 6

2.1.3 The general factors influencing web usability and how to improve it 6

2.2 Website Performance 11

2.2.1 How to measure web performance 11

2.2.2 The general factors influencing web performance 12

2.2.3 The ideas behind these 14 rules 13

Chapter 3 - Requirement Analysis 17

3.1. Requirements Gathering 17

3.2. Software Requirements 17

3.2.1 Web Content Management System 17

3.2.2 WordPress 17

3.2.3 Other Technologies and software used 18

Chapter 4 - Implementation 19

4.1 Convert PSD designs into HTML Website templates 19

4.2 Create WordPress template 23

4.2.1 How does WordPress template work? 23

4.2.2 The WordPress Page Structure 24

4.2.3 WordPress Template Hierarchy 25

4.2.4 Custom WordPress Template implementation 27

4.3 Mobile support 27
Chapter 5 - Performance Testing 29
5.1 Apache JMeter 29
5.2 Performance test plan and result interpretation 30
Chapter 6 - Conclusions and Future Enhancements 39
6.1 Conclusions 39
6.2 Future Enhancements 39
Chapter 7 - References 40

List of Figures

Figure 1.1: Unattractive current website home page 2

Figure 1.2 Contains too much information in one page 3

Figure 1.3 Confusion subscription functionality 4

Figure 1.4 Current website in Iphone and Ipad.. 5

Figure 2.1 Characteristics of a Usable Website 7

Figure 4.1 Web page template – content page, header, content and footer 20

Figure 4.2 Web page template – Home page.. 21

Figure 4.3 Web page template – Second page 22

Figure 4.4 Web page template – content page 23

Figure 4.5 WordPress Template Hierarchy 26

Figure 4.6 Custom WordPress Template structure 27

Figure 4.7 New website and current website in Iphone 28

Figure 4.8 Mobile site of the new grace Baptist church website 29

Figure 5.1 JMeter test result - Home page (5 user)... 31

Figure 5.2 JMeter test result - Home page (10 user)... 31

Figure 5.3 JMeter test result - Home page (20 user)... 32

Figure 5.4 JMeter test result - Home page (40 user)... 32

Figure 5.5 JMeter test result - Home page (80 user)... 33

Figure 5.6 JMeter test result - Search page (5 user) 34

Figure 5.7 JMeter test result - Search page (10 user) 35

Figure 5.8 JMeter test result - Search page (20 user) 35

Figure 5.9 JMeter test result - Search page (40 user) 36

Figure 5.10 JMeter test result - Search page (80 user) 36

Figure 5.11 Summary of the total number of HTTP requests and the total weight of the Home page as well as page weight analysis for both Empty Cache and Primed Cache perspectives 38

List of Tables

Table 2.1 Ten principles of web usability 9

Table 2.2 Checklist by web usability testing institute, University of Wisconsin-Stout 11

Table 2.3 Fourteen rules to improve website performance 13

Table 5.1 Home page test results summary 33

Table 5.2 Search page test results summary 37

Acknowledgements

I would like to take this opportunity to extend my deepest gratitude to my major professor Dr. Torben Amtoft, for all the guidance and encouragement he provided through the project. I am also grateful to my committee members Dr. Daniel Andresen and Dr. Mitchell L. Neilsen for their valuable advices and supports for this project. Furthermore, I would like to thank Mr. Dan Learned, the ministry administrator of Grace Baptist Church and Mr. Alan Agerling for providing this project and giving all the designs for the website implementation. At last, I want to thank Nikhita Addanki, we have been worked on this project as a team, and she had a lot of ideas for this project and did a lot of work on it.

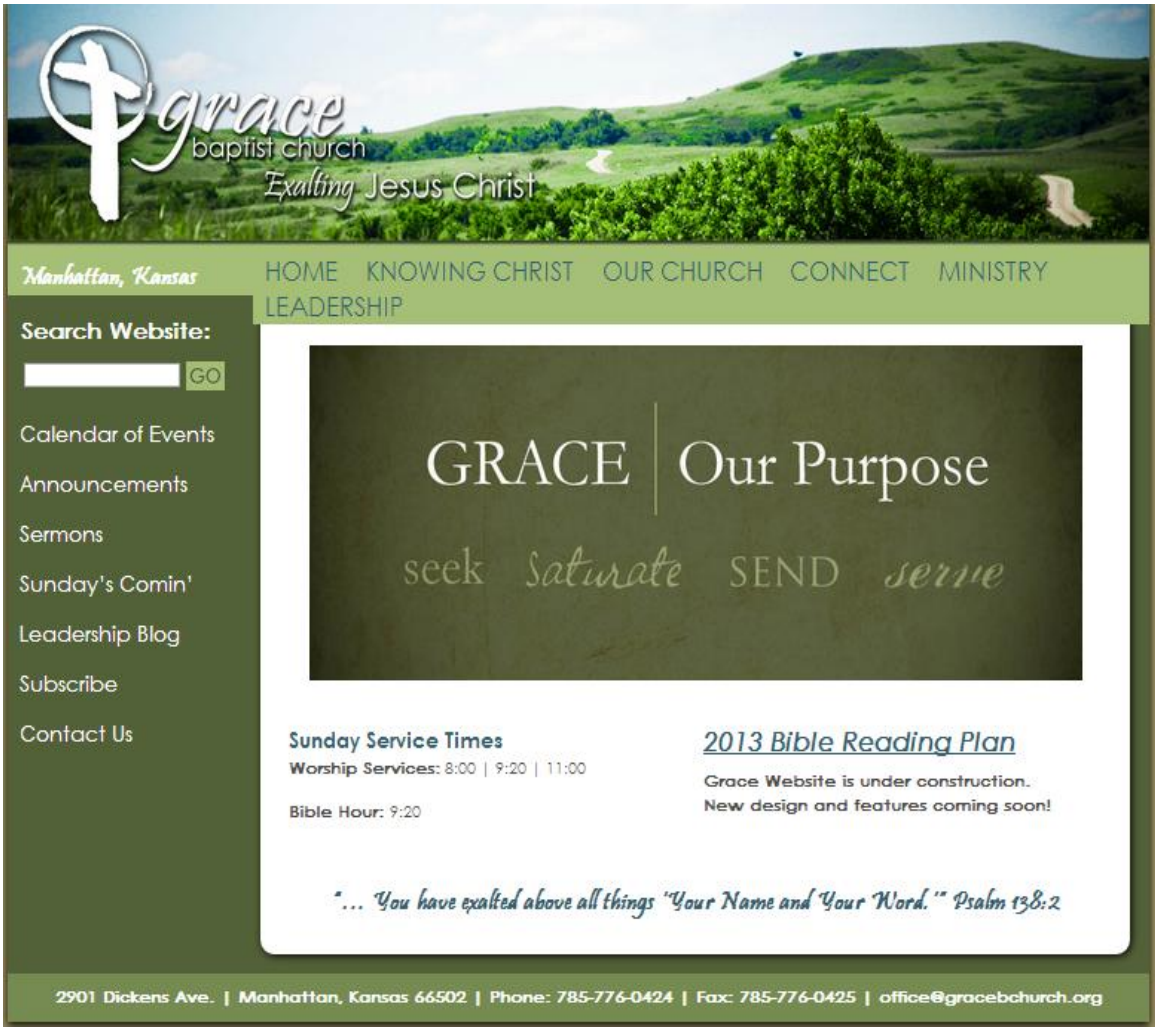
Chapter 1 - Project Description

1.1 Introduction

The current project involves the redesign and implementation of a website for Grace Baptist Church at Manhattan. This website is used to give people information about Grace Baptist Church, which includes the introduction of Grace Baptist Church, information about events, announcements and opportunity, some videos and blogs, subscribe functionality and location and contact information.

1.2 Motivation

The current website of Grace Baptist Church has some flaws such as its page appearance is not attractive (see Figure 1.1), the structure is not well organized and it is difficult to look for information (see Figure 1.2), some confusion implementation (see Figure 1.3), and mobile support insufficiency (see Figure 1.4).



Sunday Service Times
Worship Services: 8:00 | 9:20 | 11:00

Bible Hour: 9:20

2013 Bible Reading Plan

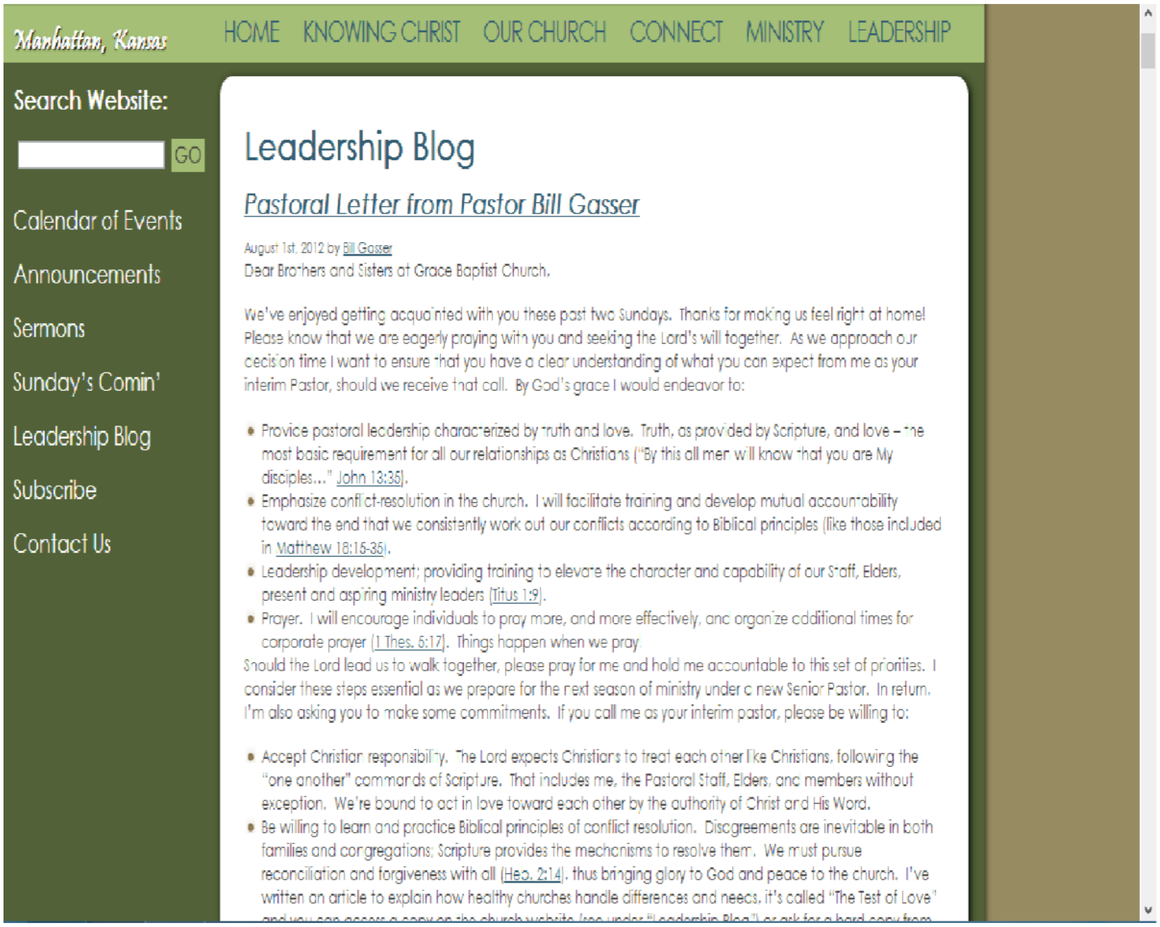
Grace Website is under construction.
New design and features coming soon!

"… You have exalted above all things 'Your Name and Your Word.'" Psalm 138:2

2901 Dickens Ave. | Manhattan, Kansas 66502 | Phone: 785-776-0424 | Fax: 785-776-0425 | office@gracebchurch.org

Figure 1.1: Unattractive current website home page

This is the home page of the current website, there are too many texts on the page instead of beautiful images, the font and color are not attractive and the navigation bar organized menus in this unprofessional way.



Leadership Blog

Pastoral Letter from Pastor Bill Gasser

August 1st, 2012 by Bill Gasser

Dear Brothers and Sisters at Grace Baptist Church,

We've enjoyed getting acquainted with you these past two Sundays. Thanks for making us feel right at home! Please know that we are eagerly praying with you and seeking the Lord's will together. As we approach our decision time I want to ensure that you have a clear understanding of what you can expect from me as your interim Pastor, should we receive that call. By God's grace I would endeavor to:

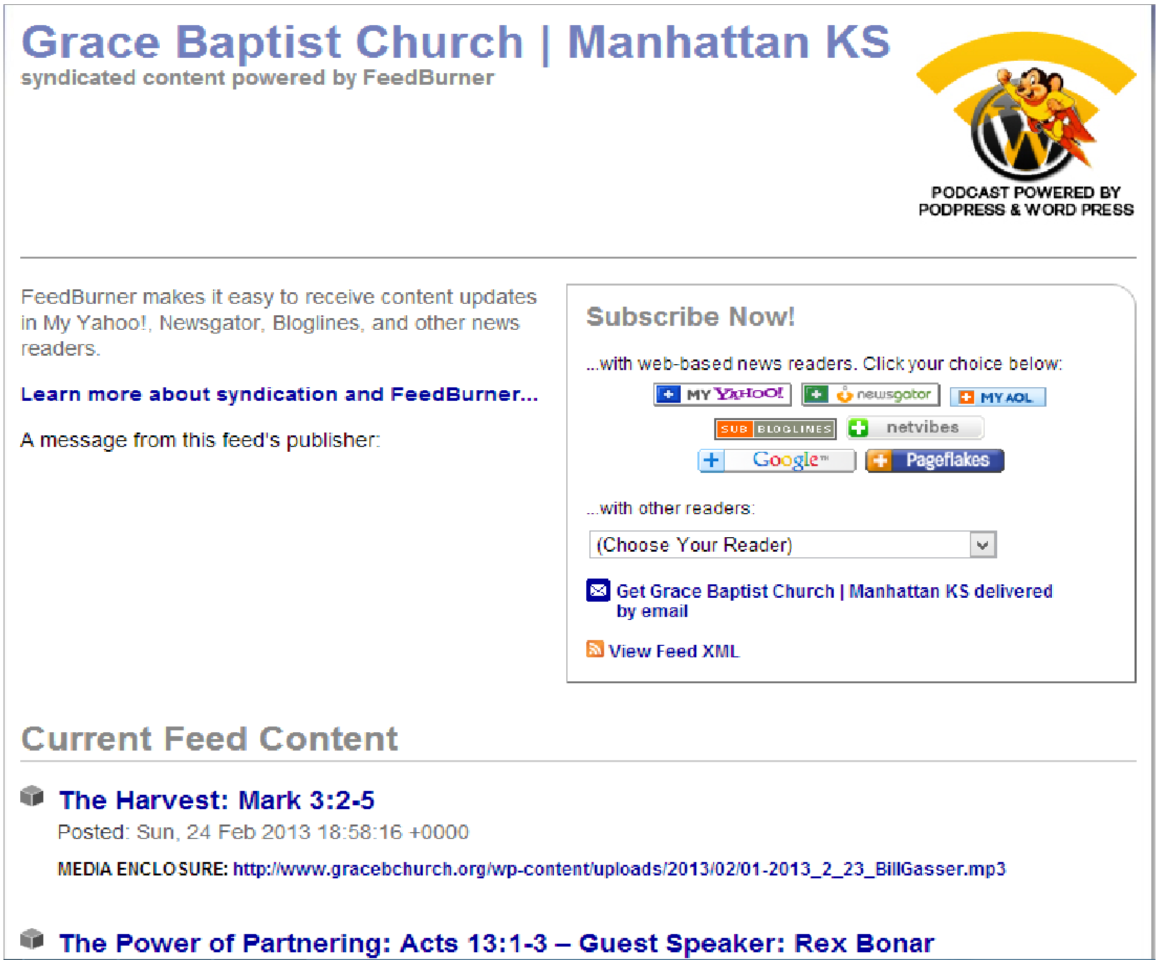
- Provide pastoral leadership characterized by truth and love. Truth, as provided by Scripture, and love – the most basic requirement for all our relationships as Christians ("By this all men will know that you are My disciples..." John 13:35).
- Emphasize conflict-resolution in the church. I will facilitate training and develop mutual accountability toward the end that we consistently work out our conflicts according to Biblical principles (like those included in Matthew 18:15-35).
- Leadership development; providing training to elevate the character and capability of our Staff, Elders, present and aspiring ministry leaders (Titus 1:9).
- Prayer. I will encourage individuals to pray more, and more effectively, and organize additional times for corporate prayer (1 Thes. 5:17). Things happen when we pray.

