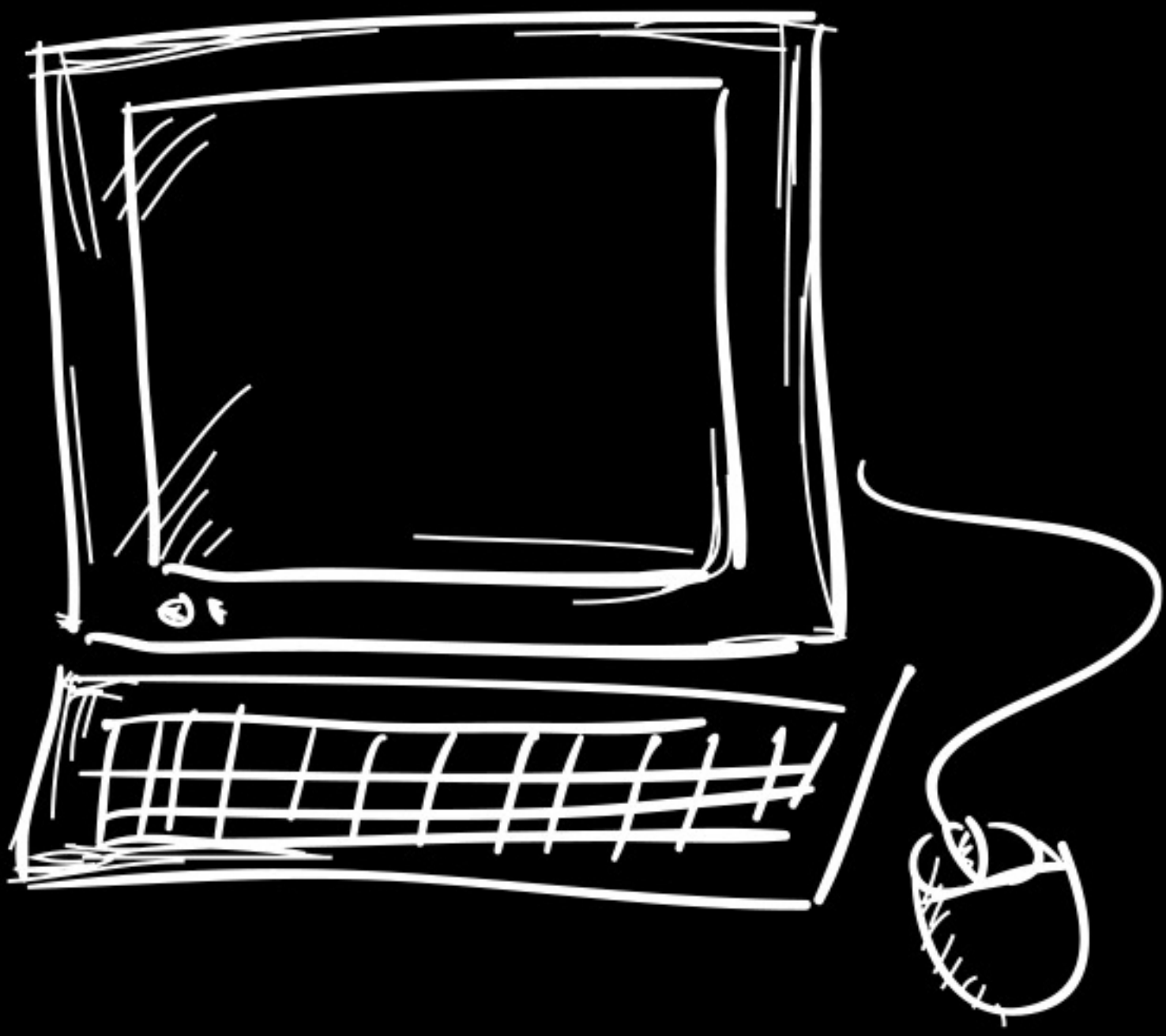
THE NON-TECHNICAL GUIDE TO

WEB TECHNOLOGIES



TOMMY CHHENG

Non-Technical Guide to Web Technologies

Tommy Chheng

Contents

Chapter 1

Introduction

Who is this book for?

In the past 15 years, I've worked with a variety of non-technical co-workers in big and small technology companies. I wrote this book to pass on knowledge which could be helpful for those working in the internet field. This book is for everyone who wants to understand the basics of web technologies. You may be a student interested in a career with a software company, a recruiter seeking new talent, a designer creating beautiful web experiences, a non-technical startup founder, a salesperson seeking new leads, or perhaps a technology reporter. The list goes on!

