# UI Design/Development Environment Setup

Table of Contents

[UI Design/Development Environment Setup 1](#_Toc379896079)

[Overview 1](#_Toc379896080)

[This Document 1](#_Toc379896081)

[Basic UI development environment using HTML/CSS/Javascript 1](#_Toc379896082)

[Tools & Components 2](#_Toc379896083)

[File Directory 2](#_Toc379896084)

[ST (State or Project level specific files) 2](#_Toc379896085)

[Web Components…. 3](#_Toc379896086)

[Installation of your environment 3](#_Toc379896087)

[1. Create a work directory 3](#_Toc379896088)

[2. Setup Dreamweaver 3](#_Toc379896089)

[3. Install Xamp (If you don’t have Apache) 10](#_Toc379896090)

## Overview

### About This Document

This document describes the installation of the UI Design/Development environment. It is intended for designers/developers who will be producing rapid prototypes, proofs of concept, and enhancements to user interfaces for testing screen designs and flows in html/javascript/css and preparing code for transition to the production environment, whatever it becomes.

### Basic UI development environment using HTML/CSS/Javascript

#### Purpose

* This is a simple structure made up primarily of CSS HTML and Javascript, using a couple of server benefits to make the design process faster and easier.
* It is designed so that anyone with front end skills can work productively with UIs prior to sending them to developers.
* The purpose of this project, is to insert a basic low-level web application design environment ahead of the development process.

#### Needs and Advantages

**Need**: To iterate and test design with customer/owners independently from development overhead

**Advantage:** Vets concepts, styles and labelling and iterating in a low cost, high availability environment.

**Need:** Simplicity   
**Advantage:** Environment is low bandwidth, self-contained on a single PC and can be managed easily, without high dependence on Development for support/access

**Need:** Communicate designs and interactions to developers  
**Advantage:** Allows designers to provide developers with CSS/HTML in a Semantic/adaptive structure, styles and script libs should be “plug and play”

#### Structural Design

Although the files in the design environment will change and iterate, the basic structure should remain somewhat constant upon implementation.

#### Coding Style/Standards

W3c/CSS 3 / HTML 5 (XHTML 1.0?)/Twitter Bootstrap 3

#### Scope and Boundaries

This code can form the foundation for screens in a live system, but will take a varying level of re-locating/chopping up to map to whatever development platform we land on.

## Tools & Components

### File Directory Structure Description

#### Overview

Understanding the file structure will help you develop pages more easily, and make content more findable by others who work in the system. Every file should use absolute paths, (unless linking to child resources). That way they can be re-purposed throughout the system without having to change paths to dependencies. It is a lower maintenance way to manage linked resources. For an example, navigate to the page URL listed at the bottom of this guide and review its source.

**Each page/screen requires 5-6 includes…**

* Global Nav
* Global Header
* Secondary Nav
* 1 CSS file using cascade
* 1 Global JS file – this includes all the JS required for every page on the site(usually just below the CSS tag)
* 1 Conditional or local JS file (optional) – contains all the conditional JS for the page (usually at the page bottom)

#### JS

##### **DT\_JS** – Our own shared js code

##### **Vendor**- Global JS from vendors like JQuery and Twitter

#### Font

##### **Misc Web Fonts**

##### **Fontawesome**

#### CSS

##### **DT** – Our own CSS

##### **Vendor** – Bootstrap, JQuery etc.

##### **Global.css** – file that collects everything

#### Img

Image files (if)

#### Inc

Include files (some question about the need for this rather than say storing js includes in the js directory and html includes in the pages directory… etc.)

#### Components

##### Each component directory can have its own css/js/img/ whatever in any structure. Consistency with the resources folder is one thing, but I think this can be kept as shallow as possible with a js file, include file, and css file all in one directory, if its that simple.