Should the Lord lead us to walk together, please pray for me and hold me accountable to this set of priorities. I consider these steps essential as we prepare for the next season of ministry under a new Senior Pastor. In return, I'm also asking you to make some commitments. If you call me as your interim pastor, please be willing to:

- Accept Christian responsibility. The Lord expects Christians to treat each other like Christians, following the "one another" commands of Scripture. That includes me, the Pastoral Staff, Elders, and members without exception. We're bound to act in love toward each other by the authority of Christ and His Word.
- Be willing to learn and practice Biblical principles of conflict resolution. Disagreements are inevitable in both families and congregations; Scripture provides the mechanisms to resolve them. We must pursue reconciliation and forgiveness with all (Heb. 2:14), thus bringing glory to God and peace to the church. I've written an article to explain how healthy churches handle differences and needs, it's called "The Test of Love" and you can access a copy on the church website (see under "Leadership Blog") or ask for a hard copy from

Figure 1.2 Contains too much information in one page

This is the leadership blog page. It is a single page with lots of content in it. We have to scroll the whole page to see the content. It is difficult for people to quick search what they are looking for. The page contains both titles and the content of those titles. A better solution is to design it in a way that the page contains all the "Titles" at one place and then has links to the content on different pages. And also, the navigation bar on the top and the side bar on the left are a little confused, for example, the navigation bar has a bottom leadership and sidebar has a bottom leadership blog. It is better to organize all the bottoms at the same place.



Grace Baptist Church | Manhattan KS

syndicated content powered by FeedBurner

FeedBurner makes it easy to receive content updates in My Yahoo!, Newsgator, Bloglines, and other news readers.

Learn more about syndication and FeedBurner...

A message from this feed's publisher:

Subscribe Now!

...with web-based news readers. Click your choice below:

...with other readers:

(Choose Your Reader)

Get Grace Baptist Church | Manhattan KS delivered by email

View Feed XML

Current Feed Content

The Harvest: Mark 3:2-5

Posted: Sun, 24 Feb 2013 18:58:16 +0000

MEDIA ENCLOSURE: http://www.gracebchurch.org/wp-content/uploads/2013/02/01-2013_2_23_BillGasser.mp3

The Power of Partnering: Acts 13:1-3 – Guest Speaker: Rex Bonar

Figure 1.3 Confusion subscription functionality

This page is the subscription page. But it is a bit confusing here. It redirected user to the FeedBurner page without any instructions, inexperienced users may not have knowledge about how to use this. A better solution is to provide with a small text box for users to input their email addresses and let the system automatically add their emails into subscription list, like many other websites did.

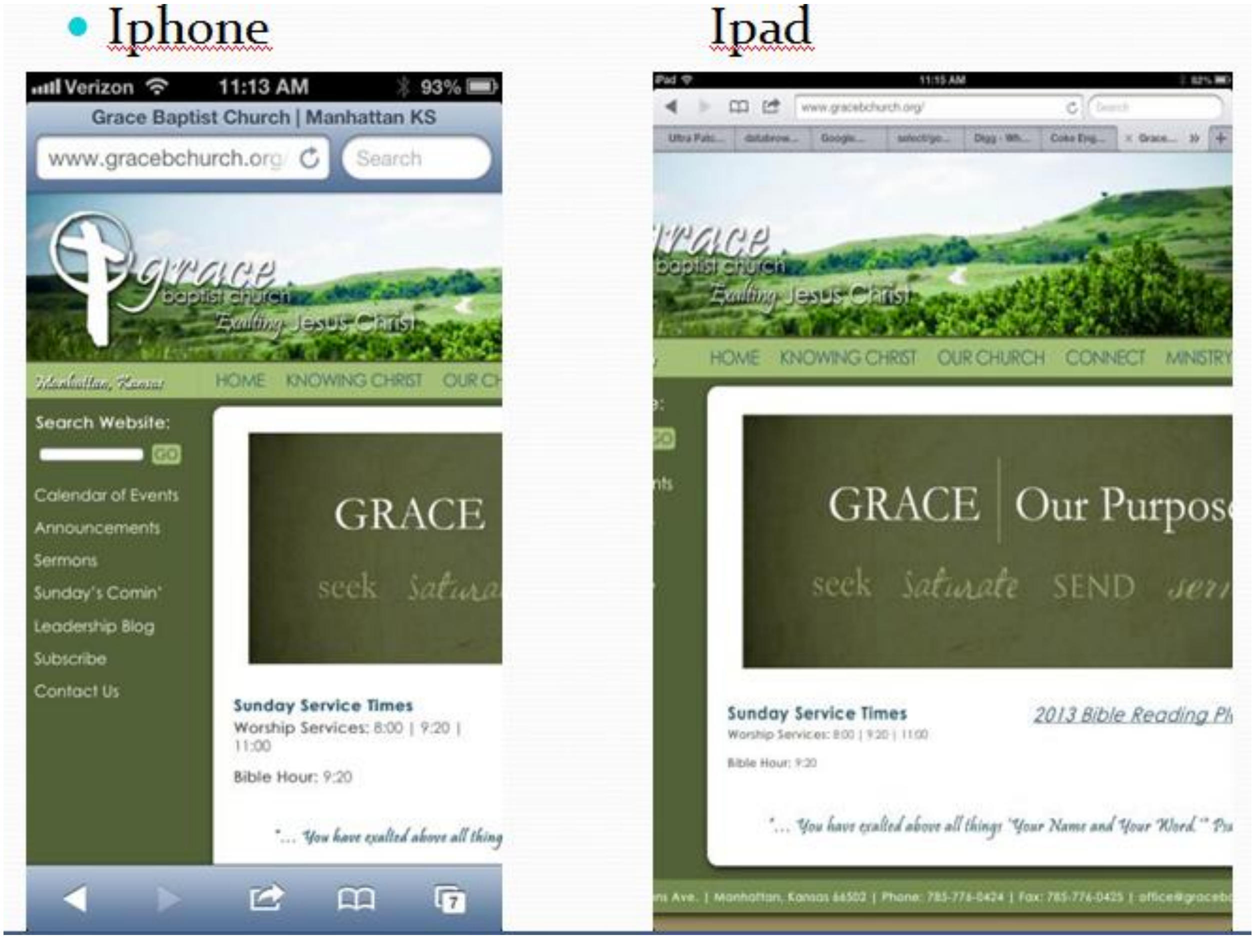


Figure 1.4 Current website in Iphone and Ipad

These are two snapshots of the homepage opened in Iphone and Ipad. It is very clear that the website doesn't have mobile support; half of the page was cut off from the screen.

The administrator of Grace Baptist Church decided to upgrade the entire website to make it more user friendly and more functionable to better represent the church. They wanted to use a new design for the website and add several new features. They are looking for students from Kansas State University to help them implementing the new website.

The new website has better page appearance, well organized structure and mobile support and it is implemented as a content management system, so that people from the church can early update the information without knowing much about web techniques.

Chapter 2 - Literature Review

2.1 Web usability

2.1.1 The Common definitions of Web Usability

With the advent of the computer, studies on usability have found a wide basis for application and today, following the mass introduction of the personal computer and software into the home, concepts such as "user friendly" or "ergonomic design" has become part of everyday language.

Usability is a quality attribute that assesses how easy user interfaces are to use which can be defined by 5 quality components: learnability, efficiency, memorability, errors, satisfaction.

Paraphrasing the definition of usability supplied by ISO 9241, web usability is the efficient, effective and satisfying completion of a specified task by any web user. Usable Web sites are those which help users to accomplish a goal easily, quickly, and pleasantly.

2.1.2 How and why website usability has been important?

Millions of new web pages are published daily, designed and developed by people who give little thought to how the information will be used and by whom. Usability is one of the most important quality factors for web applications. Unusable web applications cause users to reject them. For example, people will leave when they feel a website is difficult to use or when a website's information is hard to read or doesn't answer user's key questions. There's no such thing as a user reading a website manual or otherwise spending much time trying to figure out an interface. There are plenty of other websites available; leaving is the first line of defense when users encounter a difficulty. So if a developer wants to develop a website and make it successful, the usability is the top factor under consideration.

2.1.3 The general factors influencing web usability and how to improve it

Creating a usable website is not a trivial task. Brinck et al. (2002) outlines the detail systematic process for creating usable websites. They present their expertise gained through years of web usability and web design projects and practices. In their book, they have given the concept of pervasive usability which says that usability can be factored into every stage of the web site design process. Characteristics (See Figure 2.1) of a usable website include following factors.

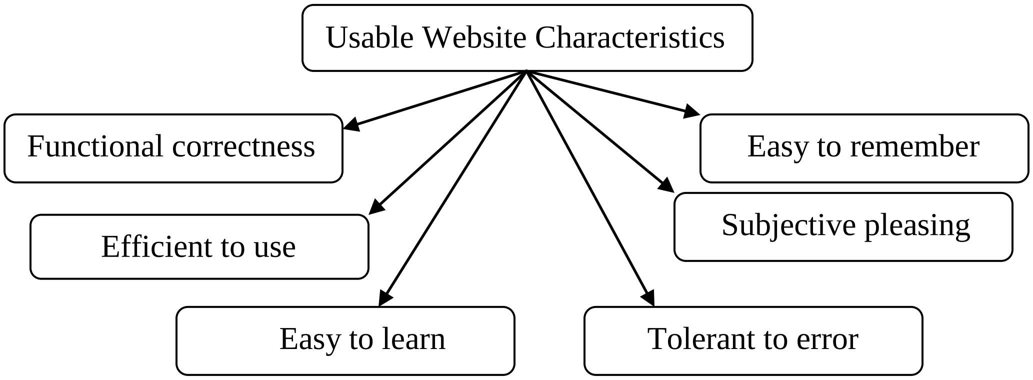


Figure 2.1 Characteristics of a Usable Website

There are many methods for studying usability, but the most basic and useful is user testing. The usability expert Jacob Nielsen has identified user testing as three components:

- Get hold of some representative users, such as customers for an e-commerce site
- Ask the users to perform representative tasks with the design.
- Observe what the users do, where they succeed, and where they have difficulties with the user interface.

According to Jacob, testing 5 users is typically enough to identify a design's most important usability problems. The best way to improve the usability is to run many small tests and revise the design between each one so you can fix the usability flaws as you identify them. Iterative design is the best way to increase the quality of user experience. The more versions and interface ideas you test with users, the better. For example, Jacob Nielsen has shown rapidly in his studies that web user skim or scan the web pages text rather than reading it. Reading on the web can be painful that is why Jacob Nielsen and others advocate that the web does require its own style of writing, a style which facilitates scanning.

H. Shahizan and Li Feng (2006) advocate the benchmarking approach for evaluating web usability. Benchmarking is a technique performed by an organization to compare their website with its competitors'. According to them, usability is a broad concept covering at least seven factors. They are screen appearance, consistency, accessibility, and navigation, media use, interactivity and content. It is up to an organization if it wants to benchmark all seven factors or some selected factors. Their purpose of research was only to test the applicability of the framework. They think that for using only one method is not adequate to access the quality of a website. Combining several approaches in web evaluation would produce better result.

K.Guenther (2003) advocates engaging users' early one. He says that although web usability has a significant priority for web development, there are very few organizations who take time to formally test usability or engage potential users early enough in development stage of project. He also writes that usability testing does not need to be sophisticated or expensive in order to be successful.

Human Factors International, Inc. (HFI) claims to be world leader in user-centered design. They offer a complete usability suit, consulting, training, and products to help customers in creating intuitive, easy-to-use Web sites and applications. They have made 10 principles of web usability (See table 2.1).

1. Motivate Design your site to meet specific user needs and goals. Use motivators to draw different user "personae" into specific parts of your site.
2. User taskflow Who are your users? What are their tasks and online environment? For a site to be usable, page flow must match workflow.
3. Architecture — it's 80% of usability Build an efficient navigational structure. Remember — if they can't find it in three 3 clicks, they're gone.
4. Affordance means obvious Make controls understandable. Avoid confusion between emblems, banners, and buttons.
5. Replicate Why reinvent the wheel? Use ergonomically designed templates for the most

common 8–12 page types.
6. Usability test along the way Test users with low fidelity prototypes early in design. Don’t wait until the end when it’s too late.
7. Know the technology limitations Identify and optimize for target browsers and user hardware. Test HTML, JavaScript, etc for compatibility.
8. Know user tolerances Users are impatient. Design for a 2–10 second maximum download. Reuse header graphics so they can load from cache. Avoid excessive scrolling.
9. Multimedia — be discriminating Good animation attracts attention to specific information, then stops. Too much movement distracts reading and slows comprehension.
10. Use a stats package Monitor traffic through your site. Which pages pique user interest? Which pages make users leave? Adjust your site accordingly.

Table 2.1 Ten principles of web usability

There is a Web Usability Testing Institute in University WISCONSIN-STOUT USA. They have made a good checklist for web usability evaluation (See Table 2.2).

Characteristics	**Checks**
1. **Load Time**	
adheres to 8-second rule	
progress bar is shown during load time	
images optimized properly	
reasonable picture response	
hit counter load times are reasonable if counter is used at all	
2. **Navigation**	
organization of navigation labels	

limited amount of links in list(5-9)	
uses hyperlink text accurately to describe the linked pages	
uses anchors on large documents/return to top	
always has back to home on every page	
hyperlinks are standard color	
hyperlinks change with viewing	
forms of navigation feed back are employed	
no dead links	
3 **Structure/Layout**	
arrangement is prioritized on the screen	
has constant design	
consistent in organization	
pages are designed in a way to maximize the use of space	
layout is not fixed width	
uses no more than one extra browser window and only when necessary	
avoids frames except for navigation	
avoids long scrolling pages by using columns	
no “under construction” pages	
3 **Content**	
site is well focused around it’s uses and content	
provides access to the webmaster and site owner	
avoids redundancy	
has up to date information	
tells user when last updated	
4 **Visuals**	
pages fit onto low resolution screen (800x600)	
font is readable size	
background is not busy	
has appropriate contrast in color between text and background	

colors are controlled in an aesthetically pleasing way	
no excessive glitz	
visuals serve purpose and are not strictly decorative	
5 **Learnability**	
is designed with different user levels in mind	
memorable URL for home page	
6 **Non-HTML Features**	
site permits access to any software required to view it or operate it.	
no JavaScript errors	
no applet errors	
javaScript is only used when appropriate	

Table 2.2 Checklist by web usability testing institute, University of Wisconsin-Stout

2.2 Website Performance

2.2.1 How to measure web performance

The performance of the website is highly associated with the user experience. A slow online shopping site will definitely lost the potential buyers. The website performance can be measured by the speed in which web pages are downloaded and displayed on the user's web browser. The less time user spent on waiting, the better the website performed.

