

Creating a Simple Chrome Extension

Instructions

Submitted to

Thomas M. Keating
Assistant Teaching Professor
School of Computer Science
Carnegie Mellon University

Prepared by

Justin Gallagher
Ted Li

School of Computer Science
Carnegie Mellon University
October 28, 2014

Contents

1	Overview	1
1.1	Introduction	1
1.2	Motivation	1
1.3	Requirements	1
1.4	Goal	1
1.5	Cautions and Warnings	1
2	Understanding the Constituents of a Google Chrome Extension	2
3	Creating a Chrome Extension	2
3.1	Step 1: Create a Folder For Our Project	2
3.2	Step 2: Declaring the Manifest File	2
3.3	Step 3: Adding In the Resources For the Chrome Extension	3
4	Adding the Extension to Chrome	4
5	Uploading the Extension to the Chrome Web Store	7

1 Overview

1.1 Introduction

Chrome extensions allow you to add features and functionality to Google's Chrome browser without modifying source code. This guide will walk you through creating a simple extension and uploading it to the Chrome Web Store for the public to download.

The tutorial has 5 steps and can be completed in approximately 15 minutes.

1.2 Motivation

Chrome extensions allow you to add features and functionality to Google's Chrome browser without modifying source code. The framework utilizes web development technologies such as HTML, CSS, and JavaScript which may already be familiar to you. Extensions are also modular, so you can build many which work in their own environments without conflict. Finally, Chrome extensions can be easily distributed to users through the Chrome Web Store, allowing wide adoption of your creations.

1.3 Requirements

For this guide, you will need:

1. A Internet connected Windows / Mac computer with Google Chrome installed.
2. A text editor in which you are proficient.
3. A Google account.
4. \$5 to pay Google's developer registration fee.
5. A simple logo for your Chrome extension.
6. Elementary skills in HTML and JavaScript.

1.4 Goal

By the end of this guide, you will have created a simple Chrome extension which produces a pop-up window reading "Hello world!" when a button in the top right toolbar is clicked. The extension will also be available for the general public to download onto their browser from the Chrome Web Store.

1.5 Cautions and Warnings

An incorrectly written extension can cause undesired behavior resulting in an unstable browser. Uploading such an extension can cause it to be removed from the Chrome Web Store and your developer account to be banned. Remember to test your extension thoroughly before uploading!

2 Understanding the Constituents of a Google Chrome Extension

This section covers the basic constituents that are required for any Google Chrome extension. Usually, Chrome extensions are packaged as a zipped archive of a folder. The archive consists of all the resources that a Chrome extension requires to run its functionalities. The files in the archive may include images, web pages or source code that the extension will find useful.

3 Creating a Chrome Extension

3.1 Step 1: Create a Folder For Our Project

For this tutorial, we will be putting everything related to our extension under the My Chrome Extension folder on the Desktop. As the first step, create a new folder on the Desktop named My Chrome Extension.

