

**UJIAN AKHIR SEMESTER
SERVICE ORIENTED ARCHITECTURE**



Oleh:

Teguh Maulana Sidik (41155050190071)

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS LANGLANGBUANA
BANDUNG
2023**

BAGIAN 1

1. Apa bedanya Google Services dan Amazon Web Services?

Jawab:

- a. Wilayah Ketersediaan (*Availability Zone*)
 - AWS memiliki 66 *availability zone* dengan 12 lainnya yang masih *on going*.
 - Google Cloud Platform telah tersedia di 20 wilayah di seluruh dunia dengan 3 lainnya masih *on-going*.
- b. Layanan (*Service*)
 - AWS menawarkan sekitar 200+ layanan.
 - Google Cloud menawarkan sekitar 60+ layanan.
- c. Harga
 - AWS menyediakan percobaan gratis menggunakan salah satu layanan yang mereka sediakan selama satu tahun.
 - Google Cloud Platform menyediakan percobaan gratis dengan menggunakan skema pemberian *credits* sebanyak \$300 yang dapat digunakan untuk layanan apapun pada Google Cloud Platform.
- d. Stack
 - AWS memiliki pilihan region terbanyak (14 region) di antara penyedia lain. Meski begitu, region terdekat dengan Indonesia adalah Singapura.
 - Google Cloud Platform memiliki pilihan region yang merata (10 region) dengan region terdekat adalah region Singapore (SouthEast Asia).

2. Sebutkan apa saja aplikasi yang ada pada Amazon Web Services!

Jawab:

- a. Migratrion (Migrasi)

Contoh layanan migration adalah DMS (Database Migration Service) dan Snowball
- b. Cloud Computing (Komputasi Awan)

Contoh layanan Cloud Computing adalah EC2 (Elastic Compute Cloud), Elastic Beanstalk dan Lightsail.
- c. Storage (Penyimpanan)

Contoh layanan Storage Amazon Elastic Block Store (EBS), AWS Storage Gateway, AWS Backup dsb.
- d. Security services (Layanan keamanan)

Contoh layanan Security adalah Amazon Cognito, Amazon GuardDuty AWS Identify and Access Management dsb.
- e. Database services (Layanan database)

Contoh layanan Database adalah Amazon Aurora, Amazon DynamoDB, Amazon ElastiCache dsb.
- f. Analytics (Analitik)

Contoh layanan Analytics adalah Amazon Athena, Amazon CloudSearch, Amazon Kinesis dsb.

g. Internet of Things

Contoh layanan Internet of Things adalah Amazon FreeRTOS, AWS IoT 1-Click, AWS IoT Core dsb.

h. Manajemen Keuangan Cloud

Contoh layanan Manajemen Keuangan Cloud adalah AWS Budgets, AWS Cost Explorer, Savings Plans dsb.

i. Machine Learning

Contoh layanan Machine Learning adalah Amazon Comprehend, Amazon Polly, Amazon Rekognition dsb.

3. Apa itu framework Laravel?

Jawab:

Laravel merupakan suatu Framework dari PHP, Framework PHP adalah suatu kerangka yang dibuat untuk membangun atau membuat sebuah aplikasi dengan menggunakan PHP sebagai bahasa pemrogramannya. Kehadiran framework Laravel menjadikan bahasa pemrograman PHP menjadi lebih powerful. Perlu kita ketahui bahwa kehadiran framework Laravel selalu menghadirkan fitur-fitur terbaru dibandingkan framework lainnya.

4. Laravel menggunakan arsitektur MVC, yang merupakan turunan dari arsitektur orientasi objek. Apa itu MVC?

Jawab:

MVC merupakan suatu arsitektur aplikasi dengan cara memisahkan aplikasi menjadi tiga bagian yang saling terhubung dan sesuai dengan bagaimana informasi disajikan kepada pengguna dan didapat dari pengguna.

MVC terdiri dari 3 bagian, yaitu:

- a. **Model** : Adalah bagian yang memiliki fungsi-fungsi untuk mengelola database.
- b. **View** : adalah bagian yang bertugas menampilkan data
- c. **Controller** : adalah bagian yang menjadi penghubung antara model dan view. Controller memiliki perintah-perintah yang berfungsi untuk memproses bagaimana data ditampilkan dari Model ke View, atau bagaimana data dikirimkan dari View ke Model.

5. Restful API apa yang ada di Laravel?

Jawab:

- a. **GET** : Berfungsi untuk membaca data atau sumber dari Rest Server.
- b. **POST** : Berfungsi untuk membuat data/sumber baru di Rest Server.
- c. **DELETE** : Berfungsi untuk menghapus data / sumber dari Rest Server.
- d. **PUT** : Berfungsi untuk memperbaharui data/sumber di Rest Server.
- e. **OPTION** : Berfungsi untuk mendapatkan operasi yang didukung pada sumber dari Rest Server.

6. Microservice apa saja yang ada di Laravel?

Jawab:

- a. **View** : Berfungsi menampilkan tampilan kepada user.
- b. **Model** : Berfungsi untuk mengolah dan pengambilan data pada *database*.
- c. **Controller** : Berfungsi mengatur perpindahan data pada *model* dan *view*.
- d. **Router** : Mengatur *controller* dan *function* mana yang dipanggil Ketika mengakses *url* tertentu.
- e. **Middleware** : Mengatur dan melakukan *validasi* diantara *router* dan *controller*.
- f. **Package** : *Service* untuk membuat *Code Modular*.

7. Apa bedanya instalasi Laravel di server lokal (XAMPP) dengan di *cloud hosting*?

Jawab:

Pada *local server* untuk menjalankan Laravel *project* dapat dilakukan dengan menjalankan command `php artisan serve`. Sedangkan pada *cloud hosting* harus menggunakan *web server* seperti *nginx* atau *apache* dan mengarahkan *root* ke folder *public* pada *project laravel*.

