LAMPIRAN

Lampiran-1 Login

```
<?php

namespace
App\Http\Controllers\Api;

use
App\Http\Controllers\Controller
;
use Exception;
use
Facade\FlareClient\Http\Respons
e;
use Illuminate\Http\Request;
use
Illuminate\Support\Facades\Auth
;
use
Illuminate\Support\Facades\DB;

class LoginRegisterController
extends Controller
{
    public function
login(Request $request)
    {
        $username = $request-
>username;
        $pwd   = $request-
>password;
        if
(Auth::attempt(['username' =>
$username, 'password' => $pwd,
'aktif' => 1])) {
            $users =
DB::table('users')
                -
>select("users.*", "users.id as
userid")
                -
>where('username', $username)
                ->first();
            return Response()-
>json($users);
        } else {
            return Response()-
>json(["error" => true]);
        }
    }

    public function
updateProfil(Request $request)
    {
        $id = $request->id;
        $username = $request-
>username;
        $password = $request-
>password;
        $nomor = $request-
>nomor;
        $update =
DB::table('users')
            ->where("id", $id)
            ->update([
                'username' =>
$username,
                'password' =>
bcrypt($password),
                'nomor_hp' =>
$nomor,
            ]);

        // if($password){
        //     $update =
DB::table('users')
        //     ->where("id",
$id)
        //     ->update([
        //         'password'
=> bcrypt($password),
        //     ]);
        // }


        return response()-
>json(["status" => "success",
"message" => "Berhasil", "data"
=> $update]);
    }

    public function register()
    {
        $managerExist =
DB::table('users')-
>where("username",
request('username_manager'))-
>get();
        if
(count($managerExist) != 0) {
            return response()-
>json(["status" => "error",
"message" => "Akun Manager
Sudah Ada"]);
        }
```

```
        $adminExist =
DB::table('users')-
>where("username",
request('username_admin'))-
>get();
        if (count($adminExist)
!= 0) {
            return response()-
>json(["status" => "error",
"message" => "Akun Admin Sudah
Ada"]);
        }

        if (request('logo') !=
null) {
            $imageName = time()
. '.' . request('logo')-
>extension();
            request('logo')-
>move(public_path('img'),
$imageName);
            //request('logo')-
>storeAs('/public/images',
$imageName);
        } else {
            $imageName =
'none.jpg';
        }

        try {
            $insertPerusahaan =
DB::table('perusahaan')-
>insertGetId([
                'nama' =>
request('nama_usaha'),
                'logo' =>
$imageName,
                'jenis_usaha'
=> request('jenis_usaha'),
                'email' =>
request('email'),
                'telepon' =>
request('telepon'),
                'alamat' =>
request('alamat')
            ]);
        } catch (Exception $e)
{
            $errorCode = $e-
>errorInfo[1];
            if ($errorCode ==
1062) {
                return
response()->json(["status" =>
"error", "message" => "Nama
Usaha Sudah Pernah
Didaftarkan"]);
            }
        }

        $insertManager =
DB::table('users')-
>insertGetId([
            'username' =>
request('username_manager'),
            'password' =>
bcrypt(request('password_manage
r')),
            'id_perusahaan' =>
$insertPerusahaan,
            'user_level' => 2
        ]);

        $insertAdmin =
DB::table('users')-
>insertGetId([
            'username' =>
request('username_admin'),
            'password' =>
bcrypt(request('password_admin'
)),
            'id_perusahaan' =>
$insertPerusahaan,
            'user_level' => 3
        ]);

        $insertPelanggan =
DB::table('pelanggan')-
>insert([
            'id_perusahaan' =>
$insertPerusahaan,
            'nama' =>
'Penjualan Tunai',
            'telepon' => '-',
            'email' => '-',
            'alamat' =>
request('alamat'),
        ]);

        $insertPemasok =
DB::table('pemasok')->insert([
            'id_perusahaan' =>
$insertPerusahaan,
```

```
            'nama' =>
'Pembelian Tunai',
            'telepon' => '-',
            'email' => '-',
            'alamat' =>
request('alamat'),
        ]);

        return response()-
>json(["status" => "success",
"message" => "Berhasil"]);
    }

    public function
jenisusaha()
    {
        $jenis_usaha =
DB::table('jenis_usaha')-
>get();
        return Response()-
>json($jenis_usaha);
    }
}
```

