File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\.gitignore

```
# Miscellaneous
*.class
*.log
*.pyc
*.swp
.DS_Store
.atom/
.build/
.buildlog/
.history
.svn/
.swiftpm/
migrate_working_dir/

# IntelliJ related
*.iml
*.ipr
*.iws
.idea/

# The .vscode folder contains launch configuration and tasks you configure in
# VS Code which you may wish to be included in version control, so this line
# is commented out by default.
#.vscode/

# Flutter/Dart/Pub related
**/doc/api/
**/ios/Flutter/.last_build_id
.dart_tool/
.flutter-plugins
.flutter-plugins-dependencies
.pub-cache/
.pub/
/build/

# Symbolication related
app.*.symbols

# Obfuscation related
app.*.map.json

# Android Studio will place build artifacts here
/android/app/debug
/android/app/profile
/android/app/release
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\.metadata

```
# This file tracks properties of this Flutter project.
# Used by Flutter tool to assess capabilities and perform upgrades etc.
#
# This file should be version controlled and should not be manually edited.

version:
  revision: "6fba2447e95c451518584c35e25f5433f14d888c"
  channel: "stable"

project_type: app

# Tracks metadata for the flutter migrate command
migration:
  platforms:
    - platform: root
      create_revision: 6fba2447e95c451518584c35e25f5433f14d888c
      base_revision: 6fba2447e95c451518584c35e25f5433f14d888c
    - platform: android
      create_revision: 6fba2447e95c451518584c35e25f5433f14d888c
      base_revision: 6fba2447e95c451518584c35e25f5433f14d888c
    - platform: ios
      create_revision: 6fba2447e95c451518584c35e25f5433f14d888c
      base_revision: 6fba2447e95c451518584c35e25f5433f14d888c
    - platform: linux
      create_revision: 6fba2447e95c451518584c35e25f5433f14d888c
      base_revision: 6fba2447e95c451518584c35e25f5433f14d888c
    - platform: macos
      create_revision: 6fba2447e95c451518584c35e25f5433f14d888c
      base_revision: 6fba2447e95c451518584c35e25f5433f14d888c
    - platform: web
      create_revision: 6fba2447e95c451518584c35e25f5433f14d888c
      base_revision: 6fba2447e95c451518584c35e25f5433f14d888c
    - platform: windows
      create_revision: 6fba2447e95c451518584c35e25f5433f14d888c
      base_revision: 6fba2447e95c451518584c35e25f5433f14d888c

  # User provided section

  # List of Local paths (relative to this file) that should be
  # ignored by the migrate tool.
  #
  # Files that are not part of the templates will be ignored by default.
  unmanaged_files:
    - 'lib/main.dart'
    - 'ios/Runner.xcodeproj/project.pbxproj'
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\analysis_options.yaml

```
# This file configures the analyzer, which statically analyzes Dart code to
# check for errors, warnings, and lints.
#
# The issues identified by the analyzer are surfaced in the UI of Dart-enabled
# IDEs (https://dart.dev/tools#ides-and-editors). The analyzer can also be
# invoked from the command line by running `flutter analyze`.

# The following line activates a set of recommended lints for Flutter apps,
# packages, and plugins designed to encourage good coding practices.
include: package:flutter_lints/flutter.yaml

linter:
  # The lint rules applied to this project can be customized in the
  # section below to disable rules from the `package:flutter_lints/flutter.yaml`
  # included above or to enable additional rules. A list of all available lints
  # and their documentation is published at https://dart.dev/lints.
  #
  # Instead of disabling a lint rule for the entire project in the
  # section below, it can also be suppressed for a single line of code
  # or a specific dart file by using the `// ignore: name_of_lint` and
  # `// ignore_for_file: name_of_lint` syntax on the line or in the file
  # producing the lint.
  rules:
    # avoid_print: false  # Uncomment to disable the `avoid_print` rule
    # prefer_single_quotes: true  # Uncomment to enable the `prefer_single_quotes` rule

# Additional information about this file can be found at
# https://dart.dev/guides/language/analysis-options
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\devtools_options.yaml

```
description: This file stores settings for Dart & Flutter DevTools.
documentation: https://docs.flutter.dev/tools/devtools/extensions#configure-extension-enablement-states
extensions:
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\pubspec.lock

```
# Generated by pub
# See https://dart.dev/tools/pub/glossary#lockfile
packages:
  args:
    dependency: transitive
    description:
      name: args
      sha256: d0481093c50b1da8910eb0bb301626d4d8eb7284aa739614d2b394ee09e3ea04
      url: "https://pub.dev"
    source: hosted
    version: "2.7.0"
  async:
    dependency: transitive
    description:
      name: async
      sha256: "758e6d74e971c3e5aceb4110bfd6698efc7f501675bcfe0c775459a8140750eb"
      url: "https://pub.dev"
    source: hosted
    version: "2.13.0"
  bloc:
    dependency: transitive
    description:
      name: bloc
      sha256: "52c10575f4445c61dd9e0cafcc6356fdd827c4c64dd7945ef3c4105f6b6ac189"
      url: "https://pub.dev"
    source: hosted
    version: "9.0.0"
  boolean_selector:
    dependency: transitive
    description:
      name: boolean_selector
      sha256: "8aab1771e1243a5063b8b0ff68042d67334e3feab9e95b9490f9a6ebf73b42ea"
      url: "https://pub.dev"
    source: hosted
    version: "2.1.2"
  characters:
    dependency: transitive
    description:
      name: characters
      sha256: f71061c654a3380576a52b451dd5532377954cf9dbd272a78fc8479606670803
      url: "https://pub.dev"
    source: hosted
    version: "1.4.0"
  clock:
    dependency: transitive
    description:
      name: clock
      sha256: fddb70d9b5277016c77a80201021d40a2247104d9f4aa7bab7157b7e3f05b84b
      url: "https://pub.dev"
    source: hosted
    version: "1.1.2"
  collection:
    dependency: transitive
    description:
      name: collection
      sha256: "2f5709ae4d3d59dd8f7cd309b4e023046b57d8a6c82130785d2b0e5868084e76"
      url: "https://pub.dev"
    source: hosted
    version: "1.19.1"
  crypto:
    dependency: transitive
    description:
      name: crypto
      sha256: "1e445881f28f22d6140f181e07737b22f1e099a5e1ff94b0af2f9e4a463f4855"
      url: "https://pub.dev"
    source: hosted
    version: "3.0.6"
  cupertino_icons:
    dependency: "direct main"
    description:
      name: cupertino_icons
      sha256: ba631d1c7f7bef6b729a622b7b752645a2d076dba9976925b8f25725a30e1ee6
      url: "https://pub.dev"
    source: hosted
    version: "1.0.8"
  dartz:
    dependency: "direct main"
    description:
      name: dartz
      sha256: e6acf34ad2e31b1eb00948692468c30ab48ac8250e0f0df661e29f12dd252168
      url: "https://pub.dev"
    source: hosted
    version: "0.10.1"
  equatable:
    dependency: "direct main"
    description:
      name: equatable
      sha256: "567c64b3cb4cf82397aac55f4f0cbd3ca20d77c6c03bedbc4ceaddc08904aef7"
      url: "https://pub.dev"
    source: hosted
    version: "2.0.7"
  fake_async:
    dependency: transitive
    description:
      name: fake_async
      sha256: "5368f224a74523e8d2e7399ea1638b37aecfca824a3cc4dfdf77bf1fa905ac44"
      url: "https://pub.dev"
    source: hosted
    version: "1.3.3"
  ffi:
    dependency: transitive
    description:
      name: ffi
      sha256: "289279317b4b16eb2bb7e271abccd4bf84ec9bdcbe999e278a94b804f5630418"
      url: "https://pub.dev"
    source: hosted
    version: "2.1.4"
  file:
    dependency: transitive
    description:
      name: file
      sha256: a3b4f84adafef897088c160faf7dfffb7696046cb13ae90b508c2cbc95d3b8d4
      url: "https://pub.dev"
    source: hosted
    version: "7.0.1"
  flutter:
    dependency: "direct main"
    description: flutter
    source: sdk
    version: "0.0.0"
  flutter_bloc:
    dependency: "direct main"
    description:
      name: flutter_bloc
      sha256: cf51747952201a455a1c840f8171d273be009b932c75093020f9af64f2123e38
      url: "https://pub.dev"
    source: hosted
    version: "9.1.1"
  flutter_lints:
    dependency: "direct dev"
    description:
      name: flutter_lints
      sha256: "5398f14efa795ffb7a33e9b6a08798b26a180edac4ad7db3f231e40f82ce11e1"
      url: "https://pub.dev"
    source: hosted
    version: "5.0.0"
  flutter_svg:
    dependency: "direct main"
    description:
      name: flutter_svg
      sha256: cd57f7969b4679317c17af6fd16ee233c1e60a82ed209d8a475c54fd6fd6f845
      url: "https://pub.dev"
    source: hosted
    version: "2.2.0"
  flutter_test:
    dependency: "direct dev"
    description: flutter
    source: sdk
    version: "0.0.0"
  flutter_web_plugins:
    dependency: transitive
    description: flutter
    source: sdk
    version: "0.0.0"
  get_it:
    dependency: "direct main"
    description:
      name: get_it
      sha256: f126a3e286b7f5b578bf436d5592968706c4c1de28a228b870ce375d9f743103
      url: "https://pub.dev"
    source: hosted
    version: "8.0.3"
  google_fonts:
    dependency: "direct main"
    description:
      name: google_fonts
      sha256: b1ac0fe2832c9cc95e5e88b57d627c5e68c223b9657f4b96e1487aa9098c7b82
      url: "https://pub.dev"
    source: hosted
    version: "6.2.1"
  http:
    dependency: "direct main"
    description:
      name: http
      sha256: "2c11f3f94c687ee9bad77c171151672986360b2b001d109814ee7140b2cf261b"
      url: "https://pub.dev"
    source: hosted
    version: "1.4.0"
  http_parser:
    dependency: transitive
    description:
      name: http_parser
      sha256: "178d74305e7866013777bab2c3d8726205dc5a4dd935297175b19a23a2e66571"
      url: "https://pub.dev"
    source: hosted
    version: "4.1.2"
  intl:
    dependency: "direct main"
    description:
      name: intl
      sha256: "3df61194eb431efc39c4ceba583b95633a403f46c9fd341e550ce0bfa50e9aa5"
      url: "https://pub.dev"
    source: hosted
    version: "0.20.2"
  leak_tracker:
    dependency: transitive
    description:
      name: leak_tracker
      sha256: "6bb818ecbdffe216e81182c2f0714a2e62b593f4a4f13098713ff1685dfb6ab0"
      url: "https://pub.dev"
    source: hosted
    version: "10.0.9"
  leak_tracker_flutter_testing:
    dependency: transitive
    description:
      name: leak_tracker_flutter_testing
      sha256: f8b613e7e6a13ec79cfdc0e97638fddb3ab848452eff057653abd3edba760573
      url: "https://pub.dev"
    source: hosted
    version: "3.0.9"
  leak_tracker_testing:
    dependency: transitive
    description:
      name: leak_tracker_testing
      sha256: "6ba465d5d76e67ddf503e1161d1f4a6bc42306f9d66ca1e8f079a47290fb06d3"
      url: "https://pub.dev"
    source: hosted
    version: "3.0.1"
  lints:
    dependency: transitive
    description:
      name: lints
      sha256: c35bb79562d980e9a453fc715854e1ed39e24e7d0297a880ef54e17f9874a9d7
      url: "https://pub.dev"
    source: hosted
    version: "5.1.1"
  matcher:
    dependency: transitive
    description:
      name: matcher
      sha256: dc58c723c3c24bf8d3e2d3ad3f2f9d7bd9cf43ec6feaa64181775e60190153f2
      url: "https://pub.dev"
    source: hosted
    version: "0.12.17"
  material_color_utilities:
    dependency: transitive
    description:
      name: material_color_utilities
      sha256: f7142bb1154231d7ea5f96bc7bde4bda2a0945d2806bb11670e30b850d56bdec
      url: "https://pub.dev"
    source: hosted
    version: "0.11.1"
  meta:
    dependency: transitive
    description:
      name: meta
      sha256: e3641ec5d63ebf0d9b41bd43201a66e3fc79a65db5f61fc181f04cd27aab950c
      url: "https://pub.dev"
    source: hosted
    version: "1.16.0"
  nested:
    dependency: transitive
    description:
      name: nested
      sha256: "03bac4c528c64c95c722ec99280375a6f2fc708eec17c7b3f07253b626cd2a20"
      url: "https://pub.dev"
    source: hosted
    version: "1.0.0"
  path:
    dependency: transitive
    description:
      name: path
      sha256: "75cca69d1490965be98c73ceaea117e8a04dd21217b37b292c9ddbec0d955bc5"
      url: "https://pub.dev"
    source: hosted
    version: "1.9.1"
  path_parsing:
    dependency: transitive
    description:
      name: path_parsing
      sha256: "883402936929eac138ee0a45da5b0f2c80f89913e6dc3bf77eb65b84b409c6ca"
      url: "https://pub.dev"
    source: hosted
    version: "1.1.0"
  path_provider:
    dependency: transitive
    description:
      name: path_provider
      sha256: "50c5dd5b6e1aaf6fb3a78b33f6aa3afca52bf903a8a5298f53101fdaee55bbcd"
      url: "https://pub.dev"
    source: hosted
    version: "2.1.5"
  path_provider_android:
    dependency: transitive
    description:
      name: path_provider_android
      sha256: d0d310befe2c8ab9e7f393288ccbb11b60c019c6b5afc21973eeee4dda2b35e9
      url: "https://pub.dev"
    source: hosted
    version: "2.2.17"
  path_provider_foundation:
    dependency: transitive
    description:
      name: path_provider_foundation
      sha256: "4843174df4d288f5e29185bd6e72a6fbdf5a4a4602717eed565497429f179942"
      url: "https://pub.dev"
    source: hosted
    version: "2.4.1"
  path_provider_linux:
    dependency: transitive
    description:
      name: path_provider_linux
      sha256: f7a1fe3a634fe7734c8d3f2766ad746ae2a2884abe22e241a8b301bf5cac3279
      url: "https://pub.dev"
    source: hosted
    version: "2.2.1"
  path_provider_platform_interface:
    dependency: transitive
    description:
      name: path_provider_platform_interface
      sha256: "88f5779f72ba699763fa3a3b06aa4bf6de76c8e5de842cf6f29e2e06476c2334"
      url: "https://pub.dev"
    source: hosted
    version: "2.1.2"
  path_provider_windows:
    dependency: transitive
    description:
      name: path_provider_windows
      sha256: bd6f00dbd873bfb70d0761682da2b3a2c2fccc2b9e84c495821639601d81afe7
      url: "https://pub.dev"
    source: hosted
    version: "2.3.0"
  petitparser:
    dependency: transitive
    description:
      name: petitparser
      sha256: "07c8f0b1913bcde1ff0d26e57ace2f3012ccbf2b204e070290dad3bb22797646"
      url: "https://pub.dev"
    source: hosted
    version: "6.1.0"
  platform:
    dependency: transitive
    description:
      name: platform
      sha256: "5d6b1b0036a5f331ebc77c850ebc8506cbc1e9416c27e59b439f917a902a4984"
      url: "https://pub.dev"
    source: hosted
    version: "3.1.6"
  plugin_platform_interface:
    dependency: transitive
    description:
      name: plugin_platform_interface
      sha256: "4820fbfdb9478b1ebae27888254d445073732dae3d6ea81f0b7e06d5dedc3f02"
      url: "https://pub.dev"
    source: hosted
    version: "2.1.8"
  provider:
    dependency: transitive
    description:
      name: provider
      sha256: "4abbd070a04e9ddc287673bf5a030c7ca8b685ff70218720abab8b092f53dd84"
      url: "https://pub.dev"
    source: hosted
    version: "6.1.5"
  shared_preferences:
    dependency: "direct main"
    description:
      name: shared_preferences
      sha256: "6e8bf70b7fef813df4e9a36f658ac46d107db4b4cfe1048b477d4e453a8159f5"
      url: "https://pub.dev"
    source: hosted
    version: "2.5.3"
  shared_preferences_android:
    dependency: transitive
    description:
      name: shared_preferences_android
      sha256: "20cbd561f743a342c76c151d6ddb93a9ce6005751e7aa458baad3858bfbfb6ac"
      url: "https://pub.dev"
    source: hosted
    version: "2.4.10"
  shared_preferences_foundation:
    dependency: transitive
    description:
      name: shared_preferences_foundation
      sha256: "6a52cfcdaeac77cad8c97b539ff688ccfc458c007b4db12be584fbe5c0e49e03"
      url: "https://pub.dev"
    source: hosted
    version: "2.5.4"
  shared_preferences_linux:
    dependency: transitive
    description:
      name: shared_preferences_linux
      sha256: "580abfd40f415611503cae30adf626e6656dfb2f0cee8f465ece7b6defb40f2f"
      url: "https://pub.dev"
    source: hosted
    version: "2.4.1"
```

```
shared_preferences_platform_interface:
  dependency: transitive
  description:
    name: shared_preferences_platform_interface
    sha256: "57cbf196c486bc2cf1f02b85784932c6094376284b3ad5779d1b1c6c6a816b80"
    url: "https://pub.dev"
  source: hosted
  version: "2.4.1"
shared_preferences_web:
  dependency: transitive
  description:
    name: shared_preferences_web
    sha256: c49bd060261c9a3f0ff445892695d6212ff603ef3115edbb448509d407600019
    url: "https://pub.dev"
  source: hosted
  version: "2.4.3"
shared_preferences_windows:
  dependency: transitive
  description:
    name: shared_preferences_windows
    sha256: "94ef0f72b2d71bc3e700e025db3710911bd51a71cefb65cc609dd0d9a982e3c1"
    url: "https://pub.dev"
  source: hosted
  version: "2.4.1"
sky_engine:
  dependency: transitive
  description: flutter
  source: sdk
  version: "0.0.0"
source_span:
  dependency: transitive
  description:
    name: source_span
    sha256: "254ee5351d6cb365c859e20ee823c3bb479bf4a293c22d17a9f1bf144ce86f7c"
    url: "https://pub.dev"
  source: hosted
  version: "1.10.1"
stack_trace:
  dependency: transitive
  description:
    name: stack_trace
    sha256: "8b27215b45d22309b5cddda1aa2b19bdfec9df0e765f2de506401c071d38d1b1"
    url: "https://pub.dev"
  source: hosted
  version: "1.12.1"
stream_channel:
  dependency: transitive
  description:
    name: stream_channel
    sha256: "969e04c80b8bcdf826f8f16579c7b14d780458bd97f56d107d3950fdbeef059d"
    url: "https://pub.dev"
  source: hosted
  version: "2.1.4"
string_scanner:
  dependency: transitive
  description:
    name: string_scanner
    sha256: "921cd31725b72fe181906c6a94d987c78e3b98c2e205b397ea399d4054872b43"
    url: "https://pub.dev"
  source: hosted
  version: "1.4.1"
term_glyph:
  dependency: transitive
  description:
    name: term_glyph
    sha256: "7f554798625ea768a7518313e58f83891c7f5024f88e46e7182a4558850a4b8e"
```

```
      url: "https://pub.dev"
    source: hosted
    version: "1.2.2"
  test_api:
    dependency: transitive
    description:
      name: test_api
      sha256: fb31f383e2ee25fbbfe06b40fe21e1e458d14080e3c67e7ba0acfde4df4e0bbd
      url: "https://pub.dev"
    source: hosted
    version: "0.7.4"
  typed_data:
    dependency: transitive
    description:
      name: typed_data
      sha256: f9049c039ebfeb4cf7a7104a675823cd72dba8297f264b6637062516699fa006
      url: "https://pub.dev"
    source: hosted
    version: "1.4.0"
  vector_graphics:
    dependency: transitive
    description:
      name: vector_graphics
      sha256: a4f059dc26fc8295b5921376600a194c4ec7d55e72f2fe4c7d2831e103d461e6
      url: "https://pub.dev"
    source: hosted
    version: "1.1.19"
  vector_graphics_codec:
    dependency: transitive
    description:
      name: vector_graphics_codec
      sha256: "99fd9fbd34d9f9a32efd7b6a6aae14125d8237b10403b422a6a6dfeac2806146"
      url: "https://pub.dev"
    source: hosted
    version: "1.1.13"
  vector_graphics_compiler:
    dependency: transitive
    description:
      name: vector_graphics_compiler
      sha256: "557a315b7d2a6dbb0aaaff84d857967ce6bdc96a63dc6ee2a57ce5a6ee5d3331"
      url: "https://pub.dev"
    source: hosted
    version: "1.1.17"
  vector_math:
    dependency: transitive
    description:
      name: vector_math
      sha256: "80b3257d1492ce4d091729e3a67a60407d227c27241d6927be0130c98e741803"
      url: "https://pub.dev"
    source: hosted
    version: "2.1.4"
  vm_service:
    dependency: transitive
    description:
      name: vm_service
      sha256: ddfa8d30d89985b96407efce8acbdd124701f96741f2d981ca860662f1c0dc02
      url: "https://pub.dev"
    source: hosted
    version: "15.0.0"
  web:
    dependency: transitive
    description:
      name: web
      sha256: "868d88a33d8a87b18ffc05f9f030ba328ffefba92d6c127917a2ba740f9cfe4a"
      url: "https://pub.dev"
    source: hosted
    version: "1.1.1"
  xdg_directories:
    dependency: transitive
    description:
      name: xdg_directories
      sha256: "7a3f37b05d989967cdddcbb571f1ea834867ae2faa29725fd085180e0883aa15"
      url: "https://pub.dev"
    source: hosted
    version: "1.1.0"
  xml:
    dependency: transitive
    description:
      name: xml
      sha256: b015a8ad1c488f66851d762d3090a21c600e479dc75e68328c52774040cf9226
      url: "https://pub.dev"
    source: hosted
    version: "6.5.0"
sdks:
  dart: ">=3.8.1 <4.0.0"
  flutter: ">=3.27.0"
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\pubspec.yaml