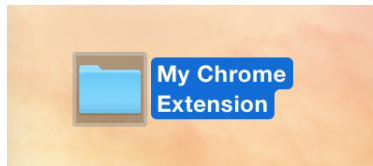


Figure 1: Creating the folder on the Desktop

3.2 Step 2: Declaring the Manifest File

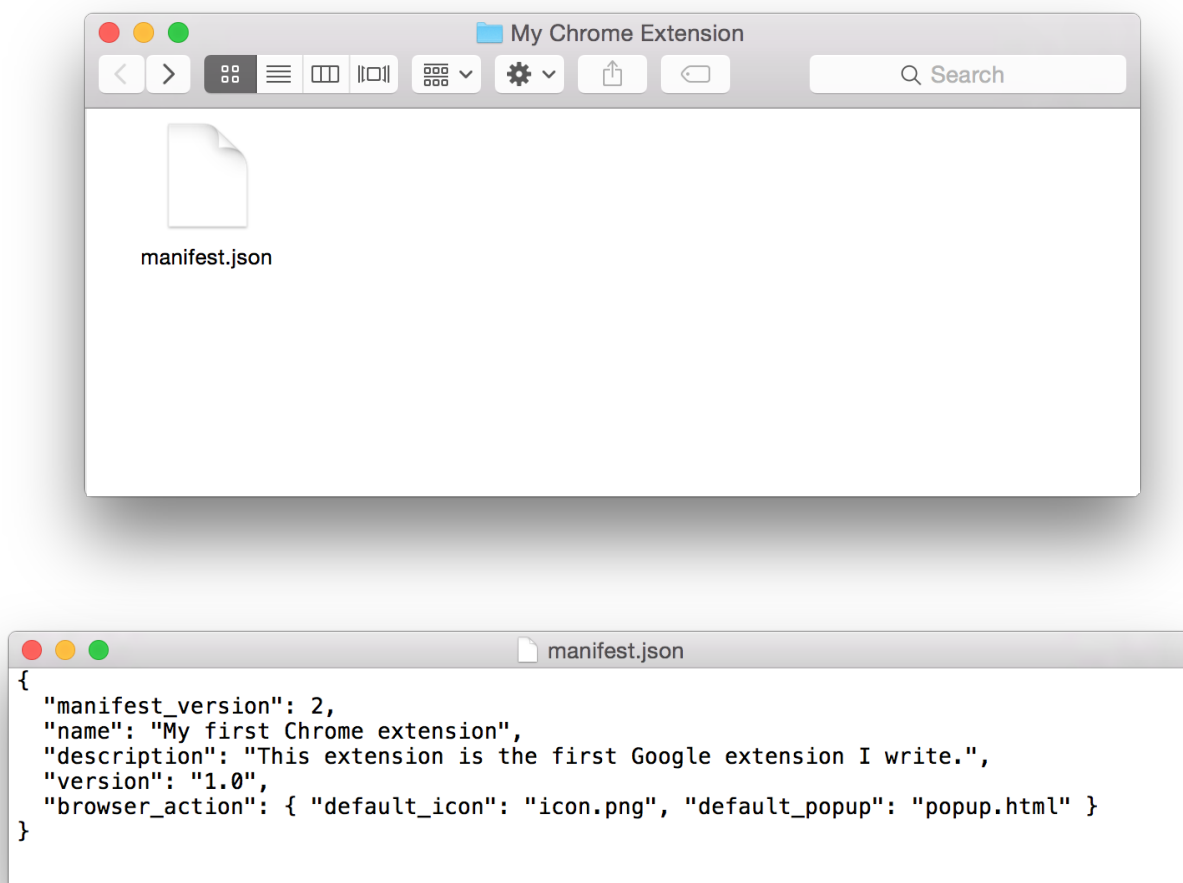
After creating a folder for our project, we'll need to create a manifest file named `manifest.json`. A manifest file can be viewed as a configuration file for any Chrome Extension that configures specific settings of our Chrome extension, such as icon, version number, extension name and description. At a high level, we'll use the manifest file to declare to Chrome its functionalities, and what permissions it requires in order to complete these functions.

For this tutorial, we will name our Chrome extension 'My Chrome extension', and we will also designate an icon for our Chrome extension. Feel free to replace the name, icon and description with your own.

Open up any text editor and save the following code snippet to the folder we just created, and name the file `manifest.json`.

```
1 {  
2   "manifest_version": 2,  
3   "name": "My Chrome extension", // The name for our Chrome extension  
4   "description": "This is a simple extension intended as a demonstration!",  
5   "version": "1.0", // The current version for our Chrome extension  
6   "browser_action": { "default_icon": "icon.png", "default_popup": "popup.html" }  
7 }
```

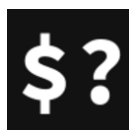
After this step, our folder should look like this:

Figure 2: Creating the `manifest.json` file

3.3 Step 3: Adding In the Resources For the Chrome Extension

By now, you probably have noticed that there were two files in the `manifest.json` file that we haven't yet created. Now, we should proceed to create the two resource files. They are called resource files because they are files that Google Chrome will eventually use to render our extension to users.

