# DIVE developers Documentation

Following steps are valid for DIVE or any project based on GRUNT / BOWER / GITHUB tools.  
To be able to successfully modify and code on DIVE project it is MANDATORY to read the following document in all his parts; having as background good skills of front-end development (html5,css3,js).

DIVE developers Documentation 1

FRONT-END environment setup (on windows) 3

Tools suggested developing: 3

Using the Terminal: 3

Install the global environment : 3

Tips and useful information: 4

GITHUB setup 5

GITHUB Tips and useful information: 5

INSTALL YOUR DEPENDENCY 6

GRUNT (http://gruntjs.com/) 6

YEOMAN (http://yeoman.io/) 6

Various: 6

DIVE Content update training support 7

Wht DIVE is using: 7

Install DIVE (to develop it) 7

Build DIVE (to deploy it on dive.pmi.com) 8

# FRONT-END environment setup (on windows)

## Tools suggested developing:

* Netbeans 7.4.x *Or* Sublime Text 2 *Or* your preferred tools

## Using the Terminal:

1. Open Start menu > Programs > Accessories > Command prompt
2. Go to your local copy repository of the project
   1. type “ls” (enter) to view the content of your position
   2. type “cd [foldername]” to acess it
   3. type “cd ..” to go to previous file
3. Once your in the root of the local repository of the project, type “grunt server”
4. (More info about command prompt on <http://www.makeuseof.com/tag/a-beginners-guide-to-the-windows-command-line/>)

## Install the global environment :

1. Install node.js downloading the windows installer from the website
   1. More information on node.js installation: [*http://nodejs.org/*](http://nodejs.org/)
2. Install the terminal “Git Bash” (this will be your terminal to insert all the necessary command)
   1. go at the website http://git-scm.com/download
   2. download the windows terminal; latest version
   3. install “git bash” and when requested
      1. select 2nd checkbox
      2. select 2nd checkbox
3. install Ruby (requested on windows to be able to install sass)
   1. go at the website <http://rubyinstaller.org/downloads/>
   2. download 2.0.0 version (select all the 3 check when the process start)

Your global tools are installed

Now some installation by terminal

1. ***install SASS***: open your “Git Bash” terminal and on your “Git Bash” terminal write: “gem install sass”
   1. more information on how toinstall SASS on windows: <https://gist.github.com/andersonaguiar/2400385>
2. ***install Compass***: on your git-bash terminal write: “gem install compass”
   1. More information on compass installation: <http://compass-style.org/install>
3. **install yeoman:** on your git-bash terminal write: “yeoman install”
   1. more information about yeoman installation: <http://yeoman.io/gettingstarted.html>

Your global environment is set up

### Install the local application environment:

1. **initiate GRUNT:** open your “Git Bash” terminal and on it type: “npm install –g grunt-cli”
2. **initiate BOWER:** open your “Git Bash” terminal and on it type: “npm install -g bower”

## Tips and useful information:

To validate and control that you have the necessary “stuff” installed:

Open your “Git Bash” terminal and on it type “THE\_NAME\_OF\_YOUR\_STUFF –v”

Example: “ compass –v “ “npm –v” or “ruby –v”

This will give you the version of the “tool” installed if not means that is not installed correctly.

# GITHUB setup

Before using GITHUB you will need (only the first time) to “tell” github that your computer is allowed to publish files. For that you need to generate a unique KEY that you will transmit to github.

1. To do that please follow the instruction on: <https://help.github.com/articles/generating-ssh-keys>

Before typing any github command please follow the below step:

1. Open your “Git Bash” terminal and on it type: “ *git config --global url."https://".insteadOf git://* “

At this point you can use github (clone, pull, push, commit….)

Git cheat sheet: [**http://www.git-tower.com/blog/assets/2013-05-22-git-cheat-sheet/cheat-sheet-large01.png**](http://www.git-tower.com/blog/assets/2013-05-22-git-cheat-sheet/cheat-sheet-large01.png)

## GITHUB Tips and useful information:

First time you will clone or pull to a project you will be requested to enther e PASSPHRASE tht will be your unique password. You can just press enter to avoid setting it up.