### ST (State or Project level specific files)

#### Pages

Folder for pages/screens/flows

##### Page specific resources

###### **CSS –** page css

###### **JS–** page js

###### **IMG –** img files (if)

#### Resources

State Specific Resources at this level

### Web Components….

#### Apache Server (part of XAMPP)

Provides absolute paths and Server Side includes

#### Adobe Dreamweaver

Design, layout, site management, etc.

#### Windows “Host” file

Lets you name your localhost and test directory without confusing it with other server apps (localhost:8080)

## Installation of your environment

To set up a site, you’ll need Dreamweaver as your working environment, (not that it matters for the files, just the working process). Apache, Virtual Hosts, SSIs and Absolute paths, a host file for efficient working with sites, and a path for local and remote files.

### Create a work directory

Example: C:\DT2\www

### Setup Dreamweaver

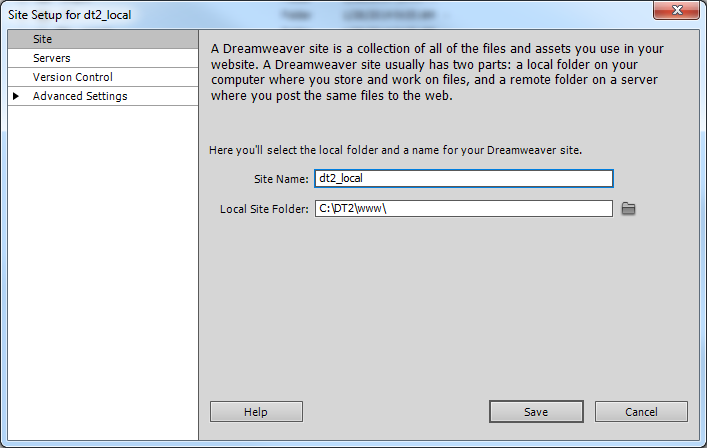
#### Import site definition file

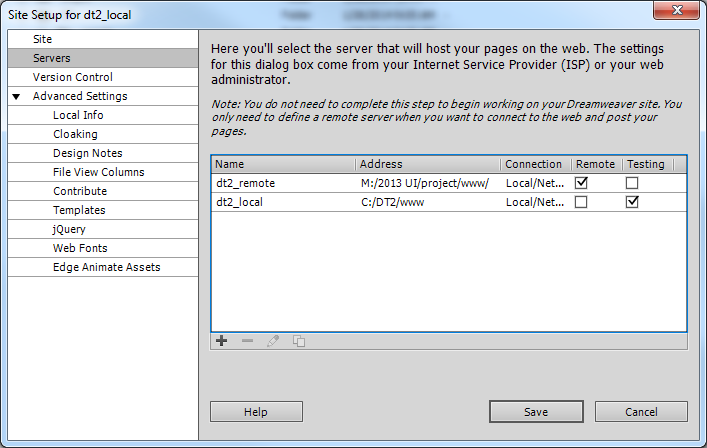
Site file is at: M:\2013 UI\project\dt2\_local.ste – Click that file, select open with “Dreamweaver” and follow the steps that the wizard takes you through

**OR**

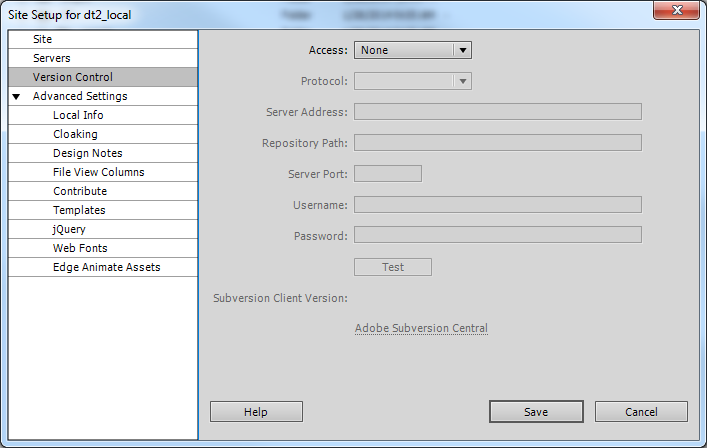
**Copy from the network : M:\2013 UI\project\www to your own computer C:\DT2\www**Then set up a new site as shown by the following screens...

