

# Udacity

**Student:** Thalbert Barbosa de Miranda

**Course:** Cloud Developer

**Project:** Engineering Full Stack Apps in the Cloud

---

## Engineering Process and Quality

*The project demonstrates an understanding of a good cloud git process*

GitHub repository:

<https://github.com/Thalberty/cloud-developer/tree/master>

*The project demonstrates an ability to use typescript and Nodejs*

```
app.get("/filteredimage", async (req: express.Request, res: express.Response) => {
  const { image_url } = req.query

  // 1. validate the image_url query
  if (!image_url) {
    res.status(400).send('Please, send a valid url')
  }

  try {

    // 2. call filterImageFromURL(image_url) to filter the image
    var imageFiltered: string = await filterImageFromURL(image_url);

    // 3. send the resulting file in the response
    res.status(200).sendFile(imageFiltered);

    res.on('finish', function () {
      // 4. deletes any files on the server on finish of the response
      deleteLocalFiles([imageFiltered])
    });

  } catch (err) {
    res.status(422).send('Processing failed, please try again')
  }
});
```

## Development Server

*The project demonstrates the ability to develop using the NodeJS framework*

```
thalbert@thalbert:~/Documentos/AWS/fullstack-apps-on-aws/project1/cloud-developer/course-02/project/image-filter-starter-code$ npm run dev

> udacity-c2-image-filter@1.0.0 dev /home/thalbert/Documentos/AWS/fullstack-apps-on-aws/project1/cloud-developer/course-02/project/image-filter-starter-code
> ts-node-dev --respawn --transpileOnly ./src/server.ts

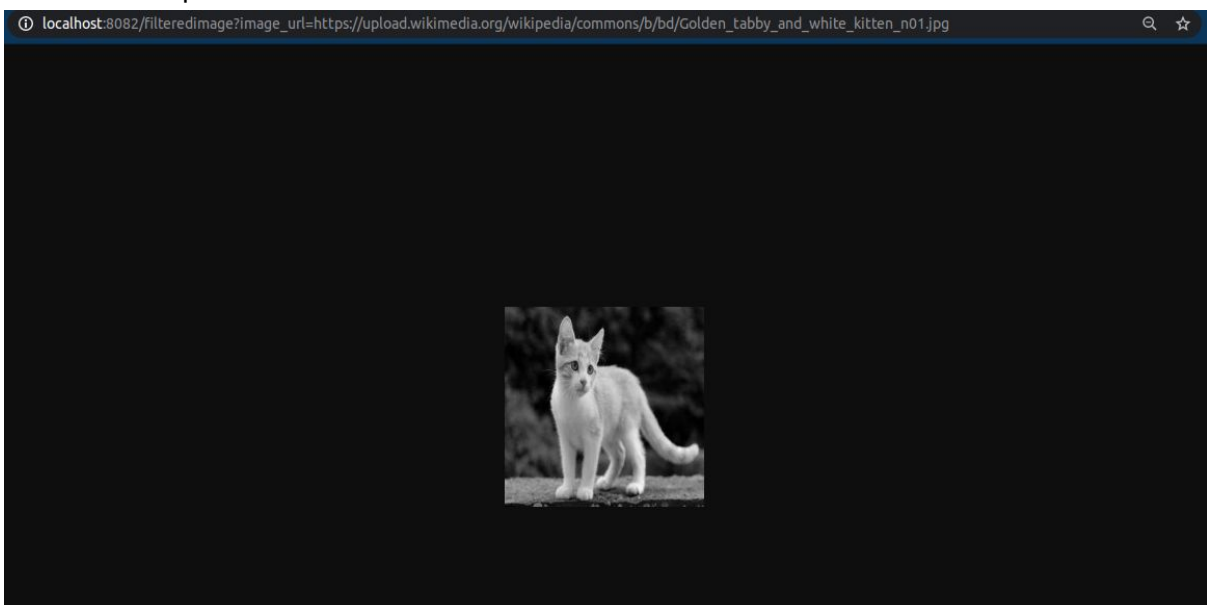
Using ts-node version 8.3.0, typescript version 3.5.3
server running http://localhost:8082
press CTRL+C to stop server
```

*The project demonstrates an understanding of RESTFUL design*

[http://localhost:8082/filteredimage?image\\_url=https://upload.wikimedia.org/wikipedia/commons/b/bd/Golden\\_tabby\\_and\\_white\\_kitten\\_n01.jpg](http://localhost:8082/filteredimage?image_url=https://upload.wikimedia.org/wikipedia/commons/b/bd/Golden_tabby_and_white_kitten_n01.jpg)

*The project demonstrates an understanding of HTTP status codes*

Successful response:



Error code for caught errors (i.e. 422):


```
} catch (err) {
  res.status(422).send('Processing failed, please try again')
}
```

## Elastic Beanstalk Deployment

The project demonstrates the ability to create functional cloud deployments


http://udacity-image-project-tm-dev.sa-east-1.elasticbeanstalk.com/filteredimage?image\_url=https://upload.wikimedia.org/wikipedia/commons/b/bd/Golden\_tabby\_and\_white\_kitten\_n01.jpg

The project demonstrates an understanding of AWS Elastic Beanstalk's CLI and Console Dashboard

**udacity-image-project-tm-dev**  
[udacity-image-project-tm-dev.sa-east-1.elasticbeanstalk.com](http://udacity-image-project-tm-dev.sa-east-1.elasticbeanstalk.com)  (e-bxyafjcbyc)  
Nome do aplicativo: **udacity-image-project-tm**

Atualizar


Ações ▼

**Integridade**  
  
OK  

Causas

**Versão em execução**  
app-201105\_223539  

Fazer upload e Implantar

**Plataforma**  
  
Node.js 12 running on 64bit  
Amazon Linux 2/5.2.2  

Alterar

**Eventos recentes**

Mostrar todos

< 1 >

Tempo	Tipo	Detalhes
05-11-2020 22:39:55 UTC-0300	INFO	Environment health has transitioned from Pending to Ok. Initialization completed 17 seconds ago and took 3 minutes.
05-11-2020 22:39:10 UTC-0300	INFO	Successfully launched environment: udacity-image-project-tm-dev
05-11-2020 22:39:09 UTC-0300	INFO	Application available at udacity-image-project-tm-dev.sa-east-1.elasticbeanstalk.com.
05-11-2020 22:38:55 UTC-0300	INFO	Added instance [i-07d9574e9c688a17d] to your environment.
05-11-2020 22:38:55 UTC-0300	INFO	Instance deployment completed successfully.