```
name: wismon_keuangan
description: "A new Flutter project."
# The following line prevents the package from being accidentally published to
# pub.dev using `flutter pub publish`. This is preferred for private packages.
publish_to: "none" # Remove this line if you wish to publish to pub.dev

# The following defines the version and build number for your application.
# A version number is three numbers separated by dots, like 1.2.43
# followed by an optional build number separated by a +.
# Both the version and the builder number may be overridden in flutter
# build by specifying --build-name and --build-number, respectively.
# In Android, build-name is used as versionName while build-number used as versionCode.
# Read more about Android versioning at https://developer.android.com/studio/publish/versioning
# In iOS, build-name is used as CFBundleShortVersionString while build-number is used as CFBundleVersion.
# Read more about iOS versioning at
#
https://developer.apple.com/library/archive/documentation/General/Reference/InfoPlistKeyReference/Articles/CoreFoundationKeys.html
# In Windows, build-name is used as the major, minor, and patch parts
# of the product and file versions while build-number is used as the build suffix.
version: 1.0.0+1

environment:
  sdk: ^3.8.1

# Dependencies specify other packages that your package needs in order to work.
# To automatically upgrade your package dependencies to the latest versions
# consider running `flutter pub upgrade --major-versions`. Alternatively,
# dependencies can be manually updated by changing the version numbers below to
# the latest version available on pub.dev. To see which dependencies have newer
# versions available, run `flutter pub outdated`.
dependencies:
  flutter:
    sdk: flutter
  google_fonts: ^6.1.0

  # The following adds the Cupertino Icons font to your application.
  # Use with the CupertinoIcons class for iOS style icons.
  cupertino_icons: ^1.0.8

  # State management
```

```yaml
  flutter_bloc: ^9.1.1
  equatable: ^2.0.5

  # HTTP requests
  http: ^1.1.0

  # Shared preferences for token storage
  shared_preferences: ^2.2.2

  # Functional programming
  dartz: ^0.10.1

  # Dependency injection
  get_it: ^8.0.2
  intl: ^0.20.2
  flutter_svg: ^2.2.0

dev_dependencies:
  flutter_test:
    sdk: flutter

  # The "flutter_lints" package below contains a set of recommended lints to
  # encourage good coding practices. The lint set provided by the package is
  # activated in the `analysis_options.yaml` file located at the root of your
  # package. See that file for information about deactivating specific lint
  # rules and activating additional ones.
  flutter_lints: ^5.0.0

# For information on the generic Dart part of this file, see the
# following page: https://dart.dev/tools/pub/pubspec

# The following section is specific to Flutter packages.
flutter:
  # The following line ensures that the Material Icons font is
  # included with your application, so that you can use the icons in
  # the material Icons class.
  uses-material-design: true

  # To add assets to your application, add an assets section, like this:
  assets:
    - assets/LOGOWHS.svg
    - assets/LogoWHNProfil.svg
    - assets/images/menu_background.svg
    - assets/images/profile_background.svg

  # An image asset can refer to one or more resolution-specific "variants", see
  # https://flutter.dev/to/resolution-aware-images

  # For details regarding adding assets from package dependencies, see
  # https://flutter.dev/to/asset-from-package

  # To add custom fonts to your application, add a fonts section here,
  # in this "flutter" section. Each entry in this list should have a
  # "family" key with the font family name, and a "fonts" key with a
  # list giving the asset and other descriptors for the font. For
  # example:
  # fonts:
  #   - family: Schyler
  #     fonts:
  #       - asset: fonts/Schyler-Regular.ttf
  #       - asset: fonts/Schyler-Italic.ttf
  #         style: italic
  #   - family: Trajan Pro
  #     fonts:
  #       - asset: fonts/TrajanPro.ttf
  #       - asset: fonts/TrajanPro_Bold.ttf
  #         weight: 700
  #
  # For details regarding fonts from package dependencies,
  # see https://flutter.dev/to/font-from-package
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\README.md

```
# wismon_keuangan

A new Flutter project.

## Getting Started

This project is a starting point for a Flutter application.

A few resources to get you started if this is your first Flutter project:

- [Lab: Write your first Flutter app](https://docs.flutter.dev/get-started/codelab)
- [Cookbook: Useful Flutter samples](https://docs.flutter.dev/cookbook)

For help getting started with Flutter development, view the
[online documentation](https://docs.flutter.dev/), which offers tutorials,
samples, guidance on mobile development, and a full API reference.
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\android\.gitignore

```
gradle-wrapper.jar
/.gradle
/captures/
/gradlew
/gradlew.bat
/local.properties
GeneratedPluginRegistrant.java
.cxx/

# Remember to never publicly share your keystore.
# See https://flutter.dev/to/reference-keystore
key.properties
**/*.keystore
**/*.jks
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\android\build.gradle.kts

```
allprojects {
    repositories {
        google()
        mavenCentral()
    }
}

val newBuildDir: Directory = rootProject.layout.buildDirectory.dir("../../build").get()
rootProject.layout.buildDirectory.value(newBuildDir)

subprojects {
    val newSubprojectBuildDir: Directory = newBuildDir.dir(project.name)
    project.layout.buildDirectory.value(newSubprojectBuildDir)
}
subprojects {
    project.evaluationDependsOn(":app")
}
```

```
tasks.register<Delete>("clean") {
    delete(rootProject.layout.buildDirectory)
}
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\android\gradle.properties

```
org.gradle.jvmargs=-Xmx8G                 -XX:MaxMetaspaceSize=4G                 -XX:ReservedCodeCacheSize=512m
-XX:+HeapDumpOnOutOfMemoryError
android.useAndroidX=true
android.enableJetifier=true
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\android\settings.gradle.kts

```
pluginManagement {
    val flutterSdkPath = run {
        val properties = java.util.Properties()
        file("local.properties").inputStream().use { properties.load(it) }
        val flutterSdkPath = properties.getProperty("flutter.sdk")
        require(flutterSdkPath != null) { "flutter.sdk not set in local.properties" }
        flutterSdkPath
    }

    includeBuild("$flutterSdkPath/packages/flutter_tools/gradle")

    repositories {
        google()
        mavenCentral()
        gradlePluginPortal()
    }
}

plugins {
    id("dev.flutter.flutter-plugin-loader") version "1.0.0"
    id("com.android.application") version "8.7.3" apply false
    id("org.jetbrains.kotlin.android") version "2.1.0" apply false
}

include(":app")
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\android\app\build.gradle.kts

```
plugins {
    id("com.android.application")
    id("kotlin-android")
    // The Flutter Gradle Plugin must be applied after the Android and Kotlin Gradle plugins.
    id("dev.flutter.flutter-gradle-plugin")
}

android {
    namespace = "com.example.wismon_keuangan"
    compileSdk = flutter.compileSdkVersion
    ndkVersion = flutter.ndkVersion

    compileOptions {
        sourceCompatibility = JavaVersion.VERSION_11
        targetCompatibility = JavaVersion.VERSION_11
    }

    kotlinOptions {
        jvmTarget = JavaVersion.VERSION_11.toString()
    }
    defaultConfig {
        // TODO: Specify your own unique Application ID (https://developer.android.com/studio/build/application-id.html).
        applicationId = "com.example.wismon_keuangan"
        // You can update the following values to match your application needs.
        // For more information, see: https://flutter.dev/to/review-gradle-config.
        minSdk = flutter.minSdkVersion
        targetSdk = flutter.targetSdkVersion
        versionCode = flutter.versionCode
        versionName = flutter.versionName
    }

    buildTypes {
        release {
            // TODO: Add your own signing config for the release build.
            // Signing with the debug keys for now, so `flutter run --release` works.
            signingConfig = signingConfigs.getByName("debug")
        }
    }
}

flutter {
    source = "../.."
}
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\android\app\src\debug\AndroidManifest.xml

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android">
    <!-- The INTERNET permission is required for development. Specifically,
         the Flutter tool needs it to communicate with the running application
         to allow setting breakpoints, to provide hot reload, etc.
    -->
    <uses-permission android:name="android.permission.INTERNET"/>
</manifest>
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\android\app\src\main\AndroidManifest.xml

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android">
    <application
        android:label="wismon_keuangan"
        android:name="${applicationName}"
        android:icon="@mipmap/ic_launcher"
        android:enableOnBackInvokedCallback="true">
        <activity
            android:name=".MainActivity"
            android:exported="true"
            android:launchMode="singleTop"
            android:taskAffinity=""
            android:theme="@style/LaunchTheme"
            android:configChanges="orientation|keyboardHidden|keyboard|screenSize|smallestScreenSize|locale|layoutDirection|fontScale|screenLayout|density|uiMode"
            android:hardwareAccelerated="true"
            android:windowSoftInputMode="adjustResize">
            <!-- Specifies an Android theme to apply to this Activity as soon as
                 the Android process has started. This theme is visible to the user
                 while the Flutter UI initializes. After that, this theme continues
                 to determine the Window background behind the Flutter UI. -->
            <meta-data
              android:name="io.flutter.embedding.android.NormalTheme"
              android:resource="@style/NormalTheme"
              />
            <intent-filter>
                <action android:name="android.intent.action.MAIN"/>
                <category android:name="android.intent.category.LAUNCHER"/>
            </intent-filter>
        </activity>
        <!-- Don't delete the meta-data below.
             This is used by the Flutter tool to generate GeneratedPluginRegistrant.java -->
        <meta-data
            android:name="flutterEmbedding"
            android:value="2" />
    </application>
    <!-- Required to query activities that can process text, see:
         https://developer.android.com/training/package-visibility and
         https://developer.android.com/reference/android/content/Intent#ACTION_PROCESS_TEXT.

         In particular, this is used by the Flutter engine in io.flutter.plugin.text.ProcessTextPlugin. -->
    <queries>
        <intent>
            <action android:name="android.intent.action.PROCESS_TEXT"/>
            <data android:mimeType="text/plain"/>
        </intent>
    </queries>
    <!-- The INTERNET permission is required for development. -->
    <uses-permission android:name="android.permission.INTERNET"/>
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
</manifest>
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\android\app\src\main\kotlin\com\example\wismon_keuangan\MainActivity.kt

```
package com.example.wismon_keuangan

import io.flutter.embedding.android.FlutterActivity

class MainActivity : FlutterActivity()
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\android\app\src\main\res\drawable\launch_background.xml

```
<¿xml version="1.0" encoding="utf-8"¿>
<!-- Modify this file to customize your launch splash screen -->
<layer-list xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:drawable="@android:color/white" />

    <!-- You can insert your own image assets here -->
    <!-- <item>
        <bitmap
            android:gravity="center"
            android:src="@mipmap/launch_image" />
    </item> -->
</layer-list>
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\android\app\src\main\res\drawable-v21\launch_background.xml

```
<¿xml version="1.0" encoding="utf-8"¿>
<!-- Modify this file to customize your launch splash screen -->
<layer-list xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:drawable="¿android:colorBackground" />

    <!-- You can insert your own image assets here -->
    <!-- <item>
        <bitmap
            android:gravity="center"
            android:src="@mipmap/launch_image" />
    </item> -->
</layer-list>
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\android\app\src\main\res\values\styles.xml

```
<¿xml version="1.0" encoding="utf-8"¿>
<resources>
    <!-- Theme applied to the Android Window while the process is starting when the OS's Dark Mode setting is
off -->
    <style name="LaunchTheme" parent="@android:style/Theme.Light.NoTitleBar">
        <!-- Show a splash screen on the activity. Automatically removed when
             the Flutter engine draws its first frame -->
        <item name="android:windowBackground">@drawable/launch_background</item>
    </style>
    <!-- Theme applied to the Android Window as soon as the process has started.
         This theme determines the color of the Android Window while your
         Flutter UI initializes, as well as behind your Flutter UI while its
         running.

         This Theme is only used starting with V2 of Flutter's Android embedding. -->
    <style name="NormalTheme" parent="@android:style/Theme.Light.NoTitleBar">
        <item name="android:windowBackground">¿android:colorBackground</item>
    </style>
</resources>
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\android\app\src\main\res\values-night\styles.xml

```
<¿xml version="1.0" encoding="utf-8"¿>
<resources>
    <!-- Theme applied to the Android Window while the process is starting when the OS's Dark Mode setting is
on -->
    <style name="LaunchTheme" parent="@android:style/Theme.Black.NoTitleBar">
        <!-- Show a splash screen on the activity. Automatically removed when
             the Flutter engine draws its first frame -->
        <item name="android:windowBackground">@drawable/launch_background</item>
    </style>
    <!-- Theme applied to the Android Window as soon as the process has started.
         This theme determines the color of the Android Window while your
         Flutter UI initializes, as well as behind your Flutter UI while its
         running.

         This Theme is only used starting with V2 of Flutter's Android embedding. -->
    <style name="NormalTheme" parent="@android:style/Theme.Black.NoTitleBar">
        <item name="android:windowBackground">¿android:colorBackground</item>
    </style>
</resources>
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\android\app\src\profile\AndroidManifest.xml

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android">
    <!-- The INTERNET permission is required for development. Specifically,
         the Flutter tool needs it to communicate with the running application
         to allow setting breakpoints, to provide hot reload, etc.
    -->
    <uses-permission android:name="android.permission.INTERNET"/>
</manifest>
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\android\gradle\wrapper\gradle-wrapper.properties

```
distributionBase=GRADLE_USER_HOME
distributionPath=wrapper/dists
zipStoreBase=GRADLE_USER_HOME
zipStorePath=wrapper/dists
distributionUrl=https\://services.gradle.org/distributions/gradle-8.12-all.zip
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\assets\LogoWHNProfil.svg

```
<svg width="402" height="391" viewBox="0 0 402 391" fill="none" xmlns="http://www.w3.org/2000/svg">
<path d="M129 297L258 396L258 198L129 297Z" fill="#207BB5"/>
<path d="M258 198L387 297L387 99L258 198Z" fill="#207BB5"/>
<path d="M258 396L387 495L387 297L258 396Z" fill="#207BB5"/>
<path d="M129 99.0002L3.91724e-05 198L4.57764e-05 0L129 99.0002Z" fill="#135EA2"/>
<path d="M387 297L258 396L258 198L387 297Z" fill="#135EA2"/>
<path d="M258 198L129 297L129 99L258 198Z" fill="#135EA2"/>
<path d="M129 297L3.91724e-05 396L4.57764e-05 198L129 297Z" fill="#135EA2"/>
<path d="M258 396L129 495L129 297L258 396Z" fill="#135EA2"/>
<path d="M-4.32743e-06 198L129 297L129 99L-4.32743e-06 198Z" fill="#207BB5"/>
<path d="M387 297L516 396L516 198L387 297Z" fill="#207BB5"/>
```

```
<path d="M516 198L387 297L387 99L516 198Z" fill="#135EA2"/>
<path d="M516 396L387 495L387 297L516 396Z" fill="#135EA2"/>
</svg>
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\assets\LOGOWHS.svg

```
<svg width="297" height="766" viewBox="0 0 297 766" fill="none" xmlns="http://www.w3.org/2000/svg">
<path d="M297 802L396 673L198 673L297 802Z" fill="#207BB5"/>
<path d="M99.0002 802L198 673L-5.63877e-06 673L99.0002 802Z" fill="#207BB5"/>
<path d="M297 544L396 415L198 415L297 544Z" fill="#207BB5"/>
<path d="M198 415L297 286L99 286L198 415Z" fill="#207BB5"/>
<path d="M99.0002 544L198 673L5.63877e-06 673L99.0002 544Z" fill="#135EA2"/>
<path d="M297 286L396 415L198 415L297 286Z" fill="#135EA2"/>
<path d="M198 415L297 544L99 544L198 415Z" fill="#135EA2"/>
<path d="M297 544L396 673L198 673L297 544Z" fill="#135EA2"/>
<path d="M198 673L297 544L99 544L198 673Z" fill="#207BB5"/>
<path d="M198 673L99 802L297 802L198 673Z" fill="#135EA2"/>
<path d="M297 28L396 157L198 157L297 28Z" fill="#135EA2"/>
<path d="M297 286L396 157L198 157L297 286Z" fill="#207BB5"/>
<path d="M198 157L297 286L99 286L198 157Z" fill="#135EA2"/>
</svg>
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\assets\images\menu_background.svg

```
<¿xml version="1.0" encoding="UTF-8"¿>
<svg width="802" height="396" xmlns="http://www.w3.org/2000/svg">
  <rect width="802" height="396" fill="#E5F3FF" opacity="0.3"/>
    <text x="401" y="198" text-anchor="middle" font-family="Arial" font-size="24" fill="#135EA2">Menu Background</text>
</svg>
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\assets\images\profile_background.svg

```
<¿xml version="1.0" encoding="UTF-8"¿>
<svg width="802" height="396" xmlns="http://www.w3.org/2000/svg">
  <rect width="802" height="396" fill="#E5F3FF" opacity="0.3"/>
    <text x="401" y="198" text-anchor="middle" font-family="Arial" font-size="24" fill="#207BB5">Profile Background</text>
</svg>
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\ios\.gitignore

```
**/dgph
*.mode1v3
*.mode2v3
*.moved-aside
*.pbxuser
*.perspectivev3
**/*sync/
.sconsign.dblite
.tags*
**/.vagrant/
**/DerivedData/
Icon¿
**/Pods/
**/.symlinks/
profile
```

```
xcuserdata
**/.generated/
Flutter/App.framework
Flutter/Flutter.framework
Flutter/Flutter.podspec
Flutter/Generated.xcconfig
Flutter/ephemeral/
Flutter/app.flx
Flutter/app.zip
Flutter/flutter_assets/
Flutter/flutter_export_environment.sh
ServiceDefinitions.json
Runner/GeneratedPluginRegistrant.*

# Exceptions to above rules.
!default.mode1v3
!default.mode2v3
!default.pbxuser
!default.perspectivev3
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\ios\Flutter\AppFrameworkInfo.plist

```
<¿xml version="1.0" encoding="UTF-8"¿>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
  <key>CFBundleDevelopmentRegion</key>
  <string>en</string>
  <key>CFBundleExecutable</key>
  <string>App</string>
  <key>CFBundleIdentifier</key>
  <string>io.flutter.flutter.app</string>
  <key>CFBundleInfoDictionaryVersion</key>
  <string>6.0</string>
  <key>CFBundleName</key>
  <string>App</string>
  <key>CFBundlePackageType</key>
  <string>FMWK</string>
  <key>CFBundleShortVersionString</key>
  <string>1.0</string>
  <key>CFBundleSignature</key>
  <string>¿¿¿¿</string>
  <key>CFBundleVersion</key>
  <string>1.0</string>
  <key>MinimumOSVersion</key>
  <string>12.0</string>
</dict>
</plist>
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\ios\Flutter\Debug.xcconfig

```
#include "Generated.xcconfig"
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\ios\Flutter\Release.xcconfig

```
#include "Generated.xcconfig"
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\ios\Runner\AppDelegate.swift

```
import Flutter
import UIKit

@main
@objc class AppDelegate: FlutterAppDelegate {
  override func application(
    _ application: UIApplication,
    didFinishLaunchingWithOptions launchOptions: [UIApplication.LaunchOptionsKey: Any]¿
  ) -> Bool {
    GeneratedPluginRegistrant.register(with: self)
    return super.application(application, didFinishLaunchingWithOptions: launchOptions)
  }
}
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\ios\Runner\Info.plist

```
<¿xml version="1.0" encoding="UTF-8"¿>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
 <key>CFBundleDevelopmentRegion</key>
 <string>$(DEVELOPMENT_LANGUAGE)</string>
 <key>CFBundleDisplayName</key>
 <string>Wismon Keuangan</string>
 <key>CFBundleExecutable</key>
 <string>$(EXECUTABLE_NAME)</string>
 <key>CFBundleIdentifier</key>
 <string>$(PRODUCT_BUNDLE_IDENTIFIER)</string>
 <key>CFBundleInfoDictionaryVersion</key>
 <string>6.0</string>
 <key>CFBundleName</key>
 <string>wismon_keuangan</string>
 <key>CFBundlePackageType</key>
 <string>APPL</string>
 <key>CFBundleShortVersionString</key>
 <string>$(FLUTTER_BUILD_NAME)</string>
 <key>CFBundleSignature</key>
 <string>¿¿¿¿</string>
 <key>CFBundleVersion</key>
 <string>$(FLUTTER_BUILD_NUMBER)</string>
 <key>LSRequiresIPhoneOS</key>
 <true/>
 <key>UILaunchStoryboardName</key>
 <string>LaunchScreen</string>
 <key>UIMainStoryboardFile</key>
 <string>Main</string>
 <key>UISupportedInterfaceOrientations</key>
 <array>
  <string>UIInterfaceOrientationPortrait</string>
  <string>UIInterfaceOrientationLandscapeLeft</string>
  <string>UIInterfaceOrientationLandscapeRight</string>
 </array>
 <key>UISupportedInterfaceOrientations~ipad</key>
 <array>
  <string>UIInterfaceOrientationPortrait</string>
  <string>UIInterfaceOrientationPortraitUpsideDown</string>
  <string>UIInterfaceOrientationLandscapeLeft</string>
  <string>UIInterfaceOrientationLandscapeRight</string>
 </array>
 <key>CADisableMinimumFrameDurationOnPhone</key>
 <true/>
 <key>UIApplicationSupportsIndirectInputEvents</key>
 <true/>
</dict>
</plist>
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\ios\Runner\Runner-Bridging-Header.h

```
#import "GeneratedPluginRegistrant.h"
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\ios\Runner\Assets.xcassets\AppIcon.appiconset\Contents.json