Lampiran2 – Dashboard

```
<?php declare(strict_types=1);
/*
 * This file is part of
phpunit/php-code-coverage.
 *
 * (c) Sebastian Bergmann
<sebastian@phpunit.de>
 *
 * For the full copyright and
license information, please
view the LICENSE
 * file that was distributed
with this source code.
 */
namespace
SebastianBergmann\CodeCoverage\
Report\Html;

use function array_values;
use function arsort;
use function asort;
use function count;
use function explode;
use function floor;
use function json_encode;
use function sprintf;
use function str_replace;
use
SebastianBergmann\CodeCoverage\
Node\AbstractNode;
use
SebastianBergmann\CodeCoverage\
Node\Directory as
DirectoryNode;
use
SebastianBergmann\Template\Temp
late;

/**
 * @internal This class is not
covered by the backward
compatibility promise for
phpunit/php-code-coverage
 */
final class Dashboard extends
Renderer
{
    public function
render(DirectoryNode $node,
string $file): void
    {
        $classes      = $node-
>classesAndTraits();
        $templateName = $this-
>templatePath . ($this-
>hasBranchCoverage ?
'dashboard_branch.html' :
'dashboard.html');
        $template     = new
Template(
            $templateName,
            '{{',
            '}}'
        );

        $this-
>setCommonTemplateVariables($te
mplate, $node);

        $baseLink             =
$node->id() . '/';
        $complexity           =
$this->complexity($classes,
$baseLink);
        $coverageDistribution =
$this-
>coverageDistribution($classes)
;
```

```
        $insufficientCoverage =
$this-
>insufficientCoverage($classes,
$baseLink);
        $projectRisks         =
$this->projectRisks($classes,
$baseLink);

        $template->setVar(
            [

'insufficient_coverage_classes'
=>
$insufficientCoverage['class'],

'insufficient_coverage_methods'
=>
$insufficientCoverage['method']
,

'project_risks_classes'
=> $projectRisks['class'],

'project_risks_methods'
=> $projectRisks['method'],

'complexity_class'
=> $complexity['class'],

'complexity_method'
=> $complexity['method'],

'class_coverage_distribution'
=>
$coverageDistribution['class'],

'method_coverage_distribution'
=>
$coverageDistribution['method']
,
            ]
        );

        $template-
>renderTo($file);
    }

    protected function
activeBreadcrumb(AbstractNode
$node): string
    {
        return sprintf(
            '         <li
class="breadcrumb-item"><a
href="index.html">%s</a></li>'
. "\n" .
            '         <li
class="breadcrumb-item
active">(Dashboard)</li>' .
"\n",
            $node->name()
        );
    }

    /**
     * Returns the data for the
Class/Method Complexity charts.
     */
    private function
complexity(array $classes,
string $baseLink): array
    {
        $result = ['class' =>
[], 'method' => []];

        foreach ($classes as
$className => $class) {
            foreach
($class['methods'] as
$methodName => $method) {
                if ($className
!== '*') {
                    $methodName
= $className . '::' .
$methodName;
                }

$result['method'][] = [

$method['coverage'],

$method['ccn'],
                    sprintf(
                        '<a
href="%s">%s</a>',

str_replace($baseLink, '',
$method['link']),

$methodName
                    ),
                ];
            }
```

```
            $result['class'][]
= [
                $class['coverage'],
                $class['ccn'],
                sprintf(
                    '<a
href="%s">%s</a>',
                    str_replace($baseLink, '',
$class['link']),
                    $className
                ),
            ];
        }

        return [
            'class'  =>
json_encode($result['class']),
            'method' =>
json_encode($result['method']),
        ];
    }

    /**
     * Returns the data for the
Class / Method Coverage
Distribution chart.
     */
    private function
coverageDistribution(array
$classes): array
    {
        $result = [
            'class'  => [
                '0%'      => 0,
                '0-10%'   => 0,
                '10-20%'  => 0,
                '20-30%'  => 0,
                '30-40%'  => 0,
                '40-50%'  => 0,
                '50-60%'  => 0,
                '60-70%'  => 0,
                '70-80%'  => 0,
                '80-90%'  => 0,
                '90-100%' => 0,
                '100%'    => 0,
            ],
            'method' => [
                '0%'      => 0,
                '0-10%'   => 0,
                '10-20%'  => 0,
                '20-30%'  => 0,
                '30-40%'  => 0,
                '40-50%'  => 0,
                '50-60%'  => 0,
                '60-70%'  => 0,
                '70-80%'  => 0,
                '80-90%'  => 0,
                '90-100%' => 0,
                '100%'    => 0,
            ],
        ];

        foreach ($classes as
$class) {
            foreach
($class['methods'] as
$methodName => $method) {
                if
($method['coverage'] === 0) {
$result['method']['0%']++;
                } elseif
($method['coverage'] === 100) {
$result['method']['100%']++;
                } else {
                    $key =
floor($method['coverage'] / 10)
* 10;
                    $key = $key
. '-' . ($key + 10) . '%';
$result['method'][$key]++;
                }
            }

            if
($class['coverage'] === 0) {
$result['class']['0%']++;
            } elseif
($class['coverage'] === 100) {
$result['class']['100%']++;
            } else {
                $key =
floor($class['coverage'] / 10)
* 10;
                $key = $key .
'-' . ($key + 10) . '%';
```

```
$result['class'][$key]++;
            }
        }

        return [
            'class'  =>
json_encode(array_values($resul
t['class'])),
            'method' =>
json_encode(array_values($resul
t['method'])),
        ];
    }

    /**
     * Returns the classes /
methods with insufficient
coverage.
     */
    private function
insufficientCoverage(array
$classes, string $baseLink):
array
    {
        $leastTestedClasses =
[];
        $leastTestedMethods =
[];
        $result             =
['class' => '', 'method' =>
''];

        foreach ($classes as
$className => $class) {
            foreach
($class['methods'] as
$methodName => $method) {
                if
($method['coverage'] < $this-
>highLowerBound) {
                    $key =
$methodName;

                    if
($className !== '*') {
                        $key =
$className . '::' .
$methodName;
                    }
$leastTestedMethods[$key] =
$method['coverage'];
                }
            }

            if
($class['coverage'] < $this-
>highLowerBound) {

$leastTestedClasses[$className]
= $class['coverage'];
            }
        }


asort($leastTestedClasses);

asort($leastTestedMethods);

        foreach
($leastTestedClasses as
$className => $coverage) {
            $result['class'] .=
sprintf(
                '
<tr><td><a
href="%s">%s</a></td><td
class="text-
right">%d%%</td></tr>' . "\n",

str_replace($baseLink, '',
$classes[$className]['link']),
                $className,
                $coverage
            );
        }

        foreach
($leastTestedMethods as
$methodName => $coverage) {
            [$class, $method] =
explode('::', $methodName);

            $result['method']
.= sprintf(
                '
<tr><td><a href="%s"><abbr
title="%s">%s</abbr></a></td><t
d class="text-
right">%d%%</td></tr>' . "\n",
```

```
str_replace($baseLink, '',
$classes[$class]['methods'][$me
thod]['link']),
                $methodName,
                $method,
                $coverage
            );
        }

        return $result;
    }

    /**
     * Returns the project
risks according to the CRAP
index.
     */
    private function
projectRisks(array $classes,
string $baseLink): array
    {
        $classRisks  = [];
        $methodRisks = [];
        $result      = ['class'
=> '', 'method' => ''];

        foreach ($classes as
$className => $class) {
            foreach
($class['methods'] as
$methodName => $method) {
                if
($method['coverage'] < $this-
>highLowerBound &&
$method['ccn'] > 1) {
                    $key =
$methodName;

                    if
($className !== '*') {
                        $key =
$className . '::' .
$methodName;
                    }


$methodRisks[$key] =
$method['crap'];
                }
            }

            if
($class['coverage'] < $this-
>highLowerBound &&
                $class['ccn'] >
count($class['methods'])) {

$classRisks[$className] =
$class['crap'];
            }
        }

        arsort($classRisks);
        arsort($methodRisks);

        foreach ($classRisks as
$className => $crap) {
            $result['class'] .=
sprintf(
                '
<tr><td><a
href="%s">%s</a></td><td
class="text-
right">%d</td></tr>' . "\n",

str_replace($baseLink, '',
$classes[$className]['link']),
                $className,
                $crap
            );
        }

        foreach ($methodRisks
as $methodName => $crap) {
            [$class, $method] =
explode('::', $methodName);

            $result['method']
.= sprintf(
                '
<tr><td><a href="%s"><abbr
title="%s">%s</abbr></a></td><t
d class="text-
right">%d</td></tr>' . "\n",

str_replace($baseLink, '',
$classes[$class]['methods'][$me
thod]['link']),
                $methodName,
                $method,
                $crap
            );
        }
```

```
        return $result;
    }
}
```