### Useful terminal (“git-bash”) Command:

**CTRL+c** To stop the terminal

**“ls”** To view the content of your position

**“cd [foldername]”** Tto acess the [foldername]

**“cd ..”** to go back of one level in the folder structure

**“mkdir [foldername]”** To create a folder [foldername]

### How to clone an existing project:

1. Open your “Git Bash” terminal and got to your local directory where your project is.
2. Type: “git clone URL\_SSH\_OF\_PROJECT” - Example: “git clone git://github.com/folder/project-name.git”

* You have to copy the link from github.com inside the project you want to clone.

### How to upload your latest version to an existing project:

1. Open your “Git Bash” terminal and got to your local directory where your github project is.
2. Confirm modifications: “git add .” or “git add -A”
3. Commit modified files: “git commit –am ‘your message’ ”
4. Push the commitment: “git push origin master”

### How to download (merging) the latest github version with your local version :

1. Open your “Git Bash” terminal and got to your local directory where your github project is.
2. Get the latest version (like a checkout). Type “git pull origin master”

# INSTALL YOUR DEPENDENCY

*To proced with the dependency and project installation it is mandatory to read previous parts (above) of this document*

### first time you get a “grunt project” on your local environment:

1. Open your “Git Bash” terminal and got to your local directory where your github project is.
2. Type “npm install“ (it pull the dependencies from the file package.json)
3. Type “git status” : to control changes
4. Type “bower install”

DONE (you can run “grunt server”)

## Various:

“gem update --system “ to update the system

### GRUNT (<http://gruntjs.com/>)

### Useful terminal (“git-bash”) Command:

**“grunt server”** to run the local server

**“grunt build”** to build the project (distribution version [dist])

### YEOMAN (http://yeoman.io/)

### Useful terminal (“git-bash”) Command:

**“yo webapp”** to create/generate a web application

# DIVE Content update training support

## Wht DIVE is using:

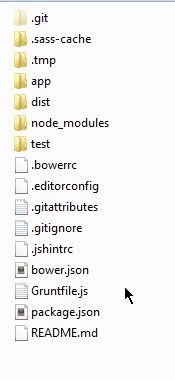
1. [**GruntJS**](http://gruntjs.com/)to compile, build, and do almost everything
2. [**Bower**](http://bower.io/)for front-end dependencies
3. [**Assemble**](http://assemble.io/) for building HTML templates and PHP-like partial/includes.
4. [**Font Awesome**](http://fontawesome.io/) for awesome font icons
5. [**PrismJS**](http://prismjs.com/index.html) for syntax highlighting
6. [**Sass**](http://sass-lang.com/) for some awesome pre-processed CSS

To be able to code and change DIVE project above technologies are mandatory to be able to successfully understand and modify the project.

## Install DIVE (to develop it)

As pre-requisite you will need to be proficient on using GRUNT + BOWER and front-end technologies.  
Once you will get the DIVE project you will need to follow the step “[INSTALL YOUR DEPENDENCY](#_INSTALL_YOUR_DEPENDENCY)”.

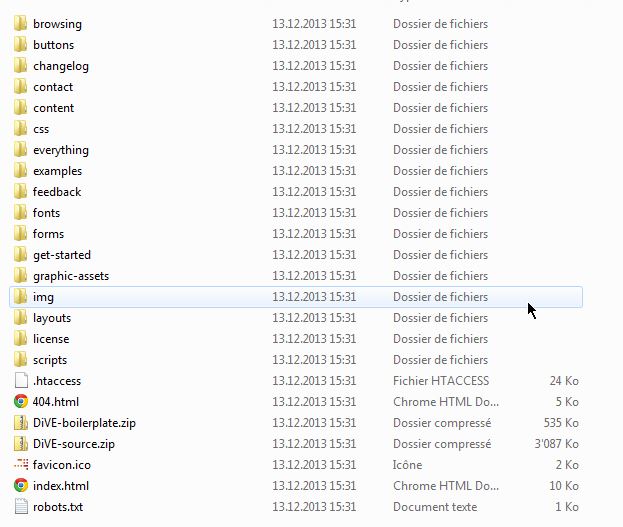
Once the DIVE project is installed on your machine your **Development folder structure** will look like this:



