

# RETROPIE LAB 3: XML/THEMES

This lab will cover the creation of an XML theme for the RetroPie, including customization.

IT 4530 Senior  
Capstone Project  
Spring 2019 Georgia  
Southern University

## Table of Contents

## Introduction

We are going to create a simple theme for our RetroPie called 'Spare'. There are many guides out there for building themes but by creating all the XML from scratch, we can get a better understanding of it.

See Figures 1,2,&3 for multiple views of the theme we plan to create.

### **Figure 1**

The System View is what you see when Emulationstation starts. It has a large white bar along the middle that houses a carousel that shows 3 System logos at a time.



**Figure 2**



This is a simple game list page. You see this if you haven't scraped any metadata for that system.

Figure 3

GAME BOY

Avenging Spirit

Balloon Kid

Battletoads

Battletoads & Double Dragon

Battletoads in Ragnarok's World

Bionic Commando

Castlevania II: Belmont's Revenge

Castlevania Legends

Castlevania: The Adventure

Contra: The Alien Wars

Darkwing Duck

Nintendo

GAME BOY

CAPCOM

BIONIC

COMMANDO



Nintendo

RATING:  
★★★★☆

PLAYERS:  
1

GENRE:  
Action

DEVELOPER:  
Minakuchi Engineering

PUBLISHER:  
Capcom

RELEASED:  
10/01/1992

TIMES PLAYED:  
0

LAST PLAYED:  
never

SWING INTO ACTION AGAINST THE FORCES OF EVIL!

A ruthless army stands ready for war and the one man who can stop it has disappeared. Now, in a desperate fight for survival, the world turns to an elite group of men and women to put an end to the threat. More than simple soldiers, they are the Bionic Commandos. Drop behind the lines with

OPTIONS

MENU

BACK

LAUNCH

SYSTEM

CHOOSE

The Detailed View is what you see if you *have* scraped metadata. Different themes can show different data, in different places or a different order, but every theme can only display these values.

## Prerequisites

What You Will Need:

- A text editor
  - Notepad will do fine, but something with syntax highlighting is helpful. Sublime Text 3 works, but others like Notepad++ are good too.
- Emulationstation
- An image editor
  - This is optional as all the images will be supplied, however you can use these images as a base to create your own, or make your own from scratch. Photoshop works fine but there are other programs like GIMP or Paint.net would work too.
- Any coding experience

## Initial Setup

### Set Up EmulationStation on PC

- Download the portable Emulationstation
  - [Watch the video first](#)
  - [Download the file](#)
- Install/unzip the file where you want it.
- Download [this](#) zip file. Within it are 3 folders: 'gb', 'nes' and 'snes'. Within each folder are 15 blank .txt files, each named after a game on that system. These files have also had their extensions changed to something that ES will see as a rom.
  - These are not real roms. They are blank text files that have been renamed to look like roms.
  - Emulationstation sees these as real games. You won't be able to play them (since they are really just blank text files), but they will scrape with the built in Emulationstation scraper. They are also extremely small, so you don't have to worry about your test-bed getting too big.
- Copy those folders into F:\emulationstation\.emulationstation\roms

Emulationstation should now be set up with 45 (fake) games on 3 Systems, and ready for you to use.

Double click Launch Portable (Windowed).bat to start Emulationstation.

## Creating the Theme

We're going to make the theme straight in the Portable Emulationstation's theme folder. This makes things a lot easier, because you can view a change straight away without having to move files each time.