8. Apa itu Metode Pengembangan Scrum?

Jawab:

SCRUM adalah salah satu metode rekayasa perangkat lunak dengan menggunakan prinsip - prinsip pendekatan AGILE, yang bertumpu pada kekuatan kolaborasi tim, incremental product dan proses iterasi untuk mewujudkan hasil akhir. Scrum sendiri bukan satu-satunya metode yang menggunakan pendekatan AGILE. Mungkin kita juga pernah mendengar metode Extreme Programming (XP) yang juga menggunakan pendekatan AGILE dalam rekayasa perangkat lunak. Masing-masing metode memiliki fokus atau penekanan yang berbeda yang tentu saja dapat dikombinasikan untuk menghasilkan proses yang optimal

9. Apa itu GitHub dalam software development (pengembangan perangkat lunak)?

Jawab:

GitHub adalah situs web dan layanan berbasis cloud yang membantu developer menyimpan dan mengelola kode yang dibuat, serta melacak dan mengontrol perubahan pada kode tersebut. GitHub mencakup dua prinsip, yaitu version control dan Git. Version control membantu developer melacak dan mengelola perubahan pada kode proyek perangkat lunak. Seiring pertumbuhan proyek perangkat lunak, version control menjadi penting.

10. Bagaimana membuat Site Map XML dengan Laravel?

Jawab:

- a. Instal proyek laravel baru.

Membuat proyek laravel dengan versi laravel terbaru. Buka terminal dan masuk ke direktori dimana program ingin di instal project laravel tersebut. Kemudian instal laravel dengan perintah **composer create-project laravel/laravel sitemap**. Dengan perintah tersebut kita akan dibuatkan project laravel baru dengan nama *project sitemap*.

- b. Membuat *database* baru di phpMyAdmin dengan nama yang bersangkutan dengan proyek (di proyek ini, Dibuat database dengan nama sitemap).
- c. Sesuaikan **DB_DATABASE** dengan nama database yang baru dibuat di langkah nomor 2.
- d. Buat file *model* dan *migration* Post dengan perintah **php artisan make:model Post -m**.
- e. Buka file 2023_01_create_posts_table.php atau file migration yang telah dibuat dengan command pada langkah nomor 4. Edit script di file tersebut menjadi seperti di bawah ini.

```
<?php
use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;
class CreatePostsTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('posts', function (Blueprint $table) {
            $table->id();
            $table->string('title');
            $table->string('slug');
            $table->longText('body');
            $table->timestamps();
        });
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::dropIfExists('posts');
    }
}
```

- f. Jalankan perintah **php artisan migrate** untuk membuat table atau migrasi *file migration* di folder **database>migration** ke database phpMyAdmin.

- g. Buat *file factory* untuk membuat data *dummy* dengan perintah **php artisan make:factory PostFactory -m Post**.
- h. Buka file *PostFactory* di dalam folder **database>factory**, kemudian ubah *script* di file tersebut menjadi seperti pada gambar di bawah ini.

```
<?php
namespace Database\Factories;
use App\Models\Post;
use Illuminate\Database\Eloquent\Factories\Factory;
use Illuminate\Support\Str;
class PostFactory extends Factory
{
    /**
     * The name of the factory's corresponding model.
     *
     * @var string
     */
    protected $model = Post::class;

    /**
     * Define the model's default state.
     *
     * @return array
     */
    public function definition()
    {
        $title = $this->faker->sentence;
        $slug = Str::slug($title);
        return [
            'title' => $title,
            'slug' => $slug,
            'body' => $this->faker->paragraph(10)
        ];
    }
}
```

```

$ php artisan tinker
> App\Models\Post::factory()->count(10)->create();
=> Illuminate\Database\Eloquent\Collection {#3370
  all: [
    App\Models\Post {#3374
      title: "Eos nostrum qui nulla aut rerum harum ducimus.",
      slug: "eos-nostrum-qui-nulla-aut-rerum-harum-ducimus",
      body: "Non fuga sit tempore placeat. Molestias occaecati sit iusto cupiditate. Ratione ex et quo fugit molestiae omnis asperiores. Et quia tempore officiis quos. Est blanditiis natus eius omnis perspiciatis. Harum laborum esse vel. Illo fuga aliquid error maxime. Quis aut quibusdam consequatur tenetur quia.",
      updated_at: "2021-01-05 02:31:37",
      created_at: "2021-01-05 02:31:37",
      id: 51,
    },
  ],
}

```

- i. Kemudian buka kembali terminal dan jalankan perintah **php artisan tinker**. Lalu masukan perintah **App\Models\Post::factory()->count(10)->create();** guna membuat *data dummy* sebanyak 10 data.
- j. Instal *package* spatie laravel sitemap dengan menjalankan perintah **composer require spatie/laravel-sitemap** pada terminal.
- k. Jalankan perintah **php artisan vendor:publish --provider="Spatie\Sitemap\SitemapServiceProvider" --tag=config** untuk menyalin file sitemap.php ke dalam folder *config*.
- l. Buka file **web.php** yang berada di folder *routes*. Kemudian ubah *script* di file tersebut menjadi seperti gambar di bawah ini.

```

<?php
use Illuminate\Support\Facades\Route;
use App\Models\Post;
use Spatie\Sitemap\Sitemap;
use Spatie\Sitemap\Tags\Url;

/*
|-----
| Web Routes
|-----
|
| Here is where you can register web routes for your application.
| routes are loaded by the RouteServiceProvider within a group which
| contains the "web" middleware group. Now create something awesome.
|
*/

Route::get('/', function () {
    return view('welcome');
});

Route::get('/sitemap', function() {
    $sitemap = Sitemap::create()
        ->add(Url::create('/about-us'))
        ->add(Url::create('/contact-us'));

    $post = Post::all();
    foreach ($post as $post) {
        $sitemap->add(Url::create("/post/{ $post->slug }"));
    }
    $sitemap->writeToFile(public_path('sitemap.xml'));
});