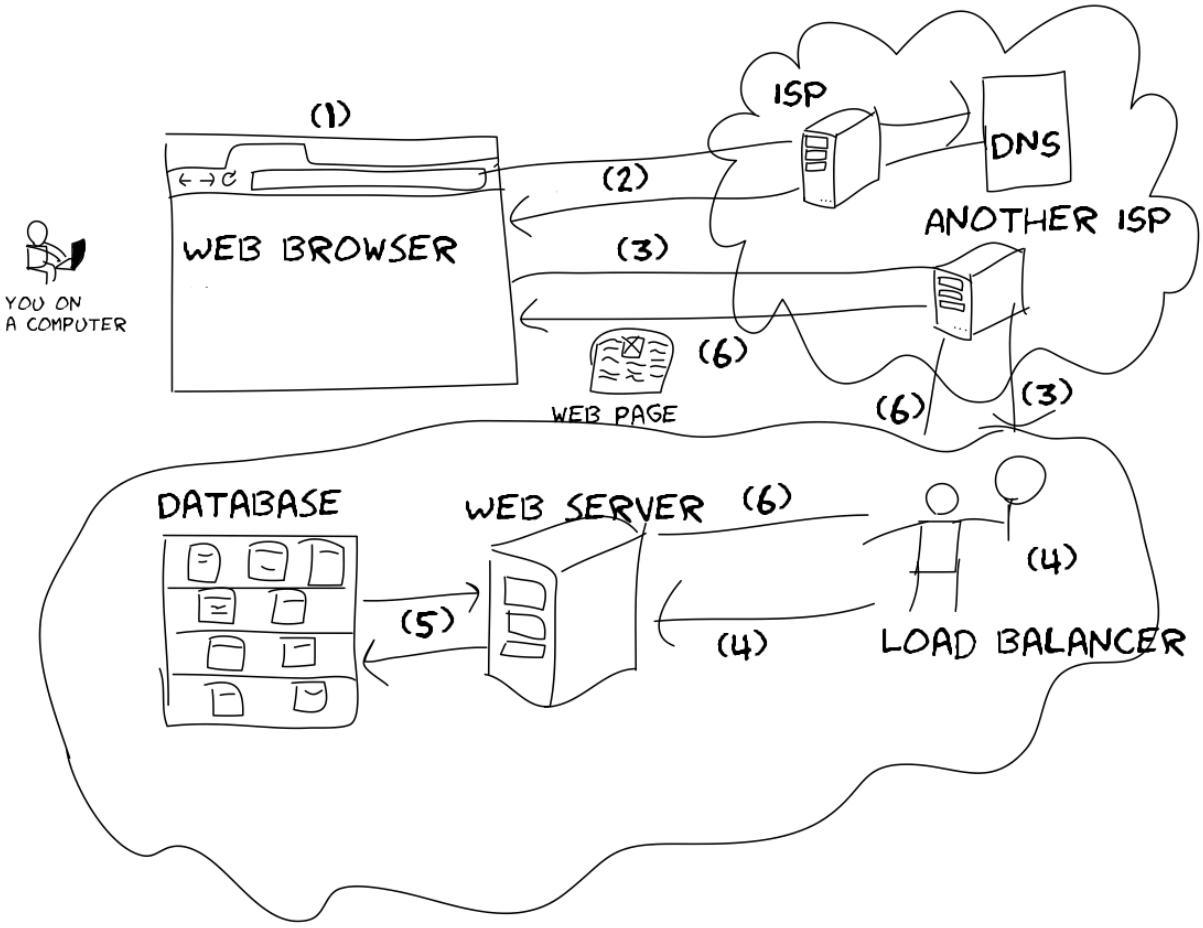
If you work in the internet world, a basic fundamental understanding of these technologies will make you more successful.

How to use this guide?

This guide is meant to be an introduction and will not dive into the detail that would be covered by an engineering book. After reading this, you should have a clearer understanding of web technologies and their use. Many concepts are simplified to give you a quick explanation and should be considered as a starting point. Each section can also be read independently. If you only wish to learn about programming languages, you can skip the internet section. I have included many exploratory links throughout the book if you are interested in learning more depth about a topic.

Overview

Web technologies can be daunting to learn because the vocabulary used to describe them is unfamiliar. Let's take a journey through visiting a website and see the web technologies we encounter along the way. Don't worry about not completely understanding this right now.



Let's say you want to visit the New York Times to read today's news.