For most of the websites, a few users wouldn’t be much of a problem, but it can be a huge challenge when there are millions of people using the website. For the Googles, Yahoos, Amazons and eBays, slow websites mean fewer users and less happy users and thus lost revenue and reputation.

Even if you don’t have millions of users (yet), consider one very important thing: people are consuming the Web nowadays less with fat connections and massive computers and more with mobile phones over slow wireless and 3G connections, but they still expect the same performance. Waiting for a slow website to load on a mobile phone is doubly annoying because the user is usually already in a hurry and is paying by the byte or second.

2.2.2 The general factors influencing web performance

The user waiting time can be divided into backend waiting time and frontend waiting time. The backend waiting time is from when the user makes the request to when the last byte of the HTML document arrives. It includes the time for the initial request to go up, the web server to stitch together the HTML, and for the response to come back. The frontend waiting time is everything after the HTML document arrives. In reality, it includes backend time (primarily reading static files) and network time, as well as true frontend activities such as parsing HTML, CSS and JavaScript, and executing JavaScript.

In the book "High Performance Web Sites", Steve Souders presented a Performance Golden Rule: "Only 10–20% of the end user response time is spent downloading the HTML document (backend).The other 80–90% is spent downloading all the components in the page (frontend)." According to Steve's research, there is more potential for improvement in focusing on the frontend. More specifically, if we were able to cut backend response times in half, the end user response time would decrease only 5-10% overall; if instead, we reduce the frontend performance by half, we would reduce overall response times by 40-45%.

Steve Souders offered precise guidelines for reducing that 80-90% end user response time (see Table 2.3). By applying these rules to the website, it is possible to make the pages 25-50% faster and improve the user experience.

Rule 1: Make Fewer HTTP Requests
Rule 2: Use a Content Delivery Network
Rule 3: Add an Expires Header
Rule 4: Gzip Components
Rule 5: Put Stylesheets at the Top
Rule 6: Put Scripts at the Bottom
Rule 7: Avoid CSS Expressions

Rule 8: Make JavaScript and CSS External
Rule 9: Reduce DNS Lookups
Rule 10: Minify JavaScript
Rule 11: Avoid Redirects
Rule 12: Remove Duplicate Scripts
Rule 13: Configure ETags
Rule 14: Make Ajax Cacheable

Table 2.3 Fourteen rules to improve website performance

2.2.3 The ideas behind these 14 rules

The first rule he explained in his book is Make Fewer HTTP Requests. The techniques include using image maps, CSS sprites, inline images, and combined scripts and stylesheets. For example, by associating multiple URLs with a single image the developer can reduce the number of HTTP requests without changing the page’s look and feel.

The second rule is to use a content delivery network. This rule is based on the fact that the closer the application web servers are to the user, the less response time of one HTTP request, but the closer component web servers to the user, the less response time of many HTTP request. So instead of redesigning the application to disperse the application web servers, it is better to first disperse the component web servers. And content delivery networks as a collection of web servers distributed across multiple locations to deliver content to users, it is the perfect implementation of this idea.

The third rule is to add an expires header. It is to configure web page components to maximize the browser’s caching capabilities. By using a future expires header, you make page components cacheable after the first-time page visit. So it avoids unnecessary HTTP requests on subsequent page views. A future expires header can be used with images, scripts, stylesheets, flash and all other components.

The fourth rule is to Gzip components. Unlike Rules 1 and 3 to address response time by eliminating unnecessary HTTP requests, it reduces response times by reducing the size of the

HTTP response. If an HTTP request results in a smaller response, the transfer time decreases because fewer packets must travel from the server to the client. It is the easiest technique for reducing page weight and it also has the biggest impact.

The fifth rule is to put stylesheets at the top. The logic behind this rule is that, by putting stylesheets at the top, it allows browsers to load the page progressively. It gives visual feedback for the user who is waiting for the page and thus improves the overall user experience, even though it does not improve the actual time to load the page’s components.

The sixth rule is to put Scripts at the Bottom. This rule is similar to the fifth rule, the goal is to allow progressive rendering of the page. However, the mechanism is opposite here: the progressive rendering is blocked for all content below the script. So by moving the scripts from the top of the page to the bottom, it enables progressive rendering and achieves greater download parallelization.

The seventh rule is to avoid CSS expressions. CSS expressions are a powerful way to set CSS properties dynamically. Although it is a useful tool to create dynamic page, it is also bad for performance. The problem with expressions is that they are evaluated more frequently than most people expect. Not only are they evaluate whenever the page is rendered and resized, but also when the page is scrolled and even when the user moves the mouse over the page. So to solve this problem, people can create one-time expressions or use event handlers instead of CSS expressions.

The eighth rule is to make JavaScript and CSS external. Although in raw terms, the response is faster when JavaScript and CSS are inlined in the page itself instead of being contained in external files, it also lost the opportunity for the JavaScript and CSS files to be cached by the browser. In the real word situation, when the HTML documents are not cached, the inline JavaScript and CSS are downloaded every time the HTML document is requested. While by caching the external JavaScript and CSS files, the size of the HTML document can be reduced without increasing the number of HTTP requests.

The ninth rule is to reduce DNS lookups. The reason for this rule is that DNS typically takes 20-120 milliseconds for the browser to look up the IP address for a given hostname. The browser can’t download anything from this hostname until the DNS lookup is completed. The response time depends on the DNS resolver, the load of requests on it, your proximity to it, and your bandwidth speed. The problem for this rule is that reducing the number of unique hostnames has the potential to reduce the amount of parallel downloading that takes place in the page. Avoiding DNS lookups cuts response times, but reducing parallel downloads may increase response times. The book provides guideline for this problem: to split these components across at least two but no more than four hostnames. This result in a good compromise between reducing DNS lookups and allowing a high degree of parallel downloads.

The tenth rule is to minify JavaScript. According to the book, by removing unnecessary characters, which includes comments and unneeded whitespace characters, one can reduce the size of the downloaded file, thus improve response time performance. The author also proposed a more complex method called Obfuscation to munge the code. Function and variable names are converted into smaller strings to make the code more compact during munging. This method further reduces the code size but at the same time make the code hard to read.

The eleventh rule is to avoid redirects. Redirects will hurt performance because they delay the delivery of the entire HTML document: nothing in the page can be rendered and no components can be downloaded until the HTML document has arrived. Inserting a redirect between the user and the HTML document delays everything in the page. The book indicated some alternatives to redirects, such as using specific URLs and handlers to handle the situation like missing trailing slash; or using more complex methods to integrate two backends.

The twelfth rule is to remove duplicate scripts. The duplicate scripts hurt performance in two ways: unnecessary HTTP requests and wasted JavaScript execution. So to avoid accidentally including the same script twice is to implement a script management module in the templating system.

The thirteenth rule is to configure ETags. Entity tags (ETags) are a mechanism that web servers and browsers use to validate cached components. The problem with ETags is that because they are constructed using attributes that make them unique to a specific server hosting a site, they won’t match when a browser gets the original component from one server and later makes a conditional get request that goes to a different server. This kind of mismatch will cause unnecessary reloading of components or degrading the effectiveness of proxy caches. This hurt the performance. The book suggests some ways to configure ETags to avoid these situations.

The fourteenth rule is to make Ajax cacheable. One of the cited benefits of Ajax is that it provides instantaneous feedback to the user because it requests information asynchronously from the backend web server. The book mentions that in some situations the user can be kept waiting for the response from Ajax requests. For example, active Ajax requests are made based on the user’s current actions instead of downloading components to the cache before user actually needed, which will cause user waiting for the results to be displayed before taking any further action. The author suggests optimizing active Ajax requests by making the responses cacheable for better performance.

Chapter 3 - Requirement Analysis

3.1. Requirements Gathering

We have collected all the information required for the project from the Grace Baptist Church. After several meetings with the people from the church, we have identified some requirements that will assure the website meets their expectation. The web page designs were given based on the discussion of the people from Grace Baptist Church. The final designs were in PSD format and they wanted the website to be implemented using content management system such as WordPress.

3.2. Software Requirements

3.2.1 Web Content Management System

A Web content management system is software that helps you manage all of the content on your website. The content includes documents, photos, music, video, simple text, or other objects. It provides a central interface for you to publish, edit and modifying content as well as maintenance. Such a system allows non-technical users to make changes to a website with little training. A WCMS typically requires a web developer to set up and add features, but it has maintenance tools for non-technical people. There are some free content management systems such as Drupal, TYPO3, Joomla and WordPress. We going to use WordPress for this project.

3.2.2 WordPress

WordPress started as a blogging system, but has evolved to be used as a full content management system and has thousands of plugins, widgets and themes. WordPress used theme system to control the look and presentation of the material of your website. A WordPress Theme is a collection of template files that work together to produce a graphical interface with an underlying unifying design for a weblog. WordPress not only provides thousands of different themes for user to choose, it is also completely customizable and can be used for almost anything. WordPress accomplishes that by allowing user to create their own templates to generate pages, which will be further explained in the implementation section.

3.2.3 Other Technologies and software used

- Photoshop have been used to slice the PSD format web design into different pieces for web development
- HTML, CSS and Javascript have been used to format the web pages
- PHP has been used to create WordPress tempelates
- FTP has been used to upload files and scripts to WordPress server
- Apache JMeter has been used to test the performance of the website

Chapter 4 - Implementation

4.1 Convert PSD designs into HTML Website templates

The website was implemented based on the photoshop page template designs given by Grace Baptist Church. So the first step is to convert the PSD format pages to HTML Website templates. So we first used photoshop to slice each PSD image to a set of layers such as background, images, logos, and then create HTML code to combine all these layers into divs containers. The template was divided into three basics components: header, contents and footer. Three containers were created by defining these divs:

```
<div id="header"></div>
<div id="contents"></div>
<div id="footer"></div>
```

The next step is to insert different components such as links, images and text into these three containers and create CSS code to define their positions, sizes, colors, fonts and other style properties.

```
.header {
      background-image: url(../images/header_background.jpg);
      font-family: "Century Gothic";
      color: #3b4557;
      text-decoration: none;
      position: relative;
      margin: 0px;
      padding: 0px;
      float: left;
      width: 1024px;
      height:228px;
      overflow:hidden;
}

.content {
      background: url(../images/bg.jpg) repeat-y;
      float: left;
      width: 974px;
      min-height: 350px;
      position:relative;
      margin-right: 5px;
      padding: 0 25px 25px 25px;
}

.footer {
      background: url(../images/bg.jpg) repeat-y;
      font: normal 1em "Century Gothic";
```

```
        width:1024px;
        height:100px;
        float:left;
}
```

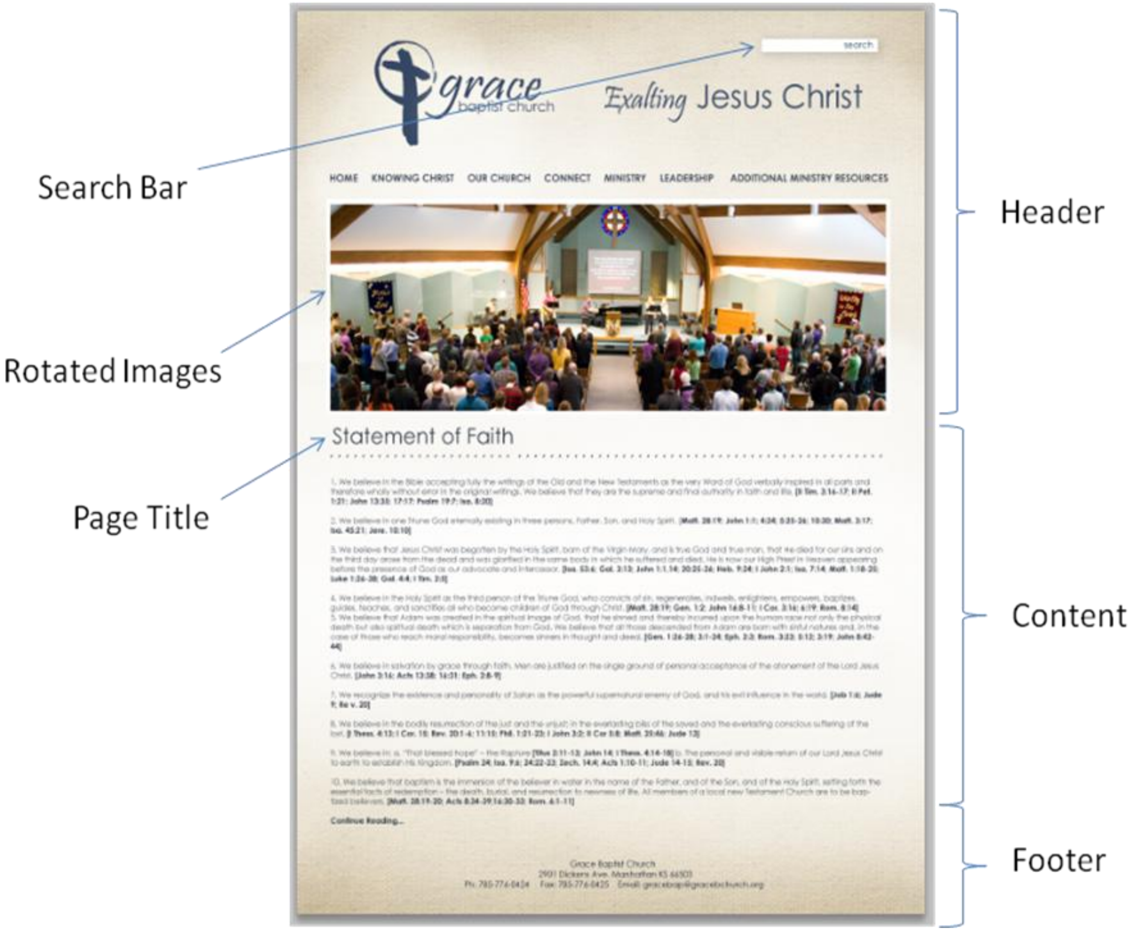
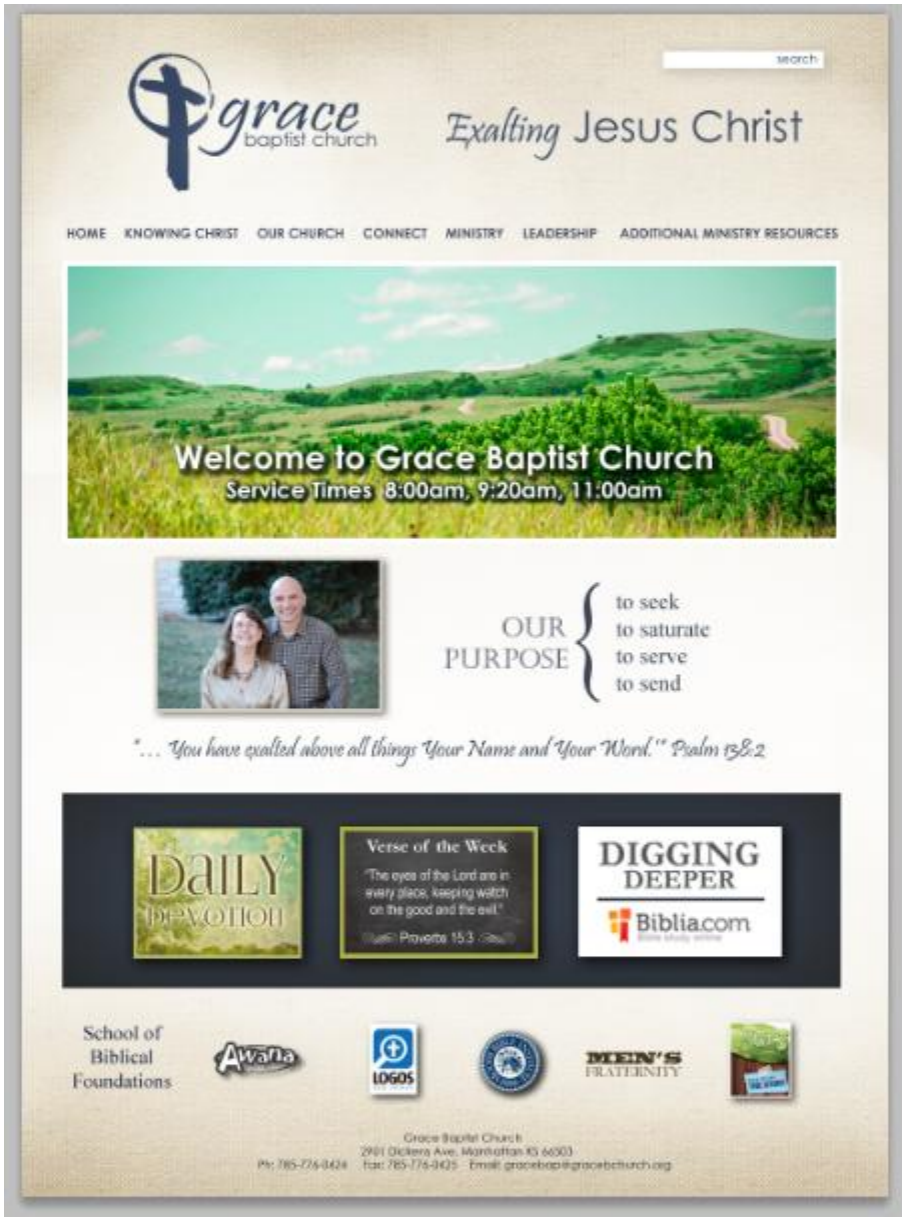