Lampiran3 – Kue

```
-- phpMyAdmin SQL Dump
-- version 5.2.0
-- https://www.phpmyadmin.net/
--
-- Host: 127.0.0.1
-- Generation Time: Jun 20,
2025 at 04:27 PM
-- Server version: 10.4.25-
MariaDB
-- PHP Version: 8.1.10

SET SQL_MODE =
"NO_AUTO_VALUE_ON_ZERO";
START TRANSACTION;
SET time_zone = "+00:00";


/*!40101 SET
@OLD_CHARACTER_SET_CLIENT=@@CHA
RACTER_SET_CLIENT */;
/*!40101 SET
@OLD_CHARACTER_SET_RESULTS=@@CH
ARACTER_SET_RESULTS */;
/*!40101 SET
@OLD_COLLATION_CONNECTION=@@COL
LATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

--
-- Database:
`marketplace_kue_db`
--

-- ----------------------------
----------------------------


--
-- Table structure for table
`diskon`
--

CREATE TABLE `diskon` (
  `id` int(15) NOT NULL,
  `id_produk` int(15) NOT NULL,
  `diskon` int(15) NOT NULL
COMMENT 'persen',
  `start_date` datetime NOT
NULL,
  `end_date` datetime NOT NULL,
  `created_at` datetime NOT
NULL DEFAULT
current_timestamp()
) ENGINE=InnoDB DEFAULT
CHARSET=utf8mb4;

--
-- Dumping data for table
`diskon`
--

INSERT INTO `diskon` (`id`,
`id_produk`, `diskon`,
`start_date`, `end_date`,
`created_at`) VALUES
(5, 16, 10, '2023-12-10
10:00:00', '2023-12-10
23:53:00', '2023-11-18
10:51:52'),
(6, 21, 2, '2024-12-03
19:02:00', '2024-12-04
19:02:00', '2024-12-03
19:02:50'),
(7, 9, 4, '2024-12-03
19:04:00', '2024-12-05
17:04:00', '2024-12-03
19:04:53');

-- ----------------------------
----------------------------

--
-- Table structure for table
`meja`
--

CREATE TABLE `meja` (
  `id` int(11) NOT NULL,
  `nama` varchar(255) NOT NULL,
  `created_at` datetime NOT
NULL DEFAULT
current_timestamp(),
  `updated_at` datetime NOT
NULL DEFAULT
current_timestamp()
) ENGINE=InnoDB DEFAULT
CHARSET=utf8mb4;

--
```

```
-- Dumping data for table
`meja`
--

INSERT INTO `meja` (`id`,
`nama`, `created_at`,
`updated_at`) VALUES
(1, 'Meja 1', '2024-12-03
12:58:46', '2024-12-03
12:58:46'),
(2, 'meja 2', '2024-12-03
12:59:46', '2024-12-03
12:59:46');

-- ----------------------------
----------------------------

--
-- Table structure for table
`orders`
--

CREATE TABLE `orders` (
  `id` int(15) NOT NULL,
  `id_user` int(15) NOT NULL,
  `id_kurir` int(11) DEFAULT
NULL,
  `kode` varchar(45) NOT NULL,
  `customer` varchar(255) NOT
NULL,
  `alamat` varchar(255) DEFAULT
'',
  `nomor_hp` varchar(50)
DEFAULT '',
  `metode_pengambilan`
enum('onplace','kurir') NOT
NULL DEFAULT 'onplace',
  `harga_total` int(15) NOT
NULL,
  `tanggal` datetime NOT NULL
DEFAULT current_timestamp(),
  `catatan` varchar(255) NOT
NULL,
  `bukti_pembayaran`
varchar(255) DEFAULT NULL,
  `status` tinyint(15) NOT NULL
DEFAULT 0 COMMENT '0:Belum
Bayar\r\n1:Sudah Bayar\r\n2:
Confirm Payment\r\n3: Kurir
Confimation\r\n4: Order Sent
Successfully\r\n5: Done\r\n-1 :
Cancel'
) ENGINE=InnoDB DEFAULT
CHARSET=utf8mb4;

--
-- Dumping data for table
`orders`
--

INSERT INTO `orders` (`id`,
`id_user`, `id_kurir`, `kode`,
`customer`, `alamat`,
`nomor_hp`,
`metode_pengambilan`,
`harga_total`, `tanggal`,
`catatan`, `bukti_pembayaran`,
`status`) VALUES
(80, 10, 7, 'TRANS-
42052025101423', 'Nino', 'jalan
kemarau', '08248524540',
'kurir', 21000, '2025-05-10
14:23:42', 'Kiman',
'1746865387_681f0ceb08ad5.jpg',
5),
(81, 10, NULL, 'TRANS-
08052025101556', 'Testing',
NULL, NULL, 'onplace', 26500,
'2025-05-10 15:56:08', 'okay',
'1746867386_681f14bac289a.jpg',
5),
(82, 10, NULL, 'TRANS-
47052025102021', 'evan',
'testing', '0845404545',
'kurir', 26500, '2025-05-10
20:21:47', 'trs', NULL, 0);

-- ----------------------------
----------------------------

--
-- Table structure for table
`order_detail`
--

CREATE TABLE `order_detail` (
  `id` int(15) NOT NULL,
  `id_order` int(15) NOT NULL,
  `id_produk` int(15) NOT NULL,
  `qty` int(15) NOT NULL
) ENGINE=InnoDB DEFAULT
CHARSET=utf8mb4;

--
```

```
-- Dumping data for table
`order_detail`
--

INSERT INTO `order_detail`
(`id`, `id_order`, `id_produk`,
`qty`) VALUES
(133, 80, 9, 1),
(134, 81, 21, 1),
(135, 81, 9, 1),
(136, 82, 9, 1),
(137, 82, 21, 1);

--
-- Triggers `order_detail`
--
DELIMITER $$
CREATE TRIGGER `add_order`
AFTER INSERT ON `order_detail`
FOR EACH ROW BEGIN
      UPDATE produk SET
stok=stok-NEW.qty WHERE
id=NEW.id_produk;
END
$$
DELIMITER ;

-- ----------------------------
----------------------------

--
-- Table structure for table
`produk`
--

CREATE TABLE `produk` (
  `id` int(15) NOT NULL,