## Build DIVE (to deploy it on dive.pmi.com)

Once you have the project correctly running on your machine you can build DIVE.  
To build (compile) the project on your Terminal you will need to write “grunt build” and the application will build your project ready to be copied and pasted into dive.pmi.com

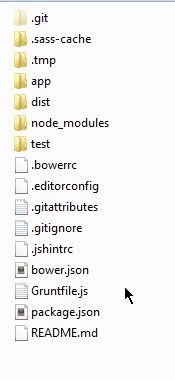
The DIVE will be generated inside the folder “dist” and the contained files/folders will looks like this:



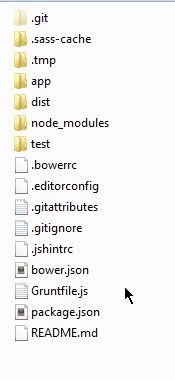
To deploy DIVE on pmi.dive.com you will need to copy and paste all the above files inside your DIVE Website root.  
*Remember that if other applications/website like “HTML5 pmi guidelins” needs to be uploaded you can copy and paste it anywhere in the same root.*

 Create content, modify content, add pages.  
First of all let’s be familiar with the **Development folder structure.**

**Two folders** will need to be know and used:

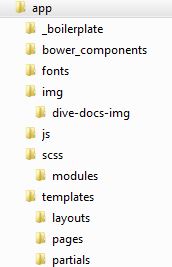


The folder where the project will be compiled. (as explained above in the document)

  
Containing **ALL** the source files where you will have to work to be able to change, add pages/contents.

**Once you will know the content off APP folder you will have all the information about where to find necessary source files to modify /create contents/pages**

### Detailed explanation of APP contents:



|  |  |
| --- | --- |
| **FOLDER** | **DESCRIPTION** |
| \_biolerplate | This folder contains files that are bearly copied and pasted inside “dist/ DiVE-boilerplate.zip” once DIVE is build (“grunt build”) |
| Bower\_components | Inside this folder thare are ALL the dependencies component for the project according to BOWER.io they are updated / added inside this folder (during “bower install”) |
| Fonts | This folder contain the icon fonts used in DIVE |
| Img | This folder contain the iamges that are exported into “dist/ DiVE-boilerplate.zip” |
| Img/dive-docs-img | This folder contain the iamges used ONLY for the DOC website that is the « dist » folder itself (not compiled inside the “dist/ DiVE-boilerplate.zip”) |
| Scss | This folder contain all the uncompiled CSS. This one are the general and global CSS classes including variabels. |
| Scss/modules | This folder contain all the uncompiled CSS of the modules 8elements of DIVE) named accordingly to their functionality and use (example the \_accrodion.scss it is the CSS source for the accordion component and so on) |
| Tempaltes | This folder contain ALL the pieces that build the DIVE DOC pages [ based on [Assemble](http://assemble.io/)] |
| Tempaltes/layouts [LAYOUTS] | This folder contain the layout page : default layout used for all pages and example layout used for the example page of DIVE |
| Tempaltes/pages [PAGE] | This folder contain all the DIVE pages that are included into one layout according to assemble code language |
| Tempaltes/partials  [PARTIAL] | This folder contain all the components of DIVE (grif,icon,datepicker, collapse ….) that are included inside one or more pages |

***IMPORTANT***: One page is build of multiple **PARTIALS** that are included inside the **PAGE** that it is attached to ONE **LAYOUT**

From this you can easily learn that to add a page or change it you will need to go inside the “templates” folder and working on the needed file.

For example to change the page contact you will search for “contacts.hbs” inside the folder Tempaltes/pages fromt this file you will see wich PARTIALS are included and you can change the contact page itself (changing componens name, order page title, description etc…)  
or

Reading the name of the paritial included you can locate them into “Tempaltes/partials” and as a normal html page you can edit them.