Development Guidelines

- Web Development
- Mobile Development
- Upgrade Notes (Laravel 9 -> 10 & Backpack 5 -> 6)
- Rules Pint

Web Development

Revision History

Date	Version	Authors	Description
2020-08-04	1.0	Adli I. Ifkar	Initial Version
2021-01-22	1.1	Adli I. Ifkar	Updated Tools
2021-05-03	2.0	Adli I. Ifkar	Removing Windows, Added ERD, and updated Issues

1. Overview

Tim IT Yayasan Vitka adalah tim yang bertanggung jawab terhadap pengembangan software di lingkungan Politeknik Pariwisata Batam, Institut Teknologi Batam, dan Yayasan Vitka. Software yang dikembangkan saat ini difokuskan pada kebutuhan akademik dan penunjang kegiatan akademik lainnya. Teknologi yang digunakan berbasis web. Software yang ditulis dengan baik dan rapi memiliki banyak kelebihan. Software tersebut akan memiliki bug lebih sedikit dan akan berjalan lebih efisien dibandingkan dengan software yang tidak ditulis dengan baik. Karena software memiliki life cycle dan Sebagian besar dari life cycle tersebut adalah maintenance, tim pengembang maupun anggota tim yang akan datang akan lebih mudah untuk melakukan maintenance dan merubah program sesuai dengan kebutuhan. Tim IT Yayasan Vitka membuat standarisasi dan panduan singkat untuk memastikan seluruh anggota tim IT yang terlibat memiliki visi yang sama dalam mengembangkan software. Standar dan panduan ini tidaklah konstan, namun dapat berevolusi sesuai dengan kebutuhan dan kesepakatan ke depannya.

2. Standard Tools

2.1 Tools & IDE

Fokus pengembangan pada Yayasan Vitka adalah web application berbasis Laravel dan Backpack for Laravel. Tools utama yang dibutuhkan adalah:
a. PHP >= 8.0
b. PHP >= 7.4.14
c. Git >= 2.27.0
d. MySQL >= 8.0.17
e. Redis >= 6.0
Adapun untuk GUI Tools & IDE yang digunakan adalah:
a. PHPStorm
b. SmartGit untuk git
c. HeidiSQL / MySQL Workbench
d. Insomnia / PostMan untuk API Testing
e. Google Chrome

2.2 Linux

Linux yang disarankan adalah Ubuntu 20.04 / POP! OS 20.04. Developer dapat memanfaatkan Laravel Homestead atau Docker Stack, atau menggunakan package dari repository.

2.3 MacOS

Developer dapat memanfaatkan Laravel Homestead atau Laravel Homestead, atau menggunakan package Homebrew.

3. Coding Standard

Coding Standard adalah tata cara developer dalam menulis kode. Dalam hal ini, beberapa standar yang harus dipenuhi dapat dilihat pada point 3.1

3.1 Penulisan kode PHP

Penulisan kode PHP harus sesuai dengan standar PSR-2 dan PSR-4. Selain itu, developer diwajibkan mengikuti naming convention yang telah disepakati, yaitu:
a. Controller
Controller menggunakan bentuk tunggal (singular) dalam Bahasa Inggris, tanpa underscore dan spasi, dan menggunakan CamelCase.
Contoh: ArticleController, DocumentController, StudentRegisterController.