  `nama` varchar(45) NOT NULL,
  `kategori` varchar(45) NOT
NULL,
  `kode` varchar(45) NOT NULL,
  `harga` int(15) NOT NULL
DEFAULT 0,
  `aktif` int(15) NOT NULL
DEFAULT 1,
  `image` varchar(45) NOT NULL
) ENGINE=InnoDB DEFAULT
CHARSET=utf8mb4;

--
-- Dumping data for table
`produk`
--

INSERT INTO `produk` (`id`,
`nama`, `kategori`, `kode`,
`harga`, `stok`, `aktif`,
`image`) VALUES
(9, 'Kue Tart', 'Ulang Tahun',
'A01', 21000, 47, 1,
'1746826549.jpg'),
(21, 'Kue Aneka', 'Aneka',
'A02', 5500, 1998, 1,
'1746826583.jpg');

-- ----------------------------
----------------------------

--
-- Table structure for table
`users`
--

CREATE TABLE `users` (
  `id` int(15) NOT NULL,
  `nama` varchar(255) NOT NULL,
  `username` varchar(45) NOT
NULL,
  `password` varchar(255) NOT
NULL,
  `nomor_hp` varchar(45) NOT
NULL,
  `role` varchar(20) NOT NULL,
  `photo` text NOT NULL DEFAULT
'default.png',
  `aktif` int(15) NOT NULL
DEFAULT 1
) ENGINE=InnoDB DEFAULT
CHARSET=utf8mb4;

--
-- Dumping data for table
`users`
--

INSERT INTO `users` (`id`,
`nama`, `username`, `password`,
`nomor_hp`, `role`, `photo`,
`aktif`) VALUES
(1, 'Admin', 'admin',
'$2y$10$7aSqbVjR1k6a64uvZXzKV.m
hzb/s4247jDIpPTPPLIDsoP0UDuF6e'
```

```
, '081245635245', 'admin',
'default.png', 1),
(7, 'Kurir', 'kurir',
'$2y$10$CIp.0w.pXFBm8zkGA6m/me.
iq/kneoAfjKqEqgptvL5IJKyjuhKIi'
, '0834534534', 'kurir',
'1733226085.jpg', 1),
(10, 'Pelanggan', 'pelanggan',
'$2y$10$uFsRTZI4EuM2a.WLl3.95uj
LIVsXLt8o9x2fEqFFj0eR6KhpATGiC'
, '083819038382', 'pelanggan',
'1692522437.jpg', 1),
(11, 'Server', 'server',
'$2y$10$.2AUyTlVl6XYRWJc4Ti6I.H
wWzK6mKgRstV79pAxICpKXbGH.gWtG'
, '089696400971', 'server',
'default.png', 1),
(16, 'Hen Hara', 'henhara',
'$2y$10$uULBDj8ZaXeWjpz96QBoler
hAEsFbZVSIdyS8tk/SCOCjR3hWmxOa'
, '08248484554', 'kurir',
'1733226117.jpg', 1),
(17, 'rina', 'rina',
'$2y$10$qld/CBbrgm1sO1mJBFJGO.j
eCDYz78SuMN/ieOp39qGcLvxtZI.cy'
, '08218884', 'kurir',
'1733314337.jpg', 1),
(18, 'testing', 'testing',
'$2y$10$4HMzQJ2NZCpwqWOS8y9dxO0
FFNd0dahkO.QKh0zfoeLUIjvyE8jvW'
, '084240484', 'pelanggan',
'default.png', 1),
(19, 'oke', 'testing2',
'$2y$10$pDjQiMBjulrcdIJw3V.du.C
IrG6rkhLW1.yY81a.Nxc62w8IH3eq.'
, '04594434', 'pelanggan',
'default.png', 1);

--
-- Indexes for dumped tables
--

--
-- Indexes for table `diskon`
--
ALTER TABLE `diskon`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `meja`
--
ALTER TABLE `meja`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `orders`
--
ALTER TABLE `orders`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table
`order_detail`
--
ALTER TABLE `order_detail`
  ADD PRIMARY KEY (`id`),
  ADD KEY `id_order`
(`id_order`),
  ADD KEY `id_produk`
(`id_produk`);

--
-- Indexes for table `produk`
--
ALTER TABLE `produk`
  ADD PRIMARY KEY (`id`),
  ADD UNIQUE KEY `kode_strict`
(`kode`);

--
-- Indexes for table `users`
--
ALTER TABLE `users`
  ADD PRIMARY KEY (`id`),
  ADD UNIQUE KEY
`username_restriction`
(`username`);

--
-- AUTO_INCREMENT for dumped
tables
--

--
-- AUTO_INCREMENT for table
`diskon`
--
ALTER TABLE `diskon`
  MODIFY `id` int(15) NOT NULL
AUTO_INCREMENT,
AUTO_INCREMENT=8;

--
```

```
-- AUTO_INCREMENT for table
`meja`
--
ALTER TABLE `meja`
  MODIFY `id` int(11) NOT NULL
AUTO_INCREMENT,
AUTO_INCREMENT=5;

--
-- AUTO_INCREMENT for table
`orders`
--
ALTER TABLE `orders`
  MODIFY `id` int(15) NOT NULL
AUTO_INCREMENT,
AUTO_INCREMENT=83;

--
-- AUTO_INCREMENT for table
`order_detail`
--
ALTER TABLE `order_detail`
  MODIFY `id` int(15) NOT NULL
AUTO_INCREMENT,
AUTO_INCREMENT=138;

--
-- AUTO_INCREMENT for table
`produk`
--
ALTER TABLE `produk`
  MODIFY `id` int(15) NOT NULL
AUTO_INCREMENT,
AUTO_INCREMENT=22;

--
-- AUTO_INCREMENT for table
`users`
--
ALTER TABLE `users`
  MODIFY `id` int(15) NOT NULL
AUTO_INCREMENT,
AUTO_INCREMENT=20;

--
-- Constraints for dumped
tables
--

--
-- Constraints for table
`order_detail`
--
ALTER TABLE `order_detail`
  ADD CONSTRAINT
`order_detail_ibfk_1` FOREIGN
KEY (`id_order`) REFERENCES
`orders` (`id`) ON DELETE
CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT
`order_detail_ibfk_2` FOREIGN
KEY (`id_produk`) REFERENCES
`produk` (`id`) ON DELETE
CASCADE ON UPDATE CASCADE;
COMMIT;

/*!40101 SET
CHARACTER_SET_CLIENT=@OLD_CHARA
CTER_SET_CLIENT */;
/*!40101 SET
CHARACTER_SET_RESULTS=@OLD_CHAR
ACTER_SET_RESULTS */;
/*!40101 SET
COLLATION_CONNECTION=@OLD_COLLA
TION_CONNECTION */;
```