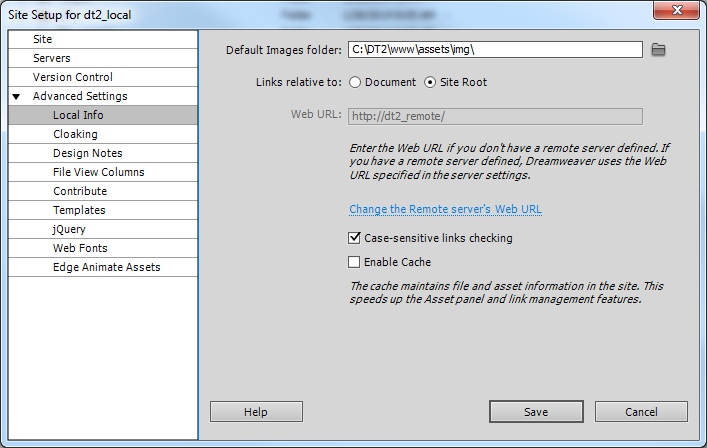
*Note, if you don’t have all of these screens in the dialogs below, that’s fine its an older version of Dreamweaver. Just fill out what it does have…*

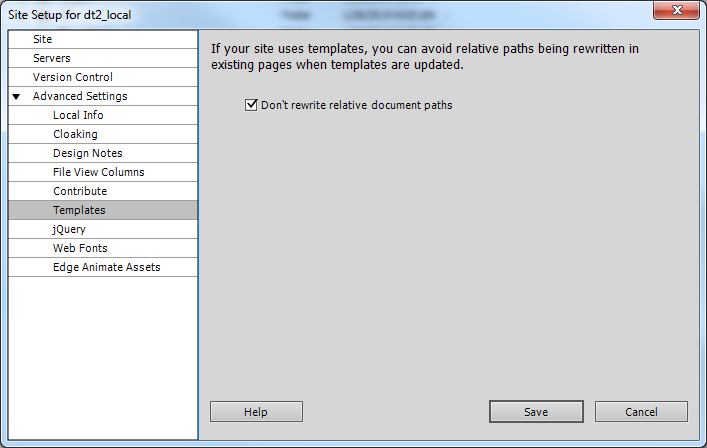


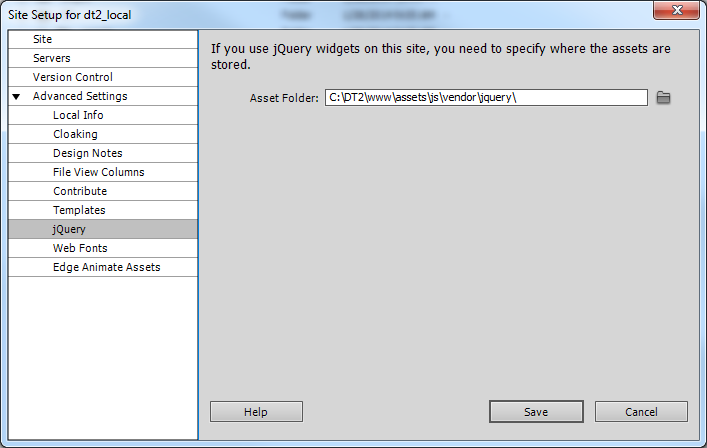


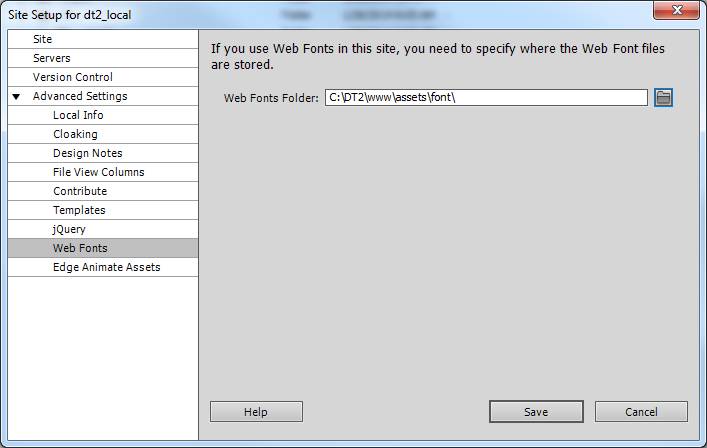
Basically you want to make sure that dreamweaver is set up not to do any code rewriting, version control, contribute, etc. Down the road we may change this but for now let’s just keep it basic.



