**Implementation Developer - Task**

**Instructions and Background**

The purpose of the task you see before you is to allow you to experience a bit of the tasks you can expect to be dealing

with on a daily basis as part of Taboola’s Implementation team.

For any question, please contact Narkis Feldman, 0584033705.

**Section 1: A day in the office**

Below is a description of typical day in the office. Describe in writing how you would manage that day’s tasks.

1. You arrive at the office at 9:30. You have to leave at 18:30 exactly, and can’t stay any longer.

**there are 13.5 hours work to squeeze into 9 hours(including lunch)**

2. On the team board you see all the team’s tasks (tickets).

3. You are the only developer that is available to do the board’s tickets because the other developers are occupied

with a strategic customer launch.

4. You have 2 open tickets left over from yesterday that have been assigned to you:

∙ 1 ticket is a week old and requires help from the team’s tech lead – you should plan at least 30 minutes for the meeting with her and another 1.5 hours to solve the issue once you do. **2H**

**This doesn’t seem in high priority and can wait till tomorrow (plus as I am the only developer today it’s seems that the tech lead isn’t available today)**

∙ 1 ticket is 2 days old and due for tomorrow. This ticket requires clarification from the project manager in NY –

you should plan 30 minutes for the call with him and another hour to resolve the issue. **1.5H**

**I would write down what I understand from the ticket and how I plan to solve it.**

**- Either on the ticketing system or email – and send it to the project manager for approval/clarification. Also I would schedule a call with him to 3-4pm (the earliest he can do it from what I already know about him) to clarify.**

5. The board has additional 7 tickets (development requests) that were supposed to be completed yesterday and the Project managers are under pressure to deliver them to their customers:

∙ 4 standard tickets – takes 30 minutes to resolve each ticket.

**2H.   
 handle those tickets**

∙ 2 tickets with missing information.

**Where should the missing info come from ? if this pending someone to answer the ticket, I**

**would send a reminder and leave it**

∙ 1 ticket for a strategic client – takes 2 hours to resolve. **2H**

**Doesn’t seems that important**

6. Additionally you need to

∙ Hand in a report about last week launches – will take 3 hours. **3H**

∙ Have a prioritization meeting with your team leader – will take 15 minutes.

∙ Call the NY team and get information about a new client that will soon have pending tickets on the board –

will take a few minutes.

**0.5H (for the last 2 tasks)**

7. There’s also a Tech talk from one of the developers today at 15:00, and you really want to attend. That talk will be

1.5 hours long, and though it won’t help you solve any of the current issues on your plate, it will help you get a

better understanding of the system.

**I would attend the beginning and see if it has an added value to me OR I would ask to record it so I can watch it later that week , in case I need to step out…**

