

Lampiran 1 Surat Kesiediaan Membimbing TA Pembimbing 1

SURAT KESEDIAAN MEMBIMBING TA

Yang bertanda tangan di bawah ini:

Nama : Safar Dwi Kurniawan, M.Kom
NIPY : 03.021.487
Jabatan Struktural : -
Jabatan Fungsional : Lektor

Dengan ini menyatakan bersedia untuk menjadi pembimbing I pada Tugas Akhir mahasiswa berikut:

Nama : Fahmi Amirudin
NIM : 22040014
Program Studi : Diploma III Teknik Komputer

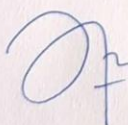
Judul TA : RANCANG BANGUN SISTEM PENJUALAN DIGITAL UNTUK TOKO PARFUM MENGGUNAKAN FLUTTER DAN LARAVEL

Dengan ini menyatakan ini dibuat agar dilaksanakan sebagaimana mestinya

Tegal, 5 Februari 2025

Mengetahui

Ka Prodi DIII Teknik Komputer, Dosen Pembimbing I,


Ida Afriliana, ST, M.Kom
NIPY. 12.013.168


Safar Dwi Kurniawan, M.Kom
NIPY. 03.021.487

Lampiran 2 Surat Kesiediaan Membimbing TA Pembimbing 2

SURAT KESEDIAAN MEMBIMBING TA

Yang bertanda tangan di bawah ini:

Nama : Arfan Haqiqi Sulasmoro, M.Kom
NIPY : 02.009.054
Jabatan Struktural : -
Jabatan Fungsional : Lektor

Dengan ini menyatakan bersedia untuk menjadi pembimbing I pada Tugas Akhir mahasiswa berikut:

Nama : Fahmi Amirudin
NIM : 22040014
Program Studi : Diploma III Teknik Komputer

Judul TA : RANCANG BANGUN SISTEM PENJUALAN DIGITAL UNTUK TOKO PARFUM MENGGUNAKAN FLUTTER DAN LARAVEL

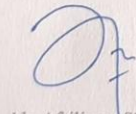
Dengan ini menyatakan ini dibuat agar dilaksanakan sebagaimana mestinya

Tegal, 5 Februari 2025

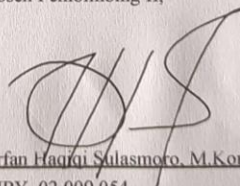
Mengetahui

Ka Prodi DIII Teknik Komputer,

Dosen Pembimbing II,



Ida Afriliana, ST, M.Kom
NIPY. 12.013.168



Arfan Haqiqi Sulasmoro, M.Kom
NIPY. 02.009.054

Lampiran 3 Surat Observasi

**POLITEKNIK HARAPAN BERSAMA**
TEKNIK HARAPAN BERSAMA

D-3 Teknik Komputer

Nomor : 296.03/KOM-PHB/IV/2025
Lampiran : -
Perihal : Permohonan Izin Observasi Tugas Akhir (TA)

Kepada Yth.
Kepala Toko Parfum Al-Hanna 2
Di Jl. Merpati, Randugunting, Kec. Tegal Sel., Kota Tegal, Jawa Tengah 52123

Dengan Hormat,

Sehubungan dengan tugas mata kuliah Tugas Akhir (TA) yang akan diselenggarakan di semester VI Program Studi Diploma III Teknik Komputer Politeknik Harapan Bersama Tegal, Maka dengan ini kami mengajukan izin observasi pengambilan data di Toko Parfum Al-Hanna 2 yang Bapak / Ibu Pimpin, untuk kepentingan dalam pembuatan produk Tugas Akhir, dengan Mahasiswa sebagai berikut:

No	NIM	NAMA	NO HP
1	22040014	FAHMI AMIRUDIN	089666853562

Demikian surat permohonan ini kami sampaikan atas izin dan kerjasamanya kami sampaikan terima kasih.

Tegal, 22 April 2025
Ka. Prodi DIII Teknik Komputer
Politeknik Harapan Bersama Tegal


Ida Afrilliana, ST, M.Kom
NIPY. 12.013.168

 J. Mutaram No. 9 Kota Tegal 52143, Jawa Tengah, Indonesia
 (0283)352090

 komputer@politektegal.ac.id
 politektegal.ac.id

Lampiran 4 Surat Keterangan Hasil Observasi

SURAT KETERANGAN OBSERVASI

Yang bertanda tangan dibawah ini:

Nama : Najib Madani
Jabatan : Kasir Toko
Toko : Al-Hanna Parfum 2
Alamat Toko : Jl. Merpati, Randugunting, Kec. Tegal Selatan,
Kota Tegal, Jawa Tengah 52123

Menyatakan bahwa nama mahasiswa yang bernama:

Nama : Fahmi Amirudin
NIM : 22040014
Jurusan / Program Studi : DIII Teknik Komputer

Benar, telah melakukan kegiatan observasi di Toko Parfum Al-Hanna 2 Jl. Merpati, Randugunting, Kec. Tegal Selatan, Kota Tegal, Jawa Tengah 52123 pada bulan mei 2025 untuk memperoleh data guna penyusunan proyek tugas akhir dengan judul "RANCANG BANGUN SISTEM PENJUALAN DIGITAL UNTUK TOKO PARFUM MENGGUNAKAN FLUTTER DAN LARAVEL".