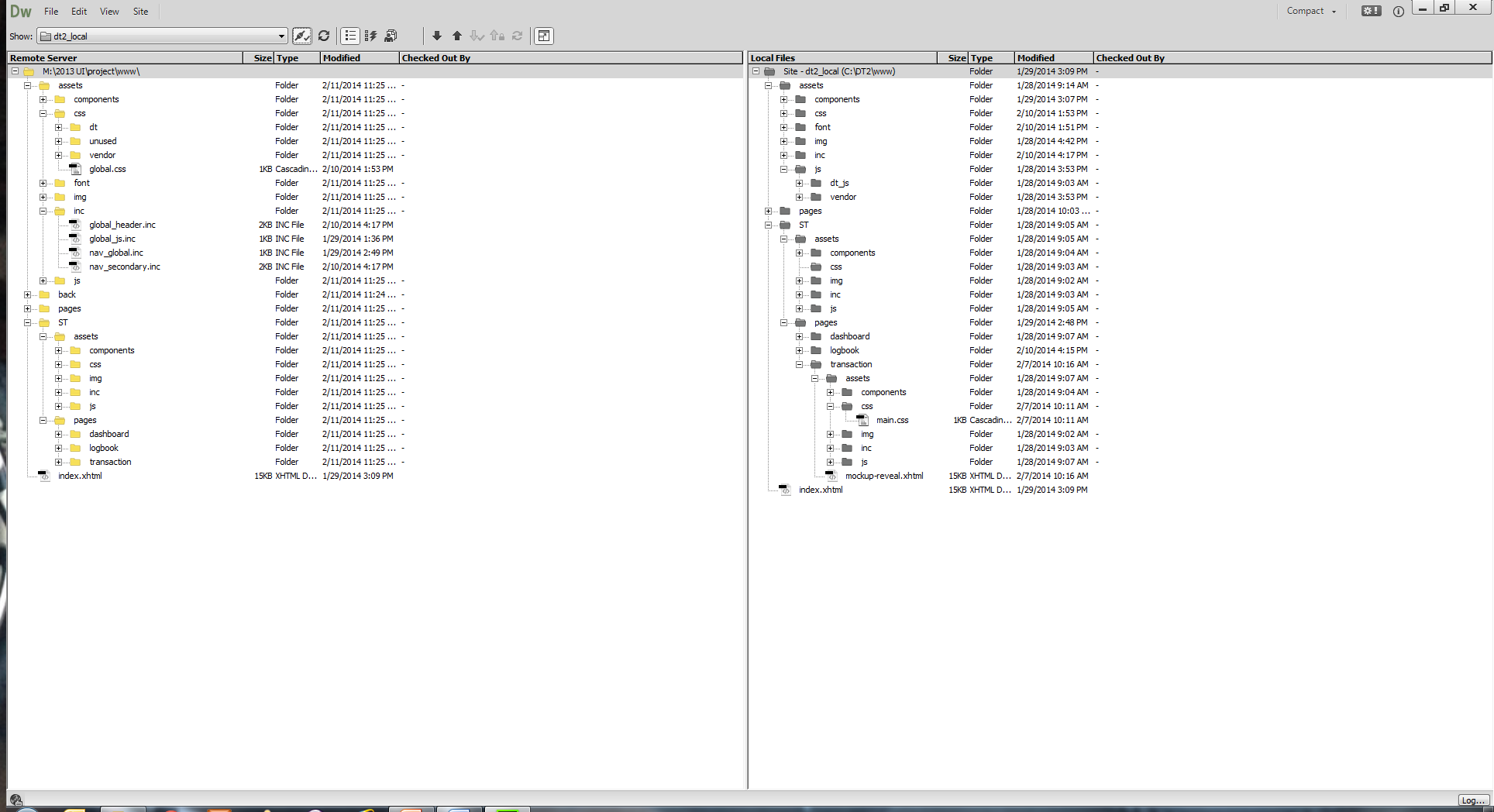








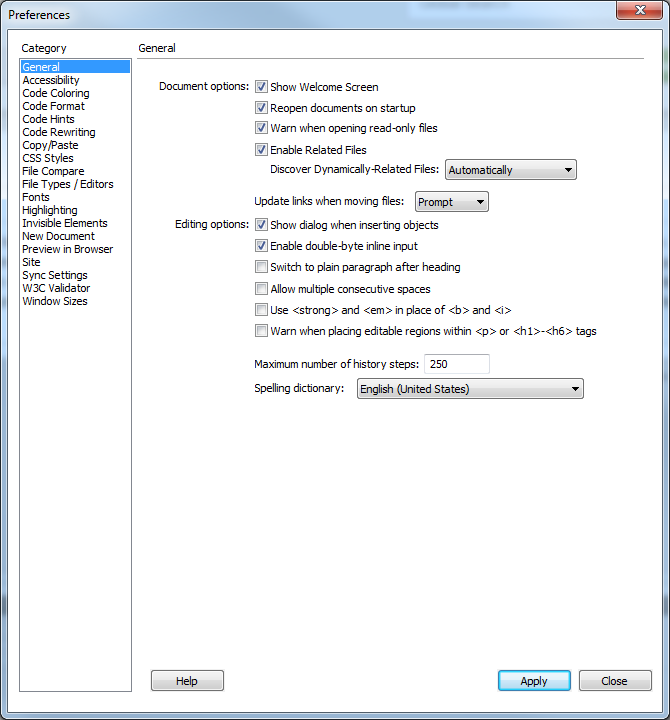
When you’re done, you should have a screen like this when you close the dialogs and click the