Figure 4.1 Web page template – content page, header, content and footer



Exalting Jesus Christ

Welcome to Grace Baptist Church
Service Times 8:00am, 9:20am, 11:00am

OUR PURPOSE
to seek
to saturate
to serve
to send

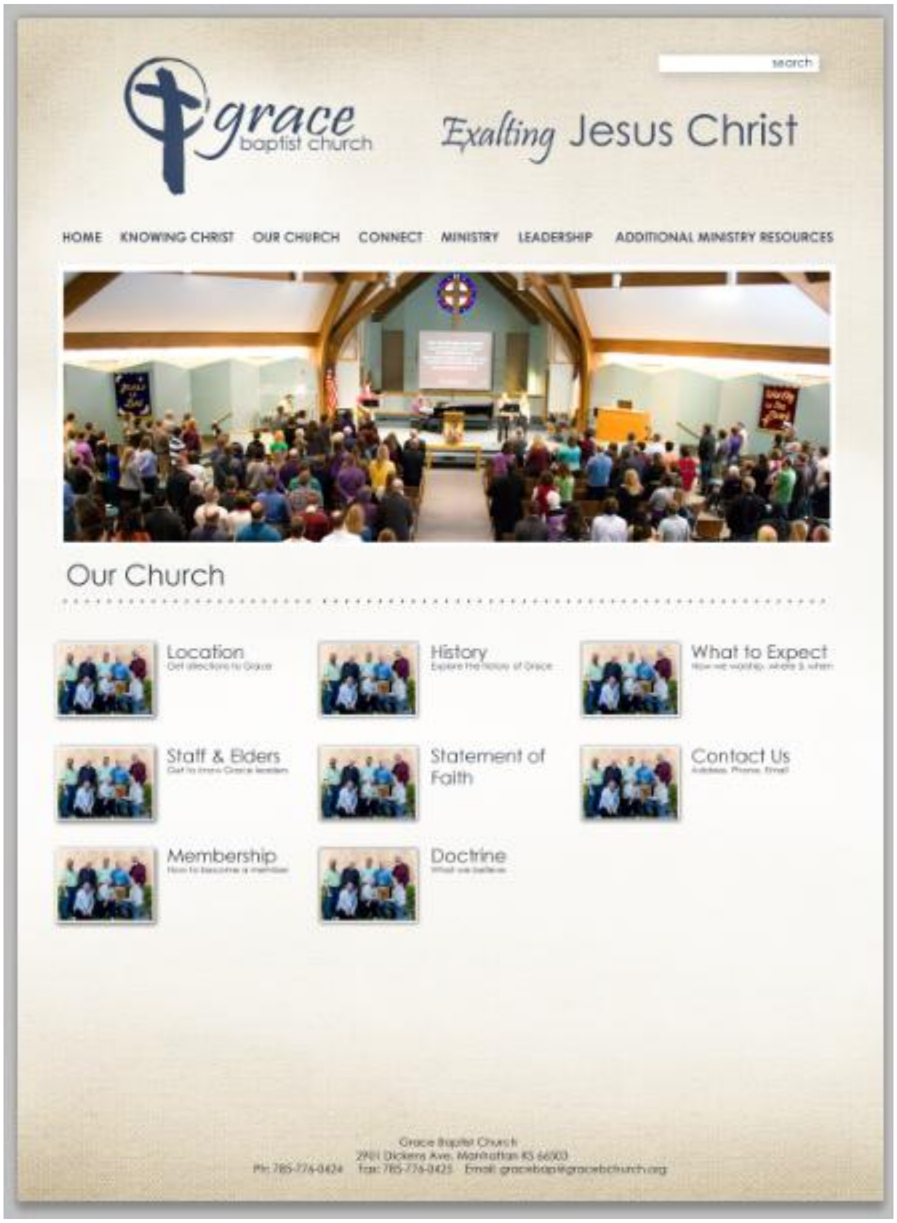
"… You have exalted above all things Your Name and Your Word." Psalm 138:2

Verse of the Week
"The eyes of the Lord are in every place, keeping watch on the good and the evil."
Proverbs 15:3

School of Biblical Foundations

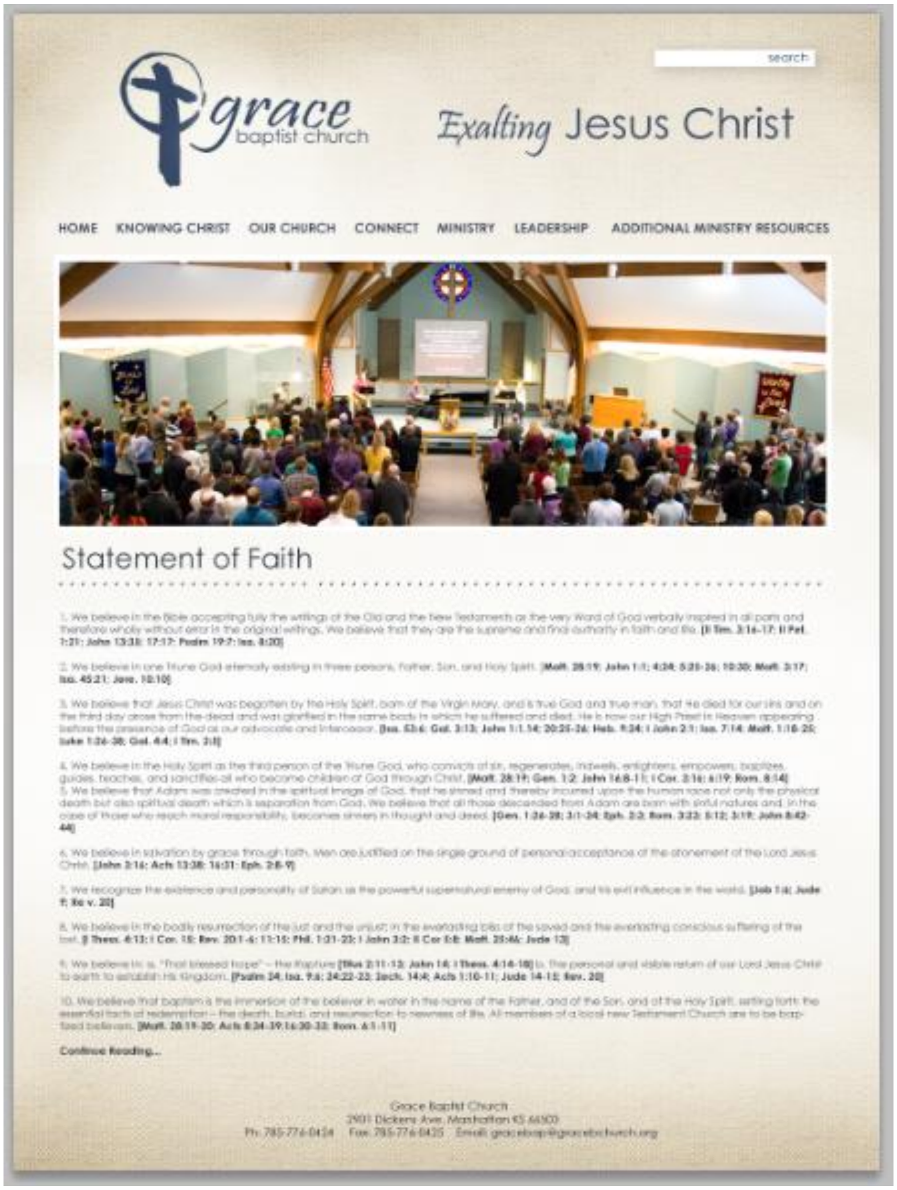
Grace Baptist Church
2901 Dickens Ave, Manhattan KS 66503
Ph: 785-776-0424 Fax: 785-776-0425 Email: gracebap@gracebchurch.org

Figure 4.2 Web page template – Home page



Grace Baptist Church
2901 Dickens Ave. Manhattan KS 66503
Ph: 785-776-0424 Fax: 785-776-0425 Email: gracebap@gracebchurch.org

Figure 4.3 Web page template – Second page



search

grace baptist church

Exalting Jesus Christ

HOME KNOWING CHRIST OUR CHURCH CONNECT MINISTRY LEADERSHIP ADDITIONAL MINISTRY RESOURCES

Statement of Faith

1. We believe in the Bible accepting fully the writings of the Old and the New Testaments as the very Word of God verbally inspired in all parts and therefore wholly without error in the original writings. We believe that they are the supreme and final authority in faith and life. **[II Tim. 3:16-17; II Pet. 1:21; John 13:38; 17:17; Psalm 19:7; Isa. 8:20]**

2. We believe in one Triune God eternally existing in three persons, Father, Son, and Holy Spirit. **[Matt. 28:19; John 1:1; 4:24; 5:25-26; 10:30; Matt. 3:17; Isa. 45:21; Jere. 10:10]**

3. We believe that Jesus Christ was begotten by the Holy Spirit, born of the Virgin Mary, and is true God and true man, that He died for our sins and on the third day arose from the dead and was glorified in the same body in which he suffered and died. He is now our High Priest in Heaven appearing before the presence of God as our advocate and intercessor. **[Isa. 53:6; Gal. 3:13; John 1:1,14; 20:25-26; Heb. 9:24; I John 2:1; Isa. 7:14; Matt. 1:18-25; Luke 1:26-38; Gal. 4:4; I Tim. 3:1]**

4. We believe in the Holy Spirit as the third person of the Triune God, who convicts of sin, regenerates, indwells, enlightens, empowers, baptizes, guides, teaches, and sanctifies all who become children of God through Christ. **[Matt. 28:19; Gen. 1:2; John 16:8-11; I Cor. 3:16; 6:19; Rom. 8:14]**
5. We believe that Adam was created in the spiritual image of God, that he sinned and thereby incurred upon the human race not only the physical death but also spiritual death which is separation from God. We believe that all those descended from Adam are born with sinful natures and, in the case of those who reach moral responsibility, becomes sinners in thought and deed. **[Gen. 1:26-28; 3:1-24; Eph. 2:2; Rom. 3:23; 5:12; 3:19; John 8:42-44]**

6. We believe in salvation by grace through faith. Men are justified on the single ground of personal acceptance of the atonement of the Lord Jesus Christ. **[John 3:16; Acts 13:38; 16:31; Eph. 2:8-9]**

7. We recognize the existence and personality of Satan as the powerful supernatural enemy of God, and his evil influence in the world. **[Job 1:6; Jude 9; Re v. 20]**

8. We believe in the bodily resurrection of the just and the unjust; in the everlasting bliss of the saved and the everlasting conscious suffering of the lost. **[I Thess. 4:13; I Cor. 15; Rev. 20:1-6; 11:15; Phil. 1:21-23; I John 3:2; II Cor 5:8; Matt. 25:46; Jude 13]**

9. We believe in a. "That blessed hope" – the Rapture **[Titus 2:11-13; John 14; I Thess. 4:14-18]** b. The personal and visible return of our Lord Jesus Christ to earth to establish His Kingdom. **[Psalm 24; Isa. 9:6; 24:22-23; Zech. 14:4; Acts 1:10-11; Jude 14-15; Rev. 20]**

10. We believe that baptism is the immersion of the believer in water in the name of the Father, and of the Son, and of the Holy Spirit, setting forth the essential facts of redemption – the death, burial, and resurrection to newness of life. All members of a local new Testament Church are to be baptized believers. **[Matt. 28:19-20; Acts 8:36-39; 16:30-33; Rom. 6:1-11]**

Continue Reading...

Grace Baptist Church
2901 Dickens Ave. Manhattan KS 66503
Ph: 785-776-0424 Fax: 785-776-0425 Email: gracebap@gracebchurch.org

Figure 4.4 Web page template – content page

4.2 Create WordPress template

4.2.1 How does WordPress template work?

WordPress uses templates to generate the pages requested by visitors, and are output as HTML. These templates are combined as the Theme system to provide control over the look and presentation of the material on the website. Themes may include customized template files, image files (*.jpg, *.gif), style sheets (*.css), custom Pages, as well as any necessary .php code files. For example, at the very minimum, a WordPress Theme consists of two files: style.css and index.php. Index.php is a template file made up of HTML, PHP, and WordPress Template Tags. The pages that are created based on Index.php will have the same page layout but different contents. A template tags are php functions telling WordPress to "do" or "get" something, which are used within Templates to display information dynamically. WordPress provides different Template Tags such as General Tags, Author Tags, Bookmark Tags, Category Tags, Link Tags, Post Tags and other type of Tags. For example, this line of code “<h2><?php the_title('Post Title: '); ?></h2>” put the entire post title into an H2 heading and added the phrase "Post Title" to the beginning of the post title. And get_header() tag is to call header.php to generate header HTML and get_footer() tag is to call footer.php to generate footer HTML.

