

# Cloud assignment with Hallvard and Vegard !

Link to api:

`flask-production-5ade.up.railway.app`

This system is a contact management system, and it is already connected to a MongoDB instance. The system allows users to upload their contact information from a VCF file to the MongoDB instance and retrieve the information in a VCF format. Thus, the system needs infrastructure that can handle a database and allow it to be queried. PaaS (Platform-as-a-Service) is the best fit for this system. This is because PaaS provides the necessary environment for developing and deploying applications without the hassle of managing the infrastructure. In PaaS, the provider handles most of the underlying infrastructure, such as server, database, and storage. Therefore, the developer can focus on developing the application and not worry about managing the infrastructure. Since this system is already connected to a MongoDB instance, a PaaS provider that supports MongoDB is a good choice, such as Heroku or AWS Elastic Beanstalk.

Our system is a API created through the flask framework. It connects to a mongodb and have functions to interact with it. The functions are activated by accessing different routes that we have defined. Here are some different routes:

- `route('/')`, this is the root route and simply gives a welcome to the api. This uses GET method.
- `route('/upload')`, this route shoes a html template where user can upload vcf files. The code formats and puts it in database as json format with help of vobject. It also tells the user what id of the data has been added. This uses GET and POST method.

- `route('/contacts')`, This route returns all the contacts as json data. This uses GET method.
- `route('/contacts/vcard')`, This route takes the JSON in database and returns it as vcard format inside json structure for the user to access. This uses GET method.
- `route('/contacts/vcard/<email>')`, This route allows the user to access specific contacts in the database by searching by email. This route uses GET method.
- `route('/contacts/<id>')`, This route allows the user to access specific contacts in the database by searching by ID generated by mongodb, and deleting the contact. This route uses DELETE method.
- `route('/contacts/export')`, this route allows the user to download the contacts vcard.