```


- m. Langkah terakhir jalankan perintah **php artisan serve** kemudian akses ke url **127.0.0.1:8000/sitemap** maka ketika kita cek di dalam folder *public* terdapat file baru dengan nama **sitemap.xml**

```
public > sitemap.xml
1 <?xml version="1.0" encoding="UTF-8"?>
2 <urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9" xmlns:xhtml="http://www.w3.org/1999/xhtml">
3   <url>
4     <loc>http://127.0.0.1:8090/about-us</loc>
5     <lastmod>2021-01-05T07:27:01+00:00</lastmod>
6     <changefreq>daily</changefreq>
7     <priority>0.8</priority>
8   </url>
9   <url>
10    <loc>http://127.0.0.1:8090/contact-us</loc>
11    <lastmod>2021-01-05T07:27:01+00:00</lastmod>
12    <changefreq>daily</changefreq>
13    <priority>0.8</priority>
14  </url>
15  <url>
16    <loc>http://127.0.0.1:8090/post/alias-recusandae-asperiores-rerum-aut-illo-ea</loc>
17    <lastmod>2021-01-05T07:27:01+00:00</lastmod>
18    <changefreq>daily</changefreq>
19    <priority>0.8</priority>
20  </url>
```

BAGIAN 2

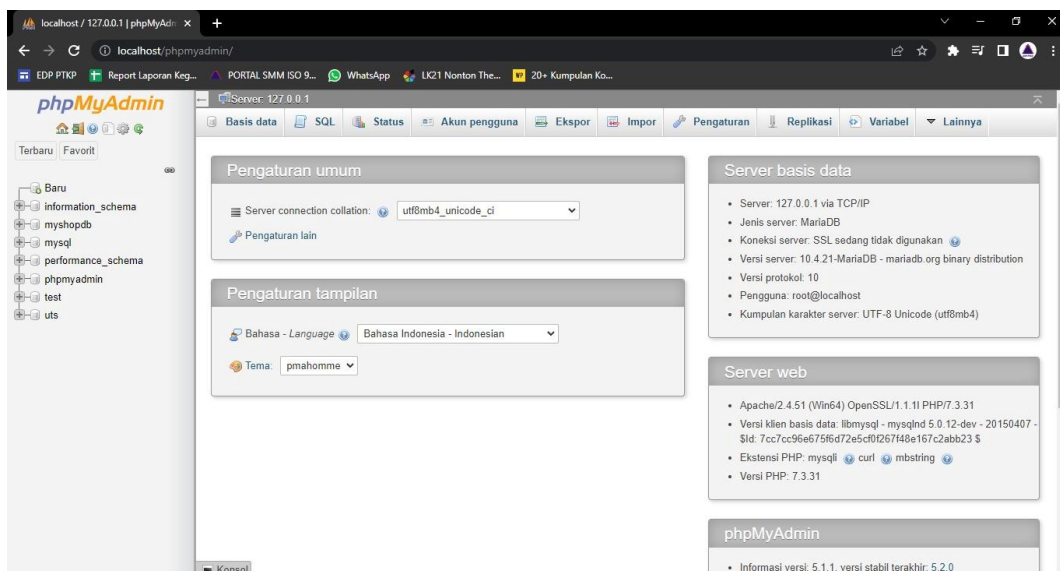
Github: <https://github.com/widianingp/Pasar-Malam-versi-1.0>

Tahapan *Sprint 0*: Perencanaan

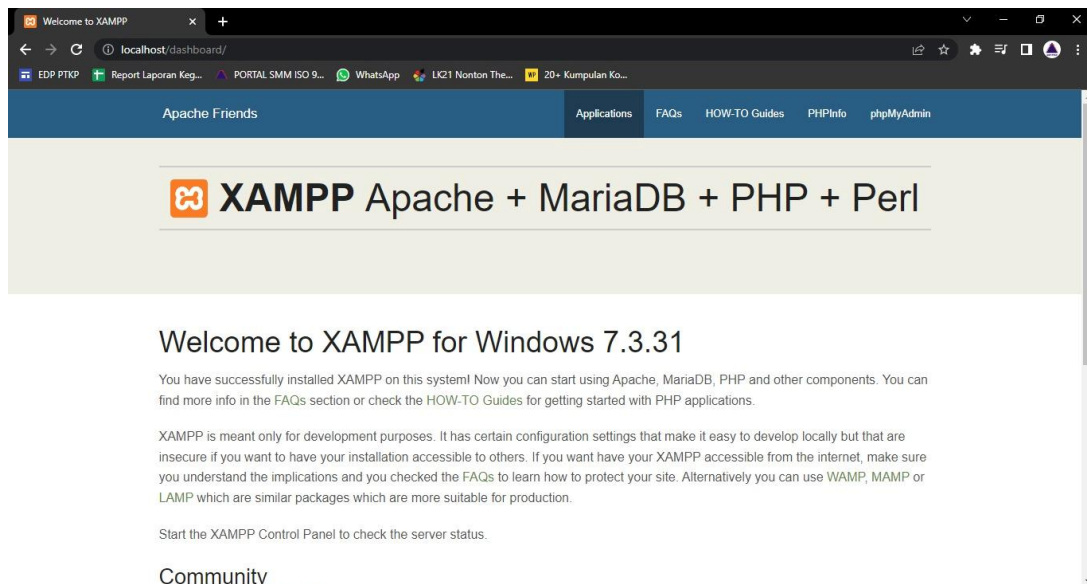
0 Day 1, Sprint 0 : 1jam, Persiapan

- 0.1. Tim Leader menyediakan XAMPP di server lokal. Prototype 0.0.1 : Screenshot page test Apache dan phpmySQL. Simpan Prototype 0.0.1 di GitHub.