For this tutorial, we will be using the following icon. You may choose any icon you like.



Save the icon into the folder named `icon.png`.

Next, create a file in the same folder named `popup.html` with the following content:

```
1 <!doctype html>
2 <html>
3   <head>
4     <title>Hello World!</title>
5   </head>
6   <body>
7     Hello World!
8   </body>
9 </html>
```

Our folder should now look like this:

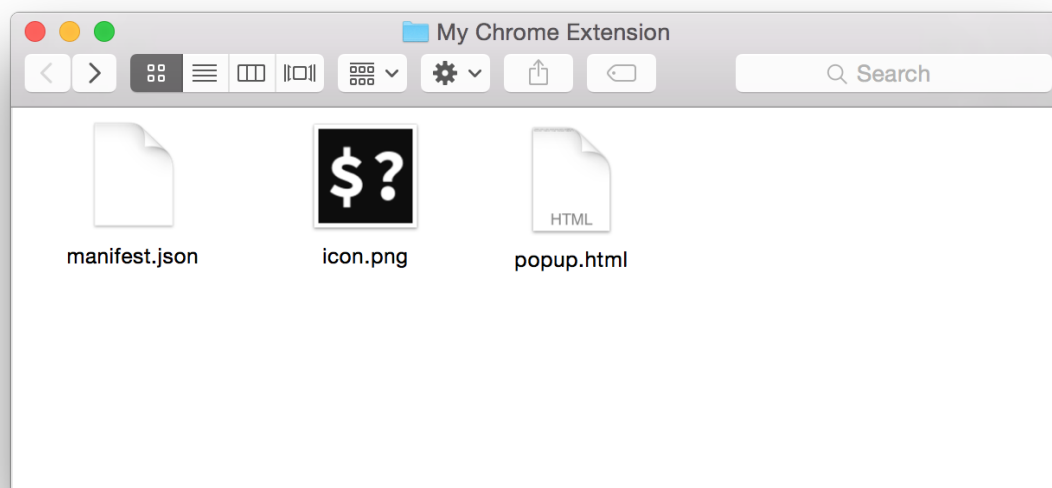


Figure 3: Adding the resource files

Now that we have created the extension, let's import into the Google Chrome browser and give it a try!

4 Adding the Extension to Chrome

This step outlines how to add the extension you wrote to your installation of Chrome, and how to run the application for yourself.

1. Open your Google Chrome browser and visit `chrome://extensions`.
2. At the top right of the page, ensure that the "Developer Mode" checkbox is checked, as shown in *Figure 4*. If it is not, click the box next to the text label to activate Developer Mode.

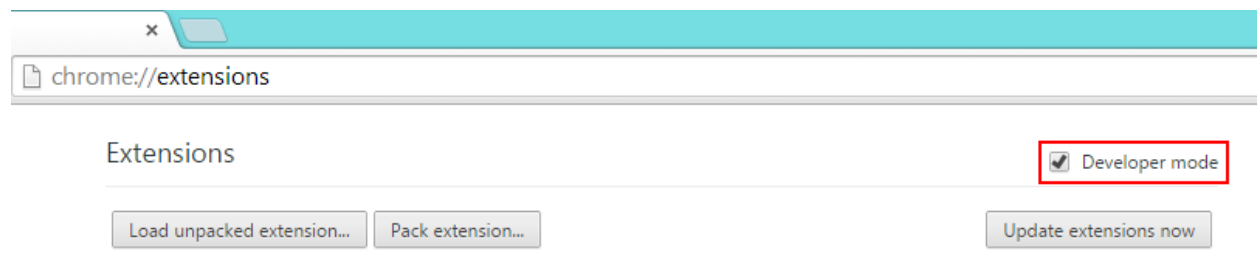


Figure 4: Enabling Chrome developer mode

3. Press the “Load unpacked extension...” button and select the folder containing your `manifest.json` file, as shown in *Figure 5*. Press “OK” to load your extension.