b. Database Table
Tabel pada database menggunakan snake_case, lowercase, menggunakan bentuk jamak (plural) dalam Bahasa Inggris.
Contoh: students, documents, student_registers.
c. Database Pivot Table
Pivot Tabel pada database menggunakan snake_case, lowercase, menggunakan bentuk tunggal (singular) dalam Bahasa Inggris.Nama tabel diurutkan sesuai abjad mengikuti tabel referensi nya.
Contoh: Pivot Table antara classes dan student_registers adalah class_studentregister.
d. Table Column
Kolom pada tabel menggunakan lowercase, dan snake_case. Kolom primary key yang menggunakan *auto_increment* adalah id.
Contoh: username, email.
e. Table Column (Foreign Key)
Jika tabel memiliki referensi dengan tabel lain (foreign key), maka nama kolom adalah nama tabel yang direferensikan dalam bentuk singular ditambah '_' dan nama kolomnya.
Sebagai contoh, tabel cities memiliki relasi dengan tabel provinces, dimana satu provinces memiliki banyak cities. Jika primary key pada tabel provinces adalah id, maka kolom referensi pada tabel cities adalah province_id. Jika primary key pada tabel provinces adalah uuid, maka kolom referensi pada tabel cities adalah province_uuid.
f. Model
Model menggunakan bentuk tunggal (singular), tidak menggunakan underscore, dan menggunakan CamelCase. Contoh: StudentRegister.
g. Model Properties
Model Properties menggunakan standar yang sama dengan Table Column. h. Model Method
Menggunakan CamelCase dengan huruf pertama lowercase. Penamaan method disesuaikan dengan value return nya. Jika return value memiliki isi lebih dari satu data (array atau collection), maka penamaan method menggunakan bentuk plural. Sedangkan jika return value hanya memiliki satu nilai atau satu record (contoh: relasi antara city belongs to province), maka penamaan method menggunakan bentuk plural. Contoh: $model->billings(), $model->billingPayments(), $model->cities(), $model->provice().
i. File Blade
File blade menggunakan snake_case.
Contoh: student_detail.blade.php
j. Translation
Untuk tampilan yang berhubungan dengan user, diwajibkan mempersiapkan file translasi untuk Bahasa Indonesia dan Bahasa Inggris.

3.2 HTML CSS JS

Indentasi pada blok kode menggunakan 4 spasi. Standar penulisan kode program pada JavaScript disesuaikan dengan PHP agar seragam.

3.3 Development Focus

Pengembangan aplikasi web oleh Tim IT Yayasan Vitka harus fungsional, mengikuti best practice baik untuk frontend (UI dan styling) maupun backend (performance dan reliability). Selain itu, dikarenakan lebih dari 60% pengguna saat ini mengakses halaman web menggunakan smartphone, maka developer harus menitik beratkan fungsionalitas pada smartphone terutama pada halaman yang berhubungan dengan pengguna.

3.4 ERD

Entity Relationship Diagram, atau ERD, adalah desain dari suatu database dan relasinya. Sebelum memulai pengembangan aplikasi dan atau modul, ERD harus dipersiapkan terlebih dahulu, dan didokumentasikan ke platform yang telah disetujui oleh IT Manager Yayasan Vitka.

4. Documentation Standard

Dokumentasi internal bertujuan untuk meningkatkan readability dari sebuah software. Standar dokumentasi untuk PHP dapat dilihat pada https://github.com/php-fig/fig-standards/blob/master/proposed/phpdoc.md. Untuk bahasa program lainnya, pada dasarnya, setiap function, class, dan atau trait harus memiliki DocBlock yang menjelaskan fungsi dari function / class / trait tersebut.

5. Repository Access & Usage

Repositori, dalam konteks ini mengacu pada Repositori Git milik Yayasan Vitka adalah sentral penyimpanan source code yang juga berfungsi sebagai pencatat perubahan dan alat bantu untuk berkolaborasi, terutama di lingkungan Software Development di Yayasan Vitka. Repositori milik Yayasan Vitka terletak di Github dengan alamat https://github.com/yayasanvitka.

5.1 Hak Akses

Akses kepada repositori diberikan kepada developer yang terlibat di dalam pengembangan software pada Yayasan Vitka dan atau pihak lainnya yang diberikan akses sesuai pertimbangan. Setiap anggota tim tidak boleh melakukan forking tanpa seizin IT Manager Yayasan Vitka, dan

dilarang untuk menunjukkan source code dengan alasan apapun kepada pihak luar kecuali dengan pertimbangan tertentu dan seizin dari IT Manager Yayasan Vitka. Jika developer tidak lagi menjadi anggota tim pengembangan software, maka hak akses kepada repository Yayasan Vitka akan dicabut.