0.0.1 Screenshot phpMySql



0.0.2 Screenshot Apache



0.2. Tim *Leader* membentuk 3 kelompok pengembang dan menugaskan untuk instalasi *Laravel Aimeos*. Daftar Kelompok Pengembang dan tugasnya simpan di *Github* sebagai *Prorotype* 0.0.2. Kelompok pengembang ini terdiri dari:

0.2.1. Kelompok 1 (pengembang *backend* dan *Controller*),

Kelompok 1 (Back End dan Controller)	
No	Nama Pengembang
1	Widi Aning Pangesti

0.2.2. Kelompok 2 (pengembang *backend* dan *Model*),

Kelompok 2 (Back End dan Model)	
No	Nama Pengembang
1	Teguh Maulana Sidik

0.2.3. Kelompok 3 (pengembang *frontend* dan *View*).

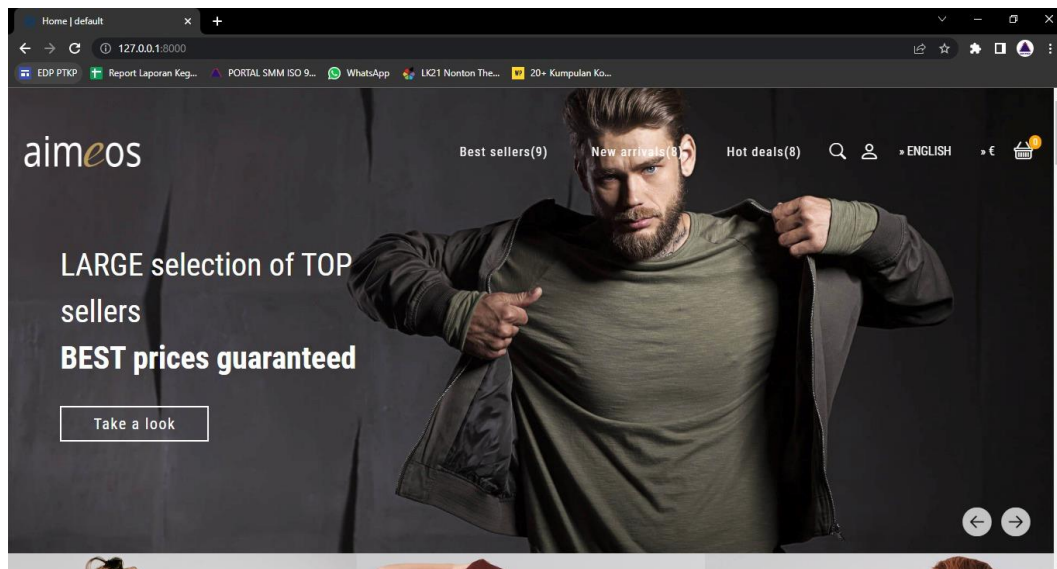
Kelompok 3 (Front End dan View)	
No	Nama Pengembang
1	Teguh Maulana Sidik
2	Widi Aning Pangesti

Tahapan *Sprint* 1: Bangun Dasar

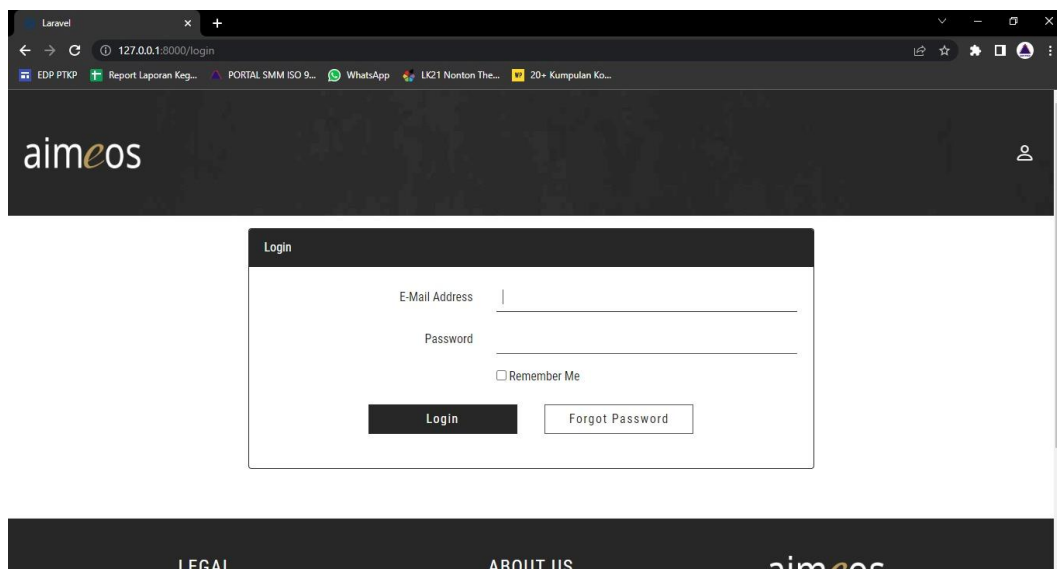
1. *Day 1, Sprint 1*: 1 jam. Instalasi dan konfigurasi *Laravel* dan mempelajari script *Aimeos* secara paralel

1.1. Kelompok 1 melakukan instalasi *Laravel. Prototype 0.1.1*. Screenshot tampilan *Laravel* yang berhasil diinstall berupa tampilan halaman awal, halaman *login*, *dashboard*. Simpan *Prototype 0.1.1*. di *Github*.