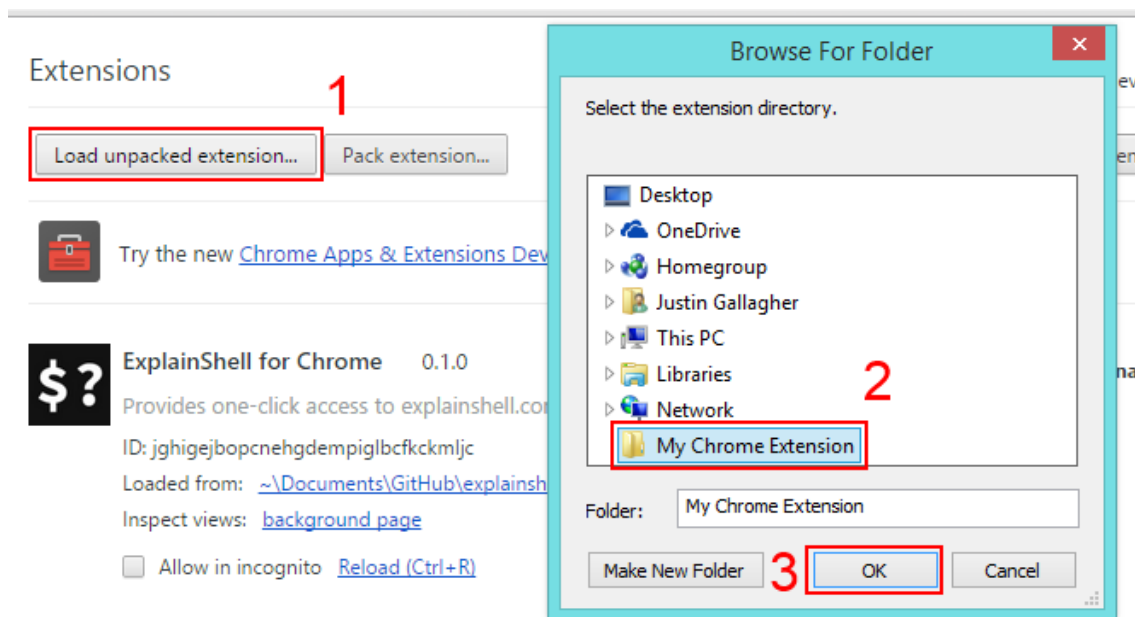


Figure 5: Adding your extension to Chrome

4. When your extension is properly loaded, you should see an entry for “My Chrome Extension” in the extensions list, as shown in *Figure 6*.

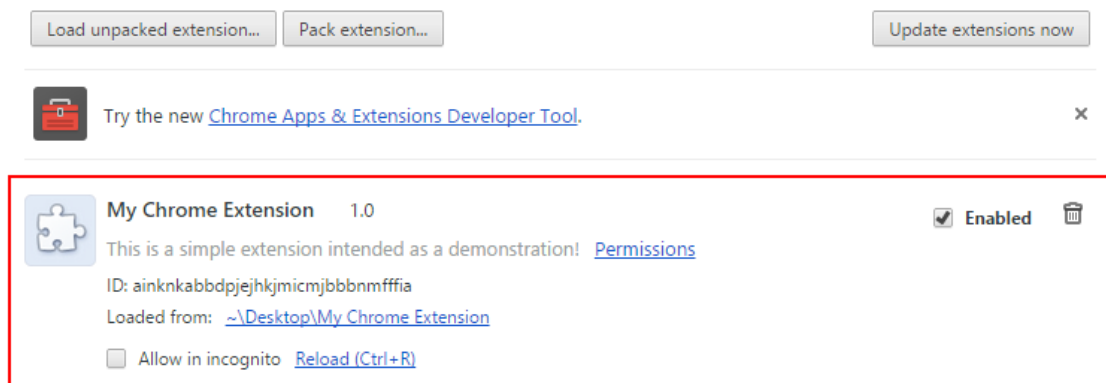


Figure 6: A successfully loaded extension

Now your extension has been loaded into the browser, and you should immediately see the extension you have written take effect.