```
{
  "images" : [
    {
      "size" : "20x20",
      "idiom" : "iphone",
      "filename" : "Icon-App-20x20@2x.png",
      "scale" : "2x"
    },
    {
      "size" : "20x20",
      "idiom" : "iphone",
      "filename" : "Icon-App-20x20@3x.png",
      "scale" : "3x"
    },
    {
      "size" : "29x29",
      "idiom" : "iphone",
      "filename" : "Icon-App-29x29@1x.png",
      "scale" : "1x"
    },
    {
      "size" : "29x29",
      "idiom" : "iphone",
      "filename" : "Icon-App-29x29@2x.png",
      "scale" : "2x"
    },
    {
      "size" : "29x29",
      "idiom" : "iphone",
      "filename" : "Icon-App-29x29@3x.png",
      "scale" : "3x"
    },
    {
      "size" : "40x40",
      "idiom" : "iphone",
      "filename" : "Icon-App-40x40@2x.png",
      "scale" : "2x"
    },
    {
      "size" : "40x40",
      "idiom" : "iphone",
      "filename" : "Icon-App-40x40@3x.png",
      "scale" : "3x"
    },
    {
      "size" : "60x60",
      "idiom" : "iphone",
      "filename" : "Icon-App-60x60@2x.png",
      "scale" : "2x"
    },
    {
      "size" : "60x60",
      "idiom" : "iphone",
      "filename" : "Icon-App-60x60@3x.png",
      "scale" : "3x"
    },
    {
      "size" : "20x20",
      "idiom" : "ipad",
      "filename" : "Icon-App-20x20@1x.png",
      "scale" : "1x"
    },
    {
      "size" : "20x20",
      "idiom" : "ipad",
      "filename" : "Icon-App-20x20@2x.png",
      "scale" : "2x"
    },
    {
      "size" : "29x29",
      "idiom" : "ipad",
      "filename" : "Icon-App-29x29@1x.png",
      "scale" : "1x"
    },
    {
      "size" : "29x29",
      "idiom" : "ipad",
      "filename" : "Icon-App-29x29@2x.png",
      "scale" : "2x"
    },
    {
      "size" : "40x40",
      "idiom" : "ipad",
      "filename" : "Icon-App-40x40@1x.png",
      "scale" : "1x"
    },
    {
      "size" : "40x40",
      "idiom" : "ipad",
      "filename" : "Icon-App-40x40@2x.png",
      "scale" : "2x"
    },
    {
      "size" : "76x76",
      "idiom" : "ipad",
      "filename" : "Icon-App-76x76@1x.png",
      "scale" : "1x"
    },
    {
      "size" : "76x76",
      "idiom" : "ipad",
      "filename" : "Icon-App-76x76@2x.png",
      "scale" : "2x"
    },
    {
      "size" : "83.5x83.5",
      "idiom" : "ipad",
      "filename" : "Icon-App-83.5x83.5@2x.png",
      "scale" : "2x"
    },
    {
      "size" : "1024x1024",
      "idiom" : "ios-marketing",
      "filename" : "Icon-App-1024x1024@1x.png",
      "scale" : "1x"
    }
  ],
  "info" : {
    "version" : 1,
    "author" : "xcode"
  }
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\ios\Runner\Assets.xcassets\LaunchImage.imageset\Contents.json

```
{
  "images" : [
    {
      "idiom" : "universal",
      "filename" : "LaunchImage.png",
      "scale" : "1x"
    },
    {
      "idiom" : "universal",
      "filename" : "LaunchImage@2x.png",
      "scale" : "2x"
    },
    {
      "idiom" : "universal",
      "filename" : "LaunchImage@3x.png",
      "scale" : "3x"
    }
  ],
  "info" : {
    "version" : 1,
    "author" : "xcode"
  }
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\ios\Runner\Assets.xcassets\LaunchImage.imageset\README.md

```
# Launch Screen Assets

You can customize the launch screen with your own desired assets by replacing the image files in this directory.

You can also do it by opening your Flutter project's Xcode project with `open ios/Runner.xcworkspace`, selecting `Runner/Assets.xcassets` in the Project Navigator and dropping in the desired images.
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\ios\Runner\Base.lproj\LaunchScreen.storyboard

```
<¿xml version="1.0" encoding="UTF-8" standalone="no"¿>
<document   type="com.apple.InterfaceBuilder3.CocoaTouch.Storyboard.XIB"   version="3.0"   toolsVersion="12121"
systemVersion="16G29"    targetRuntime="iOS.CocoaTouch"    propertyAccessControl="none"    useAutolayout="YES"
launchScreen="YES" colorMatched="YES" initialViewController="01J-lp-oVM">
    <dependencies>
        <deployment identifier="iOS"/>
        <plugIn identifier="com.apple.InterfaceBuilder.IBCocoaTouchPlugin" version="12089"/>
    </dependencies>
    <scenes>
        <!--View Controller-->
        <scene sceneID="EHf-IW-A2E">
            <objects>
                <viewController id="01J-lp-oVM" sceneMemberID="viewController">
                    <layoutGuides>
                        <viewControllerLayoutGuide type="top" id="Ydg-fD-yQy"/>
                        <viewControllerLayoutGuide type="bottom" id="xbc-2k-c8Z"/>
                    </layoutGuides>
                    <view key="view" contentMode="scaleToFill" id="Ze5-6b-2t3">
                        <autoresizingMask key="autoresizingMask" widthSizable="YES" heightSizable="YES"/>
                        <subviews>
                                         <imageView  opaque="NO"  clipsSubviews="YES"  multipleTouchEnabled="YES"
contentMode="center" image="LaunchImage" translatesAutoresizingMaskIntoConstraints="NO" id="YRO-k0-Ey4">
                            </imageView>
                        </subviews>
                         <color key="backgroundColor" red="1" green="1" blue="1" alpha="1" colorSpace="custom"
customColorSpace="sRGB"/>
                        <constraints>
                            <constraint firstItem="YRO-k0-Ey4" firstAttribute="centerX" secondItem="Ze5-6b-2t3"
secondAttribute="centerX" id="1a2-6s-vTC"/>
                            <constraint firstItem="YRO-k0-Ey4" firstAttribute="centerY" secondItem="Ze5-6b-2t3"
secondAttribute="centerY" id="4X2-HB-R7a"/>
                        </constraints>
                    </view>
                </viewController>
                       <placeholder  placeholderIdentifier="IBFirstResponder"  id="iYj-Kq-Ea1"  userLabel="First
Responder" sceneMemberID="firstResponder"/>
            </objects>
            <point key="canvasLocation" x="53" y="375"/>
        </scene>
    </scenes>
    <resources>
        <image name="LaunchImage" width="168" height="185"/>
    </resources>
</document>
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\ios\Runner\Base.lproj\Main.storyboard