Lampiran 4 – User

```
<?php

namespace
App\Http\Controllers\Api;

use
App\Http\Controllers\Controller
;
use Illuminate\Http\Request;
use
Illuminate\Support\Facades\DB;

class UserController extends
Controller
{
    public function index()
    {
        $data =
DB::table('users')
            ->where('role',
'!=', 'admin')
            ->get();

        return Response()-
>json($data);
    }
```

```
    public function add()
    {
        $data =
DB::table('users')
            ->where("username",
request('username'))
            ->get();

        if (count($data) > 0) {
            return response()-
>json(["status" => "error",
"message" => "Username sudah
ada"]);
        }

        $imageName =
'default.png';
        if (request('image') !=
null) {
            $imageName = time()
. '.' . request('image')-
>extension();
            request('image')-
>move(public_path('img/user'),
$imageName);
        }
        $insert =
DB::table('users')-
>insertGetId([
            'nama' =>
request('nama'),
            'username' =>
request('username'),
            'nomor_hp' =>
request('nomor'),
            'password' =>
bcrypt(request('username')),
            'role' =>
request('role'),
            'photo' =>
$imageName,
            'aktif' =>
request('aktif'),
        ]);

        return response()-
>json(["status" => "success",
"message" => "Berhasil", "data"
=> $insert]);
    }
    public function edit()
    {
        $id = request("id");

        $imageName =
'default.png';
        if (request('image') !=
null) {
            $imageName = time()
. '.' . request('image')-
>extension();
            request('image')-
>move(public_path('img/user'),
$imageName);
            DB::table('users')
                ->where("id",
$id)
                ->update([
                    'photo' =>
$imageName,
                ]);
        }

        DB::table('users')
            ->where("id", $id)
            ->update([
                'nama' =>
request('nama'),
                'nomor_hp' =>
request('nomor'),
                'username' =>
request('username'),
                'role' =>
request('role'),
            ]);

        if
(strlen(request("password")) >
0) {
            DB::table('users')
                ->where("id",
$id)
                ->update([
                    'password'
=> bcrypt(request('password')),
                ]);
        }

        $insert =
DB::table('users')
            ->where("id", $id)
            ->update([
                'aktif' =>
request('aktif'),
```

```
            ]);

        return response()->json(["status" => "success", "message" => "Berhasil", "data" => $insert]);
    }

    public function delete()
    {
        $id = request("id");
        $hapus = DB::table('users')
            ->where('id', $id)
            ->delete();

        return response()->json(["status" => "success", "message" => "Berhasil"]);
    }
}
```