5.2 Branching

Setiap anggota tim pengembangan software dipersilahkan membuat branch pada repository untuk memisahkan working directory antar anggota tim. Branching sangat disarankan untuk menghindari conflict, terutama jika lebih dari satu anggota tim mengerjakan modul yang sama diwaktu yang bersamaan.

5.3 Commit & Push

Commit dilakukan untuk menyimpan perubahan pada repository. Sedangkan push adalah menyimpan commit yang telah dilakukan pada source repository di Github. Setiap commit yang dilakukan harus berupa incremental changes, atau perubahan yang bersifat tahap demi tahap, dan harus memiliki commit message yang jelas dan rinci. Anggota tim tidak diperkenankan untuk melakukan commit secara langsung ke branch master dan release/latest.

5.4 Pull Request & Merging to Master

Pull request bertujuan untuk menggabungkan branch yang sedang dikembangkan dengan branch master. Pull request dilakukan pada branch yang sudah selesai dikembangkan dan sudah melewati test. Pull request harus direview oleh IT Manager terlebih dahulu sebelum di accept.

5.5 Release

Software yang telah melewati test dapat dirilis untuk pengguna. Setiap rilis harus memiliki Changelog detail, dan harus memiliki versi unik. Penomoran versi aplikasi dilakukan oleh Head Development Team dengan mengikuti standar semantic versioning. Rilis ke server dilakukan oleh PIC yang telah ditunjuk.

6. Issue & Tracking

Setiap software yang dikembangkan tidak lepas dari adanya bugs atau issue. Untuk memudahkan melakukan tracking terhadap issue, Tim IT Yayasan Vitka dapat menggunakan fitur issue pada GitHub untuk melaporkan issue yang ditemukan. Adapun guidelines untuk pembuatan issue adalah:
a. Gunakan tag yang sesuai untuk issue yang ada, seperti `[Bug]` untuk bug yang ditemukan pada program, atau `[Feature Request]` untuk permintaan penambahan fitur.
b. Issue harus memiliki data siapa yang menemukan / melakukan request, narasi dari issue, dan ekspektasi dari program.
c. Issue, Feature Request, dan hal – hal lain yang berhubungan dengan pengembangan software harus didokumentasikan di GitHub sesuai dengan repository yang bersangkutan atau pada project di organizational level. d. Komunikasi untuk issue, feature request, dan hal – hal lain yang berhubungan dengan pengembangan software harus dilakukan melalui repository software yang bersangkutan atau pada project di organizational level.

6.1 Eskalasi Issue

Jika issue tidak dapat diselesaikan oleh anggota tim yang bersangkutan, misalnya dikarenakan SOP di lapangan yang tidak sesuai dengan ekspektasi program, anggota tim dapat menghubungi head development melalui GitHub mention pada issue, Microsoft Teams, atau Email untuk di ditindak lanjuti. Eskalasi issue juga dapat dilakukan jika issue tersebut sifatnya urgent dan membutuhkan perubahan langsung pada database.

6.2 Penyelesaian

Issue yang telah diselesaikan harus ditutup, dan melakukan referencing pada commit yang menyelesaikan issue tersebut pada GitHub. Pull request dapat dilakukan untuk menggabungkan commit tersebut pada master branch di repository.

Mobile Development

Revision History

Date	Version	Authors	Description
2021-05-27	1.0	Sudiharyanto Lika	Initial Version

1. Overview

Tim IT Yayasan Vitka adalah tim yang bertanggung jawab terhadap pengembangan software di lingkungan Politeknik Pariwisata Batam, Institut Teknologi Batam, dan Yayasan Vitka. Software yang dikembangkan saat ini difokuskan pada kebutuhan akademik dan penunjang kegiatan akademik lainnya. Teknologi yang digunakan berbasis mobile. Software yang ditulis dengan baik dan rapi memiliki banyak kelebihan. Software tersebut akan memiliki bug lebih sedikit dan akan berjalan lebih efisien dibandingkan dengan software yang tidak ditulis dengan baik. Karena software memiliki life cycle dan Sebagian besar dari life cycle tersebut adalah maintenance, tim pengembang maupun anggota tim yang akan datang akan lebih mudah untuk melakukan maintenance dan merubah program sesuai dengan kebutuhan. Tim IT Yayasan Vitka membuat standarisasi dan panduan singkat untuk memastikan seluruh anggota tim IT yang terlibat memiliki visi yang sama dalam mengembangkan software. Standar dan panduan ini tidaklah konstan, namun dapat berevolusi sesuai dengan kebutuhan dan kesepakatan ke depannya.

