**README DEGROOF PETERCAM CHALLENGE**

* **Repository**: Multipage-Website-in-php
* **Team** : In a group of 5 persons
* **Duration** : 3 Days
* **Done by**
  + Antoni Dalle Nogare
  + Jean-François Geerinckx
  + Hakan Poyraz
  + Colombe Sechehaye
  + Ajay Shinde
* **When?** - From 24th September 2018 to 26th September 2018.
* **Where?** - A project done during a training at Becode (after 4 months coding)

**About the Project**

**Degroof Petercam Challenge**

**Challenge pour réaliser une application dans le cadre de la formation à BeCode pour le client : Degroof Petercam.**

Make an application so the employees can apply for voluntary work during their work hours (2hours a month). The employees would be able to choose a voluntary project in a list function of their skills (accountability, human ressources, marketing…). If their application is accepted, they receive the answer by e-mail and receive a link that run them into the app to see more information about the day they have to go there (address, hours…)

**About the Project**

This project will helps us to learn how to realise a project in real situations, but will not be given to a any client. It is just an exercise from [BeCode](https://github.com/becodeorg/) For this, we have to refresh an existing [website](http://www.abpasbl.be/-AMV-asbl-Bruxelles-), with several pages to separate the content and that the visitors to the website can send an email via a contact form with a small photo.

**Client**

Degroof Petercam

**Objectives**

The objective of this challenge is to give us the opportunity to use our new knowledge in PHP in a realistic scenario. It is therefore more important that each member of our team achieves her learning objectives than to succeed in the project. And here come steps to follow:

* Use of **DRY** method
* Separate php from html
* Use of .gitignore file
* Contact page has to contain a form that will:
  + Allow user to choose their gender
  + Ask for user's name, last name and email
  + Allow user to write a message
  + Allow user to upload a jpg, gif or png
  + Allow user to choose between *html* and *txt*
* Uploading will be done with *php upload* class
* Email will be managed with *phpmailer* or another similar class
* Form will be sanitized, validated, executed and will provide feedback
* Form sending will be logged in a *.txt* file and a *.php* page (which will remove private data)
* Errors from user will dispatch appropriate messages
* Use CCS framwork to design responsive pages
* lighthouse test result of 80 in every category