# Contribution Information

### Contribution to InvoicePlane

Every help is welcome, you don't have to be a professional PHP developer or SQL database engineer.

We appreciate any support with helping other users, translating the software or simply spreading

the word. You can find more general information in our [community contribution guide](https://go.invoiceplane.com/contribution).

If you have any questions regarding contribution please ask! Nothing is more frustrating for you

and us than misunderstandings which may lead to large amounts of time required to solve the resulting

problems.

We have a community forum and a Slack channel so please use them:

\* [Community Forum](https://community.invoiceplane.com/)

\* [Slack Channel](https://invoiceplane-slack.herokuapp.com/)

---

### Contents

\* Development Guidelines

\* Versioning and Git Branches

\* Getting Started with Development

\* Contributing Code

---

### Development Guidelines

To make sure the code is readable and well structured please follow these guidelines.

We do highly recommend using the [PhpStorm](https://www.jetbrains.com/phpstorm/) IDE

as it natively supports the correct code formatting.

#### PHP

All code is formatted according to PSR-1 and PSR-2. These two standards assure that

the whole code looks the same and is well formatted.

Variables and functions are always declared in the `$under\_score` formatting, not

`$camelCase`. Arrays are always specified in the short syntax: `['item', 'item2']`.

See [PSR-1](https://www.php-fig.org/psr/psr-1/) and [PSR-2](https://www.php-fig.org/psr/psr-2/)

#### JavaScript

Use the Standard JavaScript Code Formatting (https://standardjs.com/) with the only

change that we do use semicolons.

#### CSS

All CSS styles are written in [Sass](https://sass-lang.com/) in the SCSS style. You

do not need to specify any styles with vendor prefixes like `-moz-box-shadow` as

they are added automatically in the compilation process!

---

### Versioning and Git Branches

We do use [SemVer](https://semver.org/) to version InvoicePlane which means:

\* Bugfixes or smaller improvements go into minor versions like `v1.3.2` or `v1.5.6`.

\* New features or larger improvements go into feature versions like `v1.4.0` or `v1.6.0`.

The InvoicePlane branch structure consists of four different branch types

\* `master` This branch contains the current stable version of the application.

There will never be any development activity on this branch so please refrain

from using it for pull requests.

\* `1.5` Branches that contain only a feature version of InvoicePlane, like `1.3`

or `1.5` are archive branches that contain the last stable release of that

specific version.

\* `v1.6.0` Branches that contain a specific version number like `v1.6.0` or `v1.5.4`

are the development branches. All new code is added to these branches ONLY.

If you contribute any code make sure to select the correct branch.

Remember: bugfixes and small improvements go into minor branches like `v1.5.4`

and new features or large improvements go into feature branches like `v1.6.0`.

\* `issue` Other named branches are used to keep code for very specific tasks that do

not fit into another branch. Only push code to these branches if you are

requested to do so.

---

### Getting Started with Development

#### PHP Development

If you just want to change PHP code or HTML files, you just need to have

[Composer](https://getcomposer.org/doc/00-intro.md) installed. The tool manages

the PHP dependencies with the help of the `composer.json` file.

1. Run `composer install` to install all dependencies.

2. Follow the basic installation guide in the README.md or the InvoicePlane Wiki.

3. Make sure your editor or IDE has the correct code formatting settings set.

4. That's it. You are good to go.

#### Styles and Scripts Development

To change the actual styling of the application or scripts you may have to install

some other tools.

Please notice that InvoicePlane has a unique theme system that splits up styles

into core styles and theme styles. Please visit the [InvoicePlane Themes](https://github.com/InvoicePlane/InvoicePlane-Themes)

repository for more information.

This main repository ONLY manages core styles and scripts. If you want to develop a new

theme please use the InvoicePlane Themes repository.

\* InvoicePlane requires Node and NPM: [Install both](https://nodejs.org/en/download/)

Make sure to use the latest LTS version which is version 8 at the moment.

\* Install Grunt globally by running `npm install -g grunt-cli`.

If you have prepared your machine, run `npm install` to install all dependencies.

After that there are two commands available for development:

\* `grunt dev` Compiles all assets and starts watching your development files. If you

change a file and save it, the compilation will start automatically so

you do not have to do it manually.

All styles and scripts are not minified and sourcemaps are generated for

easier development.

\* `grunt build` This command is used to compile all assets for the production use. It

also minifies all assets to make sure pages load faster. Normally this

command is not used for development.

---

### Contributing Code

\*\*Read this carefully to prevent your pull request from being rejected!\*\*

1. Before you submit any code to the repository please take a look at the

[issue tracker](https://development.invoiceplane.com) and search for a

corresponding issue! There should be no changes to the code without an issue

filed.

\* If an issue exists and someone is working on it, please contact this person.

\* If an issue exists and no one is working on it, assign it to yourself or write a comment.

\* If no issue exists, contact the InvoicePlane team via the forums or the Slack channel first!

We would like to discuss what you want to do before you start to work.

2. ALWAYS reference the issue ID (e.g. IP-317) in all commits you create for this issue.

3. BEFORE you create a pull request, rebase from the corresponding branch. This is very important

to make sure that there are no duplicate commits and you do not revert any previous changes.

\* Add the development branch as a remote:

`git remote add ip git@github.com:InvoicePlane/InvoicePlane.git`

\* Do a rebase with the following command:

`git pull --rebase ip [branch]`

Where `ip` is the name of the remote and `[branch]` the corresponding branch.

\* Solve all conflicts and check if your code is still working.

4. Submit the pull request, reference the issue ID in the title and add a meaningful description.

Please ask questions related to this process if you are unsure. Asking before doing anything

will save both your and our time.

---

> \_The name 'InvoicePlane' and the InvoicePlane logo are both copyright by Kovah.de and InvoicePlane.com

and their usage is restricted! For more information visit invoiceplane.com/license-copyright\_