2. Standard Tools

2.1 Tools & IDE

Fokus pengembangan pada Yayasan Vitka adalah mobile application berbasis Flutter. Tools utama yang dibutuhkan adalah: a. Flutter >= 2.2.0 (Channel Stable)
b. Dart >= 2.13.0
c. Git >= 2.27.0

Adapun untuk GUI Tools & IDE yang digunakan adalah:
a. Android Studio
b. Visual Studio Code
c. XCode
d. SmartGit untuk git
e. Insomnia untuk testing API

2.2 Linux

Linux yang disarankan adalah Ubuntu 20.04 / POP! OS 20.04.

2.3 MacOS

MacOS yang disarankan adalah yang versi terbaru.

3. Coding Standard

Coding Standard adalah tata cara developer dalam menulis kode. Dalam hal ini, beberapa standar yang harus dipenuhi dapat dilihat pada point 3.1

3.1 Penulisan kode Dart

Penulisan kode Dart harus sesuai dengan standar Effective Dart: Style Flutter. Developer mewajibkan install extension Awesome Flutter Snippets di Visual Studio Code. Selain itu, developer diwajibkan mengikuti naming convention yang telah disepakati, yaitu:
a. Contracts
Contracts menggunakan tunggal (singular) Bahasa Inggris, snake_case, dan lowercase.
Contoh: attendance_contract, profile_contract, dan news_contract.
b. Converters
Converters menggunakan tunggal (singular) Bahasa Inggris,, snake_case, dan lowercase.
Contoh: wrapped_response
c. List
Lists menggunakan tunggal (singular) Bahasa Inggris, snake_case, dan lowercase.
Contoh: attendance_list.
d. Models
Models menggunakan tunggal (singular) Bahasa Inggris, snake_case, dan lowercase.
Contoh: attendance, attendance_detail, dan config.

e. Presenters
Presenters menggunakan tunggal (singular) Bahasa Inggris, snake_case, dan lowercase. Contoh: attendance_presenter, config_presenter, dan profile_presenter.
f. UI & Page
Menggunakan Bahasa Inggris , snake_case, dan lowercase.
Contoh: home, splash, home_view, profile_view, dan attendance_list_view.
g. Utils
Utils menggunakan jamak (plural) Bahasa Inggris, snake_case, dan lowercase.
Contoh: styles, assets, utils, dan widgets.
h. Web Services
Menggunakan Bahasa Inggris , snake_case, dan lowercase.
Contoh: api_services

3.2 Development Focus

Pengembangan aplikasi mobile oleh Tim IT Yayasan Vitka harus fungsional, mengikuti best practice baik dari untuk frontend (UI dan styling) maupun backend (performance dan reliability). Selain itu, dikarenakan banyak pengguna saat ini menggunakan smartphone android, maka developer harus menitik beratkan fungsionalitas pada smartphone android terutama pada halaman yang berhubungan dengan pengguna.

4. Documentation Standard

Dokumentasi internal bertujuan untuk meningkatkan readability dari sebuah software. Untuk bahasa program lainnya, pada dasarnya, setiap function, class, dan atau trait harus memiliki DocBlock yang menjelaskan fungsi dari function / class / trait tersebut.

5. Repository Access & Usage

Repositori, dalam konteks ini mengacu pada Repositori Git milik Yayasan Vitka adalah sentral penyimpanan source code yang juga berfungsi sebagai pencatat perubahan dan alat bantu untuk berkolaborasi, terutama di lingkungan Software Development di Yayasan VItka. Repositori milik Yayasan Vitka terletak di Github dengan alamat https://github.com/yayasanvitka.