Lampiramn 5 – Pesanan

```
<?xml version="1.0" encoding="utf-8" standalone="yes"?><Layout layout="activity_order_detail" modulePackage="com.project.marketplacekue" filePath="app\src\main\res\layout\activity_order_detail.xml" directory="layout" isMerge="false" isBindingData="false" rootNodeType="android.widget.LinearLayout"><Targets><Target tag="layout/activity_order_detail_0" view="LinearLayout"><Expressions/><location startLine="1" startOffset="0" endLine="389" endOffset="14"/></Target><Target id="@+id/btn_back" view="ImageView"><Expressions/><location startLine="15" startOffset="8" endLine="19" endOffset="50"/></Target><Target id="@+id/tv_kode" view="TextView"><Expressions/><location startLine="52" startOffset="20" endLine="59" endOffset="49"/></Target><Target id="@+id/tv_customer" view="TextView"><Expressions/><location startLine="65" startOffset="20" endLine="71" endOffset="49"/></Target><Target id="@+id/tv_alamat" view="TextView"><Expressions/><location startLine="77" startOffset="20" endLine="83" endOffset="49"/></Target><Target id="@+id/tv_nomor_hp" view="TextView"><Expressions/><location startLine="89" startOffset="20" endLine="95" endOffset="49"/></Target><Target id="@+id/tv_metode_pengambilan" view="TextView"><Expressions/><location startLine="101" startOffset="20" endLine="107" endOffset="49"/></Target><Target id="@+id/tv_kurir" view="TextView"><Expressions/><location startLine="113" startOffset="20" endLine="119" endOffset="49"/></Target><Target id="@+id/tv_status" view="TextView"><Expressions/><location startLine="125" startOffset="20" endLine="132" endOffset="46"/></Target><Target id="@+id/tv_lihat_bukti" view="TextView"><Expressions/><location startLine="138" startOffset="20" endLine="145" endOffset="49"/></Target><Target id="@+id/tv_tanggal" view="TextView"><Expressions/><location startLine="171" startOffset="28" endLine="178" endOffset="55"/></Target><Target id="@+id/rv_data" view="androidx.recyclerview.widget.RecyclerView"><Expressions/><location startLine="227" startOffset="20" endLine="233" endOffset="67"/></Target><Target id="@+id/imgv_not_found" view="ImageView"><Expressions/><location startLine="235" startOffset="20" endLine="241" endOffset="62"/></Target><Targe
```

```
t id="@+id/tv_total"
view="TextView"><Expressions/><
location startLine="267"
startOffset="20" endLine="273"
endOffset="45"/></Target><Targe
t id="@+id/btn_batal"
view="Button"><Expressions/><lo
cation startLine="281"
startOffset="16" endLine="293"
endOffset="44"/></Target><Targe
t
id="@+id/btn_konfirmasi_kurir"
view="Button"><Expressions/><lo
cation startLine="295"
startOffset="16" endLine="307"
endOffset="52"/></Target><Targe
t id="@+id/btn_pesanan_sampai"
view="Button"><Expressions/><lo
cation startLine="309"
startOffset="16" endLine="321"
endOffset="50"/></Target><Targe
t id="@+id/btn_bayar"
view="Button"><Expressions/><lo
cation startLine="323"
startOffset="16" endLine="335"
endOffset="41"/></Target><Targe
t id="@+id/btn_terima_pesanan"
view="Button"><Expressions/><lo
cation startLine="337"
startOffset="16" endLine="349"
endOffset="42"/></Target><Targe
t id="@+id/btn_tolak"
view="Button"><Expressions/><lo
cation startLine="351"
startOffset="16" endLine="363"
endOffset="49"/></Target><Targe
t
id="@+id/btn_selesaikan_pesanan
"
view="Button"><Expressions/><lo
cation startLine="365"
startOffset="16" endLine="376"
endOffset="51"/></Target></Targ
ets></Layout>
```