4.2.2 The WordPress Page Structure

A simple WordPress web page structure is made up of three basic building “blocks”: a header, the content, and a footer. The header typically includes the title of the website, navigation menus, a logo bar, the description of the website and other information that needs to be at the top. In the html point of view, the header part is the code inside the <head> tag such as the <doctype>, <meta> tags and links to style sheets. It also includes the opening <body> tag and the visible header of the website. The content block contains the posts and pages of the website. The footer contains the links to other pages or categories on the website in a navigation menu, copyright and contact information and other details that goes at the bottom of the page.

For this simple structure, the script files to be created in the Theme directory would be: index.php, header.php, footer.php and style.css. The files header.php and footer.php contain code that used to generate header and footer for all the pages that are created under this template. The index.php file has two main functions:

- Call header.php and footer.php using Template Tags to generate HTML output
- Include the WordPress Loop to gather information from the database (posts, pages, categories, etc.)

The WordPress Loop was placed in the index.php between the header and footer calls in order to display the posts and pages of the website. The simplest index.php structure is like this:

```
<?php
get_header();
if (have_posts()) :
   while (have_posts()) :
      the_post();
      the_content();
   endwhile;
endif;
get_sidebar();
get_footer();
?>
```

Using the Loop, WordPress processes each post to be displayed on the current page, and formats it according to how it matches specified criteria within the Loop tags. Any HTML or PHP code in the Loop will be processed on each post. Before the Loop goes into action, the WordPress connects to the database and retrieves the information based on user’s specification or the previously default setting, and stores the results in a variable. The Loop uses this variable’s value for display in the templates.

4.2.3 WordPress Template Hierarchy

To create WordPress Templates and maybe further develop WordPress themes, it is important to understand how WordPress selects template files to display the various pages on the website.

The general idea is that WordPress uses the Query String (information contained within each link on the website) to decide which template or set of templates will be used to display the page. First, WordPress matches every Query String to decide what type of page is being requested. Then Templates are chosen based on WordPress Template hierarchy (see Figure 3.2.1) to generate web page content. For example, if a visitor goes to the home page of the website, WordPress first determines whether it has a static front page. If a static front page has been set, then WordPress loads that page according to the page template hierarchy. Otherwise WordPress looks for a template file called home.php and uses it to generate the requested page. Finally, if home.php is missing, WordPress looks for index.php in the theme’s directory to generate the page. It is worth to mention that, developers can have any number of custom pages other than home page. What they need to do is to create page template file and assign it to the page to overwrite the default page.php file, then when the specific page is visited, WordPress will find the corresponding page template file to generate that page.

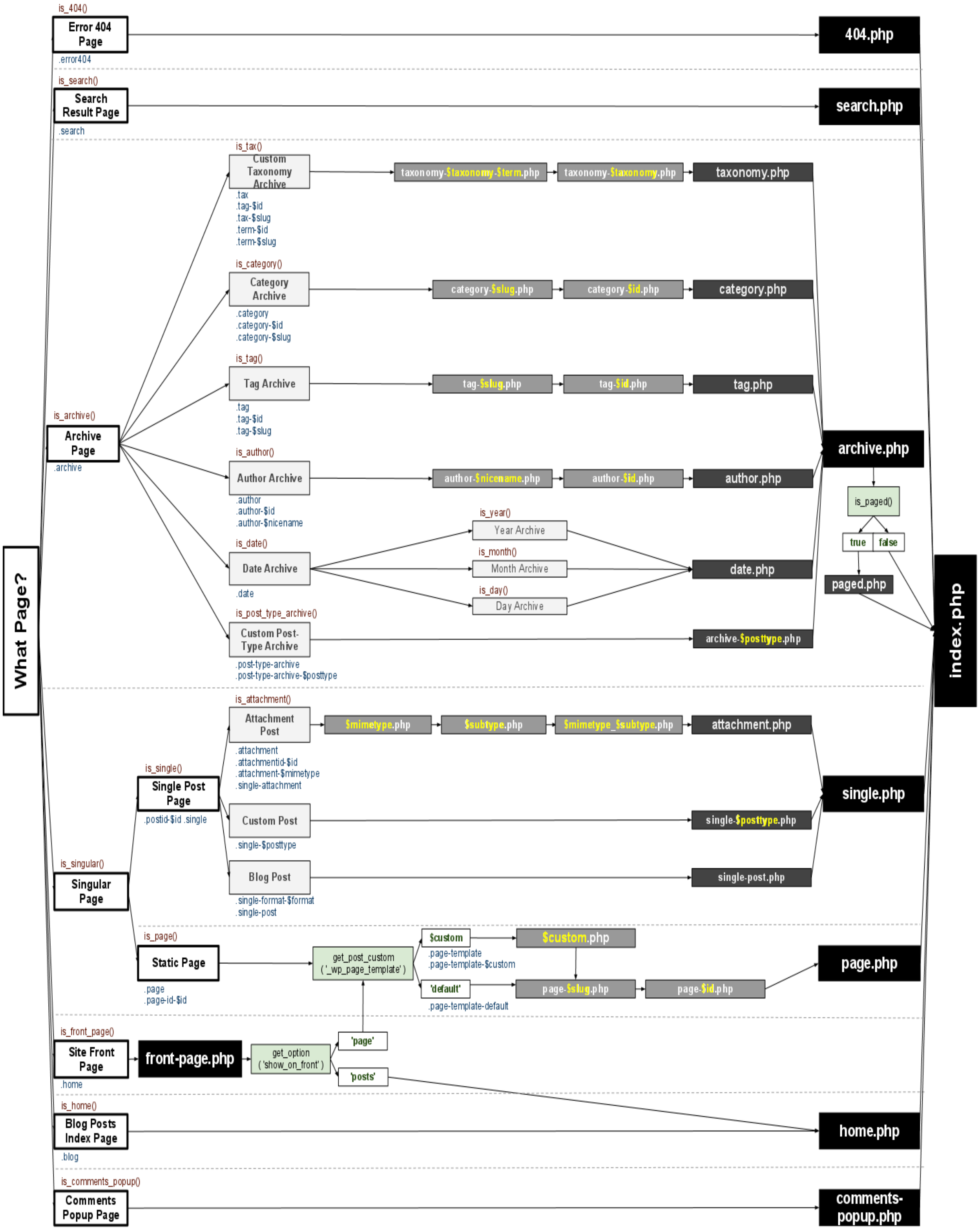


Figure 4.5 WordPress Template Hierarchy

4.2.4 Custom WordPress Template implementation

To implement the WordPress Templates for new church website, I created five template files in the theme directory (see Figure 3.2.1). Header.php and Footer.php are called inside Home.php, Ourchurch.php and Index.php template files. Based on WordPress Template Hierarchy, if visitor goes to the home page, the Home.php will be chosen to generate the home page; if visitor goes to the our church page, Ourchurch.php will be chosen to generate that page because it overwrote the default page.php; And all other pages will be generated using Index.php template to present the content.

In the same theme directory, there are images and videos folders that contain all the images and videos for generating the pages; and there are also css and javascript files to assign styles to the generated page.

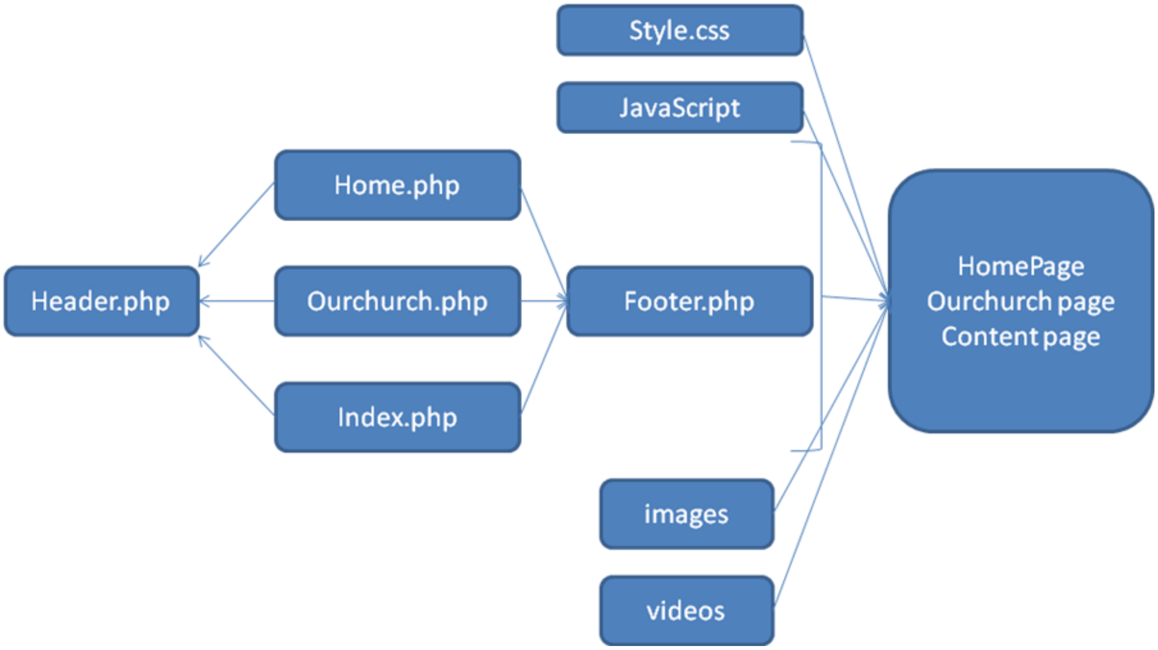


Figure 4.6 Custom WordPress Template structure

4.3 Mobile support

We used one WordPress plugin called Mobile Pack to help mobilize the new Grace Baptist Church website. This plugin has a mobile switcher functionality that will automatically suggest desktop or mobile presentation, but lets users switch to the other if required (see Figure 3.3.2). Below is the comparison of the new website and current website in Iphone:

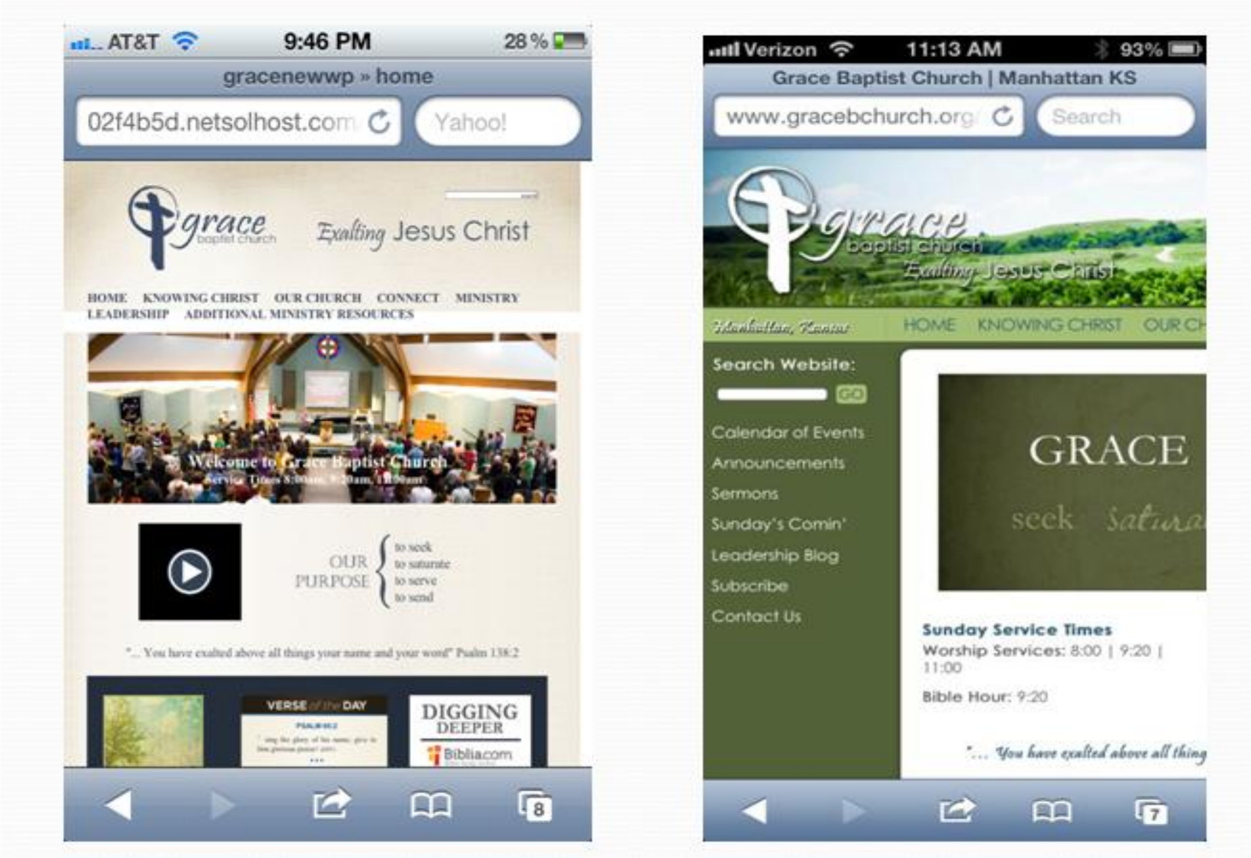


Figure 4.7 New website and current website in Iphone

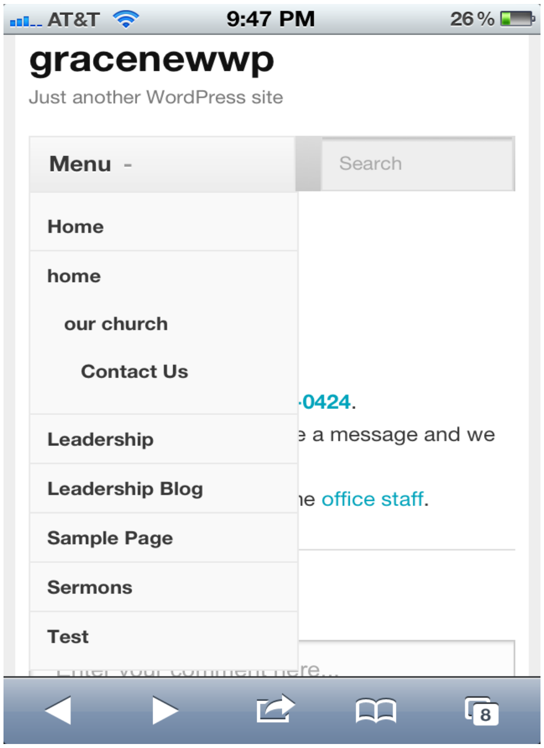


Figure 4.8 Mobile site of the new grace Baptist church website

Chapter 5 - Performance Testing

In software engineering, performance testing is used to determine how fast some aspect of a system performs under a particular workload. Performance Testing is used to verify performance behaviors for business functions under the normal and heavy work conditions. And it is aimed at assessing the speed at which the product addresses different events under different conditions. So the success criteria of this test are completion of all the test cases without any failures and within acceptable time allocation.