Demikian surat keterangan ini dibuat untuk dapat dipergunakan sebagaimana mestinya.

Tegal, 4 Juli 2025

Kasir Toko
AL HANNA PARFUME 2
JUAL BERBAGAI MACAM MINYAK WANGI
JL. MERPATI NO. 178 KAMPUNG RANDUGUNTING
RANDUGUNTING - TEGAL
HP 0878 8520 0757
Najib Madani

Lampiran 5 Source Code

Dasbor Aplikasi

```
import 'package:pos_rng_frontend/app/domain/entity/order.dart';
import
'package:pos_rng_frontend/app/presentation/detail_order/detail_order_screen.dart';
import
'package:pos_rng_frontend/app/presentation/home/home_notifier.dart';
import
'package:pos_rng_frontend/app/presentation/input_order/input_order_screen.dart';
import
'package:pos_rng_frontend/app/presentation/order/order_screen.dart';
import
'package:pos_rng_frontend/app/presentation/profil/profil_screen.dart';
import
'package:pos_rng_frontend/core/helper/date_time_helper.dart';
import 'package:pos_rng_frontend/core/helper/global_helper.dart';
import 'package:pos_rng_frontend/core/helper/number_helper.dart';
import 'package:pos_rng_frontend/core/provider/app_provider.dart';
import 'package:pos_rng_frontend/core/widget/app_widget.dart';
import 'package:Flutter/cupertino.dart';
import 'package:Flutter/material.dart';
import 'package:Flutter/src/widgets/framework.dart';
import 'package:Flutter/widgets.dart';

class HomeScreen extends StatefulWidget<HomeNotifier, void, void> {
  @override
  Widget bodyBuild(BuildContext context) {
    return SafeArea(
      child: RefreshIndicator(
        onRefresh: () => notifier.init(),
        child: ListView(
          children: [
            _headerLayout(context),
            _orderTodayLayout(context),
          ],
        ),
      ),
    );
  }

  @override
  Widget? floatingActionButtonBuild(BuildContext context) {
    return FloatingActionButton(
      onPressed: () => _onPressAddOrder(context),
      child: const Icon(Icons.add),
    );
  }

  _headerLayout(BuildContext context) {
    return Padding(
```

```

padding: const EdgeInsets.all(10),
child: Column(
  children: [
    const SizedBox(
      height: 20,
    ),
    Row(
      children: [
        InkWell(
          onTap: () => _onPressAvatar(context),
          child: CircleAvatar(
            radius: 40,
            backgroundColor:
GlobalHelper.getColorScheme(context).primary,
            child: Text(
              notifier.name.substring(0, 1),
              style: GlobalHelper.getTextTheme(context,
                appTextStyle:
AppTextStyle.HEADLINE_LARGE)
                ?.copyWith(
                  fontWeight: FontWeight.bold,
                  color:

GlobalHelper.getColorScheme(context).onPrimary),
            ),
          ),
        const SizedBox(
          width: 10,
        ),
        Expanded(
          child: Column(
            crossAxisAlignment: CrossAxisAlignment.start,
            children: [
              Text(
                notifier.name,
                style: GlobalHelper.getTextTheme(context,
                  appTextStyle:
AppTextStyle.TITLE_LARGE)
                  ?.copyWith(
                    color:

GlobalHelper.getColorScheme(context).primary,
                    fontWeight: FontWeight.bold),
              ),
              const SizedBox(
                height: 5,
              ),
              Text(
                notifier.email,
                style: GlobalHelper.getTextTheme(context,
                  appTextStyle:
AppTextStyle.LABEL_LARGE)
                  ?.copyWith(
                    color:

GlobalHelper.getColorScheme(context)

