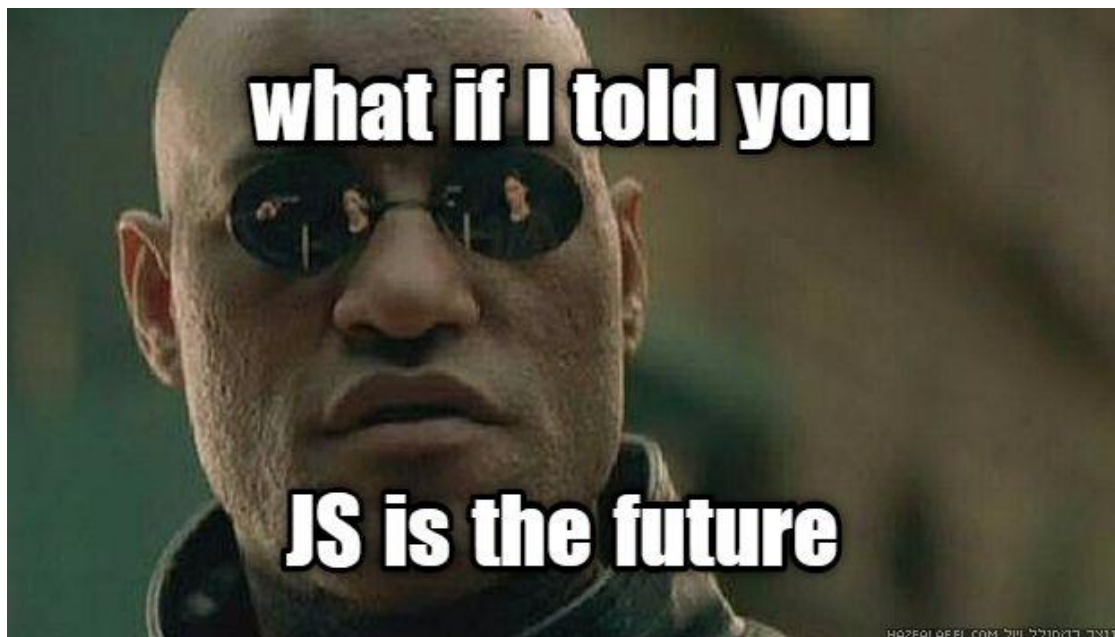


Ultimate Meme Creator

Sprint 2 Challenge

Your task is to create a Meme Generator that looks good and works well on both desktop and mobile.



Delivery Instructions

There are 3 folders to submit your code (Thursday, Saturday and Sunday evenings).

Manage your code on Git and Publish your app on Github Pages.

With your final delivery, fill in the app details in this [Submission-Form](#).

Meme Generator - App References

Play with and get to know some of the following apps on both desktop and mobile. Please learn what it is about, what to do and what not to do.

(On your app – take the freedom to create a better UI)

Main reference:

<http://g.hazfalafel.com/index.php> (line draggable, side-line editing)

Other references:

2. Search/Filter: The user should be able to search/filter the images by typing in a SEARCH-BOX letters. Relevant images shall be presented/filtered while typing.

Focusing and typing in the SEARCH-BOX shall also cause presentation of an initial list of keywords.

3. The user may also filter the list by selecting a keyword from a list where each word size is determined by the popularity of the keyword search (make an initial "random" setup so it will look good from the start):



Functionality – Meme Editor

1. It is preferred that a single set of controls ("control-box" of text line and buttons) will handle the different lines.

TIP: you will need to keep the x,y of the lines

Relating the control-box to the selected line might be done by either:

- a. "next line" button switching between the lines OR by
- b. Clicking the correct line on the canvas

Consider marking the selected line while editing

Notwithstanding with the recommendation to use a single control-box, in case time doesn't allow, you may use several control boxes as done in some of the reference examples

2. As a default, Use the common font meme "**impact**", white with black **stroke**.
3. Optional set of controls: font family, font color, font size, L-R alignment, Up-Down alignment arrows, delete line.
4. Download Button/Link of the created Meme image
5. First two lines shall be at the TOP and BOTTOM of canvas, further lines at the center
6. Line dragging is not a must and shall be implemented only in case you implemented all above functions first.