5.1 Apache JMeter

Apache JMeter is an open source Java application designed to load test functional behavior and measure performance. It can be used to simulate a heavy load on a server, network or object to test its strength or to analyze overall performance under different load types. JMeter has a multithreading framework which allows concurrent sampling by many threads and simultaneous sampling of different functions by separate thread groups. It also provides powerful visualization tools so that people can make a graphical analysis to test the server under heavy concurrent load. I used Apache JMeter to do the performance test for the new website been implemented.

5.2 Performance test plan and result interpretation

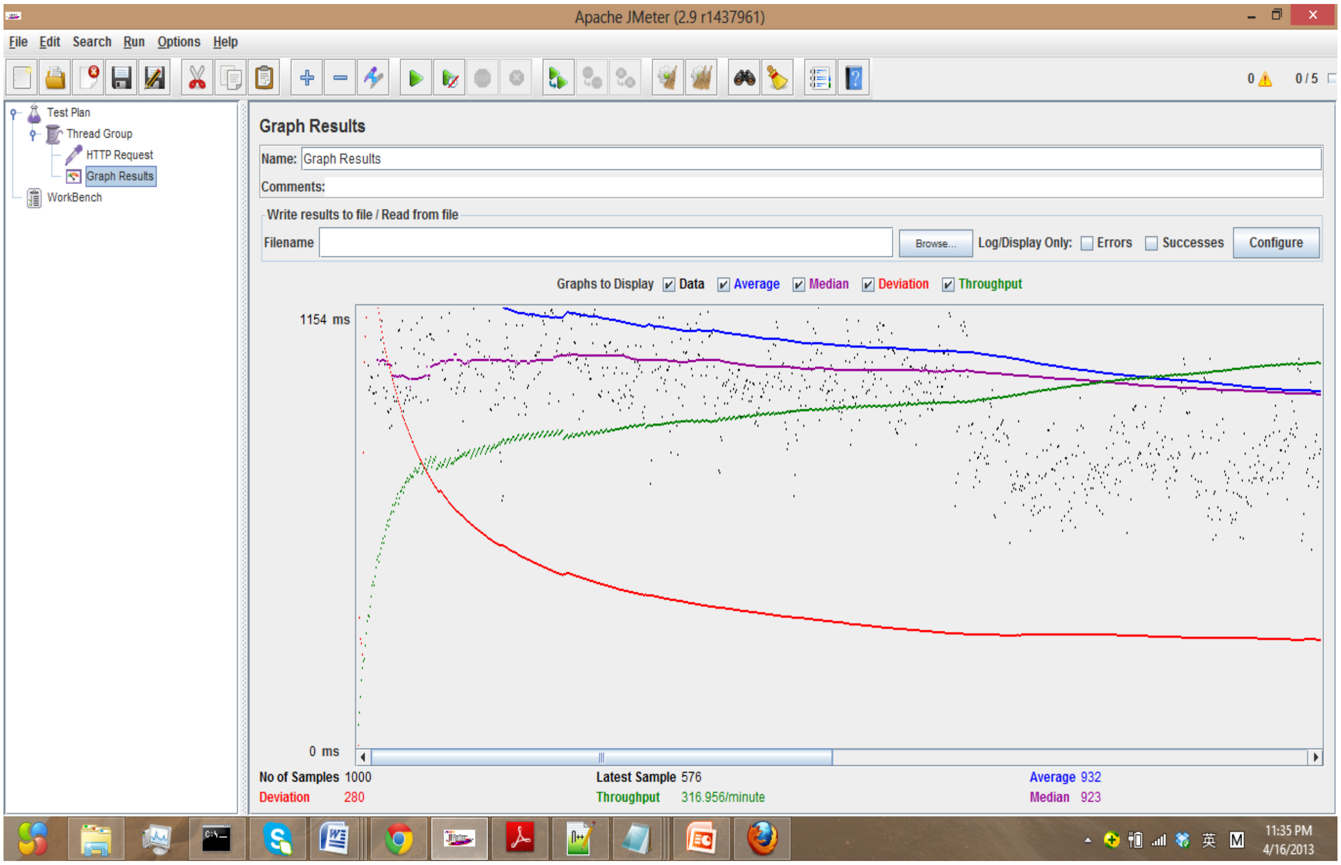
I have run performance testing in JMeter for the home page and the search page respectively in five cases: 5 user, 10 users, 20 users, 40 users and 80 users. The JMeter output graph results and recoded important values such as throughput, response time and deviation of the response time among all the samples.

- The **throughput**: is the number of requests per unit of time (seconds, minutes, hours) that are sent to your server during the test.
- The **response time**: is the elapsed time from the moment when a given request is sent to the server until the moment when the last bit of information has returned to the client

To analyze the results, I followed some basic advices regarding response times provided by Card et al. 1991:

- **0.1 second** is about the limit for having the user feel that the system is reacting instantaneously, meaning that no special feedback is necessary except to display the result.
- **1.0 second** is about the limit for the user's flow of thought to stay uninterrupted, even though the user will notice the delay. Normally, no special feedback is necessary during delays of more than 0.1 but less than 1.0 second, but the user does lose the feeling of operating directly on the data.
- **10 seconds** is about the limit for keeping the user's attention focused on the dialogue. For longer delays, users will want to perform other tasks while waiting for the computer to finish, so they should be given feedback indicating when the computer expects to be done. Feedback during the delay is especially important if the response time is likely to be highly variable, since users will then not know what to expect.

Below are the Figures of sample JMeter test-cases and analysis of their results:



Graph Results

Name: Graph Results

Comments:

Write results to file / Read from file

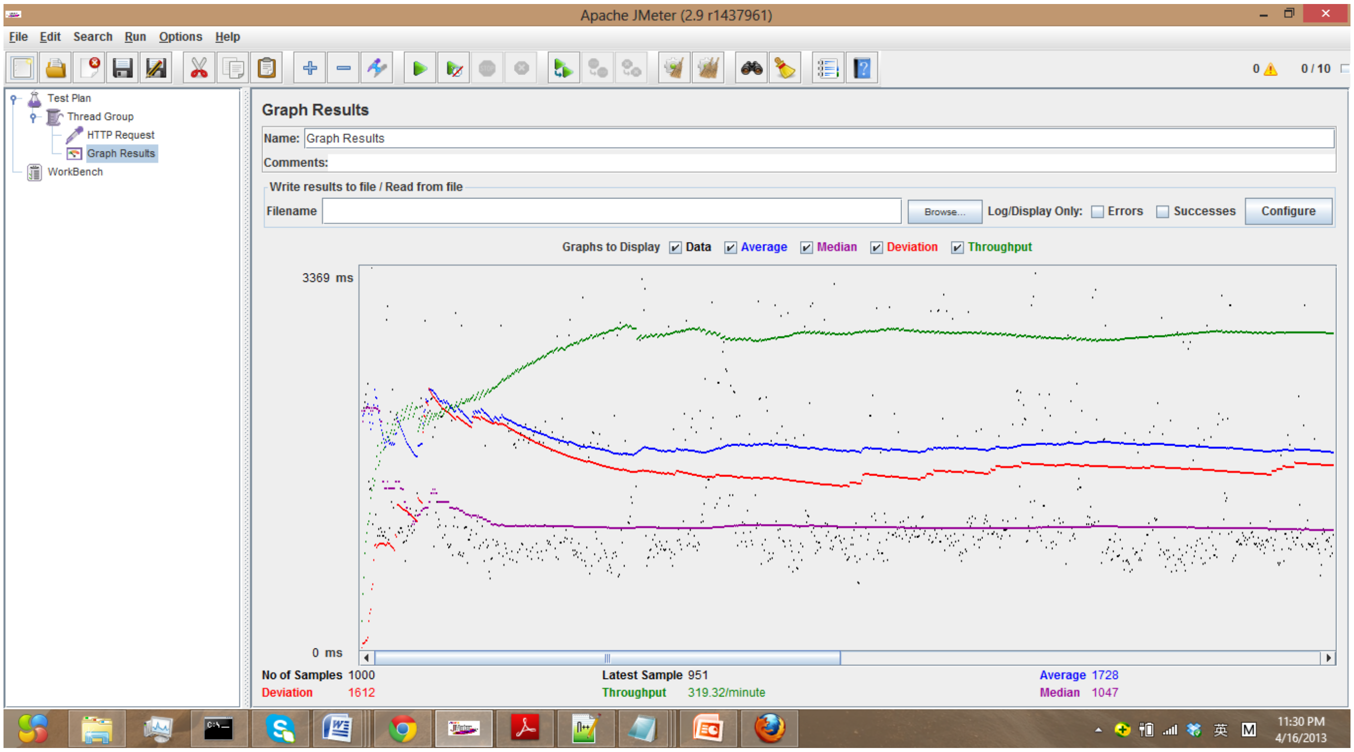
Filename

Log/Display Only: ☐ Errors ☐ Successes

Graphs to Display ☑ Data ☑ Average ☑ Median ☑ Deviation ☑ Throughput

No of Samples 1000
Latest Sample 576
Average 932
Deviation 280
Throughput 316.956/minute
Median 923

Figure 5.1 JMeter test result - Home page (5 user)



Graph Results

Name: Graph Results

Comments:

Write results to file / Read from file

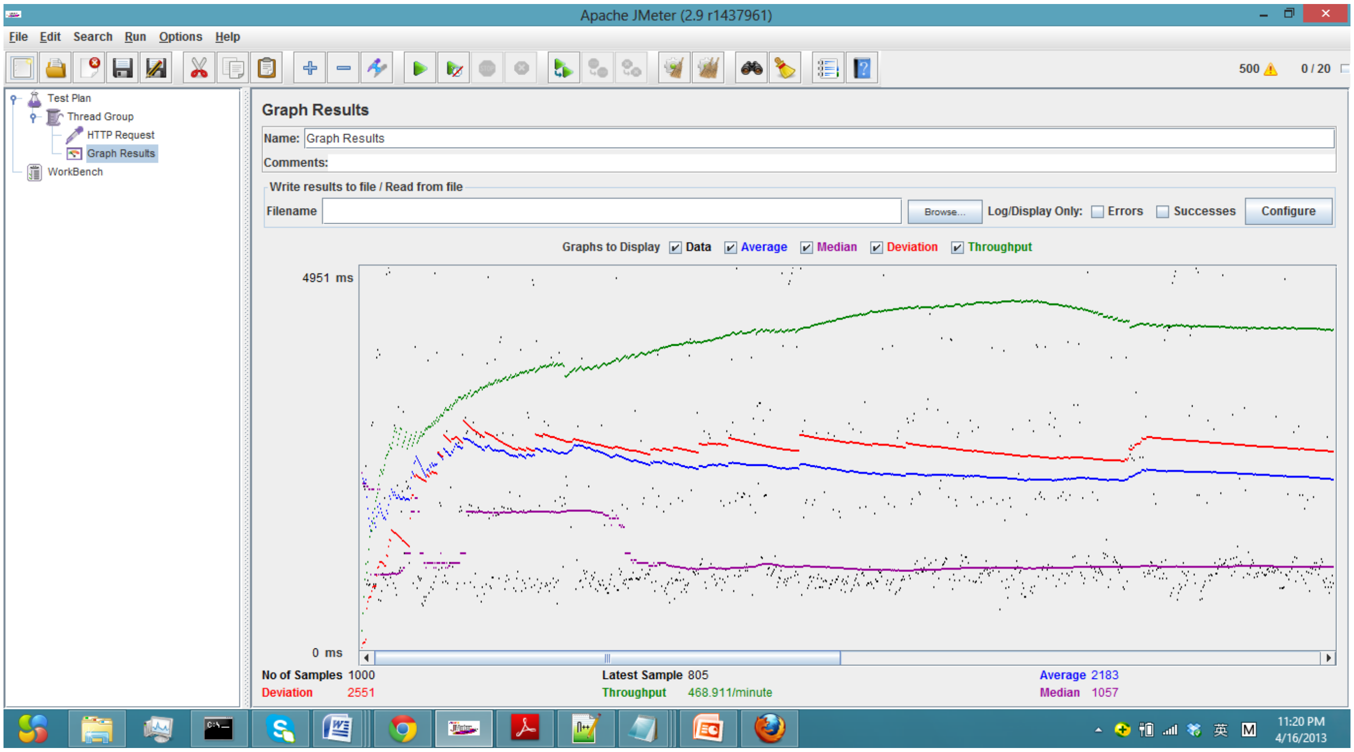
Filename

Log/Display Only: ☐ Errors ☐ Successes

Graphs to Display ☑ Data ☑ Average ☑ Median ☑ Deviation ☑ Throughput

No of Samples 1000
Latest Sample 951
Average 1728
Deviation 1612
Throughput 319.32/minute
Median 1047

Figure 5.2 JMeter test result - Home page (10 user)



Graph Results

Name: Graph Results

Comments:

Write results to file / Read from file

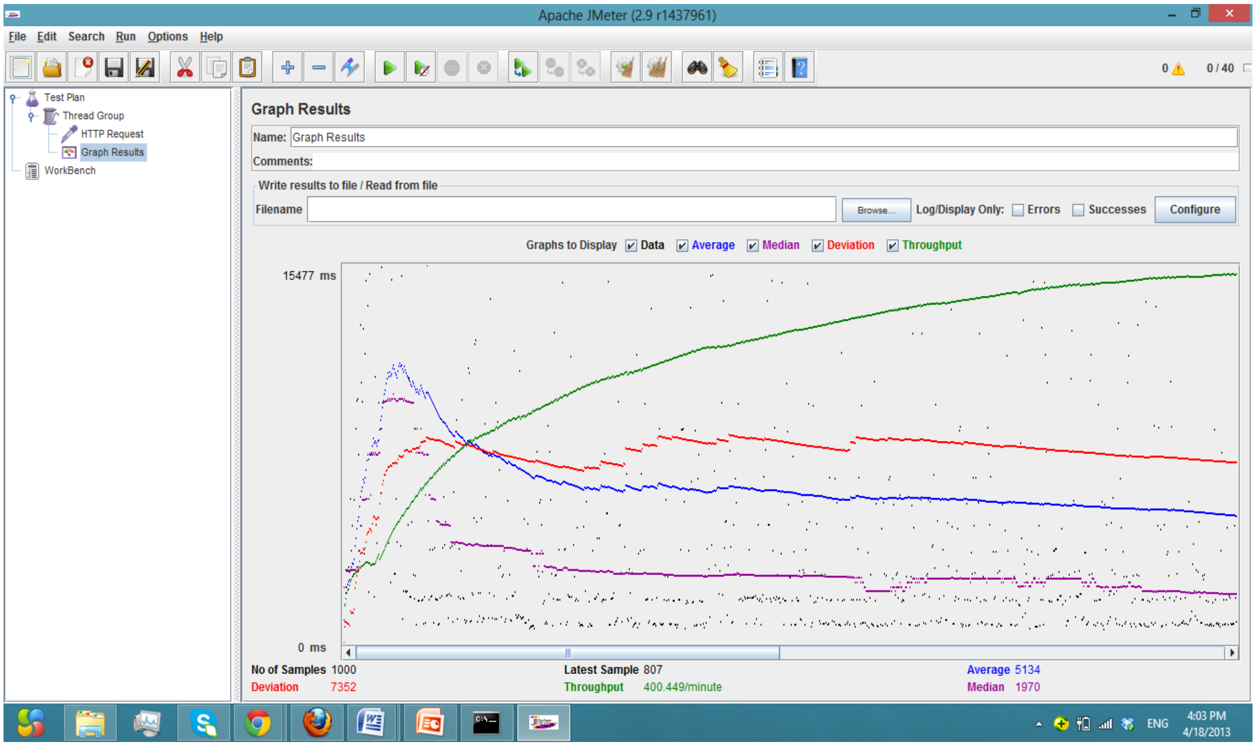
Filename Log/Display Only: □ Errors □ Successes