5.1 Hak Akses

Akses kepada repositori diberikan kepada developer yang terlibat di dalam pengembangan software pada Yayasan Vitka dan atau pihak lainnya yang diberikan akses sesuai pertimbangan. Setiap anggota tim tidak boleh melakukan forking tanpa seizin IT Manager Yayasan Vitka, dan dilarang untuk menunjukkan source code dengan alasan apapun kepada pihak luar kecuali dengan pertimbangan tertentu dan seizin dari IT Manager Yayasan Vitka.
Jika developer tidak lagi menjadi anggota tim pengembangan software, maka hak akses kepada repository Yayasan Vitka akan dicabut.

5.2 Branching

Setiap anggota tim pengembangan software dipersilahkan membuat branch pada repository untuk memisahkan working directory antar anggota tim. Branching sangat disarankan untuk menghindari conflict, terutama jika lebih dari satu anggota tim mengerjakan modul yang sama diwaktu yang bersamaan.

5.3 Commit & Push

Commit dilakukan untuk menyimpan perubahan pada repository. Sedangkan push adalah menyimpan commit yang telah dilakukan pada source repository di Github. Setiap commit yang dilakukan harus berupa incremental changes, atau perubahan yang bersifat tahap demi tahap, dan harus memiliki commit message yang jelas dan rinci. Anggota tim tidak diperkenankan untuk melakukan commit secara langsung ke branch master dan release/latest.

5.4 Pull Request & Merging to Master

Pull request bertujuan untuk menggabungkan branch yang sedang dikembangkan dengan branch master. Pull request dilakukan pada branch yang sudah selesai dikembangkan dan sudah melewati test. Pull request harus direview oleh IT Manager terlebih dahulu sebelum di accept.

5.5 Release

Software yang telah melewati test dapat dirilis untuk pengguna. Setiap rilis harus memiliki Changelog detail, dan harus memiliki versi unik. Penomoran versi aplikasi dilakukan oleh Head Development Team dengan mengikuti standar semantic versioning. Rilis ke server dilakukan oleh PIC yang telah ditunjuk.

Untuk releasenya menggunakan compile App Bundle

6. Issue & Tracking

Setiap software yang dikembangkan tidak lepas dari adanya bugs atau issue. Untuk memudahkan melakukan tracking terhadap issue, Tim IT Yayasan Vitka dapat menggunakan fitur issue pada GitHub untuk melaporkan issue yang ditemukan. Adapun guidelines untuk pembuatan issue adalah:
a. Gunakan tag yang sesuai untuk issue yang ada, seperti `[Bug]` untuk bug yang ditemukan pada program, atau `[Feature Request]` untuk permintaan penambahan fitur.
b. Issue harus memiliki data siapa yang menemukan / melakukan request, narasi dari issue, dan ekspektasi dari program.
c. Issue, Feature Request, dan hal – hal lain yang berhubungan dengan pengembangan software harus didokumentasikan di GitHub sesuai dengan repository yang bersangkutan atau pada project di organizational level. d. Komunikasi untuk issue, feature request, dan hal – hal lain yang berhubungan dengan pengembangan software harus dilakukan melalui repository software yang bersangkutan atau pada project di organizational level.

6.1 Eskalasi Issue

Jika issue tidak dapat diselesaikan oleh anggota tim yang bersangkutan, misalnya dikarenakan SOP di lapangan yang tidak sesuai dengan ekspektasi program, anggota tim dapat menghubungi head development melalui GitHub mention pada issue, Microsoft Teams, atau Email untuk di ditindak lanjuti. Eskalasi issue juga dapat dilakukan jika issue tersebut sifatnya urgent dan membutuhkan perubahan langsung pada database.