connect button and maximize button….



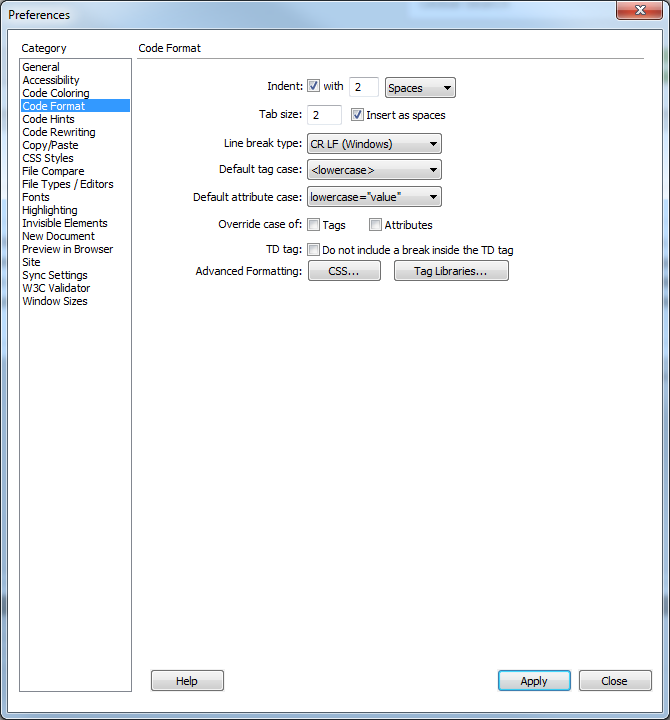
#### Set DW Preferences

Close the site files dialog and open the Preferences window, located under Edit.