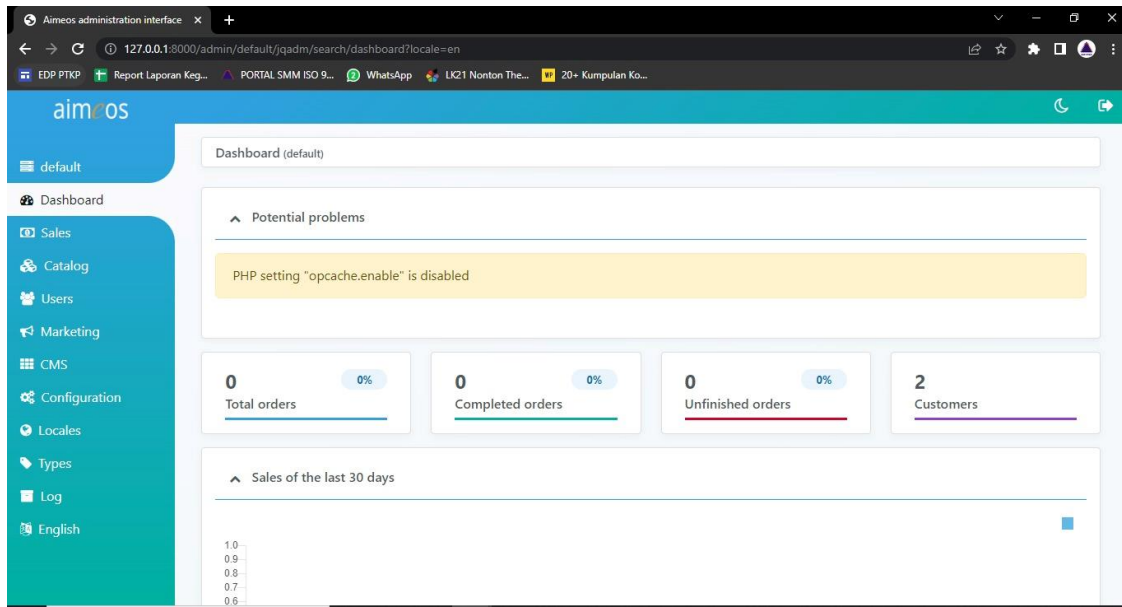
1.1.1 Screenshot Halaman Awal



1.1.2 Screenshot Halaman Login



1.1.3 Screenshot Dashboard



1.2. Kelompok 2
mempelajari *database* dan *script* di *folder Model* dari <https://github.com/aimeos/aimeos-laravel>. Siap melakukan Sprint 2

1.3. Kelompok 3
mempelajari *frontend* dan *script* di *folder View* dari <https://github.com/aimeos/aimeos-laravel>. Siap melakukan Sprint 2

Tahapan Sprint 2: Bangun Badan

2. Day 2, Sprint 2: 3 jam. Instalasi dan konfigurasi Aimeos secara bersama dan terintegrasi.

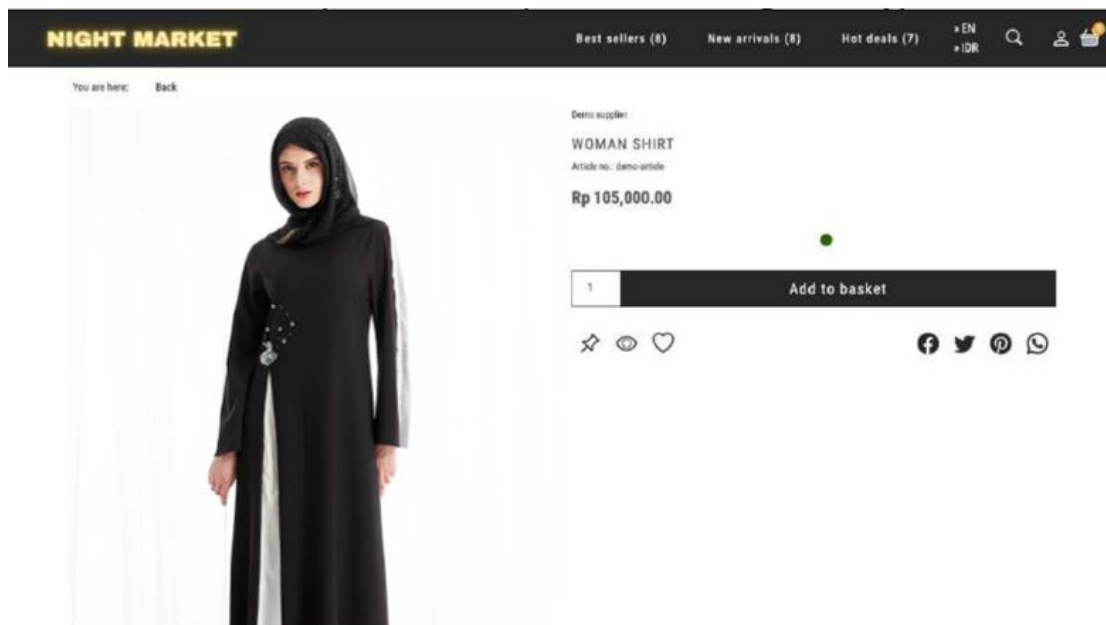
2.1. Kelompok 2
menginstal database dan script di folder Model dari <https://github.com/aimeos/aimeos-laravel> di atas Prototipe 1.1. Setelah berhasil, membuat screenshot phpMyAdmin disimpan di Github sebagai Prototype 0.2.1.