Lampiran – Laporan

```
<?xml version="1.0"
encoding="utf-8"?>
<FrameLayout
xmlns:android="http://schemas.a
ndroid.com/apk/res/android"
xmlns:tools="http://schemas.and
roid.com/tools"

android:layout_width="match_par
ent"

android:layout_height="match_pa
rent"

xmlns:app="http://schemas.andro
id.com/apk/res-auto"

android:paddingHorizontal="@dim
en/spacing_middle"

tools:context=".fragment.HomeDa
shboardFragment">

    <ScrollView

android:layout_width="match_par
ent"

android:layout_height="match_pa
rent"

android:scrollbars="none">

        <LinearLayout

android:layout_width="match_par
ent"

android:layout_height="wrap_con
tent"

android:layout_margin="@dimen/s
pacing_large"

android:orientation="vertical">

            <LinearLayout

android:layout_width="match_par
ent"

android:layout_height="wrap_con
tent"

android:layout_marginTop="@dime
n/spacing_middle"
```

```
android:visibility="gone"

android:layout_marginBottom="@d
imen/spacing_xlarge"

android:orientation="horizontal
">


<androidx.cardview.widget.CardV
iew

android:layout_width="0dp"

android:layout_weight="1"

android:layout_height="110dp"

app:cardCornerRadius="@dimen/sp
acing_large">


<LinearLayout

android:layout_width="match_par
ent"

android:layout_height="match_pa
rent"

android:background="@drawable/b
g_image_blue"

android:padding="15dp"

android:layout_gravity="center"

android:gravity="center"

android:orientation="vertical">


<TextView

android:layout_width="match_par
ent"

android:layout_height="wrap_con
tent"

android:gravity="center"

android:textColor="@color/white
"

android:textStyle="bold"

android:textSize="12sp"

android:text="Pendapatan Hari
Ini"/>

                        <View

android:layout_width="0dp"

android:layout_height="@dimen/s
pacing_middle"/>


<TextView

android:id="@+id/tv_pendapatan_
hari_ini"

android:layout_width="match_par
ent"

android:layout_height="wrap_con
tent"

android:gravity="center"

android:textColor="@color/white
"

android:textStyle="bold"

android:textSize="17sp"

android:text="Rp 0"/>


</LinearLayout>


</androidx.cardview.widget.Card
View>

                <View
```

```
android:layout_width="@dimen/spacing_large"

android:layout_height="0dp" />


<androidx.cardview.widget.CardView

android:layout_width="0dp"

android:layout_weight="1"

android:layout_height="110dp"

app:cardCornerRadius="@dimen/spacing_large">


<LinearLayout

android:layout_width="match_parent"

android:layout_height="match_parent"

android:background="@drawable/bg_image_blue"

android:padding="15dp"

android:layout_gravity="center"

android:gravity="center"

android:orientation="vertical">


<TextView

android:layout_width="match_parent"

android:layout_height="wrap_content"

android:gravity="center"

android:textColor="@color/white"

android:textStyle="bold"

android:textSize="12sp"

android:text="Pendapatan Bulan Ini"/>

                        <View

android:layout_width="0dp"

android:layout_height="@dimen/spacing_middle"/>


<TextView

android:id="@+id/tv_pendapatan_bulan_ini"

android:layout_width="match_parent"

android:layout_height="wrap_content"

android:gravity="center"

android:textColor="@color/white"

android:textStyle="bold"

android:textSize="17sp"

android:text="Rp 0"/>


</LinearLayout>


</androidx.cardview.widget.CardView>

            </LinearLayout>

            <LinearLayout

android:layout_width="match_parent"
```

```
android:layout_height="wrap_con
tent"

android:layout_marginVertical="
@dimen/spacing_medium"

android:orientation="horizontal
">


<androidx.cardview.widget.CardV
iew

android:id="@+id/cd_order"

android:layout_width="0dp"

android:layout_weight="1"

android:layout_height="130dp"

app:cardCornerRadius="@dimen/sp
acing_large">


<RelativeLayout

android:layout_width="match_par
ent"

android:layout_height="match_pa
rent"

android:background="@drawable/b
g_image_blue"

android:padding="15dp"

android:orientation="vertical">


<ImageView

android:layout_width="match_par
ent"

android:layout_height="match_pa
rent"

android:layout_marginBottom="20
dp"

android:src="@drawable/order"/>


<TextView

android:layout_width="match_par
ent"

android:layout_height="wrap_con
tent"

android:gravity="center"

android:textColor="@color/white
"

android:textStyle="bold"

android:textSize="15sp"

android:layout_alignParentBotto
m="true"

android:text="Pesanan"/>



</RelativeLayout>


</androidx.cardview.widget.Card
View>

                <View

android:layout_width="@dimen/sp
acing_large"

android:layout_height="0dp" />


<androidx.cardview.widget.CardV
iew

android:id="@+id/cd_report"

android:layout_width="0dp"

android:layout_weight="1"

android:layout_height="130dp"
```

```
app:cardCornerRadius="@dimen/spacing_large">

<RelativeLayout
android:layout_width="match_parent"
android:layout_height="match_parent"
android:background="@drawable/bg_image_blue"
android:padding="15dp"
android:orientation="vertical">

<ImageView
android:layout_width="match_parent"
android:layout_height="match_parent"
android:layout_marginBottom="20dp"
android:src="@drawable/chart"/>

<TextView
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:gravity="center"
android:textColor="@color/white"
android:textStyle="bold"
android:textSize="15sp"
android:layout_alignParentBottom="true"
android:text="Laporan"/>

</RelativeLayout>

</androidx.cardview.widget.CardView>
            </LinearLayout>
        </LinearLayout>
    </ScrollView>
</FrameLayout>
```