1. First, let's start thinking of this visit as a request for today's news web page from New York Times. You do this by typing in http://www.nytimes.com on a web browser (e.g. Google Chrome, Firefox, Internet Explorer).
2. When you press enter, the web browser will need to lookup New York Times' *IP address* from the *Domain Name System (DNS)*.
3. With this address, your web browser will send a *HTTP* request through the *internet* to New York Times.
4. Now, New York Times doesn't just run on one computer, it has hundreds in its *infrastructure*. The request arrives at the *load balancer*. The load balancer will forward your request to one of New York Times' *web servers* (computers hosting web pages are also known as servers) which are running *web applications*.
5. One of these web applications will get a list of news stories from its *database* and generate the *HTML* news web page.
6. It will then send the generated HTML page back to your web browser via the internet.

As you can see, even in this overly simplified explanation, there are many components and possibly unfamiliar terms used to describe your interactions with a website. The remaining sections of this book will cover these technology terms, how they work, and how they are used.

Chapter 2

Common Questions

Are some websites easier to develop than others?

A website can be categorized on a complexity scale. On one end we have brochure websites, and on the other we have web applications. Brochure websites, like those used by restaurants, look the same to every user. Web applications are interactive and change depending on the user. Facebook is a classic example of a web application, because the content experience is unique to each user.

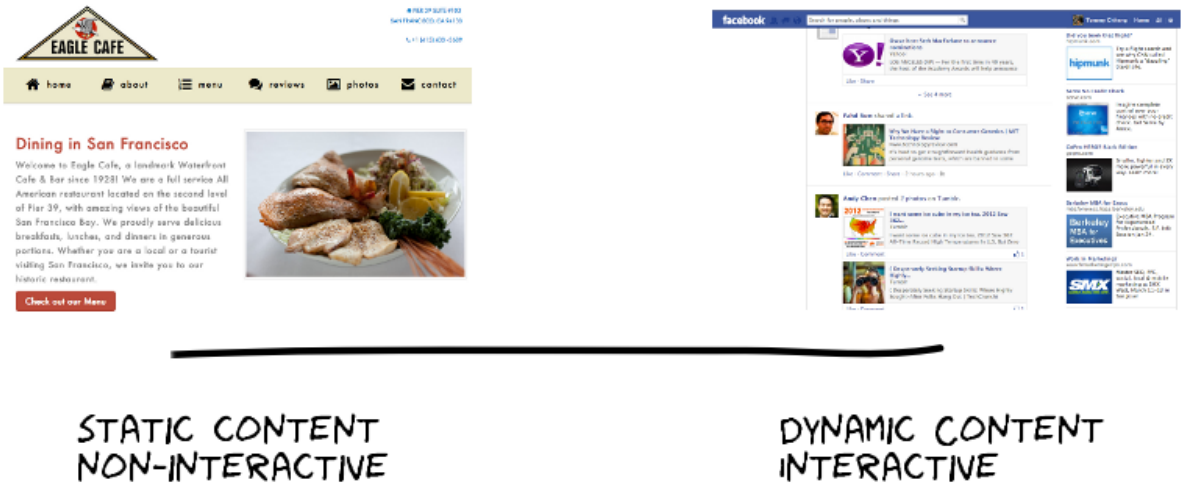


Figure 2.1: A brochure website like a restaurant site is easier to develop than a web application like Facebook.

There are websites that lie between a simple brochure site and a web application. One example could be a bicycle manufacturer's website, which can consist of a static brochure section and an interactive shopping experience. The more interactive a site is, the more difficult the site will be to develop and maintain.

Because the brochure website does not involve many components, we won't spend much time discussing it. Instead, when we refer to websites in this book, we're actually referring to those that are web applications.

If you're looking to create a website for your business, be mindful of which type will best suit your needs.

Why is it difficult to develop a web application?

Web applications are challenging to develop because it is difficult to conceive the end product. To understand web development, let's use the analogy of designing an automobile.

Bob starts building a platform for a car but, halfway through, changes his mind and wants a truck instead. He doesn't want to start from scratch and decides to use the car platform as the foundation for his new truck. The result might be usable but will require compromises. This is an inevitable part of the discovery process in choosing the right idea to pursue. The development of web applications often follow this pattern.

When the different groups involved in the development process have a better understanding of the underlying systems and can communicate in the same language, they are able to reduce misunderstandings that can lead products astray and will therefore save time and money.

Beyond developing the product, there are additional concerns to keep in mind:

- scaling the application (So it doesn't slow down or crash.)
- scaling the number of developers (Is the code maintainable by more than one person?)
- scaling the development process (The more functionality an application supports, the more difficult it is for a programmer to add new features in the future.)

Why is it difficult to scale a web application?

