Comp2Dust Project Overview

**1.Description**

Project **Comp2Dust**

Made by Yiğit Şık

Contact: [yigit6958@gmail.com](mailto:yigit6958@gmail.com)

Git: <https://github.com/YigitSIK/Comp2Dust>

**2.Elevator Pitch**

Comp2Dust online image optimizer uses lossy compression algorithms to shrink JPEG and PNG images to the minimum possible size while keeping the required level of quality.

**3.Overview**

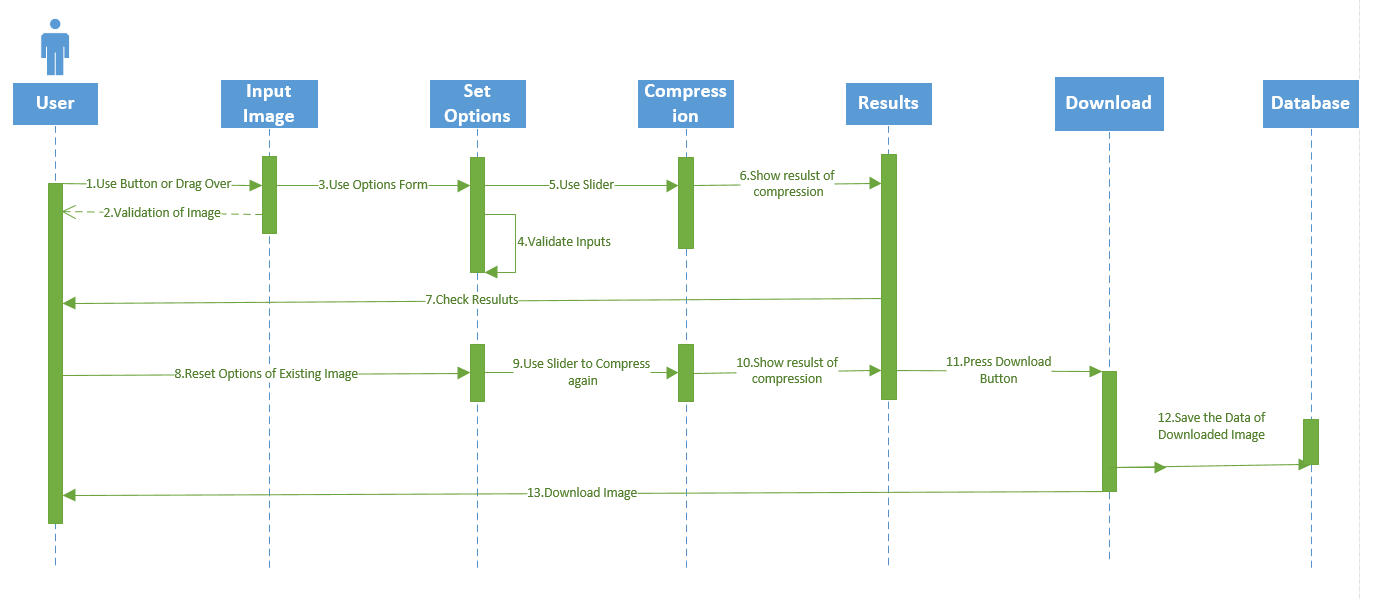
JPEG, and PNG formats make most of the entire internet’s image traffic. That is why this web site will be aiming these two format. Due to increasing upload numbers each day, it become a need to optimize content

**4.Requirements**

Comp2Dust web site is an online image compressor available for every user who has an internet connection and any popular browser such as Firefox,Mozilla,Chrome... Our intended users are people who need smaller sizes for their environment. And since we have covered two main formats that todays internet, we have common users.

**5.Functionality**

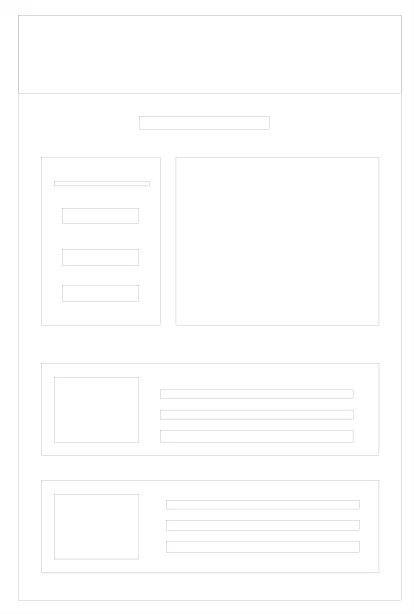
Here is an example to general flow of the application