```
<¿xml version="1.0" encoding="UTF-8" standalone="no"¿>
<document   type="com.apple.InterfaceBuilder3.CocoaTouch.Storyboard.XIB"   version="3.0"   toolsVersion="10117"
systemVersion="15F34"    targetRuntime="iOS.CocoaTouch"    propertyAccessControl="none"    useAutolayout="YES"
useTraitCollections="YES" initialViewController="BYZ-38-t0r">
    <dependencies>
        <deployment identifier="iOS"/>
        <plugIn identifier="com.apple.InterfaceBuilder.IBCocoaTouchPlugin" version="10085"/>
    </dependencies>
    <scenes>
        <!--Flutter View Controller-->
        <scene sceneID="tne-QT-ifu">
            <objects>
                <viewController id="BYZ-38-t0r" customClass="FlutterViewController" sceneMemberID="viewController">
                    <layoutGuides>
                        <viewControllerLayoutGuide type="top" id="y3c-jy-aDJ"/>
                        <viewControllerLayoutGuide type="bottom" id="wfy-db-euE"/>
                    </layoutGuides>
                    <view key="view" contentMode="scaleToFill" id="8bC-Xf-vdC">
                        <rect key="frame" x="0.0" y="0.0" width="600" height="600"/>
                        <autoresizingMask key="autoresizingMask" widthSizable="YES" heightSizable="YES"/>
                        <color key="backgroundColor" white="1" alpha="1" colorSpace="custom" customColorSpace="calibratedWhite"/>
                    </view>
                </viewController>
                <placeholder placeholderIdentifier="IBFirstResponder" id="dkx-z0-nzr" sceneMemberID="firstResponder"/>
            </objects>
        </scene>
    </scenes>
</document>
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\ios\Runner.xcodeproj\project.pbxproj

```
// !$*UTF8*$!
{
 archiveVersion = 1;
 classes = {
 };
 objectVersion = 54;
 objects = {

/* Begin PBXBuildFile section */
  1498D2341E8E89220040F4C2 /* GeneratedPluginRegistrant.m in Sources */ = {isa = PBXBuildFile; fileRef = 1498D2331E8E89220040F4C2 /* GeneratedPluginRegistrant.m */; };
  331C808B294A63AB00263BE5 /* RunnerTests.swift in Sources */ = {isa = PBXBuildFile; fileRef = 331C807B294A618700263BE5 /* RunnerTests.swift */; };
  3B3967161E833CAA004F5970 /* AppFrameworkInfo.plist in Resources */ = {isa = PBXBuildFile; fileRef = 3B3967151E833CAA004F5970 /* AppFrameworkInfo.plist */; };
  74858FAF1ED2DC5600515810 /* AppDelegate.swift in Sources */ = {isa = PBXBuildFile; fileRef = 74858FAE1ED2DC5600515810 /* AppDelegate.swift */; };
  97C146FC1CF9000F007C117D /* Main.storyboard in Resources */ = {isa = PBXBuildFile; fileRef = 97C146FA1CF9000F007C117D /* Main.storyboard */; };
  97C146FE1CF9000F007C117D /* Assets.xcassets in Resources */ = {isa = PBXBuildFile; fileRef = 97C146FD1CF9000F007C117D /* Assets.xcassets */; };
  97C147011CF9000F007C117D /* LaunchScreen.storyboard in Resources */ = {isa = PBXBuildFile; fileRef = 97C146FF1CF9000F007C117D /* LaunchScreen.storyboard */; };
/* End PBXBuildFile section */

/* Begin PBXContainerItemProxy section */
  331C8085294A63A400263BE5 /* PBXContainerItemProxy */ = {
   isa = PBXContainerItemProxy;
   containerPortal = 97C146E61CF9000F007C117D /* Project object */;
   proxyType = 1;
   remoteGlobalIDString = 97C146ED1CF9000F007C117D;
   remoteInfo = Runner;
  };
/* End PBXContainerItemProxy section */

/* Begin PBXCopyFilesBuildPhase section */
  9705A1C41CF9048500538489 /* Embed Frameworks */ = {
   isa = PBXCopyFilesBuildPhase;
   buildActionMask = 2147483647;
   dstPath = "";
   dstSubfolderSpec = 10;
   files = (
   );
   name = "Embed Frameworks";
   runOnlyForDeploymentPostprocessing = 0;
  };
/* End PBXCopyFilesBuildPhase section */

/* Begin PBXFileReference section */
  1498D2321E8E86230040F4C2 /* GeneratedPluginRegistrant.h */ = {isa = PBXFileReference; lastKnownFileType =
sourcecode.c.h; path = GeneratedPluginRegistrant.h; sourceTree = "<group>"; };
  1498D2331E8E89220040F4C2 /* GeneratedPluginRegistrant.m */ = {isa = PBXFileReference; fileEncoding = 4;
lastKnownFileType = sourcecode.c.objc; path = GeneratedPluginRegistrant.m; sourceTree = "<group>"; };
  331C807B294A618700263BE5 /* RunnerTests.swift */ = {isa = PBXFileReference; lastKnownFileType =
sourcecode.swift; path = RunnerTests.swift; sourceTree = "<group>"; };
  331C8081294A63A400263BE5 /* RunnerTests.xctest */ = {isa = PBXFileReference; explicitFileType =
wrapper.cfbundle; includeInIndex = 0; path = RunnerTests.xctest; sourceTree = BUILT_PRODUCTS_DIR; };
  3B3967151E833CAA004F5970 /* AppFrameworkInfo.plist */ = {isa = PBXFileReference; fileEncoding = 4;
lastKnownFileType = text.plist.xml; name = AppFrameworkInfo.plist; path = Flutter/AppFrameworkInfo.plist;
sourceTree = "<group>"; };
  74858FAD1ED2DC5600515810 /* Runner-Bridging-Header.h */ = {isa = PBXFileReference; lastKnownFileType =
sourcecode.c.h; path = "Runner-Bridging-Header.h"; sourceTree = "<group>"; };
  74858FAE1ED2DC5600515810 /* AppDelegate.swift */ = {isa = PBXFileReference; fileEncoding = 4;
lastKnownFileType = sourcecode.swift; path = AppDelegate.swift; sourceTree = "<group>"; };
  7AFA3C8E1D35360C0083082E /* Release.xcconfig */ = {isa = PBXFileReference; lastKnownFileType = text.xcconfig;
name = Release.xcconfig; path = Flutter/Release.xcconfig; sourceTree = "<group>"; };
  9740EEB21CF90195004384FC /* Debug.xcconfig */ = {isa = PBXFileReference; fileEncoding = 4; lastKnownFileType
= text.xcconfig; name = Debug.xcconfig; path = Flutter/Debug.xcconfig; sourceTree = "<group>"; };
  9740EEB31CF90195004384FC /* Generated.xcconfig */ = {isa = PBXFileReference; fileEncoding = 4;
lastKnownFileType = text.xcconfig; name = Generated.xcconfig; path = Flutter/Generated.xcconfig; sourceTree =
"<group>"; };
  97C146EE1CF9000F007C117D /* Runner.app */ = {isa = PBXFileReference; explicitFileType = wrapper.application;
includeInIndex = 0; path = Runner.app; sourceTree = BUILT_PRODUCTS_DIR; };
  97C146FB1CF9000F007C117D /* Base */ = {isa = PBXFileReference; lastKnownFileType = file.storyboard; name =
Base; path = Base.lproj/Main.storyboard; sourceTree = "<group>"; };
  97C146FD1CF9000F007C117D /* Assets.xcassets */ = {isa = PBXFileReference; lastKnownFileType =
folder.assetcatalog; path = Assets.xcassets; sourceTree = "<group>"; };
  97C147001CF9000F007C117D /* Base */ = {isa = PBXFileReference; lastKnownFileType = file.storyboard; name =
Base; path = Base.lproj/LaunchScreen.storyboard; sourceTree = "<group>"; };
  97C147021CF9000F007C117D /* Info.plist */ = {isa = PBXFileReference; lastKnownFileType = text.plist.xml; path
= Info.plist; sourceTree = "<group>"; };
/* End PBXFileReference section */

/* Begin PBXFrameworksBuildPhase section */
  97C146EB1CF9000F007C117D /* Frameworks */ = {
   isa = PBXFrameworksBuildPhase;
   buildActionMask = 2147483647;
   files = (
   );
   runOnlyForDeploymentPostprocessing = 0;
  };
/* End PBXFrameworksBuildPhase section */

/* Begin PBXGroup section */
  331C8082294A63A400263BE5 /* RunnerTests */ = {
   isa = PBXGroup;
   children = (
    331C807B294A618700263BE5 /* RunnerTests.swift */,
   );
   path = RunnerTests;
   sourceTree = "<group>";
  };
  9740EEB11CF90186004384FC /* Flutter */ = {
   isa = PBXGroup;
   children = (
    3B3967151E833CAA004F5970 /* AppFrameworkInfo.plist */,
    9740EEB21CF90195004384FC /* Debug.xcconfig */,
    7AFA3C8E1D35360C0083082E /* Release.xcconfig */,
    9740EEB31CF90195004384FC /* Generated.xcconfig */,
   );
   name = Flutter;
   sourceTree = "<group>";
  };
  97C146E51CF9000F007C117D = {
   isa = PBXGroup;
   children = (
    9740EEB11CF90186004384FC /* Flutter */,
    97C146F01CF9000F007C117D /* Runner */,
    97C146EF1CF9000F007C117D /* Products */,
    331C8082294A63A400263BE5 /* RunnerTests */,
   );
   sourceTree = "<group>";
  };
  97C146EF1CF9000F007C117D /* Products */ = {
   isa = PBXGroup;
   children = (
    97C146EE1CF9000F007C117D /* Runner.app */,
    331C8081294A63A400263BE5 /* RunnerTests.xctest */,
   );
   name = Products;
   sourceTree = "<group>";
  };
  97C146F01CF9000F007C117D /* Runner */ = {
   isa = PBXGroup;
   children = (
    97C146FA1CF9000F007C117D /* Main.storyboard */,
    97C146FD1CF9000F007C117D /* Assets.xcassets */,
    97C146FF1CF9000F007C117D /* LaunchScreen.storyboard */,
    97C147021CF9000F007C117D /* Info.plist */,
    1498D2321E8E86230040F4C2 /* GeneratedPluginRegistrant.h */,
    1498D2331E8E89220040F4C2 /* GeneratedPluginRegistrant.m */,
    74858FAE1ED2DC5600515810 /* AppDelegate.swift */,
    74858FAD1ED2DC5600515810 /* Runner-Bridging-Header.h */,
   );
   path = Runner;
   sourceTree = "<group>";
  };
/* End PBXGroup section */

/* Begin PBXNativeTarget section */
  331C8080294A63A400263BE5 /* RunnerTests */ = {
   isa = PBXNativeTarget;
   buildConfigurationList = 331C8087294A63A400263BE5 /* Build configuration list for PBXNativeTarget
"RunnerTests" */;
   buildPhases = (
    331C807D294A63A400263BE5 /* Sources */,
    331C807F294A63A400263BE5 /* Resources */,
   );
   buildRules = (
   );
   dependencies = (
    331C8086294A63A400263BE5 /* PBXTargetDependency */,
   );
   name = RunnerTests;
   productName = RunnerTests;
   productReference = 331C8081294A63A400263BE5 /* RunnerTests.xctest */;
   productType = "com.apple.product-type.bundle.unit-test";
  };
  97C146ED1CF9000F007C117D /* Runner */ = {
   isa = PBXNativeTarget;
   buildConfigurationList = 97C147051CF9000F007C117D /* Build configuration list for PBXNativeTarget "Runner"
*/;
   buildPhases = (
    9740EEB61CF901F6004384FC /* Run Script */,
    97C146EA1CF9000F007C117D /* Sources */,
    97C146EB1CF9000F007C117D /* Frameworks */,
    97C146EC1CF9000F007C117D /* Resources */,
    9705A1C41CF9048500538489 /* Embed Frameworks */,
    3B06AD1E1E4923F5004D2608 /* Thin Binary */,
   );
   buildRules = (
   );
   dependencies = (
   );
   name = Runner;
   productName = Runner;
   productReference = 97C146EE1CF9000F007C117D /* Runner.app */;
   productType = "com.apple.product-type.application";
  };
/* End PBXNativeTarget section */

/* Begin PBXProject section */
  97C146E61CF9000F007C117D /* Project object */ = {
   isa = PBXProject;
   attributes = {
    BuildIndependentTargetsInParallel = YES;
    LastUpgradeCheck = 1510;
    ORGANIZATIONNAME = "";
    TargetAttributes = {
     331C8080294A63A400263BE5 = {
      CreatedOnToolsVersion = 14.0;
      TestTargetID = 97C146ED1CF9000F007C117D;
     };
     97C146ED1CF9000F007C117D = {
      CreatedOnToolsVersion = 7.3.1;
      LastSwiftMigration = 1100;
     };
    };
   };
   buildConfigurationList = 97C146E91CF9000F007C117D /* Build configuration list for PBXProject "Runner" */;
   compatibilityVersion = "Xcode 9.3";
   developmentRegion = en;
   hasScannedForEncodings = 0;
   knownRegions = (
    en,
    Base,
   );
   mainGroup = 97C146E51CF9000F007C117D;
   productRefGroup = 97C146EF1CF9000F007C117D /* Products */;
   projectDirPath = "";
   projectRoot = "";
   targets = (
    97C146ED1CF9000F007C117D /* Runner */,
    331C8080294A63A400263BE5 /* RunnerTests */,
   );
  };
/* End PBXProject section */

/* Begin PBXResourcesBuildPhase section */
  331C807F294A63A400263BE5 /* Resources */ = {
   isa = PBXResourcesBuildPhase;
   buildActionMask = 2147483647;
   files = (
   );
   runOnlyForDeploymentPostprocessing = 0;
  };
  97C146EC1CF9000F007C117D /* Resources */ = {
   isa = PBXResourcesBuildPhase;
   buildActionMask = 2147483647;
   files = (
    97C147011CF9000F007C117D /* LaunchScreen.storyboard in Resources */,
    3B3967161E833CAA004F5970 /* AppFrameworkInfo.plist in Resources */,
    97C146FE1CF9000F007C117D /* Assets.xcassets in Resources */,
    97C146FC1CF9000F007C117D /* Main.storyboard in Resources */,
   );
   runOnlyForDeploymentPostprocessing = 0;
  };
/* End PBXResourcesBuildPhase section */

/* Begin PBXShellScriptBuildPhase section */
  3B06AD1E1E4923F5004D2608 /* Thin Binary */ = {
   isa = PBXShellScriptBuildPhase;
   alwaysOutOfDate = 1;
   buildActionMask = 2147483647;
   files = (
   );
   inputPaths = (
    "${TARGET_BUILD_DIR}/${INFOPLIST_PATH}",
   );
   name = "Thin Binary";
   outputPaths = (
   );
   runOnlyForDeploymentPostprocessing = 0;
   shellPath = /bin/sh;
   shellScript = "/bin/sh \"$FLUTTER_ROOT/packages/flutter_tools/bin/xcode_backend.sh\" embed_and_thin";
  };
  9740EEB61CF901F6004384FC /* Run Script */ = {
   isa = PBXShellScriptBuildPhase;
   alwaysOutOfDate = 1;
   buildActionMask = 2147483647;
   files = (
   );
   inputPaths = (
   );
   name = "Run Script";
   outputPaths = (
   );
   runOnlyForDeploymentPostprocessing = 0;
   shellPath = /bin/sh;
   shellScript = "/bin/sh \"$FLUTTER_ROOT/packages/flutter_tools/bin/xcode_backend.sh\" build";
  };
/* End PBXShellScriptBuildPhase section */

/* Begin PBXSourcesBuildPhase section */
  331C807D294A63A400263BE5 /* Sources */ = {
   isa = PBXSourcesBuildPhase;
   buildActionMask = 2147483647;
   files = (
    331C808B294A63AB00263BE5 /* RunnerTests.swift in Sources */,
   );
   runOnlyForDeploymentPostprocessing = 0;
  };
  97C146EA1CF9000F007C117D /* Sources */ = {
   isa = PBXSourcesBuildPhase;
   buildActionMask = 2147483647;
   files = (
    74858FAF1ED2DC5600515810 /* AppDelegate.swift in Sources */,
    1498D2341E8E89220040F4C2 /* GeneratedPluginRegistrant.m in Sources */,
   );
   runOnlyForDeploymentPostprocessing = 0;
  };
/* End PBXSourcesBuildPhase section */
/* Begin PBXTargetDependency section */
  331C8086294A63A400263BE5 /* PBXTargetDependency */ = {
   isa = PBXTargetDependency;
   target = 97C146ED1CF9000F007C117D /* Runner */;
   targetProxy = 331C8085294A63A400263BE5 /* PBXContainerItemProxy */;
  };
/* End PBXTargetDependency section */

/* Begin PBXVariantGroup section */
  97C146FA1CF9000F007C117D /* Main.storyboard */ = {
   isa = PBXVariantGroup;
   children = (
    97C146FB1CF9000F007C117D /* Base */,
   );
   name = Main.storyboard;
   sourceTree = "<group>";
  };
  97C146FF1CF9000F007C117D /* LaunchScreen.storyboard */ = {
   isa = PBXVariantGroup;
   children = (
    97C147001CF9000F007C117D /* Base */,
   );
   name = LaunchScreen.storyboard;
   sourceTree = "<group>";
  };
/* End PBXVariantGroup section */

/* Begin XCBuildConfiguration section */
  249021D3217E4FDB00AE95B9 /* Profile */ = {
   isa = XCBuildConfiguration;
   buildSettings = {
    ALWAYS_SEARCH_USER_PATHS = NO;
    ASSETCATALOG_COMPILER_GENERATE_SWIFT_ASSET_SYMBOL_EXTENSIONS = YES;
    CLANG_ANALYZER_NONNULL = YES;
    CLANG_CXX_LANGUAGE_STANDARD = "gnu++0x";
    CLANG_CXX_LIBRARY = "libc++";
    CLANG_ENABLE_MODULES = YES;
    CLANG_ENABLE_OBJC_ARC = YES;
    CLANG_WARN_BLOCK_CAPTURE_AUTORELEASING = YES;
    CLANG_WARN_BOOL_CONVERSION = YES;
    CLANG_WARN_COMMA = YES;
    CLANG_WARN_CONSTANT_CONVERSION = YES;
    CLANG_WARN_DEPRECATED_OBJC_IMPLEMENTATIONS = YES;
    CLANG_WARN_DIRECT_OBJC_ISA_USAGE = YES_ERROR;
    CLANG_WARN_EMPTY_BODY = YES;
    CLANG_WARN_ENUM_CONVERSION = YES;
    CLANG_WARN_INFINITE_RECURSION = YES;
    CLANG_WARN_INT_CONVERSION = YES;
    CLANG_WARN_NON_LITERAL_NULL_CONVERSION = YES;
    CLANG_WARN_OBJC_IMPLICIT_RETAIN_SELF = YES;
    CLANG_WARN_OBJC_LITERAL_CONVERSION = YES;
    CLANG_WARN_OBJC_ROOT_CLASS = YES_ERROR;
    CLANG_WARN_RANGE_LOOP_ANALYSIS = YES;
    CLANG_WARN_STRICT_PROTOTYPES = YES;
    CLANG_WARN_SUSPICIOUS_MOVE = YES;
    CLANG_WARN_UNREACHABLE_CODE = YES;
    CLANG_WARN__DUPLICATE_METHOD_MATCH = YES;
    "CODE_SIGN_IDENTITY[sdk=iphoneos*]" = "iPhone Developer";
    COPY_PHASE_STRIP = NO;
    DEBUG_INFORMATION_FORMAT = "dwarf-with-dsym";
    ENABLE_NS_ASSERTIONS = NO;
    ENABLE_STRICT_OBJC_MSGSEND = YES;
    ENABLE_USER_SCRIPT_SANDBOXING = NO;
    GCC_C_LANGUAGE_STANDARD = gnu99;
    GCC_NO_COMMON_BLOCKS = YES;
    GCC_WARN_64_TO_32_BIT_CONVERSION = YES;
    GCC_WARN_ABOUT_RETURN_TYPE = YES_ERROR;
    GCC_WARN_UNDECLARED_SELECTOR = YES;
    GCC_WARN_UNINITIALIZED_AUTOS = YES_AGGRESSIVE;
    GCC_WARN_UNUSED_FUNCTION = YES;
    GCC_WARN_UNUSED_VARIABLE = YES;
    IPHONEOS_DEPLOYMENT_TARGET = 12.0;
    MTL_ENABLE_DEBUG_INFO = NO;
    SDKROOT = iphoneos;
    SUPPORTED_PLATFORMS = iphoneos;
    TARGETED_DEVICE_FAMILY = "1,2";
    VALIDATE_PRODUCT = YES;
   };
   name = Profile;
  };
  249021D4217E4FDB00AE95B9 /* Profile */ = {
   isa = XCBuildConfiguration;
   baseConfigurationReference = 7AFA3C8E1D35360C0083082E /* Release.xcconfig */;
   buildSettings = {
    ASSETCATALOG_COMPILER_APPICON_NAME = AppIcon;
    CLANG_ENABLE_MODULES = YES;
    CURRENT_PROJECT_VERSION = "$(FLUTTER_BUILD_NUMBER)";
    ENABLE_BITCODE = NO;
    INFOPLIST_FILE = Runner/Info.plist;
    LD_RUNPATH_SEARCH_PATHS = (
     "$(inherited)",
     "@executable_path/Frameworks",
    );
    PRODUCT_BUNDLE_IDENTIFIER = com.example.wismonKeuangan;
    PRODUCT_NAME = "$(TARGET_NAME)";
    SWIFT_OBJC_BRIDGING_HEADER = "Runner/Runner-Bridging-Header.h";
    SWIFT_VERSION = 5.0;
    VERSIONING_SYSTEM = "apple-generic";
   };
   name = Profile;
  };
  331C8088294A63A400263BE5 /* Debug */ = {
   isa = XCBuildConfiguration;
   buildSettings = {
    BUNDLE_LOADER = "$(TEST_HOST)";
    CODE_SIGN_STYLE = Automatic;
    CURRENT_PROJECT_VERSION = 1;
    GENERATE_INFOPLIST_FILE = YES;
    MARKETING_VERSION = 1.0;
    PRODUCT_BUNDLE_IDENTIFIER = com.example.wismonKeuangan.RunnerTests;
    PRODUCT_NAME = "$(TARGET_NAME)";
    SWIFT_ACTIVE_COMPILATION_CONDITIONS = DEBUG;
    SWIFT_OPTIMIZATION_LEVEL = "-Onone";
    SWIFT_VERSION = 5.0;
    TEST_HOST = "$(BUILT_PRODUCTS_DIR)/Runner.app/$(BUNDLE_EXECUTABLE_FOLDER_PATH)/Runner";
   };
   name = Debug;
  };
  331C8089294A63A400263BE5 /* Release */ = {
   isa = XCBuildConfiguration;
   buildSettings = {
    BUNDLE_LOADER = "$(TEST_HOST)";
    CODE_SIGN_STYLE = Automatic;
    CURRENT_PROJECT_VERSION = 1;
    GENERATE_INFOPLIST_FILE = YES;
    MARKETING_VERSION = 1.0;
    PRODUCT_BUNDLE_IDENTIFIER = com.example.wismonKeuangan.RunnerTests;
    PRODUCT_NAME = "$(TARGET_NAME)";
    SWIFT_VERSION = 5.0;
    TEST_HOST = "$(BUILT_PRODUCTS_DIR)/Runner.app/$(BUNDLE_EXECUTABLE_FOLDER_PATH)/Runner";
   };
   name = Release;
  };
  331C808A294A63A400263BE5 /* Profile */ = {
   isa = XCBuildConfiguration;
   buildSettings = {
    BUNDLE_LOADER = "$(TEST_HOST)";
    CODE_SIGN_STYLE = Automatic;
    CURRENT_PROJECT_VERSION = 1;
    GENERATE_INFOPLIST_FILE = YES;
    MARKETING_VERSION = 1.0;
    PRODUCT_BUNDLE_IDENTIFIER = com.example.wismonKeuangan.RunnerTests;
    PRODUCT_NAME = "$(TARGET_NAME)";
    SWIFT_VERSION = 5.0;
    TEST_HOST = "$(BUILT_PRODUCTS_DIR)/Runner.app/$(BUNDLE_EXECUTABLE_FOLDER_PATH)/Runner";
   };
   name = Profile;
  };
  97C147031CF9000F007C117D /* Debug */ = {
   isa = XCBuildConfiguration;
   buildSettings = {
    ALWAYS_SEARCH_USER_PATHS = NO;
    ASSETCATALOG_COMPILER_GENERATE_SWIFT_ASSET_SYMBOL_EXTENSIONS = YES;
    CLANG_ANALYZER_NONNULL = YES;
    CLANG_CXX_LANGUAGE_STANDARD = "gnu++0x";
    CLANG_CXX_LIBRARY = "libc++";
    CLANG_ENABLE_MODULES = YES;
    CLANG_ENABLE_OBJC_ARC = YES;
    CLANG_WARN_BLOCK_CAPTURE_AUTORELEASING = YES;
    CLANG_WARN_BOOL_CONVERSION = YES;
    CLANG_WARN_COMMA = YES;
    CLANG_WARN_CONSTANT_CONVERSION = YES;
    CLANG_WARN_DEPRECATED_OBJC_IMPLEMENTATIONS = YES;
    CLANG_WARN_DIRECT_OBJC_ISA_USAGE = YES_ERROR;
    CLANG_WARN_EMPTY_BODY = YES;
    CLANG_WARN_ENUM_CONVERSION = YES;
    CLANG_WARN_INFINITE_RECURSION = YES;
    CLANG_WARN_INT_CONVERSION = YES;
    CLANG_WARN_NON_LITERAL_NULL_CONVERSION = YES;
    CLANG_WARN_OBJC_IMPLICIT_RETAIN_SELF = YES;
    CLANG_WARN_OBJC_LITERAL_CONVERSION = YES;
    CLANG_WARN_OBJC_ROOT_CLASS = YES_ERROR;
    CLANG_WARN_RANGE_LOOP_ANALYSIS = YES;
    CLANG_WARN_STRICT_PROTOTYPES = YES;
    CLANG_WARN_SUSPICIOUS_MOVE = YES;
    CLANG_WARN_UNREACHABLE_CODE = YES;
    CLANG_WARN__DUPLICATE_METHOD_MATCH = YES;
    "CODE_SIGN_IDENTITY[sdk=iphoneos*]" = "iPhone Developer";
    COPY_PHASE_STRIP = NO;
    DEBUG_INFORMATION_FORMAT = dwarf;
    ENABLE_STRICT_OBJC_MSGSEND = YES;
    ENABLE_TESTABILITY = YES;
    ENABLE_USER_SCRIPT_SANDBOXING = NO;
    GCC_C_LANGUAGE_STANDARD = gnu99;
    GCC_DYNAMIC_NO_PIC = NO;
    GCC_NO_COMMON_BLOCKS = YES;
    GCC_OPTIMIZATION_LEVEL = 0;
    GCC_PREPROCESSOR_DEFINITIONS = (
     "DEBUG=1",
     "$(inherited)",
    );
    GCC_WARN_64_TO_32_BIT_CONVERSION = YES;
    GCC_WARN_ABOUT_RETURN_TYPE = YES_ERROR;
    GCC_WARN_UNDECLARED_SELECTOR = YES;
    GCC_WARN_UNINITIALIZED_AUTOS = YES_AGGRESSIVE;
    GCC_WARN_UNUSED_FUNCTION = YES;
    GCC_WARN_UNUSED_VARIABLE = YES;
    IPHONEOS_DEPLOYMENT_TARGET = 12.0;
    MTL_ENABLE_DEBUG_INFO = YES;
    ONLY_ACTIVE_ARCH = YES;
    SDKROOT = iphoneos;
    TARGETED_DEVICE_FAMILY = "1,2";
   };
   name = Debug;
  };
  97C147041CF9000F007C117D /* Release */ = {
   isa = XCBuildConfiguration;
   buildSettings = {
    ALWAYS_SEARCH_USER_PATHS = NO;
    ASSETCATALOG_COMPILER_GENERATE_SWIFT_ASSET_SYMBOL_EXTENSIONS = YES;
    CLANG_ANALYZER_NONNULL = YES;
    CLANG_CXX_LANGUAGE_STANDARD = "gnu++0x";
    CLANG_CXX_LIBRARY = "libc++";
    CLANG_ENABLE_MODULES = YES;
    CLANG_ENABLE_OBJC_ARC = YES;
    CLANG_WARN_BLOCK_CAPTURE_AUTORELEASING = YES;
    CLANG_WARN_BOOL_CONVERSION = YES;
    CLANG_WARN_COMMA = YES;
    CLANG_WARN_CONSTANT_CONVERSION = YES;
    CLANG_WARN_DEPRECATED_OBJC_IMPLEMENTATIONS = YES;
    CLANG_WARN_DIRECT_OBJC_ISA_USAGE = YES_ERROR;
    CLANG_WARN_EMPTY_BODY = YES;
    CLANG_WARN_ENUM_CONVERSION = YES;
    CLANG_WARN_INFINITE_RECURSION = YES;
    CLANG_WARN_INT_CONVERSION = YES;
    CLANG_WARN_NON_LITERAL_NULL_CONVERSION = YES;
    CLANG_WARN_OBJC_IMPLICIT_RETAIN_SELF = YES;
    CLANG_WARN_OBJC_LITERAL_CONVERSION = YES;
    CLANG_WARN_OBJC_ROOT_CLASS = YES_ERROR;
    CLANG_WARN_RANGE_LOOP_ANALYSIS = YES;
    CLANG_WARN_STRICT_PROTOTYPES = YES;
    CLANG_WARN_SUSPICIOUS_MOVE = YES;
    CLANG_WARN_UNREACHABLE_CODE = YES;
    CLANG_WARN__DUPLICATE_METHOD_MATCH = YES;
    "CODE_SIGN_IDENTITY[sdk=iphoneos*]" = "iPhone Developer";
    COPY_PHASE_STRIP = NO;
    DEBUG_INFORMATION_FORMAT = "dwarf-with-dsym";
    ENABLE_NS_ASSERTIONS = NO;
    ENABLE_STRICT_OBJC_MSGSEND = YES;
    ENABLE_USER_SCRIPT_SANDBOXING = NO;
    GCC_C_LANGUAGE_STANDARD = gnu99;
    GCC_NO_COMMON_BLOCKS = YES;
    GCC_WARN_64_TO_32_BIT_CONVERSION = YES;
    GCC_WARN_ABOUT_RETURN_TYPE = YES_ERROR;
    GCC_WARN_UNDECLARED_SELECTOR = YES;
    GCC_WARN_UNINITIALIZED_AUTOS = YES_AGGRESSIVE;
    GCC_WARN_UNUSED_FUNCTION = YES;
    GCC_WARN_UNUSED_VARIABLE = YES;
    IPHONEOS_DEPLOYMENT_TARGET = 12.0;
    MTL_ENABLE_DEBUG_INFO = NO;
    SDKROOT = iphoneos;
    SUPPORTED_PLATFORMS = iphoneos;
    SWIFT_COMPILATION_MODE = wholemodule;
    SWIFT_OPTIMIZATION_LEVEL = "-O";
    TARGETED_DEVICE_FAMILY = "1,2";
    VALIDATE_PRODUCT = YES;
   };
   name = Release;
  };
  97C147061CF9000F007C117D /* Debug */ = {
   isa = XCBuildConfiguration;
   baseConfigurationReference = 9740EEB21CF90195004384FC /* Debug.xcconfig */;
   buildSettings = {
    ASSETCATALOG_COMPILER_APPICON_NAME = AppIcon;
    CLANG_ENABLE_MODULES = YES;
    CURRENT_PROJECT_VERSION = "$(FLUTTER_BUILD_NUMBER)";
    ENABLE_BITCODE = NO;
    INFOPLIST_FILE = Runner/Info.plist;
    LD_RUNPATH_SEARCH_PATHS = (
     "$(inherited)",
     "@executable_path/Frameworks",
    );
    PRODUCT_BUNDLE_IDENTIFIER = com.example.wismonKeuangan;
    PRODUCT_NAME = "$(TARGET_NAME)";
    SWIFT_OBJC_BRIDGING_HEADER = "Runner/Runner-Bridging-Header.h";
    SWIFT_OPTIMIZATION_LEVEL = "-Onone";
    SWIFT_VERSION = 5.0;
    VERSIONING_SYSTEM = "apple-generic";
   };
   name = Debug;
  };
  97C147071CF9000F007C117D /* Release */ = {
   isa = XCBuildConfiguration;
   baseConfigurationReference = 7AFA3C8E1D35360C0083082E /* Release.xcconfig */;
   buildSettings = {
    ASSETCATALOG_COMPILER_APPICON_NAME = AppIcon;
    CLANG_ENABLE_MODULES = YES;
    CURRENT_PROJECT_VERSION = "$(FLUTTER_BUILD_NUMBER)";
    ENABLE_BITCODE = NO;
    INFOPLIST_FILE = Runner/Info.plist;
    LD_RUNPATH_SEARCH_PATHS = (
     "$(inherited)",
     "@executable_path/Frameworks",
    );
    PRODUCT_BUNDLE_IDENTIFIER = com.example.wismonKeuangan;
    PRODUCT_NAME = "$(TARGET_NAME)";
    SWIFT_OBJC_BRIDGING_HEADER = "Runner/Runner-Bridging-Header.h";
    SWIFT_VERSION = 5.0;
    VERSIONING_SYSTEM = "apple-generic";
   };
   name = Release;
  };
/* End XCBuildConfiguration section */

/* Begin XCConfigurationList section */
  331C8087294A63A400263BE5 /* Build configuration list for PBXNativeTarget "RunnerTests" */ = {
   isa = XCConfigurationList;
   buildConfigurations = (
    331C8088294A63A400263BE5 /* Debug */,
    331C8089294A63A400263BE5 /* Release */,
    331C808A294A63A400263BE5 /* Profile */,
   );
   defaultConfigurationIsVisible = 0;
   defaultConfigurationName = Release;
  };
  97C146E91CF9000F007C117D /* Build configuration list for PBXProject "Runner" */ = {
   isa = XCConfigurationList;
   buildConfigurations = (
    97C147031CF9000F007C117D /* Debug */,
    97C147041CF9000F007C117D /* Release */,
    249021D3217E4FDB00AE95B9 /* Profile */,
   );
   defaultConfigurationIsVisible = 0;
   defaultConfigurationName = Release;
  };
  97C147051CF9000F007C117D /* Build configuration list for PBXNativeTarget "Runner" */ = {
   isa = XCConfigurationList;
   buildConfigurations = (
    97C147061CF9000F007C117D /* Debug */,
    97C147071CF9000F007C117D /* Release */,
    249021D4217E4FDB00AE95B9 /* Profile */,
   );
   defaultConfigurationIsVisible = 0;
   defaultConfigurationName = Release;
  };
/* End XCConfigurationList section */
 };
 rootObject = 97C146E61CF9000F007C117D /* Project object */;
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\ios\Runner.xcodeproj\project.xcworkspace\contents.xcworkspacedata

```
<¿xml version="1.0" encoding="UTF-8"¿>
<Workspace
   version = "1.0">
   <FileRef
      location = "self:">
   </FileRef>
</Workspace>
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\ios\Runner.xcodeproj\project.xcworkspace\xcshareddata\IDEWorkspaceChecks.plist

```
<¿xml version="1.0" encoding="UTF-8"¿>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
 <key>IDEDidComputeMac32BitWarning</key>
 <true/>
</dict>
</plist>
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\ios\Runner.xcodeproj\project.xcworkspace\xcshareddata\WorkspaceSettings.xcsettings