Simply open up a new browser window, and you should already see the icon of our extension available on the toolbar. If you click the icon, a small popup will appear that says "Hello World!", as shown in Figure 7.

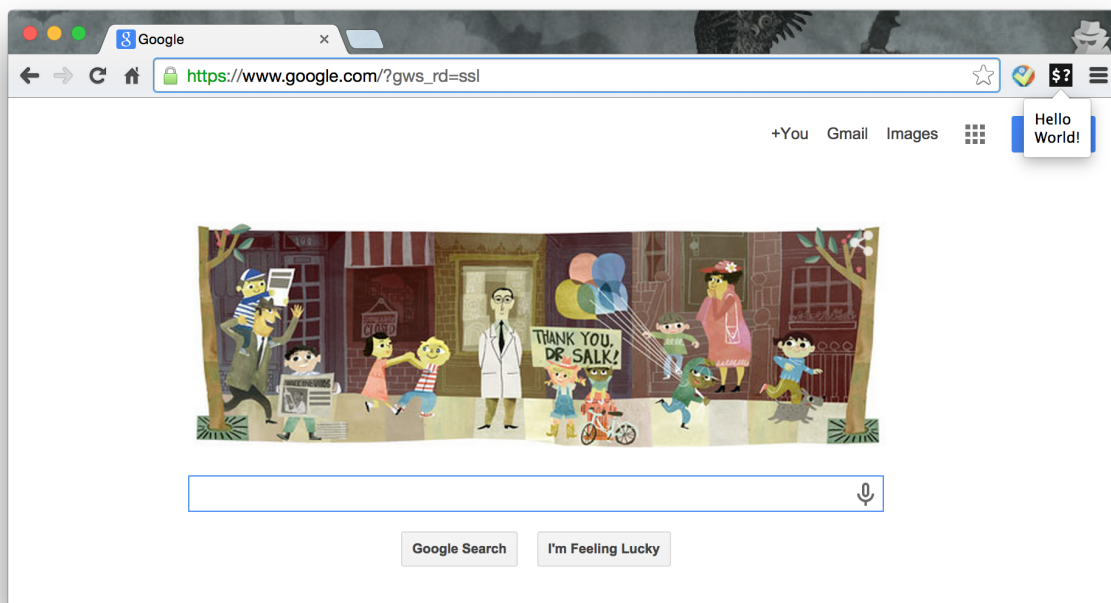


Figure 7: Our Chrome Extension in Action

Proceed to the next step to learn how to upload your extension to the Chrome Web Store for others to download.

5 Uploading the Extension to the Chrome Web Store

This step shows you how to upload your app to the Chrome Web Store for others to download and use.

1. Zip your extension directory. To do this in Windows, open Windows Explorer, right click on the directory, and choose “Send to”, then click “Compressed (zipped) folder”, as shown in *Figure 8*.

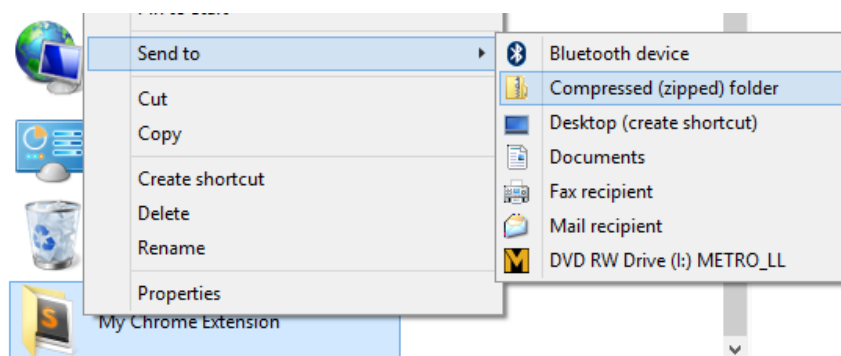


Figure 8: Zipping a directory in Windows Explorer

2. Navigate to the Chrome Web Store Developer Dashboard at <https://chrome.google.com/webstore/developer/dashboard>, and sign in to your Google Account.
3. If this is your first time publishing an application through Google, you will need to pay Google's \$5 registration fee. A yellow banner will be displayed on the dashboard if you need to pay this fee, as in *Figure 9* - press the “Pay this fee now” button and follow Google's instructions to enable your developer account.