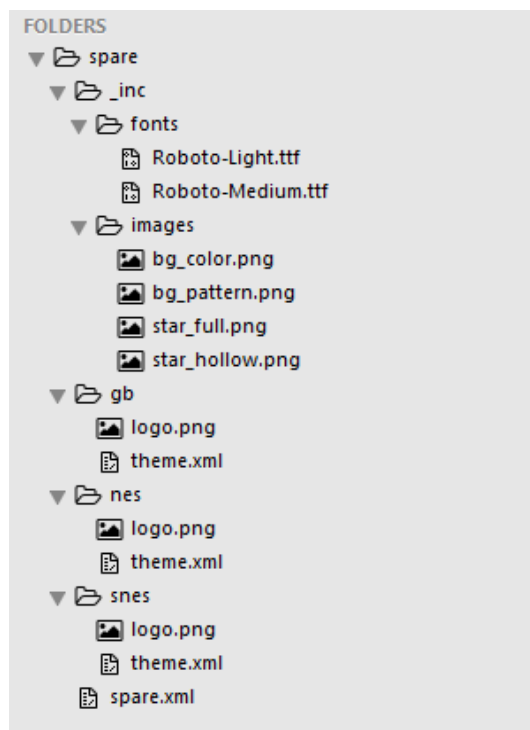
We have installed the portable Emulationstation on F:\ Drive, so the path to the themes folder is F:\emulationstation\emulationstation\themes.

First, [download the full Spare theme from GitHub](#), so you have the images and fonts you need, and also so you have something to reference.

We'll start by setting up a simple file structure, with folders for the 3 console systems; Gameboy (gb), Nintendo Entertainment System (nes) and Super Nintendo Entertainment System (snes).

- Within the themes folder make a folder called 'spare'.
  - Within the 'spare' folder make a new file called spare.xml.
- Within the 'spare' folder make folders called 'nes', 'snes', 'gb' and '\_inc'.
  - Within each of 'nes', 'snes', 'gb' make a new file called theme.xml.
  - Also copy the system images from the downloaded file into the system folders
- '\_inc' is your includes folder where you will hold your fonts and theme images. You can call it anything you like. Some people use "art", others use "common". I like "\_inc", and the underscore keeps it at the top of the file tree.
  - Within '\_inc' create 2 folders, called 'fonts' and 'images'. Copy the fonts and images from the downloaded file into their respective folders.

Your folder structure should now look like this:





Now onto the code.

Open spare.xml and add these lines:

```
<!--  
theme name:      Spare  
version:         1.0  
author:          Matt Kennedy  
email:  
website:  
license:         creative commons CC-BY-NC-SA  
based on:        "Carbon" by Eric Hettervik  
-->  
  
<theme>  
  <formatVersion>4</formatVersion>  
  <view name="system"></view>  
  <view name="basic"></view>  
  <view name="detailed"></view>  
  <view name="video"></view>  
</theme>
```

When you are making your own theme you would obviously change the 'theme name' and 'author' fields to suit.

- The theme details at the top just tell anyone looking who made the theme.
- Anything within `<!-- x -->` is a comment, visible to humans who read the code, but invisible to Emulationstation.
- The `<theme></theme>` fields tell Emulationstation that the code within is for a theme.
- The `<formatVersion>4</formatVersion>` sets the theme version.
- `<view name="x"></view>` governs what happens in each of those views.

Save spare.xml

Open spare/gb/theme.xml and add these lines:

```
<theme>
```

```
<formatVersion>4</formatVersion>

<include>../spare.xml</include>

<view name="system"></view>

<view name="basic"></view>

<view name="detailed"></view>

<view name="video"></view>

</theme>
```

Notice it's almost identical to spare.xml, except for a few things. You don't need to specify any theme details, because we are using this line:

```
<include>../spare.xml</include>
```

That line means, find a file up one level from here, called spare.xml, and include it in this file.

This is handy because it means you can have the bulk of your code in spare.xml and just call the file in each theme.xml instead of having to put the code in each individual theme.xml file.

Save theme.xml

Once you're done, locate the IP address of the pi under RetroPie and 'Show IP'. Now, make sure WinSCP is installed. For hostname, put in the ip address of the pi. The username is 'pi' and the password is 'raspberrypi' by default. The theme will go into the .emulationstation directory located in RetroPie/.emulationstation. You will need to create a new folder called 'themes'. This will make the new theme appear in the theme selection in emulation station.

**This completes the basic view set up of the xml themes.**