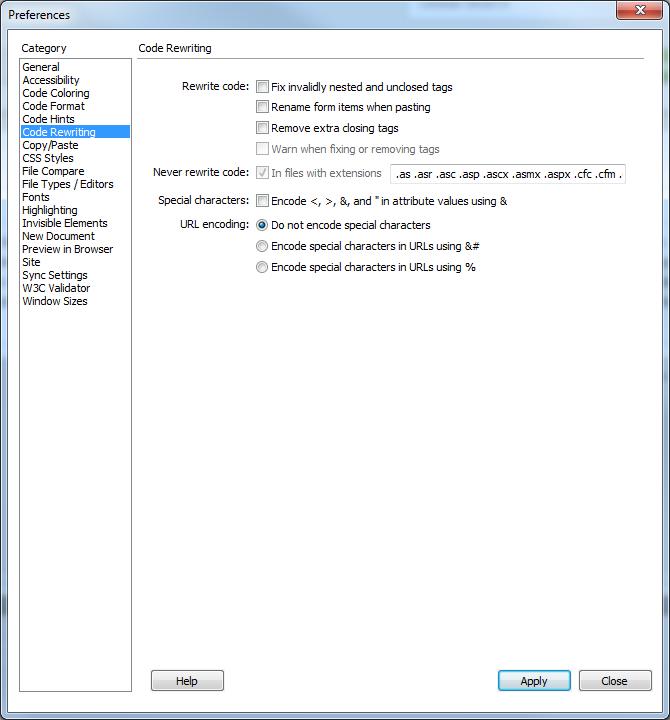
1. Bump up your undos to a healthy number…



1. Try to match developments code formatting standards. This will automatically do it when you use Commands > Format



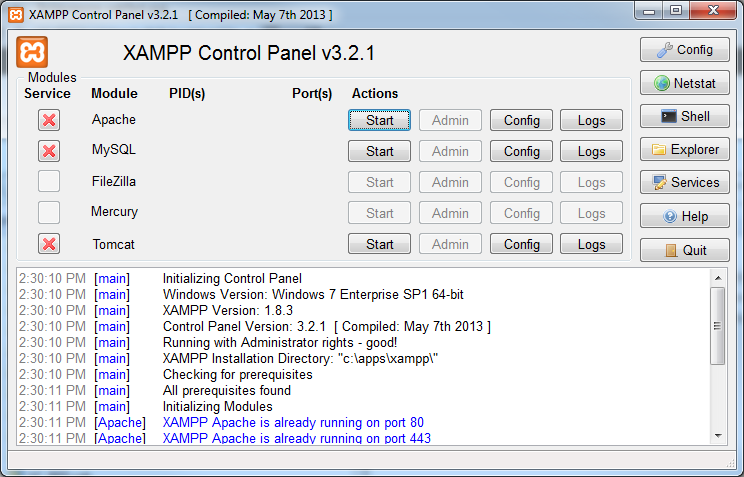
I don’t like it to rewrite my code…



You can explore other settings, just stay away from contribute, versioning, etc. It generates painful files.

### Install Xamp (If you don’t have Apache)

1. Go to M:\2013 UI\tools and copy the xamp win installer
2. Run it on your machine. Accept defaults. All you really need is Apache
3. Fire it up, you should see this…



Click start. Apache should fire. The name Apache will highlight green if its running.

#### Configure HTTPD.conf file to allow includes and find your web project

Click Config > Apache (httpd.conf)

Run “Find and look for </Directory>

Below that line, insert the following:

############# DT web server vhosts (vhost file in xampp)

##### DT 2 Local

<Directory "c:/dt2/www">

Options All

AllowOverride All

Require all granted

</Directory>

##### DT2 Remote

<Directory "M:\2013 UI\project\www">

Options All

AllowOverride All

Require all granted

</Directory>

The first one will point to your local server, so if you don’t have the same path I do, change that to your path…

Go to the line below that looks like this….