KARTU MONITORING BIMBINGAN

MAHASISWA PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH PAREPARE

PROPOSAL

Mahasiswa : MUH. FADEL HERMAN	Pembimbing I : Marlina, S.Kom.,M.Kom.
NIM : 218 280 210	Pembimbing II : Andi Wafiah, S.Kom., M.Kom.
Judul Skripsi : PENGEMBANGAN APLIKASI PEMESANAN KUE PADA PALANRO CAKE BERBASIS ANDROID	

ARAHAN PEMBIMBING I	HARI/TGL & PARAF PEMBIMBING	ARAHAN PEMBIMBING II	HARI/TGL & PARAF PEMBIMBING
Konsultasi 1 - Cari penelitian yg terkait yg ada di UMPAR. - Pelajari Metode pembayaran	21/12/23	Konsultasi 1 - Lengkapi Flowchart - Lengkapi Judul tabel setiap halaman	
Konsultasi 2		Konsultasi 2 - Lengkapi kerangka pikir	
Konsultasi 3 Acc		Konsultasi 3 Tempat Penelitian di Perpus	
Konsultasi 4		Konsultasi 4 Acc	
Konsultasi 5		Konsultasi 5	

Lanjut ke halaman sebelah...

Perhatian :

1. Mahasiswa wajib konsultasi minimal 5 kali
2. Kartu ini wajib dibawa oleh mahasiswa disetiap konsultasi dan diisi oleh Pembimbing
3. Kartu ini wajib dilampirkan pada laporan skripsi dan menjadi salah satu persyaratan untuk ikut seminar proposal/ujian skripsi
4. Kartu ini dicetak di atas kertas karton berwarna hijau muda dan dicetak timbal balik

Lanjutan ...

ARAHAN PEMBIMBING I	HARI/TGL & PARAF PEMBIMBING	ARAHAN PEMBIMBING II	HARI/TGL & PARAF PEMBIMBING
Konsultasi 6 ACC hasil		Konsultasi 6 Kerjakan Revisi dari Penguji	
Konsultasi 7 - Revisi sudah ditinolak lanjuti. - ACC ujian tutup		Konsultasi 7	
Konsultasi 8		Konsultasi 8	
Konsultasi 9		Konsultasi 9 ACC Ujian Tutup	
Konsultasi 10		Konsultasi 10	

Parepare, 5 Juli 2025

Mengetahui
Ketua Program Studi

Mughaffir Yunus, ST.,MT
NBM. 1252 809

Mahasiswa

MUHAMMAD FADEL HERMAN
NIM. 218 280 210

Perhatian :

1. Mahasiswa wajib konsultasi minimal 5 kali
2. Kartu ini wajib dibawa oleh mahasiswa disetiap konsultasi dan diisi oleh Pembimbing
3. Kartu ini wajib dilampirkan pada laporan skripsi dan menjadi salah satu persyaratan untuk ikut seminar proposal/ujian skripsi
4. Kartu ini dicetak di atas kertas karton berwarna hijau muda dan dicetak timbal balik

KARTU MONITORING BIMBINGAN

MAHASISWA PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH PAREPARE

SKRIPSI

Mahasiswa : MUHAMMAD FADEL HERMAN	Pembimbing I : Andi Wafiah, S.Kom., M.Kom
NIM : 218 280 210	Pembimbing II : Marlina, S.Kom., M.Kom
Judul Skripsi : PENGEMBANGAN APLIKASI PPEMESANAN KUE PADA PALANRO CAKE BERBASIS ANDROID	

ARAHAN PEMBIMBING I	HARI/TGL & PARAF PEMBIMBING	ARAHAN PEMBIMBING II	HARI/TGL & PARAF PEMBIMBING
Konsultasi 1 Revisi Aplikasi	4.	Konsultasi 1 Lanjutkan buatkan Dokumen Bab 4 dan Bab 5	
Konsultasi 2 ikuti pedoman penulisan	4.	Konsultasi 2 Bahasa Asing dicetak Miring	
Konsultasi 3 Revisi aplikasi	4.	Konsultasi 3 -Cari literatur minimal [illegible] Terbaru	
Konsultasi 4 Lengkapi lampiran	4.	Konsultasi 4 /	
Konsultasi 5 Ikuti pedoman penulisan (Format penggunaan bahasa asing)	4	Konsultasi 5 Acc Ujian Hasil	

Lanjut ke halaman sebelah

Perhatian :

1. Mahasiswa wajib konsultasi minimal 5 kali
2. Kartu ini wajib dibawa oleh mahasiswa disetiap konsultasi dan diisi oleh Pembimbing
3. Kartu ini wajib dilampirkan pada laporan skripsi dan menjadi salah satu persyaratan untuk ikut seminar proposal/ujian skripsi
4. Kartu ini dicetak di atas kertas karton berwarna hijau muda dan dicetak timbal balik

Lanjutan ...

ARAHAN PEMBIMBING I	HARI/TGL & PARAF PEMBIMBING	ARAHAN PEMBIMBING II	HARI/TGL & PARAF PEMBIMBING
Konsultasi 6		Konsultasi 6	
Konsultasi 7		Konsultasi 7	
Konsultasi 8		Konsultasi 8	
Konsultasi 9		Konsultasi 9	
Konsultasi 10		Konsultasi 10	

Parepare, 11 November 2023

Mengetahui
Ketua Program Studi

Marlina, S.Kom.,M.Kom.
NBM. 1162 680

Mahasiswa

MUH. FADEL HERMAN
NIM. ***218 280 210***

Perhatian :

1. Mahasiswa wajib konsultasi minimal 5 kali
2. Kartu ini wajib dibawa oleh mahasiswa disetiap konsultasi dan diisi oleh Pembimbing
3. Kartu ini wajib dilampirkan pada laporan skripsi dan menjadi salah satu persyaratan untuk ikut seminar proposal/ujian skripsi
4. Kartu ini dicetak di atas kertas karton berwarna hijau muda dan dicetak timbal balik