Design and Responsivity

Both the Image-Gallery and the Meme-Editor shall look good on both Desktop and Mobile.

QA mobile UI on your mobile devices using GITHUB pages

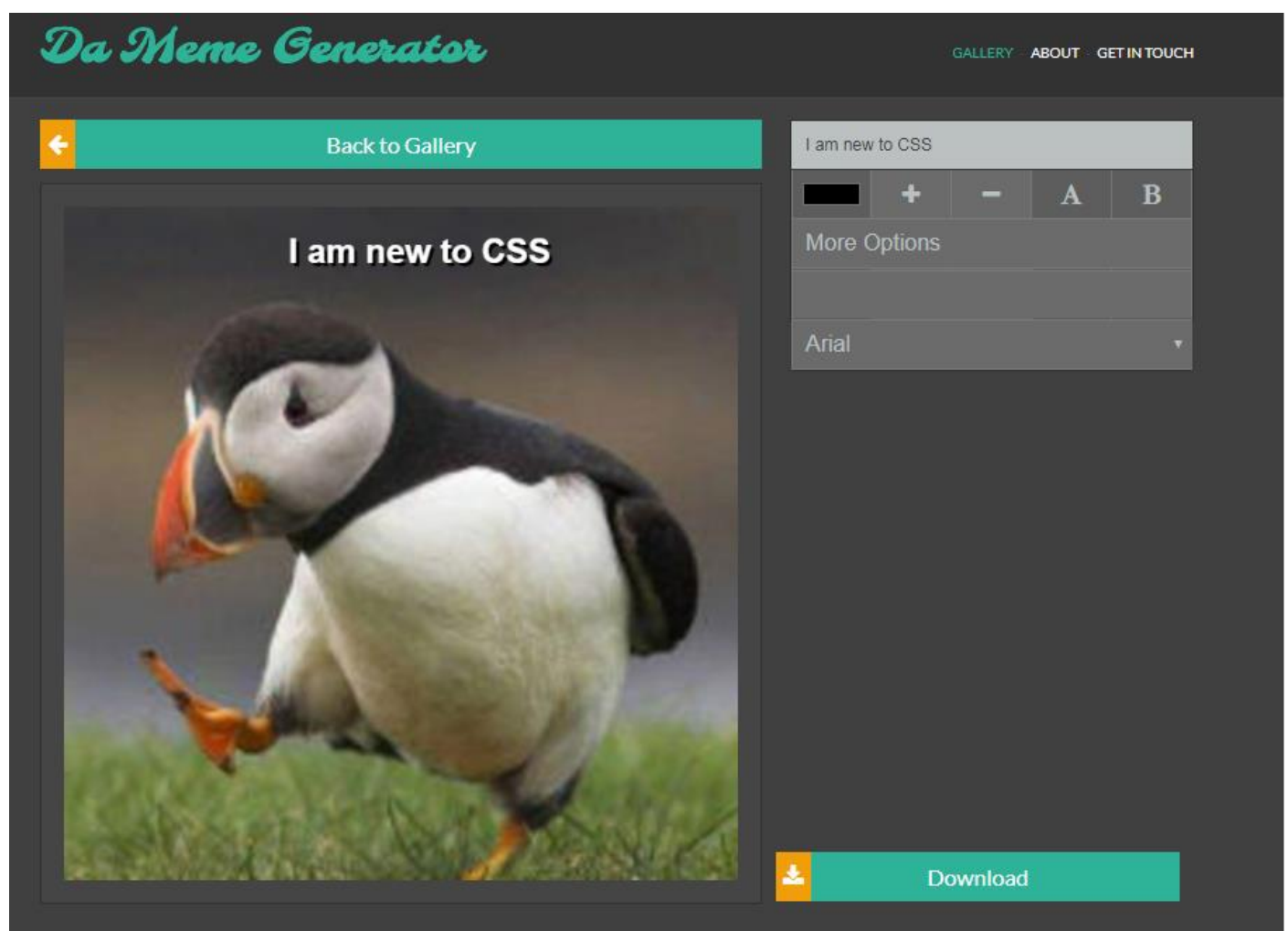
Keep original proportion of images at both Canvas and Gallery