A one-time developer registration fee of US\$5.00 is required to verify your account and publish items. [Pay this fee now](#) - [Learn more](#)

Figure 9: Google developer registration fee notice

4. Click the “Add New Item” button on the dashboard. Here, you will be prompted to upload your extension's .zip file, created in substep 1. Click “Choose file”, and select the .zip file, as shown in *Figure 10*. Press the “Upload” button to complete the transaction.
5. You will now be redirected to a page where you can configure the store appearance of your extension. There are many options available on this page, but most are optional. Make sure you set the “Category” and “Language” fields of the form to appropriate values. Click “Publish changes” to publish your app to the Chrome store. It may take up to an hour for your extension to appear to others. When you are done, your extension will have a page in the Web Store as in *Figure 11*.

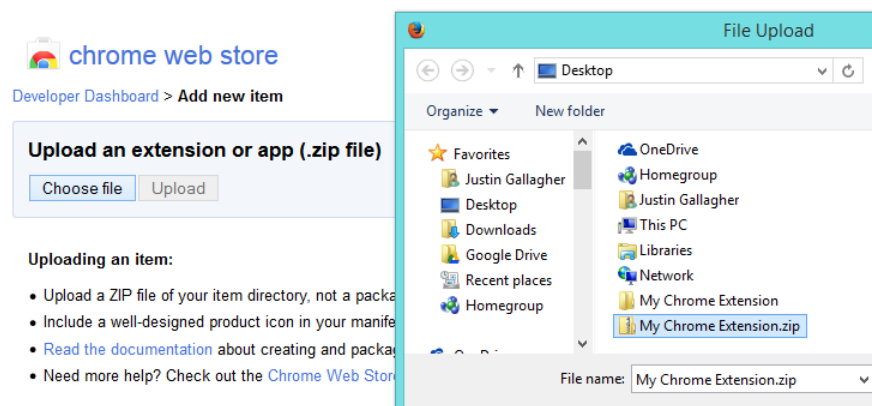


Figure 10: Uploading your extension to Google

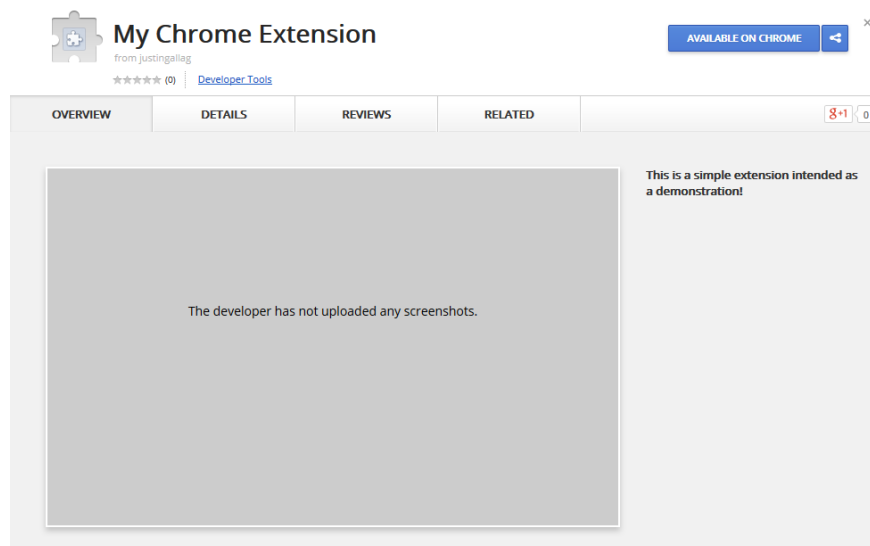


Figure 11: Store page for a completed extension

Your extension will now be available for public download, and can be found in the Chrome Web Store's search.