2.1.1 Screenshot phpMyAdmin

Table	Action	Rows	Type	Collation	Size	Overhead
failed_jobs	✱ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 K1B	-
madmin_cache	✱ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	96.0 K1B	-
madmin_cache_tag	✱ Browse Structure Search Insert Empty Drop	26	InnoDB	utf8mb4_unicode_ci	32.0 K1B	-
madmin_job	✱ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	48.0 K1B	-
madmin_log	✱ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 K1B	-
madmin_queue	✱ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32.0 K1B	-
migrations	✱ Browse Structure Search Insert Empty Drop	4	InnoDB	utf8mb4_unicode_ci	16.0 K1B	-
mshop_attribute	✱ Browse Structure Search Insert Empty Drop	19	InnoDB	utf8mb4_unicode_ci	128.0 K1B	-
mshop_attribute_list	✱ Browse Structure Search Insert Empty Drop	61	InnoDB	utf8mb4_unicode_ci	64.0 K1B	-
mshop_attribute_list_type	✱ Browse Structure Search Insert Empty Drop	4	InnoDB	utf8mb4_unicode_ci	80.0 K1B	-
mshop_attribute_property	✱ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	64.0 K1B	-
mshop_attribute_property_type	✱ Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	80.0 K1B	-
mshop_catalog	✱ Browse Structure Search Insert Empty Drop	12	InnoDB	utf8mb4_unicode_ci	80.0 K1B	-
mshop_catalog_list	✱ Browse Structure Search Insert Empty Drop	4	InnoDB	utf8mb4_unicode_ci	64.0 K1B	-
mshop_catalog_list_type	✱ Browse Structure Search Insert Empty Drop	57	InnoDB	utf8mb4_unicode_ci	96.0 K1B	-
mshop_cms	✱ Browse Structure Search Insert Empty Drop	5	InnoDB	utf8mb4_unicode_ci	60.0 K1B	-
mshop_cms_list	✱ Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_unicode_ci	64.0 K1B	-
mshop_cms_list_type	✱ Browse Structure Search Insert Empty Drop	1	InnoDB	utf8mb4_unicode_ci	64.0 K1B	-
mshop_coupon	✱ Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_unicode_ci	80.0 K1B	-
mshop_coupon_code	✱ Browse Structure Search Insert Empty Drop	3	InnoDB	utf8mb4_unicode_ci	96.0 K1B	-
mshop_coupon_code_type	✱ Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_unicode_ci	96.0 K1B	-

2.2. Kelompok 3
 menginstal Frontend dan script di folder View dari <https://github.com/aimeos/aimeos-laravel> di atas Prototipe 1.1. Setelah berhasil, membuat screenshot tampilan web disimpan di Github sebagai Prototype 0.2.2.

2.2.1 Screenshot tampilan web

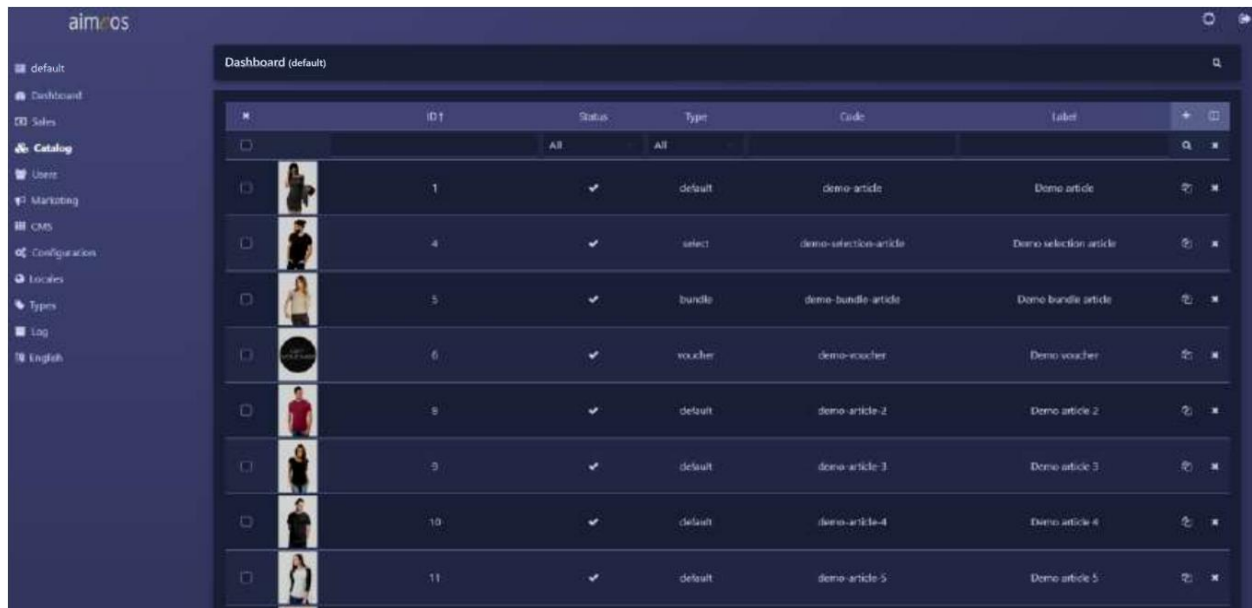


2.3. Kelompok 1
 menginstal script di folder Controller dari <https://github.com/aimeos/aimeos-laravel> di atas Prototipe 1.1. Setelah berhasil,

membuat screenshot testing Dashboard dan di Github sebagai Prototype 0.2.3.