```

```

                .secondary),
            ),
        ],
    ),
),
],
),
],
),
);
}

_orderTodayLayout(BuildContext context) {
return Column(
  children: [
    Padding(
      padding: const EdgeInsets.symmetric(horizontal: 10),
      child: Row(
        children: [
          Expanded(
            child: Text(
              'Penjualan hari ini',
              style: GlobalHelper.getTextTheme(context,
                appTextStyle: AppTextStyle.TITLE_LARGE)
                ?.copyWith(
                  color:
GlobalHelper.getColorScheme(context).primary,
                  fontWeight: FontWeight.bold),
            ),
          ),
          FilledButton(
            onPressed: () => _onPressShowAllOrder(context),
            child: const Text('Lihat Semua'),
          ),
        ],
      ),
    ),
    const SizedBox(
      height: 10,
    ),
    ListView.separated(
      physics: const NeverScrollableScrollPhysics(),
      shrinkWrap: true,
      separatorBuilder: (context, index) => const SizedBox(
        height: 5,
      ),
      itemCount: notifier.listOrder.length,
      itemBuilder: (context, index) {
        final item = notifier.listOrder[index];
        return _itemOrderLayout(context, item);
      },
    ),
    const SizedBox(
      height: 90,
    ),
  ],
)

```

```

    );
}

_itemOrderLayout(BuildContext context, OrderEntity item) {
  return InkWell(
    onTap: () => _onPressItemOrder(context, item),
    child: Container(
      padding: const EdgeInsets.all(10),
      decoration: const BoxDecoration(color: Colors.white),
      child: Column(
        children: [
          Row(
            mainAxisAlignment: MainAxisAlignment.spaceBetween,
            children: [
              Text(
                item.name,
                style: GlobalHelper.getTextTheme(context,
                  appTextStyle: AppTextStyle.BODY_LARGE)
                  ?.copyWith(
                    color:
GlobalHelper.getColorScheme(context).primary,
                    fontWeight: FontWeight.bold),
              ),
              Text(
                DateTimeHelper.formatDateTimeFromString(
                  dateTimeString: item.updatedAt!,
                  format: 'dd MMM yyyy HH:mm:ss'),
                style: GlobalHelper.getTextTheme(context,
                  appTextStyle: AppTextStyle.BODY_SMALL)
                  ?.copyWith(
                    color:
GlobalHelper.getColorScheme(context).secondary),
              ),
            ],
          ),
          const SizedBox(
            height: 5,
          ),
          Row(
            mainAxisAlignment: MainAxisAlignment.spaceBetween,
            children: [
              Text(
                '${NumberHelper.formatIdr(item.totalPrice!)}
(${'item.items.length'} item)',
                style: GlobalHelper.getTextTheme(context,
                  appTextStyle: AppTextStyle.BODY_LARGE)
                  ?.copyWith(
                    color:
GlobalHelper.getColorScheme(context).primary,
                    fontWeight: FontWeight.bold),
              ),
            ],
          ),
        ],
      ),
    ),
  ),

```



```

        ),
    );
}

_onPressAvatar(context) {
  Navigator.push(
    context,
    CupertinoPageRoute(
      builder: (context) => ProfilScreen(),
    ),
  );
}

_onPressAddOrder(BuildContext context) async {
  await Navigator.push(
    context,
    CupertinoPageRoute(
      builder: (context) => InputOrderScreen(),
    ),
  );
  notifier.init();
}

_onPressShowAllOrder(BuildContext context) async {
  await Navigator.push(
    context,
    CupertinoPageRoute(
      builder: (context) => OrderScreen(),
    ),
  );
  notifier.init();
}

_onPressItemOrder(BuildContext context, OrderEntity item) {
  Navigator.push(
    context,
    CupertinoPageRoute(
      builder: (context) => DetailOrderScreen(
        param1: item.id,
      ),
    ),
  );
}
}

