

# NotArmLeg documentation

William Kelly, Alex Thomas, Charles, Luke

February 23, 2020



# 1 Introduction

This project is designed to train users to not click on spam emails.

They will add their email using the website and choose the maximum number of emails they wish to have sent per month and then we will send them fake spam. On this fake spam is a link which if pressed they get redirected to our website which will display information about the features of the email that suggests that it is spam. This is designed to educate the user and hopefully mitigate them accidentally clicking on spam emails.

# 2 Module interaction

The project consists of 4 modules:

- Web server
- Database
- Spam repo
- Email server
- Repository generator

The web server uses flask for its backend. We are using MySQL for the data base. The spam repo is a file server with a defined hierarchy for storing spam emails and finally the email server is written in python.

### 3 Repo hirearchy

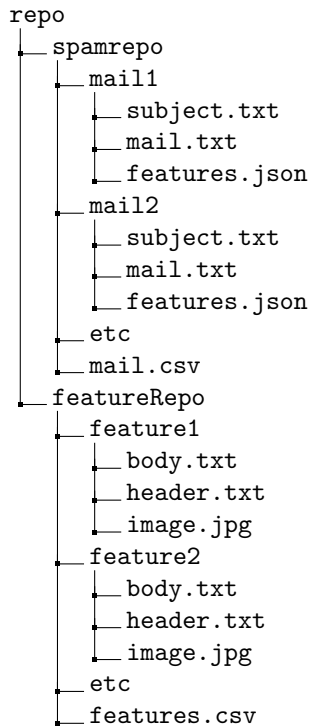


Figure 1: Repository hirearchy for spam emails

The spam repo is used to store all the emails that would be sent to the users. Inside this directory exists a mail.txt file that stores the html of the email. The features.json file contains metadata on the email. This metadata is information that says what about the email indicates that it is spam. Features can be found inside the feature repo. Each feature has a body, header along with an image that is fetched once a link inside a spam email has been pressed. This can then be displayed on the webpage as the information used to educate the user.

Listing 1: features.json

```
{
  "features": [
    {
      "name": "funkyAddress",
      "id": "funkyAddress",
      "repository": "repourl/telloffsite/featurerepo"
    },
    {
      "name": "funkySpelling",
      "id": "funkySpelling",
      "repository": "repourl/telloffsite/featurerepo"
    }
  ]
}
```

## 4 MySQL database

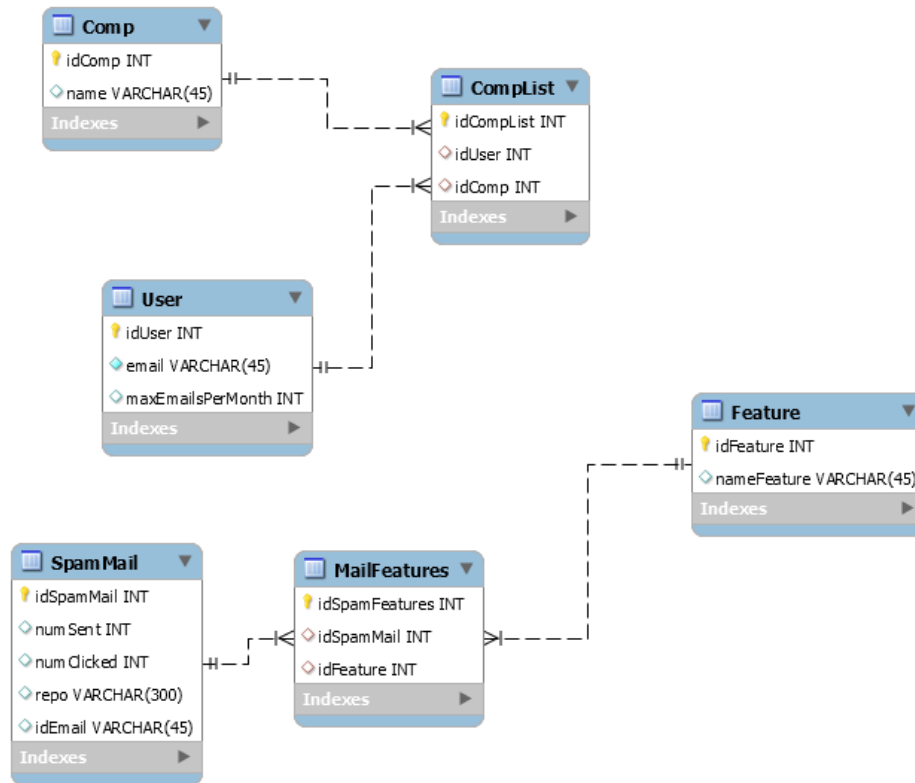


Figure 2: MySQL database layout