6.2 Penyelesaian

Issue yang telah diselesaikan harus ditutup, dan melakukan referencing pada commit yang menyelesaikan issue tersebut pada GitHub. Pull request dapat dilakukan untuk menggabungkan commit tersebut pada master branch di repository.

Upgrade Notes (Laravel 9 -> 10 & Backpack 5 -> 6)

Laravel

https://laravel.com/docs/10.x/upgrade

1. Update package version

https://github.com/yayasanvitka/saint/commit/59c1c7eee9ba0ce9ba41702d04a12333d6aa2803

2. Update minimum-stability in composer.json

https://github.com/yayasanvitka/saint/commit/db51d2e49b59fa3d7dd754f4a472f639f2179c4e

3. Update $dates to $casts

https://github.com/yayasanvitka/saint/commit/66622ead9d09e3eee61fb4bbce805cd32762f057

Backpack

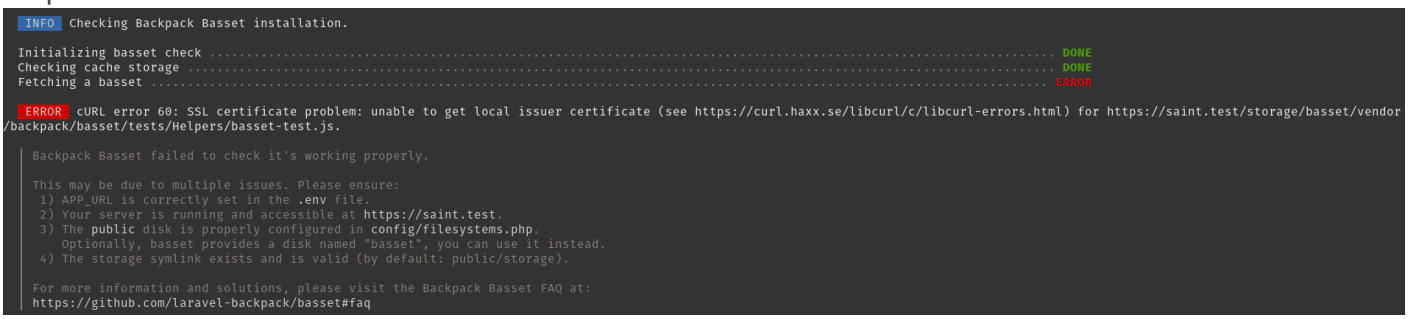
https://backpackforlaravel.com/docs/6.x/upgrade-guide

1. Step 3.2

Choose Tabler and No premium addons
Add basset in filesystems.php
The code is

'basset' => ['driver' => 'local', 'root' => public_path('storage'), 'url' => env('APP_URL') . '/storage', 'visibility' => 'public', 'throw' => false,],

Skip if error



2. Skip step 4-7

3. Skip step 9-15

4. Skip step 22

5. Step 25

Choose No to publish public/packages

6. Custom CSS

Put it into mix_styles in ui.php and webpack.mix.js

7. Update layout to vertical & enable useFluidContainers in theme-tabler.php

https://github.com/yayasanvitka/saint/commit/a58ae10287b071a1c3a49ad17103369daf3ecac3

8. Fix layout.blade.php in resource/views/errors

https://github.com/yayasanvitka/saint/commit/d764e1284460ef577179353b9d716ef4a77094aa

9. Temporary fix for multiple nested menu

https://github.com/yayasanvitka/saint/commit/d35ab9edabcc8a1863706cf0fe715e134a226029

10. Re-install elfinder

rm -rf resources/views/vendor/elfinder
php artisan backpack:filemanager:install

11. Re-publish backupmanager and add backup.blade.php

php artisan vendor:publish --provider="Backpack\BackupManager\BackupManagerServiceProvider" --tag=backup-config --tag=lang

https://github.com/yayasanvitka/saint/commit/a176fa99829a075affebd43a64b09463c74ea942

https://github.com/yayasanvitka/hrms/commit/a6ef11cdb04e3f109965ebe028d6477c2e8cd14c

https://github.com/yayasanvitka/hrms/commit/4e701ba893a2cd9e3c5f35baa89b7f13a40e9fdb