<IfModule dir\_module>

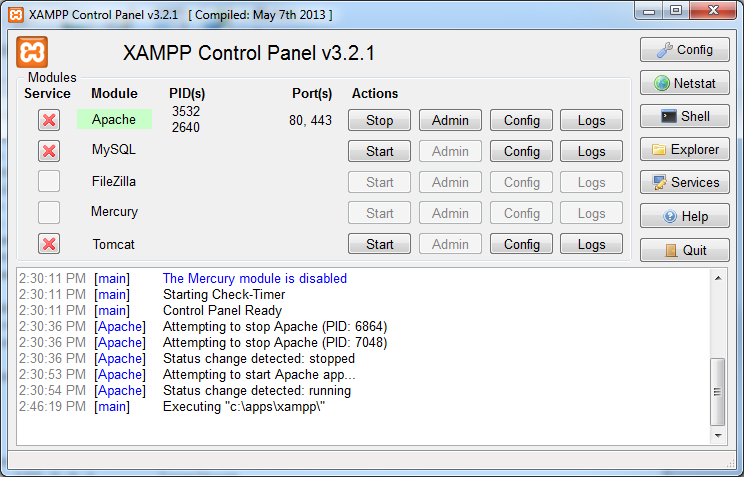
DirectoryIndex index.php index.html **index.xhtml** index.htm

Add **index.xhtml** text to the list so it can find your default index file as named by ERT

Save it and close that file.

#### Configure Virtual Server file in XAMP to point to your dev directory

1. Click Explorer from the XAMPP buttons on the right.



1. Navigate to: C:\apps\xampp\apache\conf\extra (note: C:\apps\xampp\ is where my apache is located, your machine install may differ)
2. Open **httpd-vhosts.conf**
3. Add the following lines to the bottom of the file.

<VirtualHost \*:80>

DocumentRoot "C:/DT2/www"

ServerName dt2\_local

</VirtualHost>

<VirtualHost \*:80>

DocumentRoot " M:\2013 UI\project\www"

ServerName dt2\_remote

</VirtualHost>

1. Check the paths and make sure they’re the same as your install
2. Save as text-only.

#### Configure Hosts file in windows

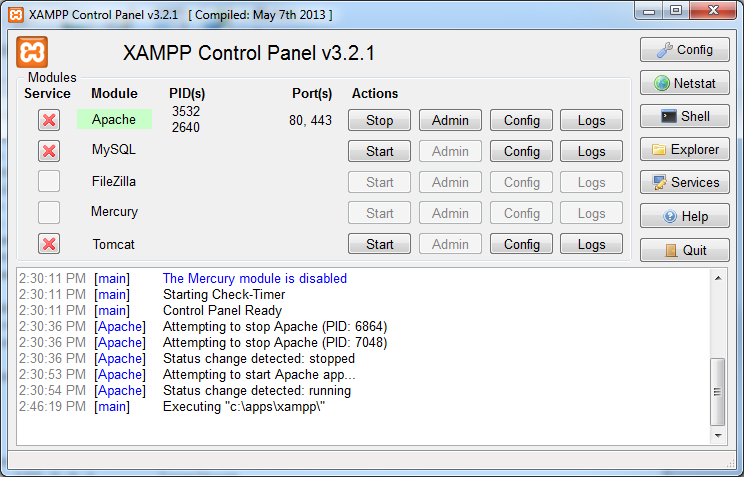
Open this file: C:\Windows\System32\drivers\etc\hosts

127.0.0.1 localhost

127.0.0.1 dt2\_local

127.0.0.1 dt2\_remote

**Stop and Start Apache….**



Paste this into your browser : http:// localhost/dt2\_local/ST/pages/logbook/index.html

Should go to this page…