```
<¿xml version="1.0" encoding="UTF-8"¿>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
 <key>PreviewsEnabled</key>
 <false/>
</dict>
</plist>
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\ios\Runner.xcodeproj\xcshareddata\xcschemes\Runner.xcscheme

```
<¿xml version="1.0" encoding="UTF-8"¿>
<Scheme
   LastUpgradeVersion = "1510"
   version = "1.3">
   <BuildAction
      parallelizeBuildables = "YES"
      buildImplicitDependencies = "YES">
      <BuildActionEntries>
         <BuildActionEntry
            buildForTesting = "YES"
            buildForRunning = "YES"
            buildForProfiling = "YES"
            buildForArchiving = "YES"
            buildForAnalyzing = "YES">
            <BuildableReference
               BuildableIdentifier = "primary"
               BlueprintIdentifier = "97C146ED1CF9000F007C117D"
               BuildableName = "Runner.app"
               BlueprintName = "Runner"
               ReferencedContainer = "container:Runner.xcodeproj">
            </BuildableReference>
         </BuildActionEntry>
      </BuildActionEntries>
   </BuildAction>
   <TestAction
      buildConfiguration = "Debug"
      selectedDebuggerIdentifier = "Xcode.DebuggerFoundation.Debugger.LLDB"
      selectedLauncherIdentifier = "Xcode.DebuggerFoundation.Launcher.LLDB"
      customLLDBInitFile = "$(SRCROOT)/Flutter/ephemeral/flutter_lldbinit"
      shouldUseLaunchSchemeArgsEnv = "YES">
      <MacroExpansion>
         <BuildableReference
            BuildableIdentifier = "primary"
            BlueprintIdentifier = "97C146ED1CF9000F007C117D"
            BuildableName = "Runner.app"
            BlueprintName = "Runner"
            ReferencedContainer = "container:Runner.xcodeproj">
         </BuildableReference>
      </MacroExpansion>
      <Testables>
         <TestableReference
            skipped = "NO"
            parallelizable = "YES">
            <BuildableReference
               BuildableIdentifier = "primary"
               BlueprintIdentifier = "331C8080294A63A400263BE5"
               BuildableName = "RunnerTests.xctest"
               BlueprintName = "RunnerTests"
               ReferencedContainer = "container:Runner.xcodeproj">
            </BuildableReference>
         </TestableReference>
      </Testables>
   </TestAction>
   <LaunchAction
      buildConfiguration = "Debug"
      selectedDebuggerIdentifier = "Xcode.DebuggerFoundation.Debugger.LLDB"
      selectedLauncherIdentifier = "Xcode.DebuggerFoundation.Launcher.LLDB"
      customLLDBInitFile = "$(SRCROOT)/Flutter/ephemeral/flutter_lldbinit"
      launchStyle = "0"
      useCustomWorkingDirectory = "NO"
      ignoresPersistentStateOnLaunch = "NO"
      debugDocumentVersioning = "YES"
      debugServiceExtension = "internal"
      enableGPUValidationMode = "1"
      allowLocationSimulation = "YES">
      <BuildableProductRunnable
         runnableDebuggingMode = "0">
         <BuildableReference
            BuildableIdentifier = "primary"
            BlueprintIdentifier = "97C146ED1CF9000F007C117D"
            BuildableName = "Runner.app"
            BlueprintName = "Runner"
            ReferencedContainer = "container:Runner.xcodeproj">
         </BuildableReference>
      </BuildableProductRunnable>
   </LaunchAction>
   <ProfileAction
      buildConfiguration = "Profile"
      shouldUseLaunchSchemeArgsEnv = "YES"
      savedToolIdentifier = ""
      useCustomWorkingDirectory = "NO"
      debugDocumentVersioning = "YES">
      <BuildableProductRunnable
         runnableDebuggingMode = "0">
         <BuildableReference
            BuildableIdentifier = "primary"
            BlueprintIdentifier = "97C146ED1CF9000F007C117D"
            BuildableName = "Runner.app"
            BlueprintName = "Runner"
            ReferencedContainer = "container:Runner.xcodeproj">
         </BuildableReference>
      </BuildableProductRunnable>
   </ProfileAction>
   <AnalyzeAction
      buildConfiguration = "Debug">
   </AnalyzeAction>
   <ArchiveAction
      buildConfiguration = "Release"
      revealArchiveInOrganizer = "YES">
   </ArchiveAction>
</Scheme>
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\ios\Runner.xcworkspace\contents.xcworkspacedata

```
<¿xml version="1.0" encoding="UTF-8"¿>
<Workspace
   version = "1.0">
   <FileRef
      location = "group:Runner.xcodeproj">
   </FileRef>
</Workspace>
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\ios\Runner.xcworkspace\xcshareddata\IDE

WorkspaceChecks.plist

```
<¿xml version="1.0" encoding="UTF-8"¿>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
 <key>IDEDidComputeMac32BitWarning</key>
 <true/>
</dict>
</plist>
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\ios\Runner.xcworkspace\xcshareddata\WorkspaceSettings.xcsettings

```
<¿xml version="1.0" encoding="UTF-8"¿>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
 <key>PreviewsEnabled</key>
 <false/>
</dict>
</plist>
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\ios\RunnerTests\RunnerTests.swift

```
import Flutter
import UIKit
import XCTest

class RunnerTests: XCTestCase {

  func testExample() {
    // If you add code to the Runner application, consider adding tests here.
    // See https://developer.apple.com/documentation/xctest for more information about using XCTest.
  }

}
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\main.dart

```
import 'package:flutter/material.dart';
import 'package:flutter/services.dart';
import 'package:google_fonts/google_fonts.dart';
import 'package:flutter_bloc/flutter_bloc.dart';
import 'core/di/injection_container.dart' as di;
import 'features/auth/presentation/bloc/auth_bloc.dart';
import 'features/auth/presentation/bloc/auth_event.dart';
import 'features/auth/presentation/bloc/auth_state.dart';
import 'features/auth/presentation/pages/login_page.dart';
import 'features/dashboard/presentation/pages/main_navigation_page.dart';
import 'features/payment/presentation/bloc/payment_bloc.dart';
import 'core/services/api_service.dart';

// Cache the text theme to prevent repeated font loading
late final TextTheme _cachedTextTheme;
late final ThemeData _cachedThemeData;
```

```dart
void main() async {
  WidgetsFlutterBinding.ensureInitialized();

  // Performance optimizations
  await SystemChrome.setPreferredOrientations([DeviceOrientation.portraitUp]);

  // Optimize memory usage
  SystemChrome.setSystemUIOverlayStyle(
    const SystemUiOverlayStyle(
      statusBarColor: Colors.transparent,
      statusBarIconBrightness: Brightness.dark,
      statusBarBrightness: Brightness.light,
    ),
  );

  // Pre-cache fonts for better performance
  await _initializeTheme();

  await di.init();
  runApp(const MyApp());
}

Future<void> _initializeTheme() async {
  // Pre-load and cache the Google Font to prevent lag
  _cachedTextTheme = GoogleFonts.plusJakartaSansTextTheme();

  _cachedThemeData = ThemeData(
    primarySwatch: Colors.blue,
    textTheme: _cachedTextTheme,
    visualDensity: VisualDensity.adaptivePlatformDensity,
    pageTransitionsTheme: const PageTransitionsTheme(
      builders: {
        TargetPlatform.android: CupertinoPageTransitionsBuilder(),
        TargetPlatform.iOS: CupertinoPageTransitionsBuilder(),
      },
    ),
    useMaterial3: true,
    colorScheme: ColorScheme.fromSeed(
      seedColor: const Color(0xFF135EA2),
      brightness: Brightness.light,
    ),
    // Optimize app bar theme
    appBarTheme: const AppBarTheme(
      elevation: 0,
      backgroundColor: Colors.transparent,
      foregroundColor: Color(0xFF121212),
      systemOverlayStyle: SystemUiOverlayStyle.dark,
    ),
    // Optimize scaffold theme
    scaffoldBackgroundColor: const Color(0xFFFBFBFB),
  );
}

class MyApp extends StatefulWidget {
  const MyApp({super.key});

  @override
  State<MyApp> createState() => _MyAppState();
}

class _MyAppState extends State<MyApp> with WidgetsBindingObserver {
  @override
  void initState() {
    super.initState();
    WidgetsBinding.instance.addObserver(this);
```

```
}

@override
void dispose() {
  WidgetsBinding.instance.removeObserver(this);
  // Clean up API service connections
  ApiService.dispose();
  super.dispose();
}

@override
void didChangeAppLifecycleState(AppLifecycleState state) {
  super.didChangeAppLifecycleState(state);
  // Optimize memory when app goes to background
  if (state == AppLifecycleState.paused) {
    // Force garbage collection when app is paused
    _performMemoryCleanup();
  }
}

void _performMemoryCleanup() {
  // Clear unnecessary cached images
  imageCache.clear();
  imageCache.clearLiveImages();
}

@override
Widget build(BuildContext context) {
  return MultiBlocProvider(
    providers: [
      BlocProvider(
        create: (_) => di.sl<AuthBloc>()..add(const CheckAuthStatusEvent()),
        lazy: false, // Load immediately for auth
      ),
      BlocProvider(
        create: (_) => di.sl<PaymentBloc>(),
        lazy: true, // Lazy load for better startup performance
      ),
    ],
    child: MaterialApp(
      title: 'Wismon Keuangan',
      debugShowCheckedModeBanner: false, // Remove debug banner
      // Performance optimizations
      builder: (context, child) {
        // Prevent text scaling beyond reasonable limits
        return MediaQuery(
          data: MediaQuery.of(context).copyWith(
            textScaler: TextScaler.linear(
              MediaQuery.of(context).textScaler.scale(1.0).clamp(0.8, 1.3),
            ),
          ),
          child: child!,
        );
      },

      theme: _cachedThemeData,

      home: const AuthWrapper(),

      // Performance monitoring in debug mode
      showPerformanceOverlay: false, // Set to true only for debugging
      // Optimize route generation
      onGenerateRoute: (settings) {
        switch (settings.name) {
          case '/login':
            return PageRouteBuilder(
                pageBuilder: (context, animation, secondaryAnimation) =>
                    const LoginPage(),
                transitionsBuilder:
                    (context, animation, secondaryAnimation, child) {
                      return FadeTransition(opacity: animation, child: child);
                    },
                transitionDuration: const Duration(milliseconds: 200),
              );
            case '/main':
              return PageRouteBuilder(
                pageBuilder: (context, animation, secondaryAnimation) =>
                    const MainNavigationPage(),
                transitionsBuilder:
                    (context, animation, secondaryAnimation, child) {
                      return FadeTransition(opacity: animation, child: child);
                    },
                transitionDuration: const Duration(milliseconds: 200),
              );
            default:
              return null;
          }
        },
      ),
    );
  }
}

class AuthWrapper extends StatelessWidget {
  const AuthWrapper({super.key});

  @override
  Widget build(BuildContext context) {
    return BlocBuilder<AuthBloc, AuthState>(
      buildWhen: (previous, current) {
        // Only rebuild when auth state actually changes
        return previous.runtimeType != current.runtimeType;
      },
      builder: (context, state) {
        if (state is AuthAuthenticated) {
          return const MainNavigationPage();
        }
        if (state is AuthUnauthenticated || state is AuthError) {
          return const LoginPage();
        }
        return const LoadingScreen();
      },
    );
  }
}

class LoadingScreen extends StatelessWidget {
  const LoadingScreen({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: const Color(0xFFFAFAFA),
      body: RepaintBoundary(
        child: Center(
          child: Column(
            mainAxisAlignment: MainAxisAlignment.center,
            children: [
              // App Logo with Hero animation
              Hero(
                tag: 'app_logo',
                child: Container(
                  width: 80,
                  height: 80,
                  decoration: BoxDecoration(
                    color: const Color(0xFF135EA2),
                    borderRadius: BorderRadius.circular(20),
                    boxShadow: const [
                      BoxShadow(
                        color: Color(0x20135EA2),
                        blurRadius: 20,
                        offset: Offset(0, 8),
                      ),
                    ],
                  ),
                  child: const Icon(
                    Icons.school,
                    color: Colors.white,
                    size: 40,
                  ),
                ),
              ),
              const SizedBox(height: 24),

              // App Title
              const Text(
                'Wismon Keuangan',
                style: TextStyle(
                  fontSize: 24,
                  fontWeight: FontWeight.bold,
                  color: Color(0xFF135EA2),
                  letterSpacing: -0.5,
                ),
              ),
              const SizedBox(height: 8),
              Text(
                'Student Payment System',
                style: TextStyle(
                  fontSize: 16,
                  color: Colors.grey[600],
                  letterSpacing: 0.2,
                ),
              ),
              const SizedBox(height: 32),

              // Loading indicator with animation
              const SizedBox(
                width: 24,
                height: 24,
                child: CircularProgressIndicator(
                  valueColor: AlwaysStoppedAnimation<Color>(Color(0xFF135EA2)),
                  strokeWidth: 2.5,
                ),
              ),
            ],
          ),
        ),
      ),
    );
  }
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\core\di\injection_container.dart

```dart
import 'package:flutter/material.dart';
import 'package:get_it/get_it.dart';
import '../services/api_service.dart';
import '../../features/auth/data/repositories/auth_repository_impl.dart';
import '../../features/auth/domain/repositories/auth_repository.dart';
import '../../features/auth/domain/usecases/login_usecase.dart';
import '../../features/auth/domain/usecases/logout_usecase.dart';
import '../../features/auth/domain/usecases/check_auth_status_usecase.dart';
import '../../features/auth/presentation/bloc/auth_bloc.dart';
import '../../features/payment/data/repositories/payment_repository_impl.dart';
import '../../features/payment/domain/repositories/payment_repository.dart';
import '../../features/payment/domain/usecases/get_payment_history_usecase.dart';
import '../../features/payment/domain/usecases/get_payment_summary_usecase.dart';
import '../../features/payment/domain/usecases/get_transaction_detail_usecase.dart';
import '../../features/payment/presentation/bloc/payment_bloc.dart';

final sl = GetIt.instance;

Future<void> init() async {
  // Core
  sl.registerLazySingleton<ApiService>(() => ApiService());
  sl.registerLazySingleton<RouteObserver<PageRoute>>(() => RouteObserver());

  // Auth Feature
  // Data sources & repositories
  sl.registerLazySingleton<AuthRepository>(
    () => AuthRepositoryImpl(apiService: sl()),
  );

  // Use cases
  sl.registerLazySingleton(() => LoginUseCase(sl()));
  sl.registerLazySingleton(() => LogoutUseCase(sl()));
  sl.registerLazySingleton(() => CheckAuthStatusUseCase(sl()));

  // Bloc
  sl.registerFactory(
    () => AuthBloc(
      loginUseCase: sl(),
      logoutUseCase: sl(),
      checkAuthStatusUseCase: sl(),
    ),
  );

  // Payment Feature
  // Data sources & repositories
  sl.registerLazySingleton<PaymentRepository>(
    () => PaymentRepositoryImpl(apiService: sl()),
  );

  // Use cases
  sl.registerLazySingleton(() => GetPaymentHistoryUseCase(sl()));
  sl.registerLazySingleton(() => GetPaymentSummaryUseCase(sl()));
  sl.registerLazySingleton(() => GetTransactionDetailUseCase(sl()));

  // Bloc
  sl.registerFactory(
    () => PaymentBloc(
      getPaymentHistoryUseCase: sl(),
      getPaymentSummaryUseCase: sl(),
      getTransactionDetailUseCase: sl(),
    ),
  );
}
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\core\error\failures.dart

```
import 'package:equatable/equatable.dart';

abstract class Failure extends Equatable {
  const Failure();
}

// General failures
class ServerFailure extends Failure {
  final String message;

  const ServerFailure(this.message);

  @override
  List<Object> get props => [message];
}

class CacheFailure extends Failure {
  final String message;

  const CacheFailure(this.message);

  @override
  List<Object> get props => [message];
}

class NetworkFailure extends Failure {
  final String message;

  const NetworkFailure(this.message);

  @override
  List<Object> get props => [message];
}

class AuthFailure extends Failure {
  final String message;

  const AuthFailure(this.message);

  @override
  List<Object> get props => [message];
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\core\services\api_service.dart

```
import 'dart:convert';
import 'dart:io';
import 'package:http/http.dart' as http;
import 'package:shared_preferences/shared_preferences.dart';
import '../../features/auth/data/models/user_model.dart';
import '../../features/payment/data/models/payment_model.dart';

class ApiService {
  // API Base URL
  // For Chrome/Web development: http://localhost:3000
  // For Android emulator: http://10.0.2.2:3000
  // For iOS simulator: http://localhost:3000
  static const String baseUrl = 'http://localhost:3000';
  static http.Client¿ _client;
  static final Map<String, dynamic> _cache = {};
  static const int _cacheTimeout = 5 * 60 * 1000; // 5 minutes

  // Regular constructor for DI, but still use singleton pattern for client
  ApiService() {
    _initializeClient();
  }

  void _initializeClient() {
    _client ¿¿= http.Client();
  }

  // Cache management for better performance
  bool _isCacheValid(String key) {
    if (!_cache.containsKey(key)) return false;
    final cachedData = _cache[key];
    final timestamp = cachedData['timestamp'] as int;
    return DateTime.now().millisecondsSinceEpoch - timestamp < _cacheTimeout;
  }

  void _setCache(String key, dynamic data) {
    _cache[key] = {
      'data': data,
      'timestamp': DateTime.now().millisecondsSinceEpoch,
    };
  }

  dynamic _getCache(String key) {
    if (_isCacheValid(key)) {
      return _cache[key]['data'];
    }
    return null;
  }

  Future<String¿> getAuthToken() async {
    try {
      final prefs = await SharedPreferences.getInstance();
      return prefs.getString('auth_token');
    } catch (e) {
      return null;
    }
  }

  Future<void> setAuthToken(String token) async {
    try {
      final prefs = await SharedPreferences.getInstance();
      await prefs.setString('auth_token', token);
    } catch (e) {
      // Handle storage error gracefully
    }
  }

  Future<void> clearAuthToken() async {
    try {
      final prefs = await SharedPreferences.getInstance();
      await prefs.remove('auth_token');
    } catch (e) {
      // Handle error gracefully
    }
  }

  Map<String, String> _getHeaders([String¿ token]) {
    final headers = {
      'Content-Type': 'application/json',
      'Accept': 'application/json',
    };
    if (token != null) {
      headers['Authorization'] = 'Bearer $token';
    }

    return headers;
  }

  // Generic GET method
  Future<Map<String, dynamic>> get(
    String endpoint, {
    bool useCache = false,
  }) async {
    final cacheKey = 'GET_$endpoint';

    // Check cache first if enabled
    if (useCache) {
      final cachedData = _getCache(cacheKey);
      if (cachedData != null) {
        return cachedData;
      }
    }

    try {
      final token = await getAuthToken();
      final response = await _client!
          .get(Uri.parse('$baseUrl$endpoint'), headers: _getHeaders(token))
          .timeout(const Duration(seconds: 30));

      final data = _handleResponse(response);

      // Cache successful responses
      if (useCache && response.statusCode == 200) {
        _setCache(cacheKey, data);
      }

      return data;
    } on SocketException {
      throw Exception('Tidak ada koneksi internet');
    } on HttpException {
      throw Exception('Gagal terhubung ke server');
    } on FormatException {
      throw Exception('Respons server tidak valid');
    } catch (e) {
      throw Exception('Terjadi kesalahan: ${e.toString()}');
    }
  }

  // Generic POST method
  Future<Map<String, dynamic>> post(
    String endpoint,
    Map<String, dynamic> data,
  ) async {
    try {
      final token = await getAuthToken();
      final response = await _client!
          .post(
            Uri.parse('$baseUrl$endpoint'),
            headers: _getHeaders(token),
            body: jsonEncode(data),
          )
          .timeout(const Duration(seconds: 30));

      return _handleResponse(response);
    } on SocketException {
      throw Exception('Tidak ada koneksi internet');
    } on HttpException {
      throw Exception('Gagal terhubung ke server');
    } on FormatException {
      throw Exception('Respons server tidak valid');
    } catch (e) {
      throw Exception('Terjadi kesalahan: ${e.toString()}');
    }
  }

  // AUTH METHODS
  Future<UserModel> login(String namamNim, String nrm) async {
    final data = await post('/api/auth/login', {
      'namam_nim': namamNim,
      'nrm': nrm,
    });

    if (data['success']) {
      await setAuthToken(data['data']['token']);
      return UserModel.fromJson(data['data']['user']);
    } else {
      throw Exception(data['message'] ¿¿ 'Login failed');
    }
  }

  Future<UserModel> getProfile() async {
    final data = await get('/api/auth/profile', useCache: true);

    if (data['success']) {
      return UserModel.fromJson(data['data']);
    } else {
      throw Exception(data['message'] ¿¿ 'Failed to get profile');
    }
  }

  // PAYMENT METHODS
  Future<List<PaymentHistoryItemModel>> getPaymentHistory({
    int page = 1,
    int limit = 20,
    String¿ startDate,
    String¿ endDate,
    String¿ type,
    String sortBy = 'tanggal',
    String sortOrder = 'desc',
    bool forceRefresh = false,
  }) async {
    final queryParams = <String, String>{
      'page': page.toString(),
      'limit': limit.toString(),
      'sortBy': sortBy,
      'sortOrder': sortOrder,
    };

    if (startDate != null) queryParams['startDate'] = startDate;
    if (endDate != null) queryParams['endDate'] = endDate;
    if (type != null) queryParams['type'] = type;

    final uri = Uri.parse(
      '$baseUrl/api/payments/history',
    ).replace(queryParameters: queryParams);

    try {
      final token = await getAuthToken();
      final response = await _client!
          .get(uri, headers: _getHeaders(token))
          .timeout(const Duration(seconds: 30));
      final data = _handleResponse(response);

      if (data['success']) {
        final List<dynamic> historyData = data['data']['data'];
        return historyData
            .map((item) => PaymentHistoryItemModel.fromJson(item))
            .toList();
      } else {
        throw Exception(data['message'] ¿¿ 'Failed to get payment history');
      }
    } on SocketException {
      throw Exception('Tidak ada koneksi internet');
    } on HttpException {
      throw Exception('Gagal terhubung ke server');
    } on FormatException {
      throw Exception('Respons server tidak valid');
    } catch (e) {
      throw Exception('Terjadi kesalahan: ${e.toString()}');
    }
  }

  Future<PaymentSummaryModel> getPaymentSummary({
    bool forceRefresh = false,
  }) async {
    final data = await get('/api/payments/summary', useCache: !forceRefresh);

    if (data['success']) {
      return PaymentSummaryModel.fromJson(data['data']);
    } else {
      throw Exception(data['message'] ¿¿ 'Failed to get payment summary');
    }
  }

  Future<TransactionDetailModel> getTransactionDetail(
    String transactionId,
  ) async {
    final data = await get('/api/payments/detail/$transactionId');

    if (data['success']) {
      return TransactionDetailModel.fromJson(data['data']);
    } else {
      throw Exception(data['message'] ¿¿ 'Failed to get transaction detail');
    }
  }

  Future<bool> refreshPaymentData() async {
    final data = await post('/api/payments/refresh', {});

    if (data['success']) {
      // Clear payment-related cache
      _cache.removeWhere((key, value) => key.contains('payment'));
      return data['data']['refreshed'] ¿¿ false;
    } else {
      throw Exception(data['message'] ¿¿ 'Failed to refresh payment data');
    }
  }

  Map<String, dynamic> _handleResponse(http.Response response) {
    final responseData = jsonDecode(response.body) as Map<String, dynamic>;

    switch (response.statusCode) {
      case 200:
      case 201:
        return responseData;
      case 400:
        throw Exception(responseData['message'] ¿¿ 'Permintaan tidak valid');
      case 401:
        throw Exception('Sesi telah berakhir, silakan login kembali');
      case 403:
        throw Exception('Akses ditolak');
      case 404:
        throw Exception('Data tidak ditemukan');
      case 500:
        throw Exception('Terjadi kesalahan pada server');
      default:
        throw Exception('Terjadi kesalahan tidak terduga');
    }
  }

  // Clean up resources
  static void dispose() {
    _client¿.close();
    _client = null;
    _cache.clear();
  }

  // Clear cache when needed
  static void clearCache() {
    _cache.clear();
  }

  // Clear expired cache entries
  static void cleanExpiredCache() {
    final now = DateTime.now().millisecondsSinceEpoch;
    _cache.removeWhere((key, value) {
      final timestamp = value['timestamp'] as int;
      return now - timestamp >= _cacheTimeout;
    });
  }
}
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\core\usecases\usecase.dart