8. Plan your day. Remember to eat ☺

**Could be good for my diet to not eat today ;-)**

**Day**

**09:30 – 09:45 – prioritize the task by myself (maybe while asking for more info on the missing tickets)**

**09:45 – 10:00 – prioritize with the team leader**

**10:00 – 12:00 – the standard 4 tickets**

**12:00 – 12:30 – Eating (start the report)**

**12:30 – 14:30 – handling the strategic ticket**

**14:30 – 16:00 – work on the report**

**16:00 – 16:30 – talk to NY**

**16:30 – 18:00 – handle the ticket**

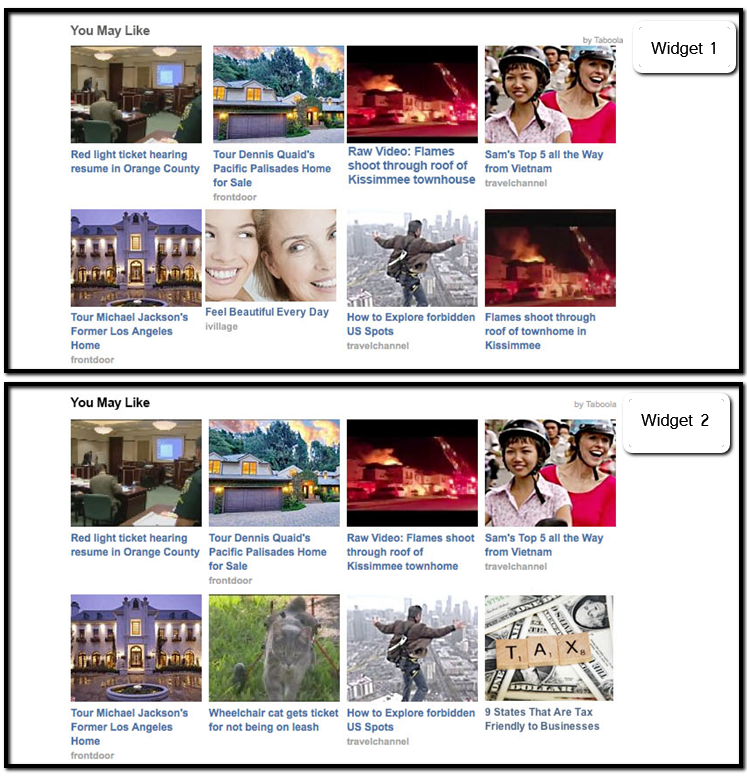
**18:00 – 18:30 – continue the report**

**Left out:**

* **Tech talk**
* **1 week old ticket**
* **Talk to NY team on the new client**

## Section 2: UI test

See 2 widgets below and describe the UI differences between both widgets:



**Section 2: Answers**

**Padding/width -** widget1 doesn’t have the same padding/width as widget2 - some images in widget1 are using padding right/left(or different width), but some doesn’t. widget2 uses the same padding/width for all images(articles)

**Images -** widget1’s images are ‘zoomed in’ whereas widget2’s images looks better.

**Fonts -** widget1 uses **bold** text for some articles and the text isn’t uniformed. (Again the text padding in widget1 isn’t well-formed - we can see different behaviors from one article to another) .  widget2 uses the same Font for every article + the padding of the text below image are uniform

**“By Taboola” -** widget1 isn’t placing the title well (should be as widget2) - not padding + the title is going out of the widget frame

**“You May Like” -** is **Bold** in widget2 + the color is different (black against gray)

In a general way, it looks as thaw that the articles in widget1 are ‘hard-coded’ (each item). On the other hand widget2 is well formed – using the same attributes in every item.

**Section 3: Widget Creation Tasks**

1. Create a widget in HTML, design it in CSS

a. In the Widget definitions directory you will find three files illustrating how the widget looks in the different states (regular, mouse over the large item, mouse over a small item). Please ignore the fact that the image and title of the small item being hovered changes; this should not be implemented. You onlyneed to implement the border, the text color change and the overlay in the Mouse over small item state.

b. The widget should look exactly as pictured in the mockups.

c. In the Resources directory you will find all the needed resources for this widget. Please do not use any external resources.

d. Please do not use external libraries (jQuery, Bootstrap, etc…).

e. The widget must look and behave well cross-browsers. Cross-browsers means at least the following:

latest version of Chrome, latest version of Firefox, IE8, IE9.

2. Review the widget for possible design mistakes and suggest (in writing, not on the widget itself) how to fix them

**Answer:**

**Hover on small –** when hovering on the small item we are adding a bottom&top border to the image.   
In the widget the image isn’t resized accordingly to the borders, which leads us to use min-height attribute for the small item if we want them to stay in the right place. I would suggest resizing the image when adding those borders.

**Main Item –** in the widget the main item width should be the same as the two small items (combained) below him. this can be fixed by setting the same margin (from the frame) to every image.

**Small Item description –** the description text (small) image isn’t using overflow , meaning – if a large text added to the html the item widget won’t behave as expected.

I would suggest to determine the text lines allowed and to use overflow accordingly.

3. Assuming the above widget was created for the home page of MarketWatch (<http://www.marketwatch.com/>), please point out any discrepancies between the design of the site and the design of the widget.

a. The widget doesn’t actually appear on the above url, so don’t go looking for it ☺. b. This too should be done in writing and not on the widget itself.

**Answer:**

* I think that the main issue is the fact that the widget is built **vertically** and in the marketWatch site all widgets are **horizontally**.
* Looking at the Most Popular widget in MarketWatch , we can see that all of the items are obviously the same (same width, height …). Our widget has the Main item above all small ones. In order to match the design of MarketWatch I would use only the small items.

**Technical guidelines that might help:**

1. Reading about CSS in [http://www.w3schools.com/css/ is](http://www.w3schools.com/css/) advisable

2. Use a separate CSS file for your styling issues - It will help you organize your code and your thoughts.

3. Use CSS selectors (IDs for unique elements and classes for elements that should have similar styling).

4. Except for the very basic properties, you should probably use the following ones: float, display, position, overflow, background-repeat.

5. For issues that you don't know how to solve - search the web. Google and stackoverflow are your friends!