Imagine a web application is a Target store that you are the proud owner of. The new iPhone was just released and your store is the only one in town that still has it in stock. You could face the situations below:

- Will too many customers come through the doors at the same time?
- Will you have enough staff and checkout lines to serve each customer?
- What if customers waiting for iPhones block customers wanting to see the new Samsung phone?

These situations are similar in a web application:

- Have you set up enough monitoring to know when your application is showing signs of increased visitors?
- Can you quickly add more servers to deal with the increased traffic?
- Will having a lot of users using one part of your application affect another part?

What happens to the web application after it's finished?

In print media, after the latest issue of a magazine has been published, it is essentially finished and the writers can move onto the next one. In software development, however, no product is ever finished as long as it is being used. Each product and its features come with overhead and maintenance costs including patching security vulnerabilities, correcting copyright violations, and scaling server load.

Chapter 3

Components

This book divides web technologies into the following sections:

- Internet: The connection between you and the website
- Application: The brains of a website
- Data: How a website stores the information
- Infrastructure: How the website is configured and managed on computers

Internet

In this section, we will discuss a few foundational internet technologies used to connect the world's computers together including your computer.

Application

We'll discuss the programming languages that web developers use to create websites. This section is divided into two parts: the front-end (the computer code that your computer runs) and back-end (the computer code that the server runs).

Data

Being able to store a user's data for a web application can make visiting the site a unique experience for each user. We discuss databases, cache and search systems used to customize a user's visit.

Infrastructure

Web applications require a good deal of configuration and management besides the web application itself. We will cover how web servers are used, as well as the operating systems powering them.