Graphs to Display ☑ Data ☑ Average ☑ Median ☑ Deviation ☑ Throughput

No of Samples 1000 Latest Sample 805 Average 2183

Deviation 2551 Throughput 468.911/minute Median 1057

Figure 5.3 JMeter test result - Home page (20 user)



Graph Results

Name: Graph Results

Comments:

Write results to file / Read from file

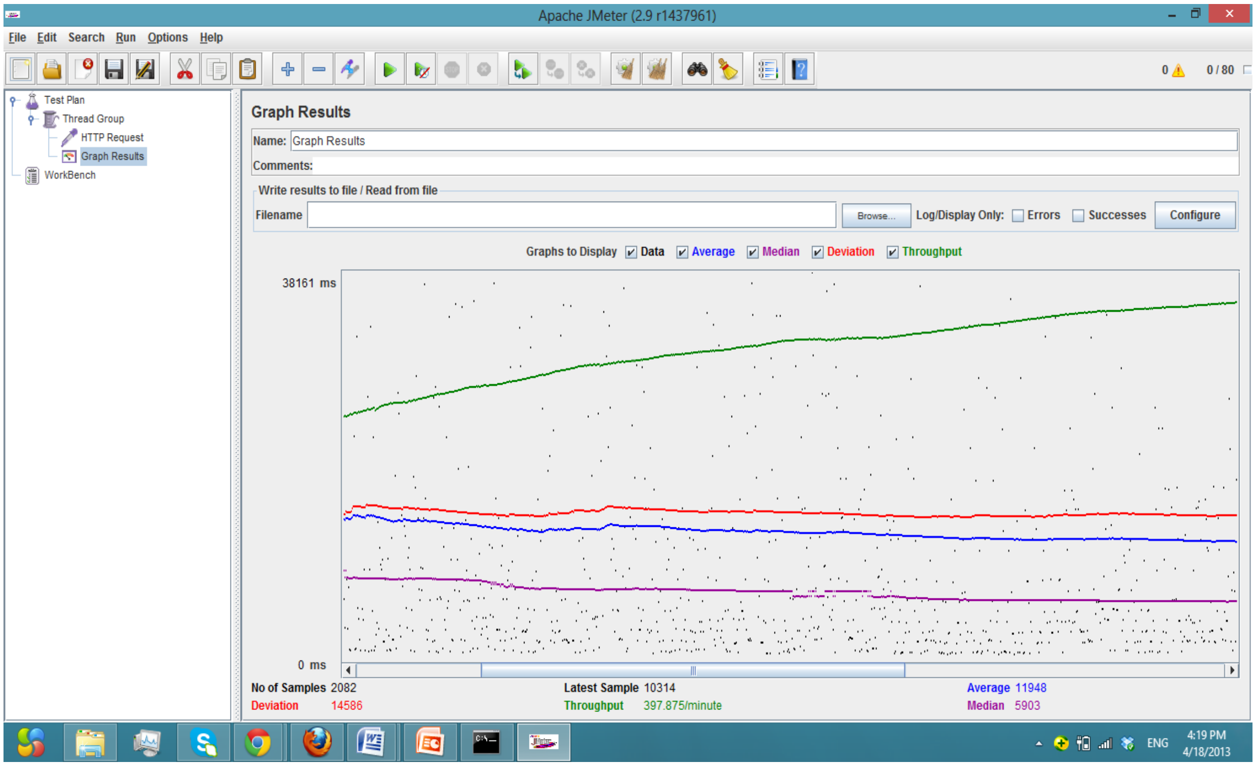
Filename Log/Display Only: □ Errors □ Successes

Graphs to Display ☑ Data ☑ Average ☑ Median ☑ Deviation ☑ Throughput

No of Samples 1000 Latest Sample 807 Average 5134

Deviation 7352 Throughput 400.449/minute Median 1970

Figure 5.4 JMeter test result - Home page (40 user)



Name: Graph Results

Comments:

Filename

Log/Display Only: □ Errors □ Successes

Graphs to Display ☑ Data ☑ Average ☑ Median ☑ Deviation ☑ Throughput

No of Samples 2082

Deviation 14586

Latest Sample 10314

Throughput 397.875/minute

Average 11948

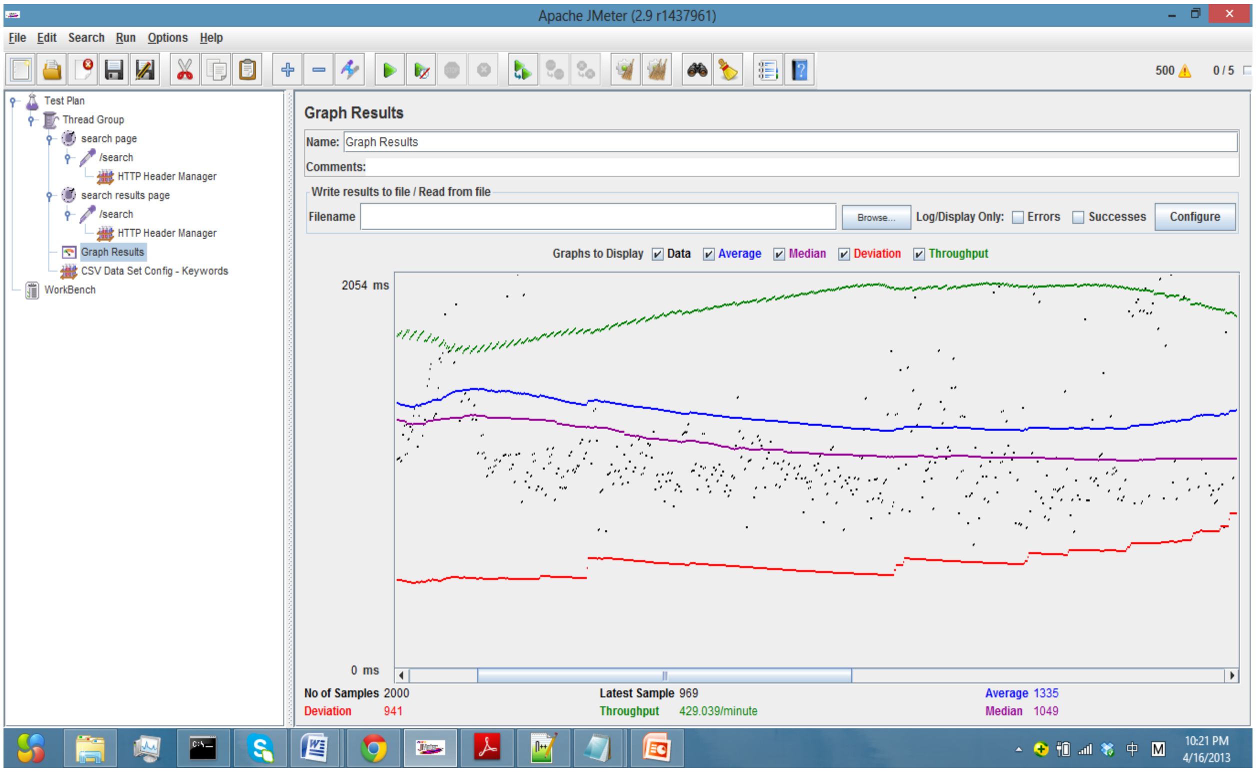
Median 5903

Figure 5.5 JMeter test result - Home page (80 user)

	Throughput	Average response time (millisecond)	Deviation
5 user	316.956/minute	932	280
10 user	319.32/minute	1728	1642
20 user	468.911/minute	2183	2551
40 user	400.449/minute	5134	7352
80 user	397.875/minute	11948	14586

Table 5.1 Home page test results summary

Table 5.2.1 lists the throughput, average response time and deviation for the home page under different number of users. The throughput increases as the total number of user for the website increases until the user number hit 40, the throughput start to decrease when the user number set to 40 and 80. The average response time and the deviation increases as the total number of the user increased. The average response times were ranged from around 1second to 12 seconds with deviation ranged from 0.3 second to 14 seconds. Based on previous advices regarding response times, if there are fewer than 40 users using this website at the same time, they may notice a little bit delay but will still keep their attention focusing on the webpage. Once the user number hit 80, the website will start to perform very slowly and will no long meet the performance requirement. And also, you can see that as we double the number of user the average response time was almost doubled as well, except from 10 users to 20 users. So it seems that the number of user and the average response time have a linear relationship.



Name: Graph Results

Comments:

Filename

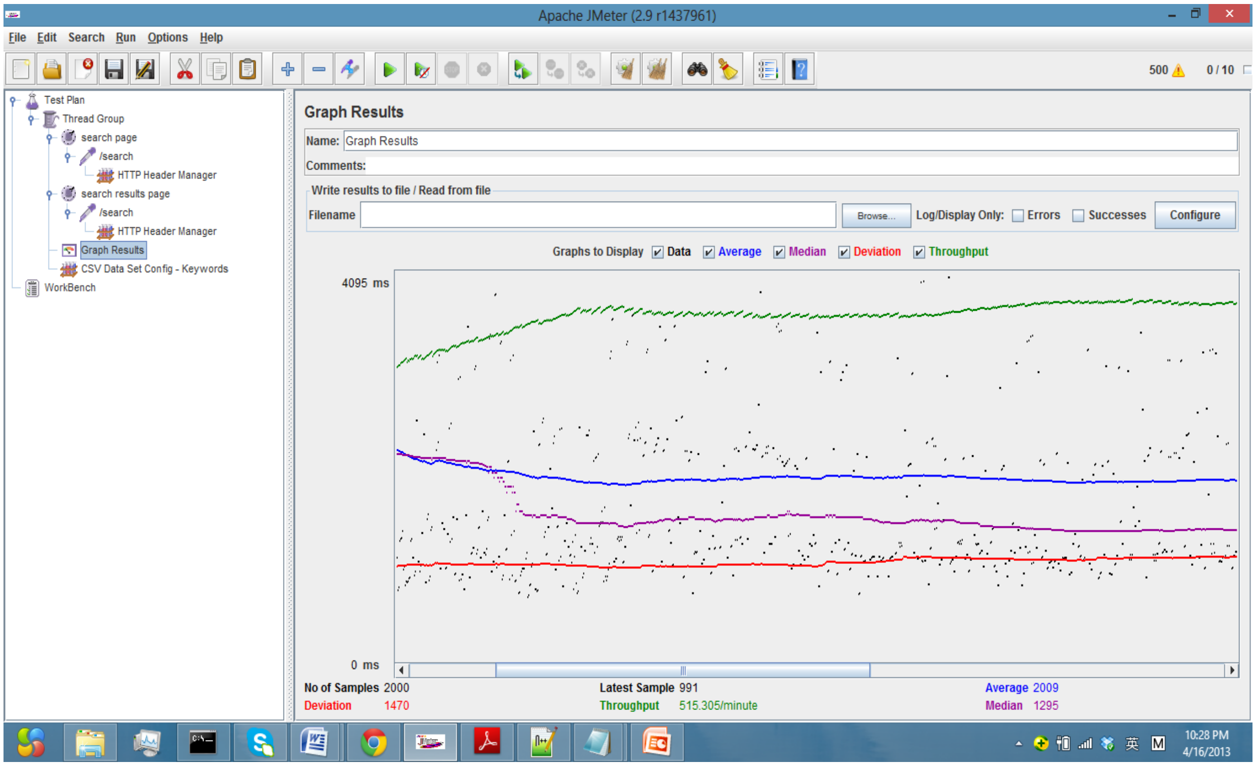
Log/Display Only: □ Errors □ Successes

Graphs to Display ☑ Data ☑ Average ☑ Median ☑ Deviation ☑ Throughput

No of Samples 2000 | Latest Sample 969 | Average 1335

Deviation 941 | Throughput 429.039/minute | Median 1049

Figure 5.6 JMeter test result - Search page (5 user)


Apache JMeter (2.9 r1437961)
File Edit Search Run Options Help
500 ⚠ 0 / 10
Browse...
Configure
4095 ms
0 ms
10:28 PM
4/16/2013

- Test Plan
  - Thread Group
    - search page
      - /search
        - HTTP Header Manager
    - search results page
      - /search
        - HTTP Header Manager
    - Graph Results
    - CSV Data Set Config - Keywords
- WorkBench

Graph Results

Name: Graph Results

Comments:

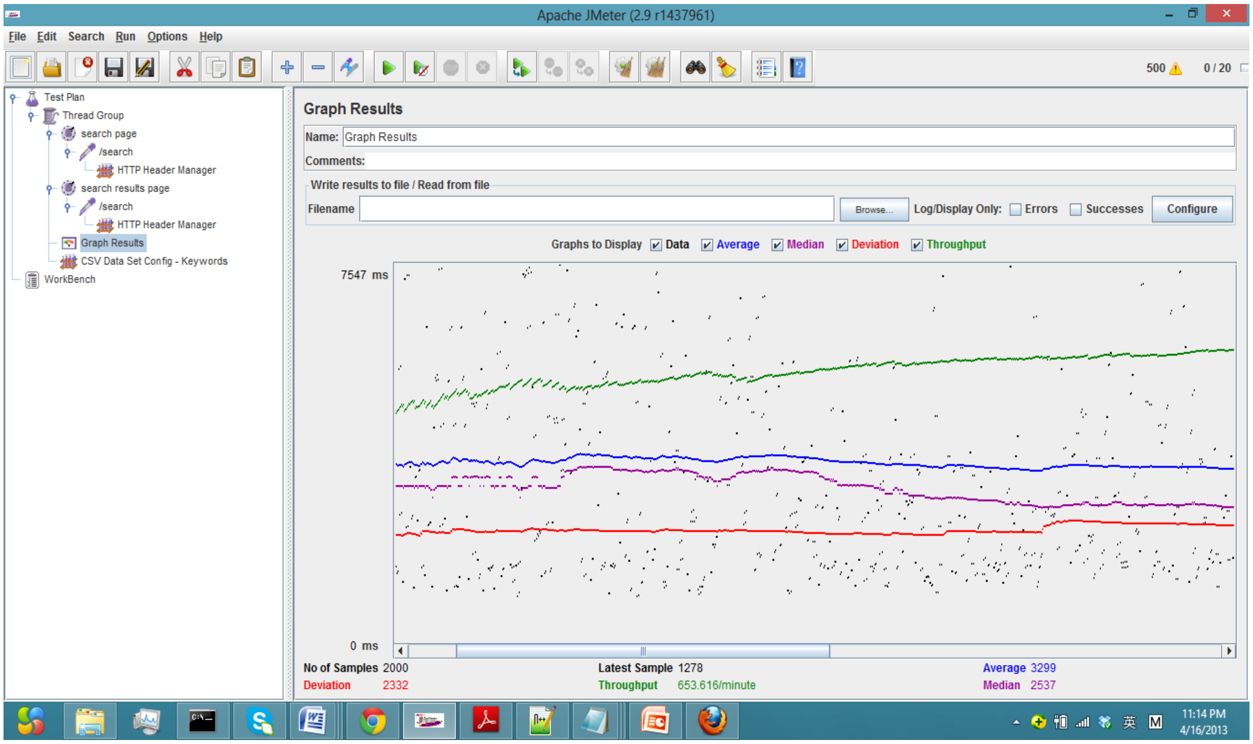
Write results to file / Read from file

Filename | Log/Display Only: □ Errors □ Successes

Graphs to Display ☑ Data ☑ Average ☑ Median ☑ Deviation ☑ Throughput

No of Samples 2000 | Latest Sample 991 | Average 2009
Deviation 1470 | Throughput 515.305/minute | Median 1295