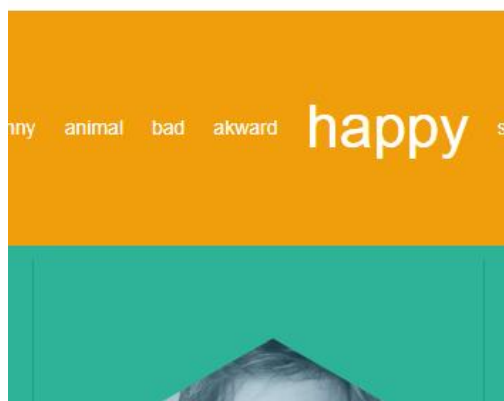
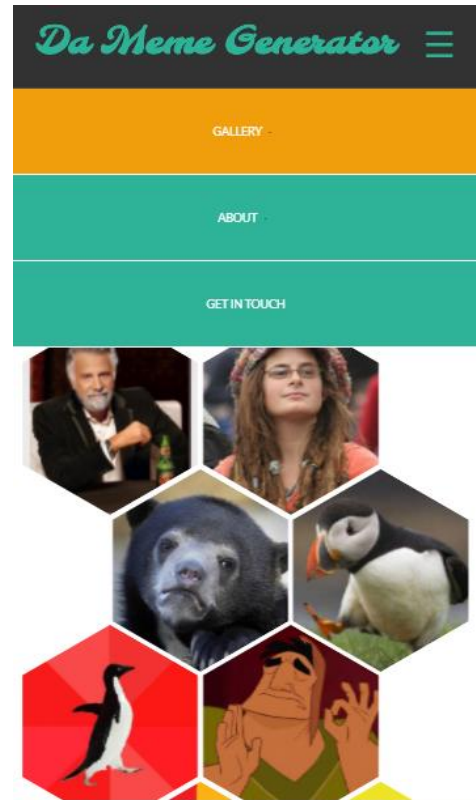
After main flow functionality is implemented and works fine, pay attention also to Canvas responsivity and sizing. We recommend using image aspect-ratio to calculate canvas height from its width.

The Home page given in the PSD and presented in Appendix2 is a concept to be inspired by. The hexagons are not a must. Nevertheless, the design presented in the PSD is challenging and can improve your CSS skills if you have time to implement it fully.

The Meme-Editor design presented below is only a possible concept. In addition, there are functional inaccuracies in it (for example, the text line control-box has different buttons than defined).

You are invited to make your own design.





Model and Code - Recommendations

Images: img objects in your model should have the properties:

```
[{id: 1, url: 'img/popo.jpg', keywords: ['happy']}
```

Keywords examples: happy, crazy, sarcastic, sad, animal...

Model: A proposed initial data structure (managed by a meme-service):

```
var gImgs = [{id: 1, url: 'img/popo.jpg', keywords: ['happy']}];
var gMeme = {
  selectedImgId: 5,
  txts: [
    {
      line: 'I never eat Falafel',
      size: 20,
      align: 'left',
      color: 'red'
    }
  ]
}
```

Search/Filer: use the [<input> list attribute](#) to show some pre-made options

Team Members Responsibility

In a separate excel document, you may find a proposed list features allocation to two team members, please consider using it to manage the tasks between you and define features implementation priority.

Recommended Development Workflow

1. Start together, build a simple project structure
2. Setup git and make sure the team can push and pull from the repository.
3. Design an initial home page (index.html, main.js, CSS files) together (don't spend more than 2 hours at this stage)
Tip: add a canvas-tries.html where you can play with a canvas
4. Commit and Push
5. Split and work separately and together whenever you see fit.
 - a. Remember to push and pull every 2-3 hours to coordinate your code and practice the workflow
 - b. Use this opportunity to improve the code by mutual code reviews
6. Remember to reserve some time for QA and UI improvements. Try to complete all functionality at least a few hours before the final delivery time and then concentrate on UI finalizations.