```
import 'package:dartz/dartz.dart';
import 'package:equatable/equatable.dart';
import '../error/failures.dart';

abstract class UseCase<Type, Params> {
  Future<Either<Failure, Type>> call(Params params);
}

class NoParams extends Equatable {
  @override
  List<Object> get props => [];
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\core\utils\geometric_background.dart

```
import 'package:flutter/material.dart';

class GeometricBackground extends StatelessWidget {
  final double opacity;
  final Alignment alignment;
  final double scale;
  const GeometricBackground({
    super.key,
    this.opacity = 0.1,
    this.alignment = Alignment.bottomRight,
    this.scale = 1.0,
  });

  @override
  Widget build(BuildContext context) {
    return RepaintBoundary(
      child: CustomPaint(
        painter: GeometricBackgroundPainter(opacity: opacity, scale: scale),
        size: Size.infinite,
      ),
    );
  }
}

class GeometricBackgroundPainter extends CustomPainter {
  final double opacity;
  final double scale;

  GeometricBackgroundPainter({required this.opacity, required this.scale});

  @override
  void paint(Canvas canvas, Size size) {
    final paint1 = Paint()
      ..color = const Color(0xFF207BB5).withOpacity(opacity)
      ..style = PaintingStyle.fill;

    final paint2 = Paint()
      ..color = const Color(0xFF135EA2).withOpacity(opacity)
      ..style = PaintingStyle.fill;

    final baseWidth = 150.0 * scale;
    final baseHeight = 120.0 * scale;

    final startX = size.width - baseWidth * 2;
    final startY = size.height * 0.15;

    _drawTriangleLayer(
      canvas,
      paint1,
      startX,
      startY,
      baseWidth,
      baseHeight,
      0,
    );
    _drawTriangleLayer(
      canvas,
      paint2,
      startX + baseWidth * 0.3,
      startY + baseHeight * 0.5,
      baseWidth,
      baseHeight,
      1,
    );
    _drawTriangleLayer(
      canvas,
      paint1,
      startX + baseWidth * 0.6,
      startY + baseHeight * 1.0,
      baseWidth,
      baseHeight,
      2,
    );
    _drawTriangleLayer(
      canvas,
      paint2,
      startX + baseWidth * 0.9,
      startY + baseHeight * 1.5,
      baseWidth,
      baseHeight,
      3,
    );
    _drawTriangleLayer(
      canvas,
      paint1,
      startX + baseWidth * 1.2,
      startY + baseHeight * 2.0,
      baseWidth,
      baseHeight,
      4,
    );
  }

  void _drawTriangleLayer(
    Canvas canvas,
    Paint paint,
    double x,
    double y,
    double width,
    double height,
    int layer,
  ) {
    final triangles = [
      [
        Offset(x, y),
        Offset(x + width * 0.8, y + height * 0.5),
        Offset(x, y + height),
      ],
      [
        Offset(x + width * 0.2, y - height * 0.3),
        Offset(x + width * 0.9, y - height * 0.1),
        Offset(x + width * 0.6, y + height * 0.4),
      ],
      [
        Offset(x - width * 0.1, y + height * 0.8),
        Offset(x + width * 0.6, y + height * 0.6),
        Offset(x + width * 0.3, y + height * 1.3),
      ],
    ];

    for (final triangle in triangles) {
      final path = Path();
      path.moveTo(triangle[0].dx, triangle[0].dy);
      path.lineTo(triangle[1].dx, triangle[1].dy);
      path.lineTo(triangle[2].dx, triangle[2].dy);
      path.close();
      canvas.drawPath(path, paint);
    }
  }

  @override
  bool shouldRepaint(covariant CustomPainter oldDelegate) => false;
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\auth\data\models\user.dart

```
class User {
  final String nrm;
  final String nim;
  final String namam;
  final String¿ tgdaftar;
  final String¿ tplahir;

  User({
    required this.nrm,
    required this.nim,
    required this.namam,
    this.tgdaftar,
    this.tplahir,
  });

  factory User.fromJson(Map<String, dynamic> json) {
    return User(
      nrm: json['nrm'] ¿¿ '',
      nim: json['nim'] ¿¿ '',
      namam: json['namam'] ¿¿ '',
      tgdaftar: json['tgdaftar'],
      tplahir: json['tplahir'],
    );
  }

  Map<String, dynamic> toJson() {
    return {
      'nrm': nrm,
      'nim': nim,
      'namam': namam,
      'tgdaftar': tgdaftar,
      'tplahir': tplahir,
    };
  }
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\auth\data\models\user_model.dart

```
import '../../domain/entities/user.dart';

class UserModel extends User {
  const UserModel({
    required super.namam,
    required super.nrm,
    super.nim,
    super.email,
    super.phone,
    super.tgdaftar,
    super.tplahir,
  });

  factory UserModel.fromJson(Map<String, dynamic> json) {
    return UserModel(
      namam: json['namam'] ¿¿ '',
      nrm: json['nrm'] ¿¿ '',
      nim: json['nim'],
      email: json['email'],
      phone: json['phone'],
      tgdaftar: json['tgdaftar'],
      tplahir: json['tplahir'],
    );
  }

  Map<String, dynamic> toJson() {
    return {
      'namam': namam,
      'nrm': nrm,
      'nim': nim,
      'email': email,
      'phone': phone,
      'tgdaftar': tgdaftar,
      'tplahir': tplahir,
    };
  }
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\auth\data\repositories\auth_repository_impl.dart

```
import 'package:dartz/dartz.dart';
import '../../../../core/error/failures.dart';
import '../../../../core/services/api_service.dart';
import '../../domain/entities/user.dart';
import '../../domain/repositories/auth_repository.dart';
import '../models/user_model.dart';

class AuthRepositoryImpl implements AuthRepository {
  final ApiService apiService;

  AuthRepositoryImpl({required this.apiService});

  @override
  Future<Either<Failure, User>> login(String namamNim, String nrm) async {
    try {
      final userModel = await apiService.login(namamNim, nrm);
      return Right(userModel);
    } catch (e) {
      return Left(AuthFailure(e.toString().replaceFirst('Exception: ', '')));
    }
  }

  @override
  Future<Either<Failure, void>> logout() async {
    try {
      await apiService.clearAuthToken();
      return const Right(null);
    } catch (e) {
      return Left(CacheFailure(e.toString()));
    }
  }

  @override
  Future<Either<Failure, User>> getCurrentUser() async {
    try {
      final userModel = await apiService.getProfile();
      return Right(userModel);
    } catch (e) {
      return Left(ServerFailure(e.toString().replaceFirst('Exception: ', '')));
    }
  }

  @override
  Future<Either<Failure, String¿>> getToken() async {
    try {
      final token = await apiService.getAuthToken();
      return Right(token);
    } catch (e) {
      return Left(CacheFailure(e.toString()));
    }
  }

  @override
  Future<Either<Failure, void>> clearToken() async {
    try {
      await apiService.clearAuthToken();
      return const Right(null);
    } catch (e) {
      return Left(CacheFailure(e.toString()));
    }
  }
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\auth\domain\entities\user.dart

```
import 'package:equatable/equatable.dart';

class User extends Equatable {
  final String namam; // Student full name
  final String nrm; // Student registration number
  final String¿ nim; // Student identification number
  final String¿ email; // Email (optional)
  final String¿ phone; // Phone (optional)
  final String¿ tgdaftar; // Registration date
  final String¿ tplahir; // Birth place

  const User({
    required this.namam,
    required this.nrm,
    this.nim,
    this.email,
    this.phone,
    this.tgdaftar,
    this.tplahir,
  });

  @override
  List<Object¿> get props => [namam, nrm, nim, email, phone, tgdaftar, tplahir];
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\auth\domain\repositories\auth_repository.dart

```
import 'package:dartz/dartz.dart';
import '../../../../core/error/failures.dart';
import '../entities/user.dart';

abstract class AuthRepository {
  Future<Either<Failure, User>> login(String namamNim, String nrm);
  Future<Either<Failure, void>> logout();
  Future<Either<Failure, User>> getCurrentUser();
  Future<Either<Failure, String¿>> getToken();
  Future<Either<Failure, void>> clearToken();
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\auth\domain\usecases\check_auth_status_usecase.dart

```
import 'package:dartz/dartz.dart';
import '../../../../core/error/failures.dart';
import '../../../../core/usecases/usecase.dart';
import '../entities/user.dart';
import '../repositories/auth_repository.dart';

class CheckAuthStatusUseCase implements UseCase<User¿, NoParams> {
  final AuthRepository repository;

  CheckAuthStatusUseCase(this.repository);

  @override
  Future<Either<Failure, User¿>> call(NoParams params) async {
    final tokenResult = await repository.getToken();

    return tokenResult.fold((failure) => Left(failure), (token) async {
      if (token == null) {
        return const Right(null);
      }

      final userResult = await repository.getCurrentUser();
      return userResult.fold((failure) => Left(failure), (user) => Right(user));
    });
  }
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\auth\domain\usecases\login_usecase.dart

```
import 'package:dartz/dartz.dart';
import 'package:equatable/equatable.dart';
import '../../../../core/error/failures.dart';
import '../../../../core/usecases/usecase.dart';
import '../entities/user.dart';
import '../repositories/auth_repository.dart';
```

```
class LoginUseCase implements UseCase<User, LoginParams> {
  final AuthRepository repository;

  LoginUseCase(this.repository);

  @override
  Future<Either<Failure, User>> call(LoginParams params) async {
    return await repository.login(params.namamNim, params.nrm);
  }
}

class LoginParams extends Equatable {
  final String namamNim;
  final String nrm;

  const LoginParams({required this.namamNim, required this.nrm});

  @override
  List<Object> get props => [namamNim, nrm];
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\auth\domain\usecases\logout_usecase.dart

```
import 'package:dartz/dartz.dart';
import '../../../../core/error/failures.dart';
import '../../../../core/usecases/usecase.dart';
import '../repositories/auth_repository.dart';

class LogoutUseCase implements UseCase<void, NoParams> {
  final AuthRepository repository;

  LogoutUseCase(this.repository);

  @override
  Future<Either<Failure, void>> call(NoParams params) async {
    return await repository.logout();
  }
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\auth\presentation\bloc\auth_bloc.dart

```
import 'package:flutter_bloc/flutter_bloc.dart';
import '../../../../core/error/failures.dart';
import '../../../../core/usecases/usecase.dart';
import '../../domain/usecases/check_auth_status_usecase.dart';
import '../../domain/usecases/login_usecase.dart';
import '../../domain/usecases/logout_usecase.dart';
import 'auth_event.dart';
import 'auth_state.dart';

class AuthBloc extends Bloc<AuthEvent, AuthState> {
  final LoginUseCase loginUseCase;
  final LogoutUseCase logoutUseCase;
  final CheckAuthStatusUseCase checkAuthStatusUseCase;

  AuthBloc({
    required this.loginUseCase,
    required this.logoutUseCase,
    required this.checkAuthStatusUseCase,
  }) : super(const AuthInitial()) {
    on<CheckAuthStatusEvent>(_onCheckAuthStatus);
    on<LoginRequestedEvent>(_onLoginRequested);
    on<LogoutRequestedEvent>(_onLogoutRequested);
  }

  Future<void> _onCheckAuthStatus(
    CheckAuthStatusEvent event,
    Emitter<AuthState> emit,
  ) async {
    emit(const AuthLoading());

    final result = await checkAuthStatusUseCase(NoParams());

    result.fold(
      (failure) => emit(AuthError(message: _mapFailureToMessage(failure))),
      (user) {
        if (user != null) {
          emit(AuthAuthenticated(user: user));
        } else {
          emit(const AuthUnauthenticated());
        }
      },
    );
  }

  Future<void> _onLoginRequested(
    LoginRequestedEvent event,
    Emitter<AuthState> emit,
  ) async {
    emit(const AuthLoading());

    final result = await loginUseCase(
      LoginParams(namamNim: event.namamNim, nrm: event.nrm),
    );

    result.fold(
      (failure) => emit(AuthError(message: _mapFailureToMessage(failure))),
      (user) => emit(AuthAuthenticated(user: user)),
    );
  }

  Future<void> _onLogoutRequested(
    LogoutRequestedEvent event,
    Emitter<AuthState> emit,
  ) async {
    final result = await logoutUseCase(NoParams());

    result.fold(
      (failure) => emit(AuthError(message: _mapFailureToMessage(failure))),
      (_) => emit(const AuthUnauthenticated()),
    );
  }

  String _mapFailureToMessage(Failure failure) {
    switch (failure.runtimeType) {
      case AuthFailure:
        return (failure as AuthFailure).message;
      case ServerFailure:
        return (failure as ServerFailure).message;
      case NetworkFailure:
        return (failure as NetworkFailure).message;
      case CacheFailure:
        return (failure as CacheFailure).message;
      default:
        return 'Unexpected error occurred';
    }
  }
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\auth\presentation\bloc\auth_event.dart

```
import 'package:equatable/equatable.dart';

abstract class AuthEvent extends Equatable {
  const AuthEvent();

  @override
  List<Object> get props => [];
}

class CheckAuthStatusEvent extends AuthEvent {
  const CheckAuthStatusEvent();
}

class LoginRequestedEvent extends AuthEvent {
  final String namamNim;
  final String nrm;

  const LoginRequestedEvent({required this.namamNim, required this.nrm});

  @override
  List<Object> get props => [namamNim, nrm];
}

class LogoutRequestedEvent extends AuthEvent {
  const LogoutRequestedEvent();
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\auth\presentation\bloc\auth_state.dart

```
import 'package:equatable/equatable.dart';
import '../../domain/entities/user.dart';

abstract class AuthState extends Equatable {
  const AuthState();

  @override
  List<Object¿> get props => [];
}

class AuthInitial extends AuthState {
  const AuthInitial();
}

class AuthLoading extends AuthState {
  const AuthLoading();
}

class AuthAuthenticated extends AuthState {
  final User user;

  const AuthAuthenticated({required this.user});

  @override
  List<Object> get props => [user];
}

class AuthUnauthenticated extends AuthState {
  const AuthUnauthenticated();
}

class AuthError extends AuthState {
  final String message;

  const AuthError({required this.message});

  @override
  List<Object> get props => [message];
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\auth\presentation\pages\login_page.dart

```
import 'package:flutter/material.dart';
import 'package:flutter_bloc/flutter_bloc.dart';
import '../../../../core/di/injection_container.dart' as di;
import '../bloc/auth_bloc.dart';
import '../bloc/auth_event.dart';
import '../bloc/auth_state.dart';

class LoginPage extends StatelessWidget {
  const LoginPage({super.key});

  @override
  Widget build(BuildContext context) {
    return const _LoginForm();
  }
}

class _LoginForm extends StatefulWidget {
  const _LoginForm();

  @override
  State<_LoginForm> createState() => _LoginFormState();
}

class _LoginFormState extends State<_LoginForm> {
  final _nimController = TextEditingController();
  final _nrmController = TextEditingController();
  final _formKey = GlobalKey<FormState>();
```

```
@override
void dispose() {
  _nimController.dispose();
  _nrmController.dispose();
  super.dispose();
}

void _onLoginPressed() {
  // Hide keyboard
  FocusScope.of(context).unfocus();
  // Dispatch event
  context.read<AuthBloc>().add(
    LoginRequestedEvent(
      namamNim: _nimController.text,
      nrm: _nrmController.text,
    ),
  );
}

@override
Widget build(BuildContext context) {
  return BlocListener<AuthBloc, AuthState>(
    listener: (context, state) {
      if (state is AuthError) {
        ScaffoldMessenger.of(context)
          ..hideCurrentSnackBar()
          ..showSnackBar(SnackBar(content: Text(state.message)));
      }
    },
    child: Scaffold(
      appBar: AppBar(title: const Text('Login Mahasiswa'), centerTitle: true),
      body: Center(
        child: SingleChildScrollView(
          padding: const EdgeInsets.all(24.0),
          child: Column(
            mainAxisAlignment: MainAxisAlignment.center,
            crossAxisAlignment: CrossAxisAlignment.stretch,
            children: [
              Icon(
                Icons.school,
                size: 80,
                color: Theme.of(context).primaryColor,
              ),
              const SizedBox(height: 24),
              TextField(
                controller: _nimController,
                decoration: const InputDecoration(
                  labelText: 'NIM/NPM',
                  border: OutlineInputBorder(),
                  prefixIcon: Icon(Icons.person),
                ),
                keyboardType: TextInputType.text,
              ),
              const SizedBox(height: 16),
              TextField(
                controller: _nrmController,
                decoration: const InputDecoration(
                  labelText: 'NRM',
                  border: OutlineInputBorder(),
                  prefixIcon: Icon(Icons.lock),
                ),
                obscureText: true,
              ),
              const SizedBox(height: 32),
              BlocBuilder<AuthBloc, AuthState>(
                builder: (context, state) {
                    final isLoading = state is AuthLoading;
                    return ElevatedButton(
                      style: ElevatedButton.styleFrom(
                        padding: const EdgeInsets.symmetric(vertical: 16),
                        shape: RoundedRectangleBorder(
                          borderRadius: BorderRadius.circular(8),
                        ),
                      ),
                      onPressed: isLoading ¿ null : _onLoginPressed,
                      child: isLoading
                          ¿ const SizedBox(
                              height: 20,
                              width: 20,
                              child: CircularProgressIndicator(
                                strokeWidth: 2,
                                color: Colors.white,
                              ),
                            )
                          : const Text('Login'),
                    );
                  },
                ),
              ],
            ),
          ),
        ),
      ),
    );
  }
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\dashboard\presentation\pages\dashboard_page.dart