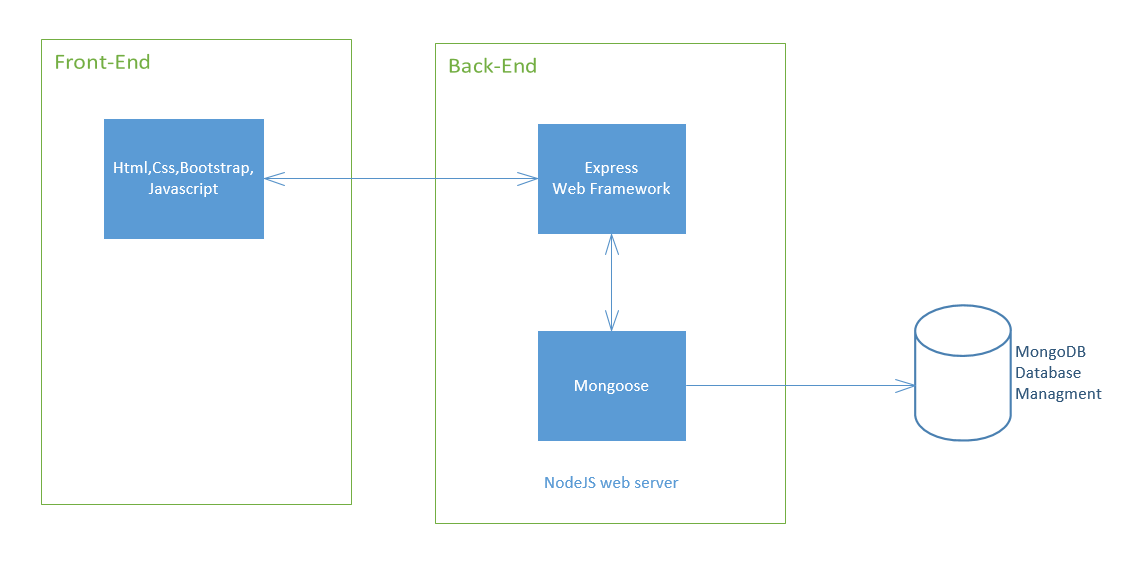
* Upload Image
* Set Compression Rate
* Set Image Width
* Set Image Height
* Reset
* Download

**6.Design**

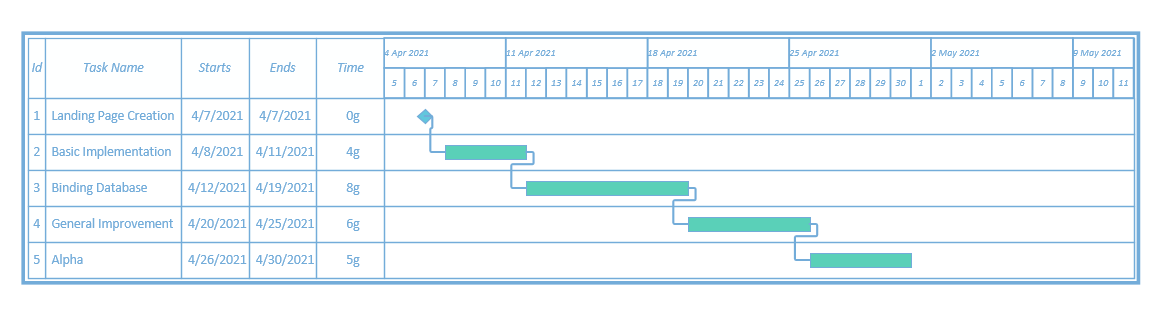
Site will designed to be a single page application. Here is a prototype of it.



* MongoDB database with Atlas cloud service to keep image data but not images themselves.
* In terms of user interface, HTML, CSS, and Bootstrap library will be used to illustrate the system attractively.
* Client side compression function depends on npm package called “compress-js”.
* Another npm package to called “body-parser” to get parsed information from HTML to server side.
* On the server side, express js will be used to handle routes and mongoose to connect database.
* Site will be hosted on Heroku servers.



**7.Milestones**



* As a first task there will be the basic design of constructed landing page.
* By the end of 2nd task application should perform the main function crudely
* At third step working algorithm should be able to send data to MongoDB database.
* At fourt step, any occurence of bugs should be fixed, input validations and exception handling task should be implemented.
* In Alpha section application should be properly working and its user interface should be revised for presentation.

**8.Risks**

* Main function will be replaced by scientific one, if this cannot be done, the existing one will be upgraded or it will be replaced by a better npm package.