IMG.Convert

General Description

For my project, I created a web application with a simple layout that allows a user to upload an image and convert it to JPEG or PNG format. The user can also optionally choose to change the size of the image by specifying the pixels for the width and height. The user must provide an email address, so that when conversion is complete, he/she receives an email with a link to the converted image.

The user can login to the web application using a Google account. In this case, the user does not have to enter an email address for the conversion, since the link is sent directly to the email that Google account is registered to.

Technical Description

When the user clicks 'Convert Images', what the backend does is retrieve the uploaded image, the new format specified, and the new width and height if provided. If the user is not logged in, the email address is retrieved from the request too, otherwise it is retrieved from Google App Engine's Users Python API.

The backend writes each uploaded image file to a Google Cloud Storage bucket, and for each file adds a conversion task to the App Engine push queue handling conversions.

A worker service is then used to process tasks on that push queue. It uses the Blobstore API to create a blob key for the GCS file, and then uses the App Engine Images Python API to transform the image using the blob key. If a new width and height were specified, it resizes the image and then transforms it to the desired end format. The converted image is then uploaded again to a different bucket in Google Cloud Storage. An email is then sent to the user, containing a link to the image, which is a serving url generated using the Images API.

The user can then proceed to download the image from the email.