Figure 5.7 JMeter test result - Search page (10 user)


Apache JMeter (2.9 r1437961)
File Edit Search Run Options Help
500 ⚠ 0 / 20
Browse...
Configure
7547 ms
0 ms
11:14 PM
4/16/2013

- Test Plan
  - Thread Group
    - search page
      - /search
        - HTTP Header Manager
    - search results page
      - /search
        - HTTP Header Manager
    - Graph Results
    - CSV Data Set Config - Keywords
- WorkBench

Graph Results

Name: Graph Results

Comments:

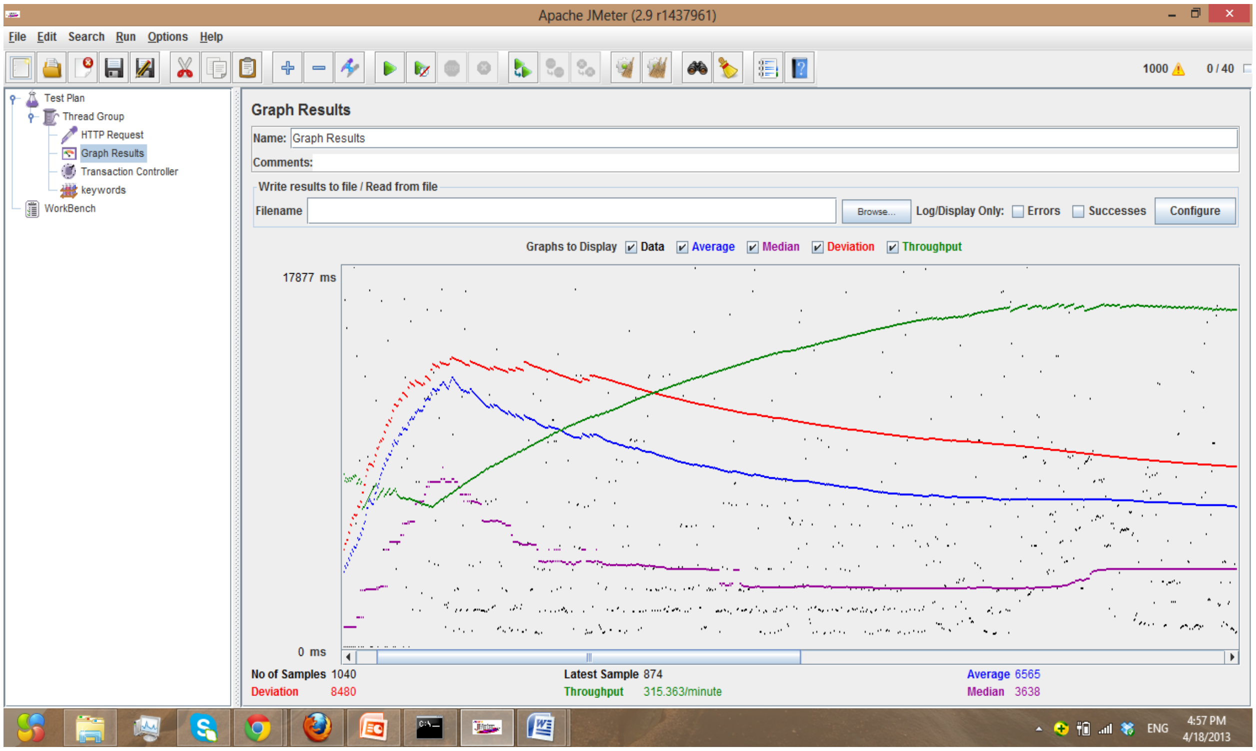
Write results to file / Read from file

Filename | Log/Display Only: □ Errors □ Successes

Graphs to Display ☑ Data ☑ Average ☑ Median ☑ Deviation ☑ Throughput

No of Samples 2000 | Latest Sample 1278 | Average 3299
Deviation 2332 | Throughput 653.616/minute | Median 2537

Figure 5.8 JMeter test result - Search page (20 user)


Apache JMeter (2.9 r1437961)
File Edit Search Run Options Help
1000
0 / 40
Test Plan
Thread Group
HTTP Request
Graph Results
Transaction Controller
keywords
WorkBench
Graph Results
Write results to file / Read from file
Browse...
Log/Display Only: □ Errors □ Successes
Configure
Graphs to Display ☑ Data ☑ Average ☑ Median ☑ Deviation ☑ Throughput
17877 ms
0 ms
ENG
4:57 PM
4/18/2013

Name: Graph Results

Comments:

Filename

No of Samples 1040

Deviation 8480

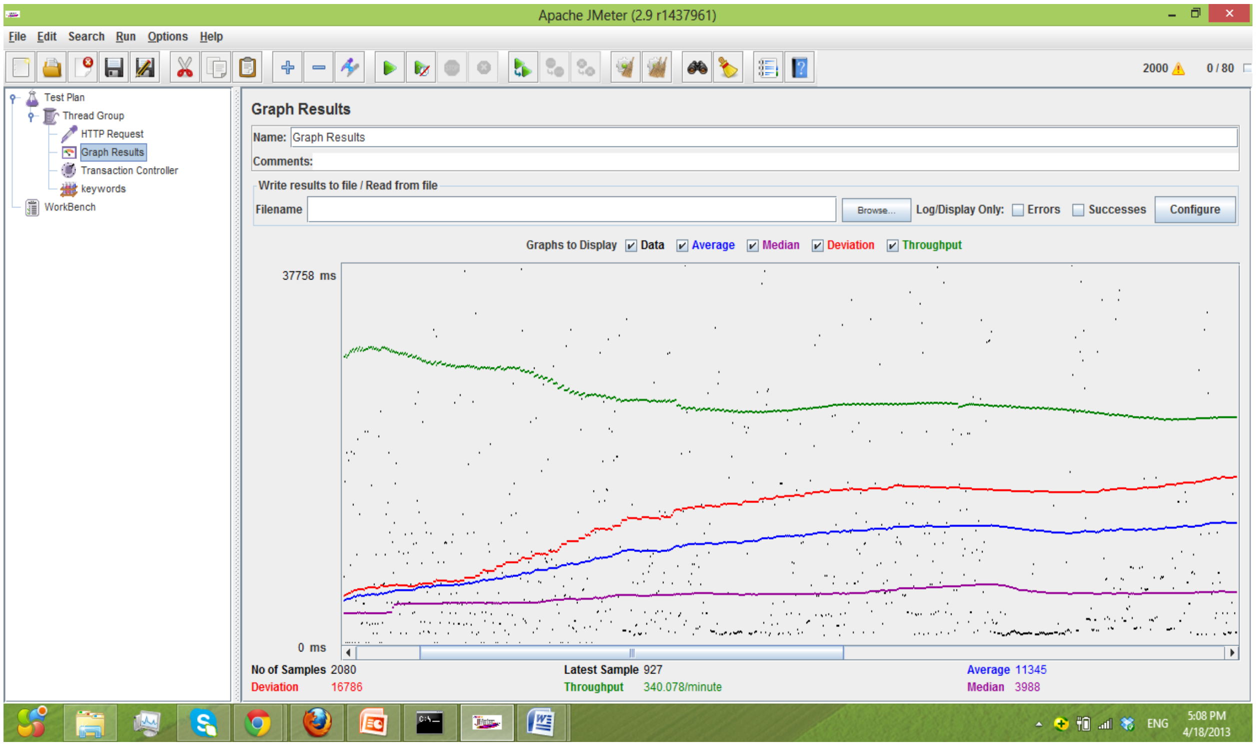
Latest Sample 874

Throughput 315.363/minute

Average 6565

Median 3638

Figure 5.9 JMeter test result - Search page (40 user)


Apache JMeter (2.9 r1437961)
File Edit Search Run Options Help
2000
0 / 80
Test Plan
Thread Group
HTTP Request
Graph Results
Transaction Controller
keywords
WorkBench
Graph Results
Write results to file / Read from file
Browse...
Log/Display Only: □ Errors □ Successes
Configure
Graphs to Display ☑ Data ☑ Average ☑ Median ☑ Deviation ☑ Throughput
37758 ms
0 ms
ENG
5:08 PM
4/18/2013

Name: Graph Results

Comments:

Filename

No of Samples 2080

Deviation 16786

Latest Sample 927

Throughput 340.078/minute

Average 11345

Median 3988

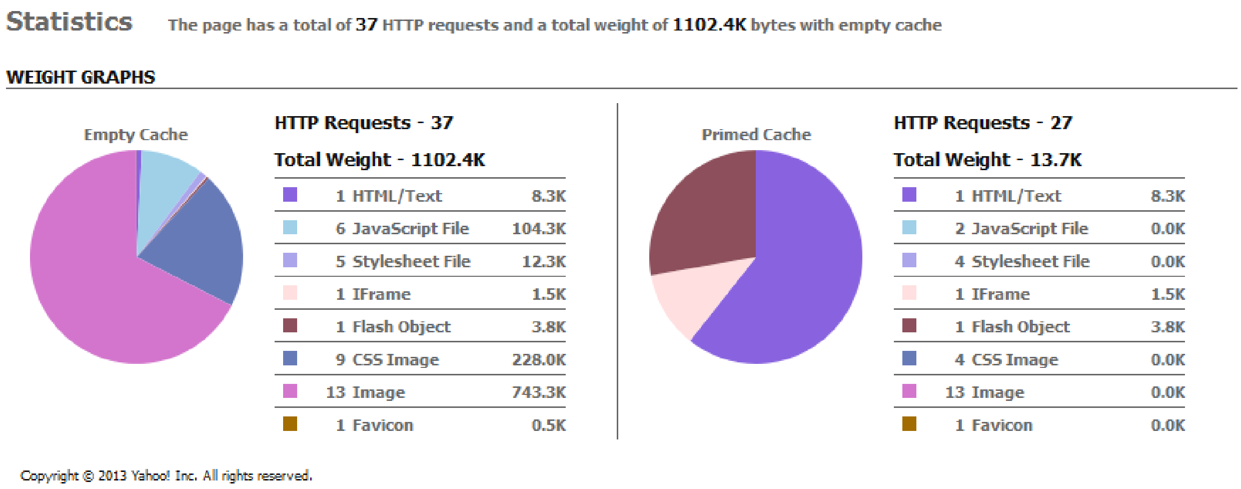
Figure 5.10 JMeter test result - Search page (80 user)

	Throughput	Average response time (millisecond)	Deviation
5 user	429.039/minute	1335	941
10 user	515.305/minute	2009	1470
20 user	653.646/minute	3299	2332
40 user	315.363/minute	6565	8480
80 user	340.078/minute	11345	16786

Table 5.2 Search page test results summary

Table 5.2.2 lists the throughput, average response time and deviation for the search page under different number of users. For the search page, the results have the same pattern, the throughout starts to decrease when the user number hit 40. The average response times were ranged from around 1second to 11 seconds with deviation ranged from 0.9 second to 17 seconds. Based on the same rule, the website was doing ok when the number of user is below 40. The users may notice a little bit delay but will not lose their patience waiting for the search results. And 80 users seem to be the limit for the website to perform well. It is reasonable to assume that in most of the cases the website will not have more than 80 users visit it at the same time, so this website will perform well under its normal working conditions.

To further break down the response time for the home page to the loading times corresponding to different components, I used a Firefox Addon called YSlow to investigate the home page of the new website.



Statistics The page has a total of **37** HTTP requests and a total weight of **1102.4K** bytes with empty cache

WEIGHT GRAPHS

Empty Cache

HTTP Requests - 37

Total Weight - 1102.4K

■	1 HTML/Text	8.3K
■	6 JavaScript File	104.3K
■	5 Stylesheet File	12.3K
■	1 IFrame	1.5K
■	1 Flash Object	3.8K
■	9 CSS Image	228.0K
■	13 Image	743.3K
■	1 Favicon	0.5K

Primed Cache

HTTP Requests - 27

Total Weight - 13.7K

■	1 HTML/Text	8.3K
■	2 JavaScript File	0.0K
■	4 Stylesheet File	0.0K
■	1 IFrame	1.5K
■	1 Flash Object	3.8K
■	4 CSS Image	0.0K
■	13 Image	0.0K
■	1 Favicon	0.0K

Figure 5.11 Summary of the total number of HTTP requests and the total weight of the Home page as well as page weight analysis for both Empty Cache and Primed Cache perspectives

Figure 5.11 summarizes the total number of HTTP requests and the total weight of the Home page as well as page weight analysis for both Empty Cache and Primed Cache perspectives. On this page over 90% of the weight is in Image files, which includes 13 images and 9 CSS images. Because this page is an image intensive web page, we can reduce over 90% the total weight of the page by using primed Cache. So users will experience a significant reducing of the waiting time after the first time of visiting the home page of the new website.

Chapter 6 - Conclusions and Future Enhancements

6.1 Conclusions

In this project, I and Nikhita Addanki developed a new website for Grace Baptist Church to replace their current website. The new website was implemented using WordPress based on the designs given by the Church. Compare to the current website, the new website improves the page appearance and has better content presentation structure and mobile support. The performance test showed that the new website has a reasonable performance under normal work load level. So the new website successfully meets all the user requirements for both usability and performance. The Grace Baptist Church will utilize the full function of this website to share church mission, events, and opportunities to people around the world.

6.2 Future Enhancements

The website can further be improved in the following way:

- Provide an option for people to subscribe for "Bible Reading Plan" and automatically send emails to subscribers according to their plan.
- Provide an option for people to post the reviews of the Christian Books they read.
- Let people register for special events online
- Add Book store or Music store to recommend good books to music, and provide links to online resources
- Create more page templates on WebPress to present different content, make the website more dynamic

Chapter 7 - References

T.Brinck, D.Gergle, and S.D. Wood, “Usability for the Web: Designing Web Sites that Work,” *San Francisco: Morgan Kaufmann*, 2002.

Jakob Nielsen's Website, available from Internet http://www.nngroup.com/

H. Shahizan, L. Feng, “Evaluating the Usability and Content Usefulness of Web Sites: A Benchmarking approach,” *Journal of Electronic Commerce in Organizations, ABI/INFORM Global*, Apr-Jun 2005, pp. 46.

K. Guenther, “Assessing Web Site Usability,” *Vol. 27 Issue 2, Ebsco*, 2003, pp 65.

Human Factors International, available from Internet, http://www.humanfactors.com/downloads/documents/HFI_brochure.pdf, Oct 2007.

Steve Souders, “High Performance Web Sites”, 2007

Card, S. K., Robertson, G. G., and Mackinlay, J. D. “The information visualizer: An information workspace.*Proc,” ACM CHI'91 Conf.*, 1991, pp181-188.

http://en.wikipedia.org/wiki/Web_content_management_system

http://codex.wordpress.org

http://jmeter.apache.org/