isi query disimpan

2.3.1 Screenshot testing Dashboard



2.3.2 Query

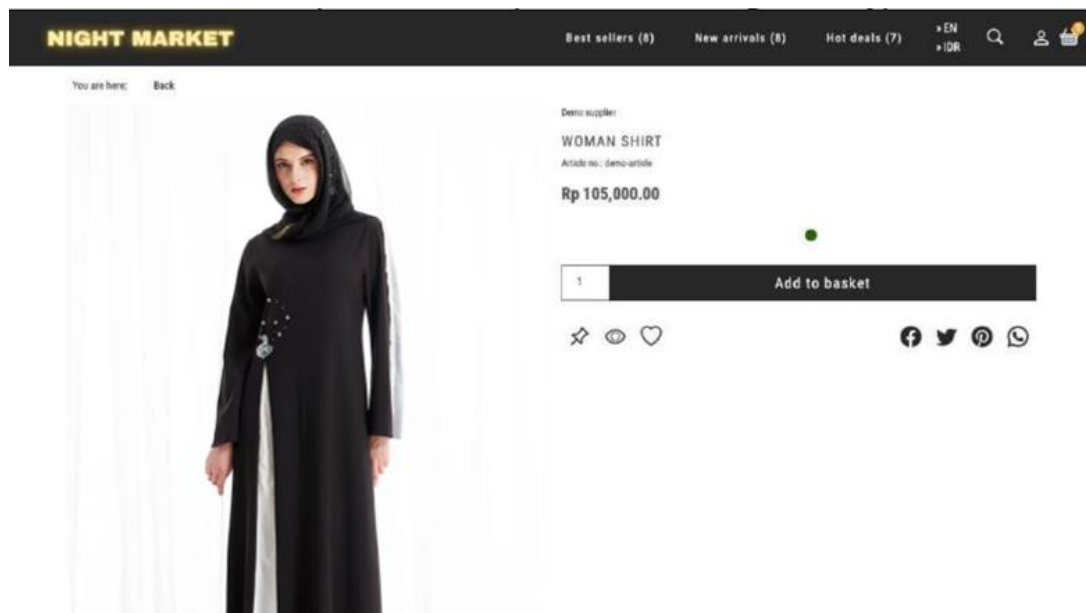
```
/**
 * Adds product IDs for filtering
 *
 * @param array|string $prodIds Product ID or list of IDs
 * @return \Aimeos\Controller\Frontend\Product\Iface Product controller for fluent interface
 * @since 2019.04
 */
public function product( $prodIds ) : Iface
{
    if( !empty( $prodIds ) && ( $ids = array_unique( $this->validateIds( (array) $prodIds ) ) ) !== [] ) {
        $this->addExpression( $this->filter->compare( '==', 'product.id', $ids ) );
    }

    return $this;
}
```

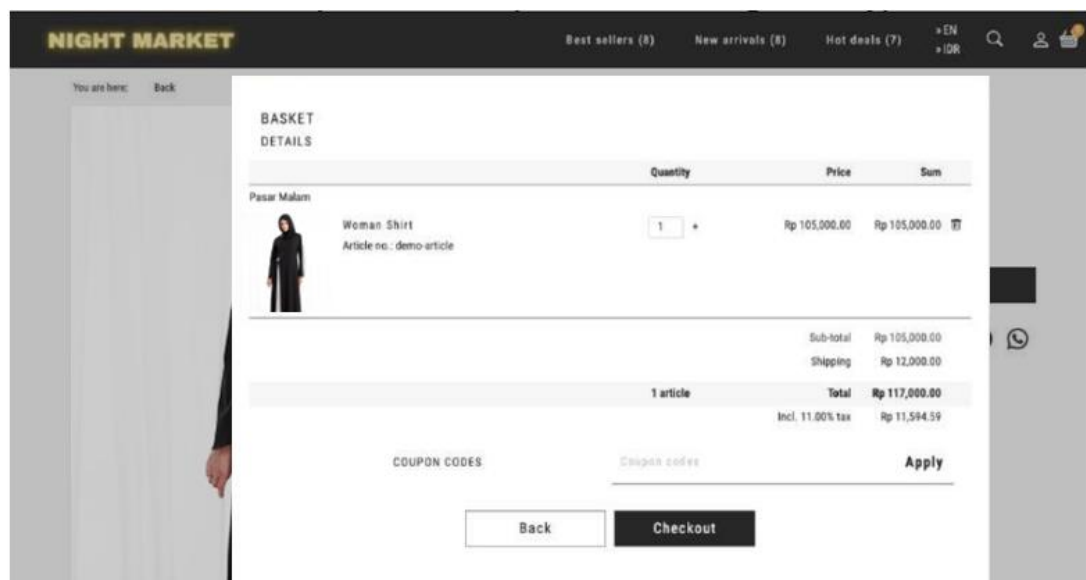
2.4. Setelah tahap 2.3. selesai, Tim Leader melakukan evaluasi fitur. Semua fitur dites dan dipastikan berjalan dengan baik. Setelah tes berhasil, maka Prototype ini disetujui dengan nama Prototype Market Place Pasar Malam versi 1.0. disimpan di Github., termasuk screenshot Prototype 1.0. dengan semua fiturnya.