Bonuses and Advanced features

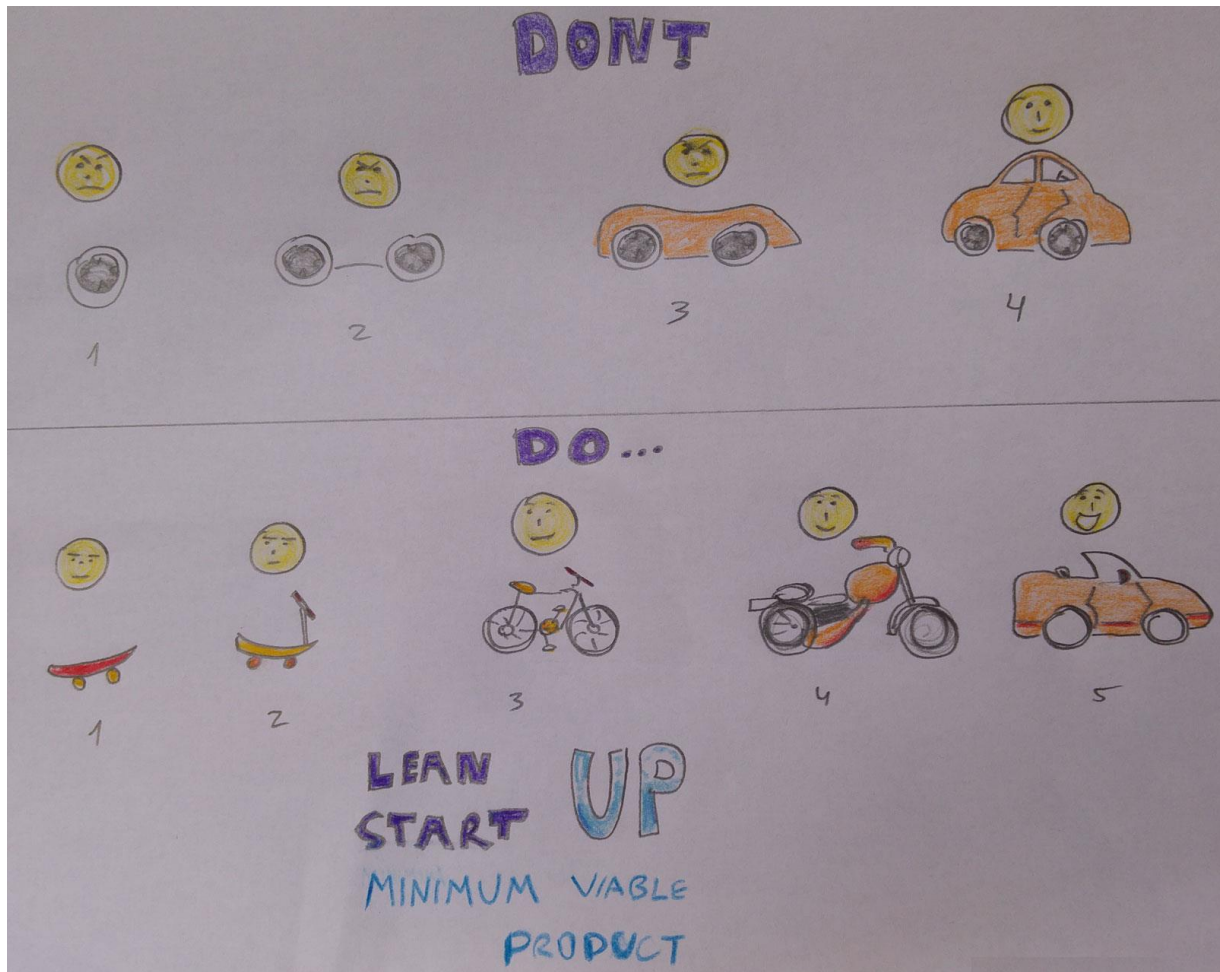
Upload img / from url

Share on Facebook (use the sample code provided)

i18n for Hebrew

Website theme: celeb-meme, politic-meme, ani-meme, kid-meme, mondeal-meme

Appendix 1 – MVP – Minimum Viable Product



Appendix 2: Gallery page – design screenshot

Please note that the designer has not completed the entire design. He is also not available, so you will need to take some decisions, please make good ones.