12. Fix error gravatar

https://github.com/yayasanvitka/saint/commit/128f9ab3955054d6c950b6544cbb8e9617edb11f

13. Remove unused js code

https://github.com/yayasanvitka/saint/commit/231dadab2edbf92f03f2a16600769375abd8ffdc

14. Fix MyAccountController

https://github.com/yayasanvitka/saint/commit/05b57f1e5a897d562ce6c0b48de367cfb3d55db7

15. Fix reorder menu

https://github.com/yayasanvitka/saint/commit/65ce077aa2f21f2db73268230c16cf307fc8756f

16. Solution for readonly field does not show gray background

Tambah `'class' => 'form-control bg-gray-600'` di dalam `attributes`

17. Update `blank.blade.php` and move to theme-tabler folder

https://github.com/yayasanvitka/saint/commit/2c9114eaea4dc1437aa16743127608e665df747a

Deploy

```
256  pacal
257  cd /var/www/saint
258  git status
259  git checkout composer.json
260  artisan down
261  artisan backup:run
262  git pull
263  ls
264  pam
265  pads --class=DataUpdater
266  cu --no-dev
267  php artisan basset:cache
268  pacal
269  artisan up
270  artisan down
271  npm install
272  npm run prod
273  ls
274  cd storage
275  ls
276  cd app
277  ls
278  cd public
279  ls
280  cd ..
281  nano .env
282  cd ..
283  nano .env
284  cd ../hrms
285  cd ../saint
286  history | grep basset
287  cd /var/www/saint
288  artisan list
289  artisan basset:install
290  ls -lah public
291  artisan basset:cache
292  artisan basset:internalize
293  ls -lah storage/app/public
294  chmod -R 777 storage/
295  pacal
296  artisan up
297  history | grep basset
```

1. Run `npm install`
2. Run `npm run prod`
3. Run `php artisan basset:install`

4. Run `php artisan basset:cache`

5. Run `chmod -R 777 public/storage`

Rules Pint

Changelog

Version 1.0.0 - August 9, 2024

- **Initial Release:** Added initial rules after discussion with the team.

Rules in pint.json


```
{
    "preset": "laravel",
    "rules": {
        "blank_line_after_namespace": true,
        "no_unused_imports": true,
        "no_useless_else": true,
        "no_trailing_comma_in_singleline_function_call": true,
        "single_line_after_imports": true,
        "single_quote": true,
        "combine_consecutive_unsets": true,
        "phpdoc_indent": true,
        "phpdoc_no_access": true,
        "phpdoc_no_package": false,
        "phpdoc_no_useless_inheritdoc": true,
        "phpdoc_scalar": true,
        "phpdoc_single_line_var_spacing": true,
        "phpdoc_summary": true,
        "phpdoc_trim": true,
        "phpdoc_trim_consecutive_blank_line_separation": true,
        "phpdoc_inline_tag_normalizer": false,
        "phpdoc_types": true,
        "phpdoc_var_without_name": true,
        "space_after_semicolon": true,
        "function_declaration": {
            "closure_fn_spacing": "none",
            "closure_function_spacing": "one"
        },
        "class_attributes_separation": {
            "elements": {
                "const": "one",
                "method": "one",
                "property": "none",
```

```
            "trait_import": "none"
        }
    },
    "visibility_required": {
        "elements": [
            "method",
            "property"
        ]
    },
    "array_syntax": {
        "syntax": "short"
    },
    "concat_space": {
        "spacing": "none"
    },
    "curly_braces_position": {
        "classes_opening_brace": "next_line_unless_newline_at_signature_end"
    },
    "phpdoc_order": {
        "order": [
            "param",
            "throws",
            "return"
        ]
    },
    "return_type_declaration": {
        "space_before": "none"
    },
    "ternary_to_null_coalescing": true,
    "void_return": false,
    "no_superfluous_phpdoc_tags": false,
    "ordered_traits": false,
    "new_with_parentheses": true
  }
}
```