

Prototype Readme

Team Alpha Q


What is it

This is a project for CSC318 course (The Design of Interactive Computational Media), a web app mockup interface designed for Toronto public transportation. The app provides information and fun for the commuters, and provide opportunities to text someone else nearby also taking public transit.

How to view

This prototype is written in HTML5, CSS, JS. So obviously you need a browser. It is strongly recommended to use **Chrome browser** because the mockup was only thoroughly tested in this browser. At current stage, only mobile devices are well supported. We apologize for the inconvenience. So,

If use Chrome on computer:

1. Double click on *demo-phone.html* to open.
2. Press F12 on your keyboard to enable **Dev Mode**.
3.  On the Dev Mode portion move your mouse to the top left and click the second tiny button to activate **Device Mode**.
4. You should see the page has been changed into a ratio same as the screen of a phone. If not, choose a preset phone from the dropdown spinner at top.
5. Optional. Adjust the zoom level right next to the dropdown.
6. Enjoy.

If you do not have Chrome and want to view in other browsers on computer:

1. Get prepared of potential incapability problems. If met one just calm down.
2. Double click on *demo-phone.html* to open.
3. Resize the browser window so it looks like the screen of a phone.
4. Adjust zoom of your browser (Keyboard short cut: hold '**Ctrl**' and tap '-') to around 50% depending on your screen resolution, until everything look right.
5. You will not be able to play with advanced mode but smile, and enjoy.
6. When done reset your browser zoom to normal (Keyboard short cut: hold '**Ctrl**' and tap '+').

If you are so kind to use a phone:

1. Download and extract the files and the *src* folder under a same directory.
2. Open *demo-phone.html* with a browser.
3. Enjoy.

Advanced

Although there is no actual server connected to the app, some functions initially implemented for fun can be called in Chrome console in Dev Mode. The following are a few operational function calls that do interesting things.

othersSend (which:int, what:str, who:str);

Sends a message with content 'what' in the role of 'who' to chat 'which'. In this prototype there are 8 (0~7) chats available for send. Chat 0 is the general chat and the others are private.

e.g. `othersSend(6, "good to see you too", "Administrator")`

Delay (where:str, length:int, why:str, whereStr:str, lengthStr:str);

Creates a delay event at 'where' for 'length' minutes for the reason of 'why' that might be notified to the user. 'where' should match a station name found on status page exactly. If you want to define your own notification string, use 'whereStr' and 'lengthStr'.

e.g. `delay("Spadina", 5, "No reason", "Spadina South Platform", "~5 min")`

Notify (what:str, blinkTimes:int, length:int);

Notifies users of 'what' and blinks the status bar for 'blinkTimes' many times. The message stays for 'length' milliseconds.

i.e. `notify("Just for fun", 2, 3000)`