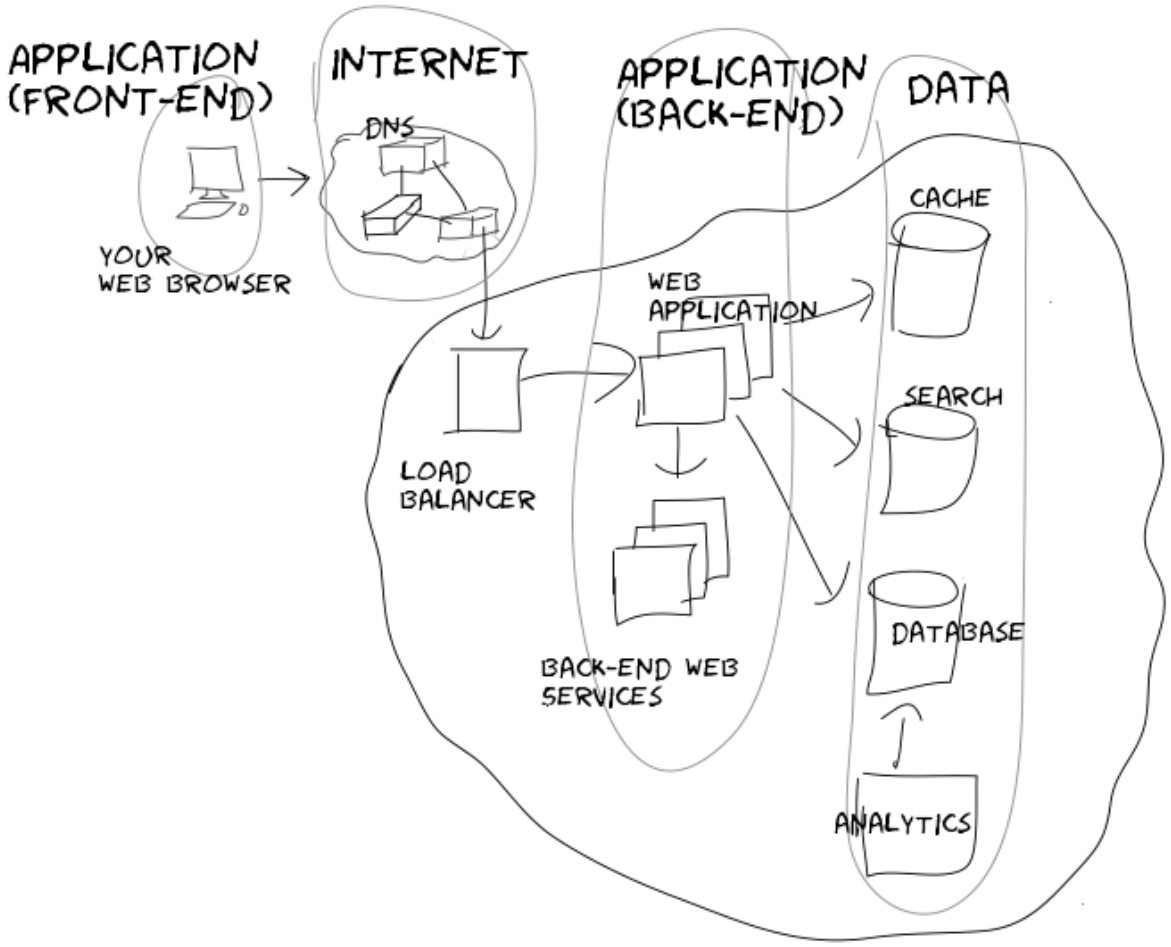
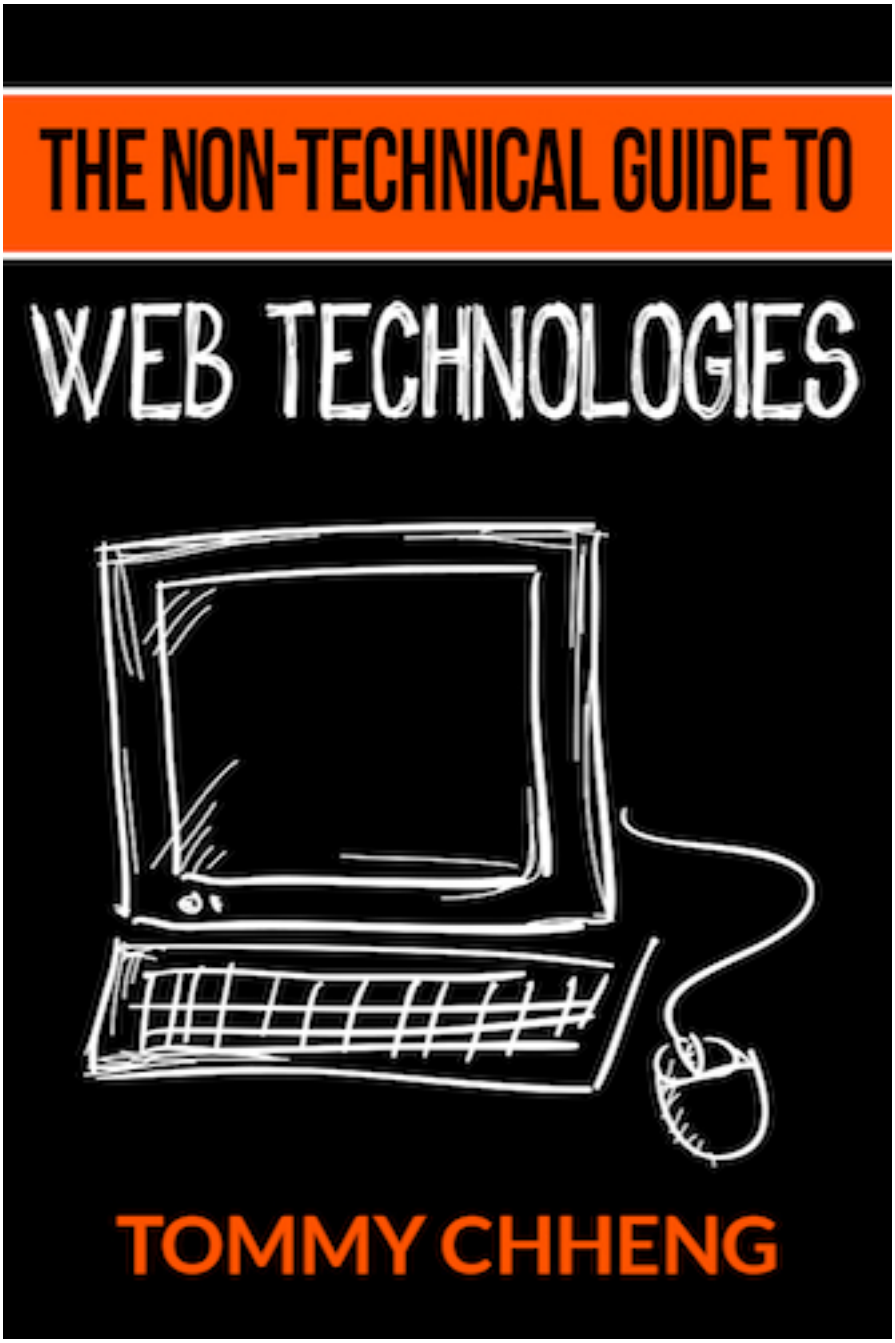


Figure 3.1: Components to a modern web application stack.

Finding the book helpful?



Finish reading at http://www.discoverbits.com/ebooks/non-technical-guide-to-web-technologies/