# Hot Beats Setup Instructions:

1. Requires TH2 builds or later.
2. Set PC into developer mode
3. Install VS Code:
4. install node.js:
5. install git:
6. install “project my screen”
7. open powershell (or cmd prompt): npm install manifoldjs -g
8. create a new folder for generating app
9. in powershell, generate apps “manifoldjs http://btdj.azurewebsites.net “
10. type “cd HotBeats” then “Manifoldjs run windows10”. This should install and open the Windows10 app.
11. If you need to install the apps on the Windows Phone and Android phone, do this with the source packages in this folder. The APK can be downloaded on the Android phone from the GitHub repo: <https://github.com/nmetulev/buildtour2016/tree/HWA-keynote>

# Hot Beats Demo setup

1. Open HotBeats in the browser at <http://btdj.azurewebsites.net>
2. Open source code of the web site in visual studio code (open folder desktop\hotbeats\hotbeats\dj), open up manifest.json for illustration
3. Open powershell (or cmd prompt) and navigate to the folder which houses the “hotBeats” folder (desktop\hotbetas)
4. Open Windows Phone and open “hot beats” app until it shows the sleigh Bells. Force Close App. This is important because the app will show the start screen every other time it’s open.

# Hot Beats Reset App

1. Navigate to the folder which houses the “hotBeats” folder (desktop\hotbetas)
2. In the new “HotBeats” folder delete the folders for web, Chrome and FireFox.
3. Open and close Windows Phone device until it shows jingle bells, the force close it (this guarantees that it will show the start screen on the next open)

# Hot Beats Demo Script

|  |  |  |
| --- | --- | --- |
| Step | Script | Screen Shot |
| 1. show in browser | > Many interesting Web Apps out there.> Hot Beats is a music performance app that lets you create tune old school style. - show the website in the browser  - choose first four instruments then hit go.  - play each of the instruments to show different sounds |  |
| 2. Build the App with VS | > let’s move this into a store app so I can take advantage of some new APIs, and release to the store > it’s quite easy to do with Visual studio  - open visual studio  - file new project, new blank app  -open manifest, change start URL to <http://btdj.azurewebsites.net>  - go to “content URI” tab and add same URL to list, and set API access to “all” |  |
| 3. demo app | > now we just hit f5 and run the app locally - open app (in the start menu), show smtc (system media transport controls) media controls with integration (press volume up/down buttons on Surface, and switch track from shell media controls) |  |
| 4. show SMTC code | - back to vs code, open smtc.js, scroll to top > I have a copy of the server code here locally, all the APIs for the media controls are exposed in JavaScript, and I have this extra file on my website to tie the functionality into my app.> at the top, you can see I’m doing a familiar “feature detection” for the appropriate Windows Objects. If they appear in the DOM, I know I’m in a windows 10 app, and execute the code, for any other platform, it never executes |  |
| 5. demo phone | > now some of these instruments are better played on different devices. I want to play these jingle bells on my windows phone> I’ve also included Project Rome in this feature that lets me keep all my apps in sync - go back to project my phone to show app open on windows phone with jingle bells > right where we left off. And I can now use the motion sensors of the windows phone to play it right (shake phone) |  |
|  |  |  |
| 6. x-plat | > now lets go to the cow bell, which can get pretty rough. I don’t want to break my windows phone, so I can actually run this web app on the anroid phone as well> I’m going to use an open source command line tool called ManifoldJS to generate these apps for each platform - in powershell, type “manifoldjs <http://btdj.azurewebsites.net> -l debug” > let the tool run |  |
| 7. show manifest | - in vs code, show manifest.json file> what’s happening is that Manfioldjs is going out to the site and downloading the W3C manifest, and from that it’s generating platform apps from it - cd into Hot beats and either LS to show platforms or show in explorer |  |
| 8. app on android | > I’ve already loaded the APK on my android phone, so now let’s jam - open app on android phone  - pull out the mallet and play the android phone like cowbell. “it needs more cowbell!!’ (be careful, you can break screen) |  |