```
import 'package:flutter/material.dart';
import 'package:flutter_bloc/flutter_bloc.dart';
import '../../../auth/presentation/bloc/auth_bloc.dart';
import '../../../auth/presentation/bloc/auth_event.dart';
import '../../../auth/presentation/bloc/auth_state.dart';
import 'package:wismon_keuangan/features/payment/presentation/pages/wismon_page.dart';

class HomePage extends StatelessWidget {
  const HomePage({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.grey[50],
      body: SafeArea(
        child: BlocListener<AuthBloc, AuthState>(
          listener: (context, state) {
            if (state is AuthError) {
              ScaffoldMessenger.of(context).showSnackBar(
                SnackBar(
                  content: Text(state.message),
                  backgroundColor: Colors.red,
                ),
              );
            }
          },
          child: BlocBuilder<AuthBloc, AuthState>(
            builder: (context, state) {
              if (state is AuthLoading) {
                return const Center(child: CircularProgressIndicator());
              }

              if (state is AuthAuthenticated) {
                return _buildHomePage(context, state.user.namam);
              }

              return const Center(child: Text('Loading...'));
            },
          ),
        ),
      ),
      bottomNavigationBar: BottomNavigationBar(
        type: BottomNavigationBarType.fixed,
        selectedItemColor: Colors.blue[600],
        unselectedItemColor: Colors.grey[600],
        currentIndex: 0,
        items: const [
          BottomNavigationBarItem(icon: Icon(Icons.home), label: 'Home'),
          BottomNavigationBarItem(icon: Icon(Icons.person), label: 'Profile'),
        ],
        onTap: (index) {
          if (index == 1) {
            _showProfile(context);
          }
        },
      ),
    );
  }

  Widget _buildHomePage(BuildContext context, String userName) {
    return Padding(
      padding: const EdgeInsets.all(24.0),
      child: Column(
        crossAxisAlignment: CrossAxisAlignment.start,
        children: [
          // Greeting
          Container(
            padding: const EdgeInsets.all(20),
            decoration: BoxDecoration(
              gradient: LinearGradient(
                colors: [Colors.blue[600]!, Colors.blue[400]!],
                begin: Alignment.topLeft,
                end: Alignment.bottomRight,
              ),
              borderRadius: BorderRadius.circular(16),
              boxShadow: [
                BoxShadow(
                  color: Colors.blue.withOpacity(0.3),
                  blurRadius: 10,
                  offset: const Offset(0, 4),
                ),
              ],
            ),
            child: Row(
              children: [
                Container(
                  width: 60,
                  height: 60,
                  decoration: BoxDecoration(
                    color: Colors.white.withValues(alpha: 0.2),
                    borderRadius: BorderRadius.circular(12),
                  ),
                  child: const Icon(
                    Icons.person,
                    color: Colors.white,
                    size: 32,
                  ),
                ),
                const SizedBox(width: 16),
                Expanded(
                  child: Column(
                    crossAxisAlignment: CrossAxisAlignment.start,
                    children: [
                      const Text(
                        'Welcome!',
                        style: TextStyle(
                          color: Colors.white,
                          fontSize: 14,
                          fontWeight: FontWeight.w500,
                        ),
                      ),
                      Text(
                        'hello, $userName',
                        style: const TextStyle(
                          color: Colors.white,
                          fontSize: 20,
                          fontWeight: FontWeight.bold,
                        ),
                      ),
                    ],
                  ),
                ),
                // Logout button
                IconButton(
                  onPressed: () => _showLogoutDialog(context),
                  icon: const Icon(Icons.logout, color: Colors.white),
                ),
              ],
            ),
          ),
          const SizedBox(height: 32),

          // Navigation Grid
          Text(
            'Quick Access',
            style: Theme.of(context).textTheme.titleLarge¿.copyWith(
              fontWeight: FontWeight.bold,
              color: Colors.grey[800],
            ),
          ),
          const SizedBox(height: 16),

          Expanded(
            child: GridView.count(
              crossAxisCount: 2,
              crossAxisSpacing: 16,
              mainAxisSpacing: 16,
              children: [
                // Wismon Button
                _FeatureCard(
                  context: context,
                  icon: Icons.monetization_on,
                  title: 'Wismon',
                  subtitle: 'Check your payment status',
                  onTap: () {
                    Navigator.push(
                      context,
                      MaterialPageRoute(
                        builder: (context) => const WismonPage(),
                      ),
                    );
                  },
                ),

                // WIS Button
                _FeatureCard(
                  context: context,
                  icon: Icons.school,
                  title: 'WIS',
                  subtitle: 'Student Information',
                  onTap: () {
                    ScaffoldMessenger.of(context).showSnackBar(
                      const SnackBar(
                        content: Text('Feature not implemented yet'),
                      ),
                    );
                  },
                ),

                // Feature 3 - Future placeholder
                _FeatureCard(
                  context: context,
                  icon: Icons.construction,
                  title: 'Feature 3',
                  subtitle: 'Coming Soon',
                  onTap: () {
                    ScaffoldMessenger.of(context).showSnackBar(
                      const SnackBar(
                        content: Text('Feature not implemented yet'),
                      ),
                    );
                  },
                ),

                // Feature 4 - Future placeholder
                _FeatureCard(
                  context: context,
                  icon: Icons.settings,
                  title: 'Feature 4',
                  subtitle: 'Coming Soon',
                  onTap: () {
                    ScaffoldMessenger.of(context).showSnackBar(
                      const SnackBar(
                        content: Text('Feature not implemented yet'),
                      ),
                    );
                  },
                ),
              ],
            ),
          ),
        ],
      ),
    );
  }

  Widget _FeatureCard({
    required BuildContext context,
    required IconData icon,
    required String title,
    required String subtitle,
    required VoidCallback onTap,
  }) {
    return Card(
      elevation: 4,
      shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(16)),
      child: InkWell(
        onTap: onTap,
        borderRadius: BorderRadius.circular(16),
        child: Padding(
          padding: const EdgeInsets.all(20),
          child: Column(
            mainAxisAlignment: MainAxisAlignment.center,
            children: [
              Container(
                width: 48,
                height: 48,
                decoration: BoxDecoration(
                  color: Colors.blue.withOpacity(0.1),
                  borderRadius: BorderRadius.circular(12),
                ),
                child: Icon(icon, color: Colors.blue, size: 24),
              ),
              const SizedBox(height: 12),
              Text(
                title,
                style: Theme.of(context).textTheme.titleMedium¿.copyWith(
                  fontWeight: FontWeight.bold,
                  color: Colors.grey[800],
                ),
              ),
              const SizedBox(height: 4),
              Text(
                subtitle,
                style: Theme.of(
                  context,
                ).textTheme.bodySmall¿.copyWith(color: Colors.grey[600]),
                textAlign: TextAlign.center,
              ),
            ],
          ),
        ),
      ),
    );
  }

  void _showProfile(BuildContext context) {
    final authState = context.read<AuthBloc>().state;
    if (authState is AuthAuthenticated) {
      showDialog(
        context: context,
        builder: (context) => AlertDialog(
          title: const Text('Profile'),
          content: Column(
            mainAxisSize: MainAxisSize.min,
            crossAxisAlignment: CrossAxisAlignment.start,
            children: [
              Text('Name: ${authState.user.namam}'),
              Text('NRM: ${authState.user.nrm}'),
              if (authState.user.email != null)
                Text('Email: ${authState.user.email}'),
              if (authState.user.phone != null)
                Text('Phone: ${authState.user.phone}'),
            ],
          ),
          actions: [
            TextButton(
              onPressed: () => Navigator.of(context).pop(),
              child: const Text('Close'),
            ),
          ],
        ),
      );
    }
  }

  void _showLogoutDialog(BuildContext context) {
    showDialog(
      context: context,
      builder: (context) => AlertDialog(
        title: const Text('Logout'),
        content: const Text('Are you sure you want to logout¿'),
        actions: [
          TextButton(
            onPressed: () => Navigator.of(context).pop(),
            child: const Text('Cancel'),
          ),
          TextButton(
            onPressed: () {
              Navigator.of(context).pop();
              context.read<AuthBloc>().add(const LogoutRequestedEvent());
            },
            child: const Text('Logout'),
          ),
        ],
      ),
    );
  }
  }
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\dashboard\presentation\pages\main_navigation_page.dart

```
import 'package:flutter/material.dart';
import 'package:flutter_bloc/flutter_bloc.dart';
import 'package:flutter_svg/flutter_svg.dart';
import 'package:wismon_keuangan/features/auth/presentation/bloc/auth_bloc.dart';
import 'package:wismon_keuangan/features/auth/presentation/bloc/auth_state.dart';
import 'package:wismon_keuangan/features/dashboard/presentation/pages/dashboard_page.dart';
import 'package:wismon_keuangan/features/menu/presentation/pages/menu_page.dart';
import 'package:wismon_keuangan/features/profile/presentation/pages/profile_page.dart';

class MainNavigationPage extends StatefulWidget {
  const MainNavigationPage({super.key});

  @override
  State<MainNavigationPage> createState() => _MainNavigationPageState();
}

class _MainNavigationPageState extends State<MainNavigationPage> {
  late PageController _pageController;
  int _selectedIndex = 0;

  final List<Widget> _pages = [
    const DashboardPage(),
    const MenuPage(),
    const ProfilePage(),
  ];

  @override
```

```dart
void initState() {
  super.initState();
  _pageController = PageController(initialPage: _selectedIndex);
}

@override
void dispose() {
  _pageController.dispose();
  super.dispose();
}

void _onItemTapped(int index) {
  if (_selectedIndex != index) {
    setState(() {
      _selectedIndex = index;
    });
    _pageController.animateToPage(
      index,
      duration: const Duration(milliseconds: 200),
      curve: Curves.easeInOut,
    );
  }
}

@override
Widget build(BuildContext context) {
  return BlocBuilder<AuthBloc, AuthState>(
    builder: (context, state) {
      if (state is AuthLoading) {
        return const Scaffold(
          body: Center(child: CircularProgressIndicator()),
        );
      }

      if (state is AuthAuthenticated) {
        return Scaffold(
          body: PageView(
            controller: _pageController,
            onPageChanged: (index) {
              setState(() {
                _selectedIndex = index;
              });
            },
            children: _pages,
          ),
          bottomNavigationBar: _buildBottomNavigationBar(),
        );
      }

      return const Center(child: Text('Loading...'));
    },
  );
}

Widget _buildBottomNavigationBar() {
  return Container(
    decoration: const BoxDecoration(
      border: Border(top: BorderSide(color: Color(0xFFE7E7E7), width: 1)),
    ),
    child: BottomNavigationBar(
      type: BottomNavigationBarType.fixed,
      backgroundColor: Colors.white,
      selectedItemColor: const Color(0xFF121212),
      unselectedItemColor: const Color(0xFF858586),
      selectedLabelStyle: const TextStyle(
        fontSize: 12,
          fontWeight: FontWeight.w800,
          letterSpacing: -0.12,
        ),
        unselectedLabelStyle: const TextStyle(
          fontSize: 12,
          fontWeight: FontWeight.w500,
          letterSpacing: -0.12,
        ),
        currentIndex: _selectedIndex,
        onTap: _onItemTapped,
        elevation: 0,
        items: const [
          BottomNavigationBarItem(
            icon: Padding(
              padding: EdgeInsets.only(bottom: 4),
              child: Icon(Icons.home_outlined, size: 24),
            ),
            activeIcon: Padding(
              padding: EdgeInsets.only(bottom: 4),
              child: Icon(Icons.home, size: 24),
            ),
            label: 'Beranda',
          ),
          BottomNavigationBarItem(
            icon: Padding(
              padding: EdgeInsets.only(bottom: 4),
              child: Icon(Icons.menu, size: 24),
            ),
            activeIcon: Padding(
              padding: EdgeInsets.only(bottom: 4),
              child: Icon(Icons.menu, size: 24),
            ),
            label: 'Menu',
          ),
          BottomNavigationBarItem(
            icon: Padding(
              padding: EdgeInsets.only(bottom: 4),
              child: Icon(Icons.person_outline, size: 24),
            ),
            activeIcon: Padding(
              padding: EdgeInsets.only(bottom: 4),
              child: Icon(Icons.person, size: 24),
            ),
            label: 'Profil',
          ),
        ],
      ),
    );
  }
}

class DashboardPage extends StatelessWidget {
  const DashboardPage({super.key});

  @override
  Widget build(BuildContext context) {
    return const Scaffold(
      body: Center(child: Text('Beranda Page (Work in Progress)')),
    );
  }
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\menu\presentation\pages\menu_page.dart

```
import 'package:flutter/material.dart';
import 'package:wismon_keuangan/features/payment/presentation/pages/wismon_page.dart';
import 'package:flutter_bloc/flutter_bloc.dart';
import 'package:wismon_keuangan/features/auth/presentation/bloc/auth_bloc.dart';
import 'package:wismon_keuangan/features/auth/presentation/bloc/auth_event.dart';
import 'package:flutter_svg/flutter_svg.dart';

class MenuPage extends StatelessWidget {
  const MenuPage({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.transparent,
      body: Stack(
        children: [
          // Correctly positioned logo background
          Positioned(
            right: 100,
            bottom: 180,
            child: RepaintBoundary(
              child: Opacity(
                opacity: 1,
                child: Transform.scale(
                  scale: 4,
                  child: SvgPicture.asset(
                    'assets/LOGOWHS.svg',
                    width: 300,
                    height: 185,
                    fit: BoxFit.contain,
                  ),
                ),
              ),
            ),
          ),
          SafeArea(
            child: Column(
              children: [
                _buildHeader(context),
                Expanded(child: _buildMenuItems(context)),
              ],
            ),
          ),
        ],
      ),
    );
  }

  Widget _buildHeader(BuildContext context) {
    return RepaintBoundary(
      child: Container(
        padding: const EdgeInsets.symmetric(horizontal: 16, vertical: 12),
        child: Row(
          mainAxisAlignment: MainAxisAlignment.spaceBetween,
          children: [
            const Text(
              'Menu Aplikasi',
              style: TextStyle(
                fontSize: 24,
                fontWeight: FontWeight.w800,
                color: Color(0xFF121212),
                letterSpacing: -0.24,
              ),
            ),
            Container(
              width: 40,
              height: 40,
              decoration: BoxDecoration(
                color: Colors.white,
                borderRadius: BorderRadius.circular(8),
                boxShadow: const [
                  BoxShadow(
                    color: Color(0x0D000000),
                    blurRadius: 10,
                    offset: Offset(0, 4),
                  ),
                ],
              ),
              child: IconButton(
                onPressed: () => _showSettingsLogoutSheet(context),
                icon: const Icon(
                  Icons.settings,
                  color: Color(0xFF121212),
                  size: 20,
                ),
                padding: EdgeInsets.zero,
              ),
            ),
          ],
        ),
      ),
    );
  }

  void _showSettingsLogoutSheet(BuildContext context) {
    showModalBottomSheet(
      context: context,
      shape: const RoundedRectangleBorder(
        borderRadius: BorderRadius.vertical(top: Radius.circular(20)),
      ),
      builder: (BuildContext sheetContext) {
        return Container(
          padding: const EdgeInsets.all(20),
          child: Column(
            mainAxisSize: MainAxisSize.min,
            children: [
              const Text(
                'Pengaturan',
                style: TextStyle(fontSize: 18, fontWeight: FontWeight.bold),
              ),
              const SizedBox(height: 20),
              const Text(
                  'Aplikasi WHN Mobile adalah sebuah aplikasi sistem informasi layanan terpadu yang memberikan
kemudahan akses akan informasi dan layanan bagi seluruh pengguna layanan Sistem Informasi Wira Husada Nusantara
yang menggunakan perangkat mobile.',
                textAlign: TextAlign.center,
                style: TextStyle(fontSize: 14, color: Colors.grey),
              ),
              const SizedBox(height: 20),
              ElevatedButton(
                onPressed: () {
                  Navigator.pop(sheetContext);
                  context.read<AuthBloc>().add(const LogoutRequestedEvent());
                },
                style: ElevatedButton.styleFrom(
                  backgroundColor: const Color(0xFF207BB5),
                  padding: const EdgeInsets.symmetric(vertical: 12),
                  shape: RoundedRectangleBorder(
                    borderRadius: BorderRadius.circular(8),
                  ),
                  minimumSize: const Size(double.infinity, 50),
                ),
                child: const Text(
                  'Keluar Akun',
                  style: TextStyle(color: Colors.white, fontSize: 16),
                ),
              ),
              const SizedBox(height: 10),
              TextButton(
                onPressed: () => Navigator.pop(sheetContext),
                child: const Text(
                  'Batal',
                  style: TextStyle(color: Colors.grey),
                ),
              ),
            ],
          ),
        );
      },
    );
  }

  Widget _buildMenuItems(BuildContext context) {
    return RepaintBoundary(
      child: SingleChildScrollView(
        padding: const EdgeInsets.symmetric(horizontal: 16),
        physics: const AlwaysScrollableScrollPhysics(),
        child: Column(
          children: [
            const SizedBox(height: 4),
            _buildMenuItem(
              context: context,
              icon: Icons.account_balance_wallet_outlined,
              title: 'Biaya Kuliah',
              subtitle: 'Detail biaya kuliah Anda.',
              onTap: () {
                Navigator.push(
                  context,
                  PageRouteBuilder(
                    pageBuilder: (context, animation, secondaryAnimation) =>
                        const WismonPage(),
                    transitionsBuilder:
                        (context, animation, secondaryAnimation, child) {
                          return FadeTransition(
                            opacity: animation,
                            child: child,
                          );
                        },
                    transitionDuration: const Duration(milliseconds: 200),
                  ),
                );
              },
            ),
            const SizedBox(height: 16),
            _buildMenuItem(
              context: context,
              icon: Icons.description_outlined,
              title: 'Transkrip Nilai',
              subtitle: 'Dokumen resmi riwayat akademik lengkap Anda.',
              onTap: () {
                _showComingSoonSnackBar(context, 'Transkrip Nilai');
              },
            ),
            const SizedBox(height: 16),
            _buildMenuItem(
              context: context,
              icon: Icons.folder_open_outlined,
              title: 'Kartu Hasil Studi (KHS)',
              subtitle: 'Lihat nilai dan progres akademik Anda dengan mudah.',
              onTap: () {
                _showComingSoonSnackBar(context, 'Kartu Hasil Studi');
              },
            ),
            const SizedBox(height: 16),
            _buildMenuItem(
              context: context,
              icon: Icons.edit_note_outlined,
              title: 'Kartu Rencana Studi (KRS)',
              subtitle: 'Rencanakan dan pilih mata kuliah selama perkuliahan.',
              onTap: () {
                _showComingSoonSnackBar(context, 'Kartu Rencana Studi');
              },
            ),
            const SizedBox(height: 16),
            _buildMenuItem(
              context: context,
              icon: Icons.calendar_today_outlined,
              title: 'Presensi Kelas',
              subtitle: 'Catat kehadiran Anda di setiap sesi perkuliahan.',
              onTap: () {
                _showComingSoonSnackBar(context, 'Presensi Kelas');
              },
            ),
            const SizedBox(height: 16),
            _buildMenuItem(
              context: context,
              icon: Icons.public_outlined,
              title: 'Layanan Perpustakaan',
              subtitle: 'Akses berbagai sumber daya akademik.',
              onTap: () {
                _showComingSoonSnackBar(context, 'Layanan Perpustakaan');
              },
            ),
            const SizedBox(height: 20), // Extra padding at bottom
          ],
        ),
      ),
    );
  }

  void _showComingSoonSnackBar(BuildContext context, String featureName) {
    ScaffoldMessenger.of(context).showSnackBar(
      SnackBar(
        content: Text('$featureName akan segera tersedia'),
        backgroundColor: const Color(0xFF207BB5),
        behavior: SnackBarBehavior.floating,
        shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(8)),
        margin: const EdgeInsets.all(16),
      ),
    );
  }

  Widget _buildMenuItem({
    required BuildContext context,
    required IconData icon,
    required String title,
    required String subtitle,
    required VoidCallback onTap,
  }) {
```

