

ZOOMOOD

A zoomable mood board.

Ulrich Lehner, KTH Royal Institute of Technology, March 2013

Nowadays online mood boards are commonly available for everybody, such as Pinterest. However these services don't allow the behavior of their analog counterparts: regrouping and arranging pictures. Instead they present it in a grid, while my approach is to collect the images on a canvas where it is possible to arrange and resize them. Furthermore it adapts the idea of Prezi, allowing to zoom in out to create fields of interest.

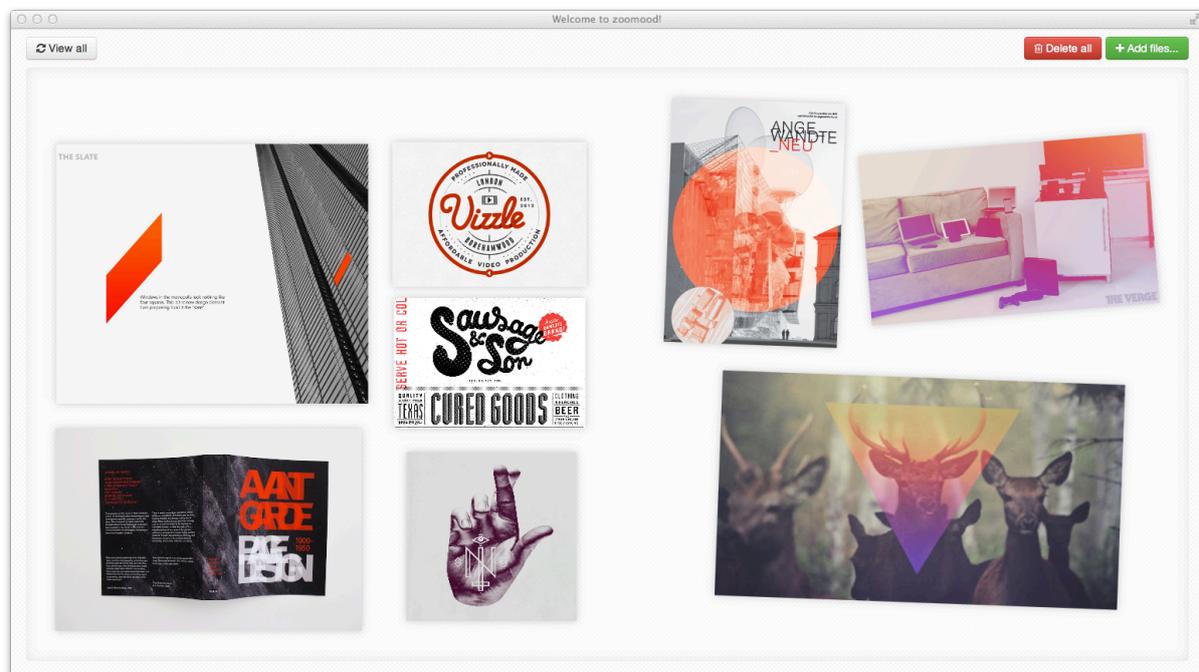


Figure 1. Zoomood canvas with all images.

With *Zoomood* you can create a mood board by simply dragging images onto a canvas. Then you can arrange your images and zoom and pan around the canvas or delete images. Further the app allows selecting multiple images at once to manipulate them at the same time. The overall design goal was to keep the interface and interaction as easy, simple and smooth as possible. For that reason the app mainly consists of a large canvas area and a small toolbar at the top, as shown in Figure 1. The only user interface elements are simple buttons to reset the viewport (in case you "lost" your content while moving or

zooming) and handle files. To upload single or multiply images you have to drag and drop images from the file explorer to the canvas (this is possible through the new HTML5 standard). Alternatively it is possible to use the upload button at the top bar. Then these images will be uploaded to a server and will be inserted on the canvas. When selecting one or more images you can then move, resize, scale or delete them, while the canvas is saved persistently in the background. By pressing and holding the space bar the user can move and zoom on the canvas, allowing exploring certain details of the mood board.

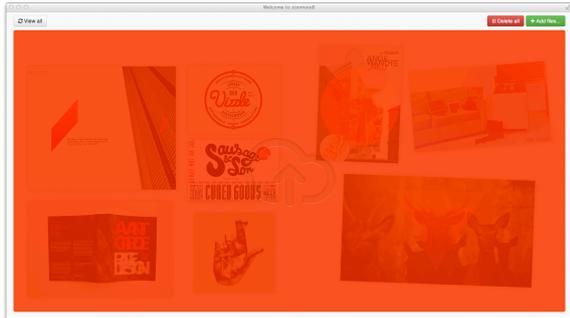


Figure 2. Canvas changes when files are dragged.

IMPLEMENTATION

The concept was implemented as a webpage using latest web development standards. The front-end layout is built on top the *Twitter Bootstrap* framework and is massively using the program language JavaScript on top of the *jQuery* framework. The images are displayed with the new HTML5 *canvas* element that provides a graphical context to draw on. The JavaScript library *fabric.js* allows handling the canvas features in an easy manner and provides basic image manipulation (such as resizing and rotating an image). The *jQuery File Upload* plugin allowed an easy way to upload multiple images via drag and drop. At the same time the canvas changes its appearance, indicating that the files can be dropped, as shown in Figure 2. The backend was developed with JavaScript using the *node.js* framework as server. Another library called *express.js* provided an easy interface for the http calls (the service follows the REST design pattern). The data is persistently stored within the NoSQL database *MongoDB*. CSS files are partly implemented with *LESS* and compiled to CSS files at run-time.

EXPERIENCE

The main motivation to implement this solution was the lack of existing solutions on the Internet. Further using open web technologies as JavaScript and HTML5 ensure that the app is technically available on many devices (PC, Mac, tablets and so on). Performance and smoothness of the interaction were critical aspects, so I experimented with some libraries to find the one that suits best. At the end I chose *fabric.js*, since this had the best overall performance and provides an easy user interface at the same time. Using the laptop's touchpad made the interaction feel quite natural (e.g. when zooming in or out, especially with the MacBook's inertia scrolling). Figure 3 shows the canvas when zoomed in.



Figure 3. Zoomed canvas.

Further the drag & drop upload was very crucial, since uploading files on a web browser was always a painful task with classic *web 1.0* applications (often limited to upload single files). I was satisfied with the app's performance and discovered that it was an easy task to concentrate on creating and arranging the collage instead of handling the app itself. The *fabric.js* library offered handy functionality such as multiple object selection and a nice UI for manipulating objects. Providing simple keyboard shortcuts for manipulating (navigating on the canvas and deleting images) means that it's not necessary to find the right button on the user interface.

FURTHER IDEAS

The developed app at this point is more a prove-of-concept, since it misses aspects needed for a full-featured product. Since the app is designed as a web service some more dimensions have to be considered: security (inspect file content, limit file size, user privileges etc.), user management, collaboration features, document management and technical improvements to scale the app's performance (e.g. techniques to reduce the memory amount when displaying a lot of images by e.g. implementing mipmapping techniques or improving the data transfer while saving). Further ideas to improve the app are including more media on the board: using text as labels, allowing to upload richer content, as videos or sound and also the possibility to tag the media and filter based on criteria, as tags, media type and so on. Grouping (or framing) multiple objects can also be considered. Hosting the app would be easy on cloud platforms such as *Heroku*, since they support all the needed technologies. The project is open source and can be downloaded at GitHub¹.

¹ <https://github.com/ulrichson/zoomood>