```

Produk Website

<?php

```
namespace App\Filament\Resources;

use App\Filament\Resources\ProductResource\Pages;
use App\Filament\Resources\ProductResource\RelationManagers;
use App\Models\Product;
use Filament\Forms;
use Filament\Forms\Form;
use Filament\Resources\Resource;
use Filament\Tables;
use Filament\Tables\Table;
use Illuminate\Database\Eloquent\Builder;
use Illuminate\Database\Eloquent\SoftDeletingScope;
use Filament\Forms\Set;
use App\Filament\Clusters\Products;
use App\Imports\ProductEditImport;
use App\Imports\ProductImport;
use Filament\Forms\Components\FileUpload;
use Filament\Notifications\Notification;
use Filament\Tables\Actions\Action;
use Filament\Tables\Actions\ExportAction;
use Filament\Tables\Actions\ImportAction;
use Filament\Tables\Filters\SelectFilter;
use Illuminate\Support\Facades\App;
use Maatwebsite\Excel\Facades\Excel;
use App\Models\StockTransaction;

class ProductResource extends Resource
{
    public static function getNavigationLabel(): string
    {
        return App::getLocale() === 'id' ? 'Produk' : 'Product';
    }
    protected static ?string $model = Product::class;

    protected static ?string $navigationIcon = 'heroicon-m-
building-storefront';

    protected static ?string $navigationGroup = 'Kasir';

    protected static ?int $navigationSort = 2;

    public static function form(Form $form): Form
    {
        return $form
            ->schema([
                Forms\Components\TextInput::make('name')
                    ->label(__('Name Product'))
                    ->required()
                    ->maxLength(255),
                Forms\Components\Select::make('category_id')
                    ->relationship('category', 'name')
                    ->label(__('Category'))
            ])
    }
}
```

```

        ->searchable()
        ->preload()
        ->native(false)
        ->required(),
Forms\Components\TextInput::make('stock')
        ->required()
        ->label(__('Stock'))
        ->numeric()
        ->default(1),
Forms\Components\TextInput::make('price')
        ->required()
        ->numeric()
        ->label(__('Price'))
        ->prefix('Rp.'),
Forms\Components\Toggle::make('is_active')
        ->default(true)
        ->required()
        ->label(__('Is Active?')),
Forms\Components\FileUpload::make('image')
        ->image()
        ->label(__('Image')),
Forms\Components\TextInput::make('barcode')
        ->label(__('Barcode')),
Forms\Components\Textarea::make('description')
        ->columnSpanFull()
        ->label(__('Description')),
Forms\Components\TextInput::make('stock_in')
        ->label('Tambah Stok')
        ->numeric()
        ->default(0)
        ->suffix('/ml')
        ->afterStateUpdated(function (Set $set,
$state, $get) {
            if ($state > 0) {
                StockTransaction::create([
                    'product_id' => $get('id'),
                    'type' => 'in',
                    'quantity' => $state,
                    'description' => 'Penambahan Stok
Manual'
                ]);
            }
        })
    });
}

public static function table(Table $table): Table
{
    return $table
        ->deferLoading()
        ->defaultPaginationPageOption(5)
        ->paginated([5, 10, 25, 50, 100, 250, 500])
        ->columns([
            Tables\Columns\TextColumn::make('name')
                ->searchable()
                ->label(__('Name Product')),

```

```

Tables\Columns\TextColumn::make('category.name')
    ->label(__('Name Category'))
    ->numeric()
    ->sortable(),
Tables\Columns\TextColumn::make('stock')
    ->numeric()
    ->sortable()
    ->label(__('Stock')),
Tables\Columns\TextColumn::make('price')
    ->searchable()
    ->money('IDR')
    ->sortable()
    ->label(__('Price')),
Tables\Columns\ToggleColumn::make('is_active')
    ->label(__('Is Active?')),
Tables\Columns\TextColumn::make('barcode')
    ->searchable()
    ->label(__('Barcode')),
Tables\Columns\ImageColumn::make('image')
    ->label(__('Image'))
    ->toggleable(isToggledHiddenByDefault: true),
Tables\Columns\TextColumn::make('created_at')
    ->dateTime()
    ->sortable()
    ->label(__('Created At'))
    ->toggleable(isToggledHiddenByDefault: true),
Tables\Columns\TextColumn::make('updated_at')
    ->dateTime()
    ->label(__('Updated At'))
    ->sortable()
    ->toggleable(isToggledHiddenByDefault: true),
Tables\Columns\TextColumn::make('deleted_at')
    ->dateTime()
    ->label(__('Deleted At'))
    ->sortable()
    ->toggleable(isToggledHiddenByDefault: true),
])
->filters([
    SelectFilter::make('category_id')
        ->label(__('Category'))
        ->relationship('category', 'name')
        ->preload()
        ->searchable()
        ->native(false)
])
->actions([
    Tables\Actions\EditAction::make(),
    Tables\Actions\DeleteAction::make(),
])
->headerActions([
    Action::make("Download Template")
        ->icon('heroicon-s-arrow-down-tray')
        ->color('success')
        ->url(route('download-template')),
    Action::make('importProducts')
        ->label('Import Product')
])

```

```

->color('danger')
->icon('heroicon-s-arrow-up-tray')
->form([
    FileUpload::make('attachment')
        ->label('Upload Template Produk')
])
->action(function (array $data) {
    $file = public_path('storage/' .
$data['attachment']);

    try {
        $import = new ProductImport();
        Excel::import($import, $file);
        $totalRows = $import->getRowCount();
        Notification::make()
            ->title('Produk diimpor')
            ->body("Produk diimpor sebanyak
{$totalRows} baris.")
            ->success()
            ->send();
    } catch (\Exception $e) {
        Notification::make()
            ->danger()
            ->title('Produk gagal diimpor')
            ->body($e->getMessage())
            ->send();
    }
}),
])

->bulkActions([
    Tables\Actions\BulkActionGroup::make([
        Tables\Actions\DeleteBulkAction::make(),
    ]),
]);
}

public static function getRelations(): array
{
    return [
        //
    ];
}

public static function getPages(): array
{
    return [
        'index' => Pages\ListProducts::route('/'),
    ];
}
}

```

Lampiran 6 Foto Dokumentasi