2.4.1 Test Order

Detail Product



Detail Keranjang



Form Pengiriman

NIGHT MARKET

1. Basket
2. Address
3. Delivery
4. Payment
5. Summary

ADDRESS

Fields with an * are mandatory

Billing address

Delivery address

☒ new address

☒ like billing address

☐ new address

Salutation

First name*

Last name*

Company

Street*

First name

Last name

Company

Street

Pilihan metode pembayaran

NIGHT MARKET

1. Basket
2. Address
3. Delivery
4. Payment
5. Summary

PAYMENT

Please choose your payment method

☐ INVOICE

Pay by invoice within 14 days

Please transfer the money within 14 days to BIC: XXX, IBAN: YYY

☐ DIRECT DEBIT

Payment via your bank account

The money will be collected from your bank account within 1-3 days

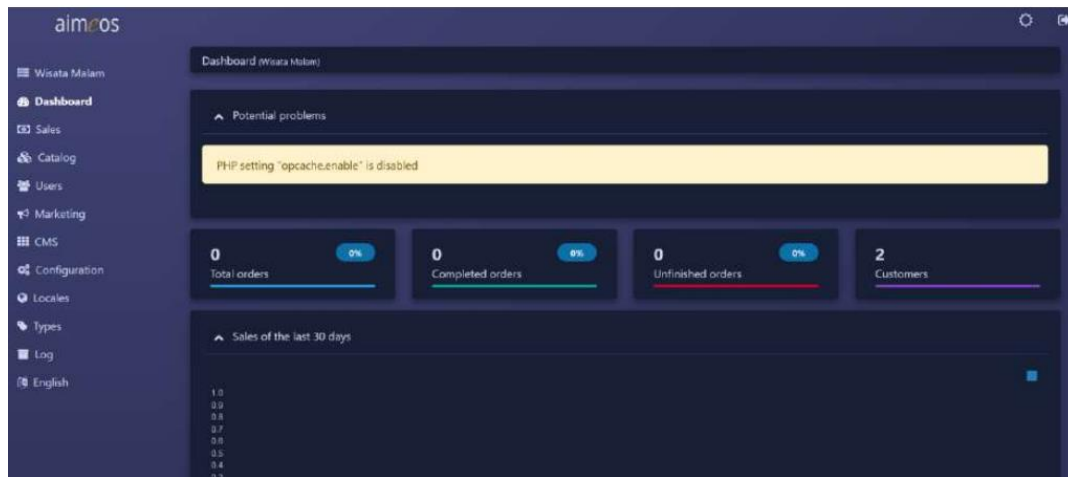
☒ PAYPAL

Payment via PayPal

Easy and secure payment with your PayPal account

2.4.2 Test Dashboard Admin

Dashboard



Menambah Catalog – Product

The screenshot shows the 'Catalog' section with the 'Products' tab selected. The 'Product: New (Wisata Malam)' form is open, displaying the 'Basic' tab. The form includes a sidebar with options: Basic, Media, Texts, Prices, Stocks, and Categories. The main content area shows the following fields:

Field	Value	Status
Status	Enabled	✓
Type	Article	✓
SKU	product-test	
Label	Product Test	

Buttons for '?', 'Cancel', and 'Save' are located at the top right of the form.

The screenshot shows the 'Product: New (Wisata Malam)' form in the 'Catalog' section, now displaying the 'Media' tab. The sidebar options remain the same. The main content area shows a 'Media' section with a 'preview.png (default)' image of a person in a black dress. To the right of the image are the following fields:

Field	Value	Status
Status	Enabled	✓
Type	Standard	✓
Title	logo-pasar-malam-removebg-pre	
Language	All	✓

Buttons for '?', 'Cancel', and 'Save' are located at the top right of the form. An 'Advanced' section is visible at the bottom.

Catalog

Products Categories Attributes Suppliers

Product: New (Warta Malang)

Basic Media Texts **Prices** Stocks Categories

Allow custom price

1 = 300000 + 30 IDR (default)

Tax rate in % 20 Status Enabled

Actual price 300000 Currency Indonesian rupiah

Granted rebate 30 Min. quantity 1

Shipping/item 30

Advanced

Catalog

Products Categories Attributes Suppliers

Product: New (Warta Malang)

Basic Media Texts Prices **Stocks** Categories

Stock level Back in stock Delivery within

0 + 100 January 23, 2023 12:00 AM Time frame (optional)

Catalog

Products Categories Attributes Suppliers

Product: New (Warta Malang)

Basic Media Texts Prices Stocks **Categories**

Default

Best sellers (demo-best)

Home (home)

Hot deals (demo-deals)

New arrivals (demo-new)

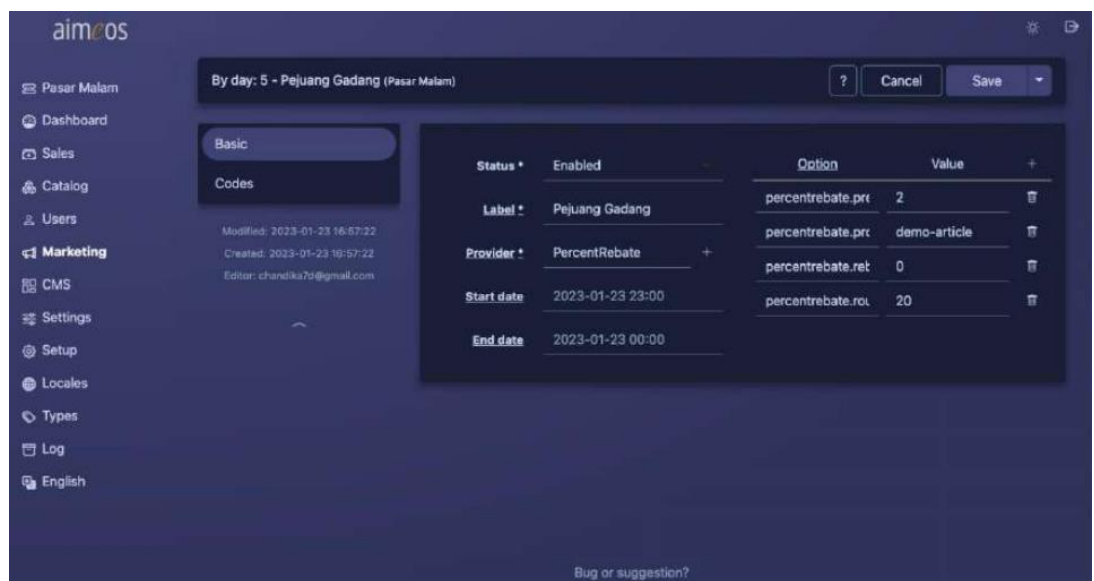
Promotion

Hasih menambah catalog

16 ✓ default product-test Product Test

Page 1 of 1 25

Menambah Voucher



Hasil menambah voucher