```
return RepaintBoundary(
  child: Container(
    decoration: BoxDecoration(
      color: Colors.white,
      borderRadius: BorderRadius.circular(8),
      boxShadow: const [
        BoxShadow(
          color: Color(0x08000000),
          blurRadius: 12,
          offset: Offset(0, 4),
        ),
      ],
    ),
    child: Material(
      color: Colors.transparent,
      child: InkWell(
        onTap: onTap,
        borderRadius: BorderRadius.circular(8),
        splashColor: const Color(0xFF207BB5).withOpacity(0.1),
        highlightColor: const Color(0xFF207BB5).withOpacity(0.05),
        child: Padding(
          padding: const EdgeInsets.all(8),
          child: Row(
            children: [
              Container(
                width: 48,
                height: 48,
                decoration: BoxDecoration(
                  border: Border.all(
                    color: const Color(0xFFCECECF),
                    width: 1,
                  ),
                  borderRadius: BorderRadius.circular(4),
                ),
                child: Icon(icon, color: const Color(0xFF121212), size: 24),
              ),
              const SizedBox(width: 12),
              Expanded(
                child: Column(
                  crossAxisAlignment: CrossAxisAlignment.start,
                  mainAxisAlignment: MainAxisAlignment.center,
                  children: [
                    Text(
                      title,
                      style: const TextStyle(
                        fontSize: 14,
                        fontWeight: FontWeight.w500,
                        color: Color(0xFF1C1D1F),
                        letterSpacing: -0.14,
                      ),
                    ),
                    const SizedBox(height: 4),
                    Text(
                      subtitle,
                      style: const TextStyle(
                        fontSize: 12,
                        fontWeight: FontWeight.w400,
                        color: Color(0xFF545556),
                        letterSpacing: -0.12,
                      ),
                      maxLines: 2,
                      overflow: TextOverflow.ellipsis,
                    ),
                  ],
                ),
              ),
                ],
              ),
            ),
          ),
        ),
      ),
    );
  }
}
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\payment\data\models\payment_model.dart

```
import '../../domain/entities/payment.dart';

// Helper function to parse currency strings
double _parseCurrency(dynamic value) {
  if (value is num) {
    return value.toDouble();
  }
  if (value is String) {
    try {
      // Remove "Rp", whitespace, and all separators (.,), then parse.
      final sanitized = value.replaceAll(RegExp(r'[Rp.,\s]'), '');
      return double.parse(sanitized);
    } catch (e) {
      return 0.0;
    }
  }
  return 0.0;
}

// --- PAYMENT HISTORY MODEL ---

class PaymentHistoryItemModel extends PaymentHistoryItem {
  const PaymentHistoryItemModel({
    required super.id,
    required super.txId,
    required super.tanggal,
    required super.tanggalFull,
    required super.type,
    required super.jumlah,
    required super.status,
    required super.method,
    required super.methodCode,
  });

  factory PaymentHistoryItemModel.fromJson(Map<String, dynamic> json) {
    return PaymentHistoryItemModel(
      id: json['id'].toString(),
      txId: json['tx_id'].toString(),
      tanggal: json['tanggal'].toString(),
      tanggalFull: json['tanggal_full'].toString(),
      type: json['type'].toString(),
      jumlah: _parseCurrency(json['jumlah']),
      status: json['status'].toString(),
      method: json['method'].toString(),
      methodCode: json['method_code'].toString(),
    );
  }
}

// --- PAYMENT SUMMARY MODEL ---

class PaymentSummaryModel extends PaymentSummary {
  const PaymentSummaryModel({
    required super.totalPembayaran,
    required super.breakdown,
  });

  factory PaymentSummaryModel.fromJson(Map<String, dynamic> json) {
    final breakdownData = json['breakdown'] as Map<String, dynamic>¿ ¿¿ {};
    final breakdown = breakdownData.map(
      (key, value) => MapEntry(key, _parseCurrency(value)),
    );

    return PaymentSummaryModel(
      totalPembayaran: _parseCurrency(json['total_pembayaran']),
      breakdown: breakdown,
    );
  }
}

// --- TRANSACTION DETAIL MODEL ---

class TransactionDetailModel extends TransactionDetail {
  const TransactionDetailModel({
    required super.id,
    required super.txId,
    required super.tanggal,
    required super.tanggalFull,
    required super.type,
    required super.jumlah,
    required super.status,
    required super.method,
    required super.studentName,
    required super.studentNim,
    required super.studentProdi,
    required super.paymentBreakdown,
  });

  factory TransactionDetailModel.fromJson(Map<String, dynamic> json) {
    final breakdownData =
        json['payment_breakdown'] as Map<String, dynamic>¿ ¿¿ {};
    final paymentBreakdown = breakdownData.map(
      (key, value) => MapEntry(key, _parseCurrency(value)),
    );

    return TransactionDetailModel(
      id: json['id'].toString(),
      txId: json['tx_id'].toString(),
      tanggal: json['tanggal'].toString(),
      tanggalFull: json['tanggal_full'].toString(),
      type: json['type'].toString(),
      jumlah: _parseCurrency(json['jumlah']),
      status: json['status'].toString(),
      method: json['method'].toString(),
      studentName: json['student_name'].toString(),
      studentNim: json['student_nim'].toString(),
      studentProdi: json['student_prodi'].toString(),
      paymentBreakdown: paymentBreakdown,
    );
  }
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\payment\data\repositories\

payment_repository_impl.dart

```
import 'package:dartz/dartz.dart';
import '../../../../core/error/failures.dart';
import '../../../../core/services/api_service.dart';
import '../../domain/entities/payment.dart';
import '../../domain/repositories/payment_repository.dart';
import '../models/payment_model.dart';

class PaymentRepositoryImpl implements PaymentRepository {
  final ApiService apiService;

  PaymentRepositoryImpl({required this.apiService});

  @override
  Future<Either<Failure, List<PaymentHistoryItem>>> getPaymentHistory({
    int page = 1,
    int limit = 20,
    String¿ startDate,
    String¿ endDate,
    String¿ type,
    String sortBy = 'tanggal',
    String sortOrder = 'desc',
  }) async {
    try {
      final historyModels = await apiService.getPaymentHistory(
        page: page,
        limit: limit,
        startDate: startDate,
        endDate: endDate,
        type: type,
        sortBy: sortBy,
        sortOrder: sortOrder,
      );

      // Convert models to entities
      final List<PaymentHistoryItem> entities = historyModels
          .cast<PaymentHistoryItem>();
      return Right(entities);
    } catch (e) {
      return Left(ServerFailure(e.toString().replaceFirst('Exception: ', '')));
    }
  }

  @override
  Future<Either<Failure, PaymentSummary>> getPaymentSummary() async {
    try {
      final summaryModel = await apiService.getPaymentSummary();
      return Right(summaryModel);
    } catch (e) {
      return Left(ServerFailure(e.toString().replaceFirst('Exception: ', '')));
    }
  }

  @override
  Future<Either<Failure, TransactionDetail>> getTransactionDetail(
    String transactionId,
  ) async {
    try {
      final detailModel = await apiService.getTransactionDetail(transactionId);
      return Right(detailModel);
    } catch (e) {
      return Left(ServerFailure(e.toString().replaceFirst('Exception: ', '')));
    }
  }

  @override
  Future<Either<Failure, bool>> refreshPaymentData() async {
    try {
      final refreshed = await apiService.refreshPaymentData();
      return Right(refreshed);
    } catch (e) {
      return Left(ServerFailure(e.toString().replaceFirst('Exception: ', '')));
    }
  }
}
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\payment\domain\entities\payment.dart

```
import 'package:equatable/equatable.dart';

// --- PAYMENT HISTORY ---

class PaymentHistoryItem extends Equatable {
  final String id;
  final String txId;
  final String tanggal;
  final String tanggalFull;
  final String type;
  final double jumlah;
  final String status;
  final String method;
  final String methodCode;

  const PaymentHistoryItem({
    required this.id,
    required this.txId,
    required this.tanggal,
    required this.tanggalFull,
    required this.type,
    required this.jumlah,
    required this.status,
    required this.method,
    required this.methodCode,
  });

  @override
  List<Object¿> get props => [
    id,
    txId,
    tanggal,
    tanggalFull,
    type,
    jumlah,
    status,
    method,
    methodCode,
  ];
}
```

```
// --- PAYMENT SUMMARY ---

class PaymentSummary extends Equatable {
  final double totalPembayaran;
  final Map<String, double> breakdown;

  const PaymentSummary({
    required this.totalPembayaran,
    required this.breakdown,
  });

  @override
  List<Object> get props => [totalPembayaran, breakdown];
}

// --- TRANSACTION DETAIL ---

class TransactionDetail extends Equatable {
  final String id;
  final String txId;
  final String tanggal;
  final String tanggalFull;
  final String type;
  final double jumlah;
  final String status;
  final String method;
  final String studentName;
  final String studentNim;
  final String studentProdi;
  final Map<String, double> paymentBreakdown;

  const TransactionDetail({
    required this.id,
    required this.txId,
    required this.tanggal,
    required this.tanggalFull,
    required this.type,
    required this.jumlah,
    required this.status,
    required this.method,
    required this.studentName,
    required this.studentNim,
    required this.studentProdi,
    required this.paymentBreakdown,
  });

  @override
  List<Object¿> get props => [
    id,
    txId,
    tanggal,
    tanggalFull,
    type,
    jumlah,
    status,
    method,
    studentName,
    studentNim,
    studentProdi,
    paymentBreakdown,
  ];
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\payment\domain\repositories\payment_repository.dart

```
import 'package:dartz/dartz.dart';
import '../../../../core/error/failures.dart';
import '../entities/payment.dart';

abstract class PaymentRepository {
  Future<Either<Failure, List<PaymentHistoryItem>>> getPaymentHistory({
    int page = 1,
    int limit = 20,
    String¿ startDate,
    String¿ endDate,
    String¿ type,
    String sortBy = 'tanggal',
    String sortOrder = 'desc',
  });

  Future<Either<Failure, PaymentSummary>> getPaymentSummary();

  Future<Either<Failure, TransactionDetail>> getTransactionDetail(
    String transactionId,
  );

  Future<Either<Failure, bool>> refreshPaymentData();
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\payment\domain\usecases\get_payment_history_usecase.dart

```
import 'package:dartz/dartz.dart';
import 'package:equatable/equatable.dart';
import '../../../../core/error/failures.dart';
import '../../../../core/usecases/usecase.dart';
import '../entities/payment.dart';
import '../repositories/payment_repository.dart';

class GetPaymentHistoryUseCase
    implements UseCase<List<PaymentHistoryItem>, PaymentHistoryParams> {
  final PaymentRepository repository;

  GetPaymentHistoryUseCase(this.repository);

  @override
  Future<Either<Failure, List<PaymentHistoryItem>>> call(
    PaymentHistoryParams params,
  ) async {
    return await repository.getPaymentHistory(
      page: params.page,
      limit: params.limit,
      startDate: params.startDate,
      endDate: params.endDate,
      type: params.type,
      sortBy: params.sortBy,
      sortOrder: params.sortOrder,
    );
  }
}

class PaymentHistoryParams extends Equatable {
  final int page;
  final int limit;
  final String¿ startDate;
  final String¿ endDate;
  final String¿ type;
  final String sortBy;
  final String sortOrder;

  const PaymentHistoryParams({
    this.page = 1,
    this.limit = 20,
    this.startDate,
    this.endDate,
    this.type,
    this.sortBy = 'tanggal',
    this.sortOrder = 'desc',
  });

  @override
  List<Object¿> get props => [
    page,
    limit,
    startDate,
    endDate,
    type,
    sortBy,
    sortOrder,
  ];
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\payment\domain\usecases\get_payment_summary_usecase.dart

```
import 'package:dartz/dartz.dart';
import '../../../../core/error/failures.dart';
import '../../../../core/usecases/usecase.dart';
import '../entities/payment.dart';
import '../repositories/payment_repository.dart';

class GetPaymentSummaryUseCase implements UseCase<PaymentSummary, NoParams> {
  final PaymentRepository repository;

  GetPaymentSummaryUseCase(this.repository);

  @override
  Future<Either<Failure, PaymentSummary>> call(NoParams params) async {
    return await repository.getPaymentSummary();
  }
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\payment\domain\usecases\ge

t_transaction_detail_usecase.dart

```
import 'package:dartz/dartz.dart';
import 'package:equatable/equatable.dart';
import '../../../../core/error/failures.dart';
import '../../../../core/usecases/usecase.dart';
import '../entities/payment.dart';
import '../repositories/payment_repository.dart';

class GetTransactionDetailUseCase
    implements UseCase<TransactionDetail, TransactionDetailParams> {
  final PaymentRepository repository;

  GetTransactionDetailUseCase(this.repository);

  @override
  Future<Either<Failure, TransactionDetail>> call(
    TransactionDetailParams params,
  ) async {
    return await repository.getTransactionDetail(params.transactionId);
  }
}

class TransactionDetailParams extends Equatable {
  final String transactionId;

  const TransactionDetailParams({required this.transactionId});

  @override
  List<Object> get props => [transactionId];
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\payment\presentation\bloc\

payment_bloc.dart

```
import 'package:flutter_bloc/flutter_bloc.dart';
import '../../../../core/error/failures.dart';
import '../../../../core/usecases/usecase.dart';
import '../../domain/entities/payment.dart';
import '../../domain/usecases/get_payment_history_usecase.dart';
import '../../domain/usecases/get_payment_summary_usecase.dart';
import '../../domain/usecases/get_transaction_detail_usecase.dart';
import 'payment_event.dart';
import 'payment_state.dart';

class PaymentBloc extends Bloc<PaymentEvent, PaymentState> {
  final GetPaymentHistoryUseCase getPaymentHistoryUseCase;
  final GetPaymentSummaryUseCase getPaymentSummaryUseCase;
  final GetTransactionDetailUseCase getTransactionDetailUseCase;

  PaymentBloc({
    required this.getPaymentHistoryUseCase,
    required this.getPaymentSummaryUseCase,
    required this.getTransactionDetailUseCase,
  }) : super(const PaymentInitial()) {
    on<LoadPaymentHistoryEvent>(_onLoadPaymentHistory);
    on<LoadPaymentSummaryEvent>(_onLoadPaymentSummary);
    on<LoadTransactionDetailEvent>(_onLoadTransactionDetail);
    on<RefreshPaymentDataEvent>(_onRefreshPaymentData);
  }
```

```
Future<void> _onLoadPaymentHistory(
  LoadPaymentHistoryEvent event,
  Emitter<PaymentState> emit,
) async {
  // Keep current history for a better refresh experience
  List<PaymentHistoryItem> currentHistory = [];
  if (state is PaymentHistoryLoaded) {
    currentHistory = (state as PaymentHistoryLoaded).historyItems;
  }

  // Show loading indicator only on initial load
  if (event.page == 1) {
    emit(const PaymentLoading());
  }

  final result = await getPaymentHistoryUseCase(
    PaymentHistoryParams(page: event.page, limit: event.limit),
  );

  await result.fold(
    (failure) async =>
        emit(PaymentError(message: _mapFailureToMessage(failure))),
    (newHistoryItems) async {
      final summaryResult = await getPaymentSummaryUseCase(NoParams());
      await summaryResult.fold(
        (summaryFailure) async => emit(
          PaymentHistoryLoaded(
            historyItems: event.page == 1
                ¿ newHistoryItems
                : (currentHistory + newHistoryItems),
          ),
        ),
        (summary) async => emit(
          PaymentHistoryLoaded(
            historyItems: event.page == 1
                ¿ newHistoryItems
                : (currentHistory + newHistoryItems),
            summary: summary,
          ),
        ),
      );
    },
  );
}

Future<void> _loadSummaryWithHistory(
  List<PaymentHistoryItem> historyItems,
  Emitter<PaymentState> emit,
) async {
  final summaryResult = await getPaymentSummaryUseCase(NoParams());

  await summaryResult.fold(
    (failure) async => emit(PaymentHistoryLoaded(historyItems: historyItems)),
    (summary) async => emit(
      PaymentHistoryLoaded(historyItems: historyItems, summary: summary),
    ),
  );
}

Future<void> _onLoadPaymentSummary(
  LoadPaymentSummaryEvent event,
  Emitter<PaymentState> emit,
) async {
  emit(const PaymentLoading());
```

```
    final result = await getPaymentSummaryUseCase(NoParams());

    await result.fold(
      (failure) async =>
          emit(PaymentError(message: _mapFailureToMessage(failure))),
      (summary) async => emit(PaymentSummaryLoaded(summary: summary)),
    );
  }

  Future<void> _onLoadTransactionDetail(
    LoadTransactionDetailEvent event,
    Emitter<PaymentState> emit,
  ) async {
    emit(const PaymentLoading());

    final result = await getTransactionDetailUseCase(
      TransactionDetailParams(transactionId: event.transactionId),
    );

    await result.fold(
      (failure) async =>
          emit(PaymentError(message: _mapFailureToMessage(failure))),
      (transactionDetail) async =>
          emit(TransactionDetailLoaded(transactionDetail: transactionDetail)),
    );
  }

  Future<void> _onRefreshPaymentData(
    RefreshPaymentDataEvent event,
    Emitter<PaymentState> emit,
  ) async {
    final result = await getPaymentHistoryUseCase(
      const PaymentHistoryParams(page: 1, limit: 20),
    );
    add(const LoadPaymentHistoryEvent());
  }

  String _mapFailureToMessage(Failure failure) {
    switch (failure.runtimeType) {
      case ServerFailure:
        return (failure as ServerFailure).message;
      case NetworkFailure:
        return (failure as NetworkFailure).message;
      case CacheFailure:
        return (failure as CacheFailure).message;
      default:
        return 'Unexpected error occurred';
    }
  }
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\payment\presentation\bloc\

payment_event.dart

```
import 'package:equatable/equatable.dart';

abstract class PaymentEvent extends Equatable {
  const PaymentEvent();

  @override
  List<Object¿> get props => [];
}

class LoadPaymentHistoryEvent extends PaymentEvent {
  final int page;
  final int limit;
  final String¿ startDate;
  final String¿ endDate;
  final String¿ type;
  final String sortBy;
  final String sortOrder;

  const LoadPaymentHistoryEvent({
    this.page = 1,
    this.limit = 20,
    this.startDate,
    this.endDate,
    this.type,
    this.sortBy = 'tanggal',
    this.sortOrder = 'desc',
  });

  @override
  List<Object¿> get props => [
    page,
    limit,
    startDate,
    endDate,
    type,
    sortBy,
    sortOrder,
  ];
}

class LoadPaymentSummaryEvent extends PaymentEvent {
  const LoadPaymentSummaryEvent();
}

class LoadTransactionDetailEvent extends PaymentEvent {
  final String transactionId;

  const LoadTransactionDetailEvent({required this.transactionId});

  @override
  List<Object> get props => [transactionId];
}

class RefreshPaymentDataEvent extends PaymentEvent {
  const RefreshPaymentDataEvent();
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\payment\presentation\bloc\

payment_state.dart

```
import 'package:equatable/equatable.dart';
import '../../domain/entities/payment.dart';

abstract class PaymentState extends Equatable {
  const PaymentState();
  @override
  List<Object¿> get props => [];
}

class PaymentInitial extends PaymentState {
  const PaymentInitial();
}

class PaymentLoading extends PaymentState {
  const PaymentLoading();
}

class PaymentHistoryLoaded extends PaymentState {
  final List<PaymentHistoryItem> historyItems;
  final PaymentSummary¿ summary;
  final TransactionDetail¿ transactionDetail;

  const PaymentHistoryLoaded({
    required this.historyItems,
    this.summary,
    this.transactionDetail,
  });

  @override
  List<Object¿> get props => [historyItems, summary, transactionDetail];
}

class PaymentSummaryLoaded extends PaymentState {
  final PaymentSummary summary;

  const PaymentSummaryLoaded({required this.summary});

  @override
  List<Object> get props => [summary];
}

class TransactionDetailLoaded extends PaymentState {
  final TransactionDetail transactionDetail;

  const TransactionDetailLoaded({required this.transactionDetail});

  @override
  List<Object> get props => [transactionDetail];
}

class PaymentRefreshed extends PaymentState {
  final bool refreshed;

  const PaymentRefreshed({required this.refreshed});

  @override
  List<Object> get props => [refreshed];
}

class PaymentError extends PaymentState {
  final String message;

  const PaymentError({required this.message});

  @override
  List<Object> get props => [message];
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\payment\presentation\components\components.dart

```
export 'payment_summary_card.dart';
export 'transaction_card.dart';
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\payment\presentation\components\payment_summary_card.dart

```
import 'package:flutter/material.dart';
import 'package:intl/intl.dart';

class PaymentSummaryCard extends StatelessWidget {
  final String title;
  final double amount;

  const PaymentSummaryCard({
    super.key,
    required this.title,
    required this.amount,
  });

  static final NumberFormat _currencyFormat = NumberFormat.currency(
    locale: 'id_ID',
    symbol: 'Rp ',
    decimalDigits: 0,
  );

  @override
  Widget build(BuildContext context) {
    return Container(
      clipBehavior: Clip.antiAlias,
      decoration: const ShapeDecoration(
        color: Color(0xFFFAFAFA),
        shape: RoundedRectangleBorder(
          side: BorderSide(width: 1, color: Color(0xFFE7E7E7)),
          borderRadius: BorderRadius.all(Radius.circular(8)),
        ),
      ),
      child: Column(
        mainAxisSize: MainAxisSize.min,
        mainAxisAlignment: MainAxisAlignment.center,
        crossAxisAlignment: CrossAxisAlignment.start,
        children: [
          Padding(
            padding: const EdgeInsets.all(8),
            child: Text(
              title,
              style: const TextStyle(
                color: Color(0xFF1C1D1F),
                fontSize: 12,
                fontWeight: FontWeight.w500,
                letterSpacing: -0.12,
              ),
            ),
          ),
          const Divider(height: 1, thickness: 1, color: Color(0xFFE7E7E7)),
          Padding(
            padding: const EdgeInsets.all(8),
            child: Text(
              _currencyFormat.format(amount),
              style: const TextStyle(
                color: Color(0xFF121111),
                fontSize: 20,
                fontWeight: FontWeight.w600,
                letterSpacing: -0.20,
              ),
            ),
          ),
        ],
      ),
    );
  }
}
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\payment\presentation\components\transaction_card.dart

```
import 'package:flutter/material.dart';
import 'package:intl/intl.dart';
import '../../domain/entities/payment.dart';

class TransactionCard extends StatelessWidget {
  final PaymentHistoryItem item;
  final VoidCallback onTap;

  const TransactionCard({super.key, required this.item, required this.onTap});

  static final NumberFormat _currencyFormat = NumberFormat.currency(
    locale: 'id_ID',
    symbol: 'Rp ',
    decimalDigits: 0,
  );

  @override
  Widget build(BuildContext context) {
    return Material(
      color: const Color(0xFFFAFAFA),
      borderRadius: BorderRadius.circular(8),
      elevation: 2,
      shadowColor: const Color(0x0C000000),
      child: InkWell(
        onTap: onTap,
        borderRadius: BorderRadius.circular(8),
        child: Padding(
          padding: const EdgeInsets.symmetric(horizontal: 12, vertical: 8),
          child: Row(
            children: [
              Expanded(
                child: Column(
                  crossAxisAlignment: CrossAxisAlignment.start,
                  children: [
                    Text(
                      item.type,
                      style: const TextStyle(
                        color: Color(0xFF121315),
                        fontSize: 16,
                        fontWeight: FontWeight.w500,
                        letterSpacing: -0.16,
                      ),
                    ),
                    const SizedBox(height: 8),
                    Row(
                      children: [
                        // Transaction ID Badge
                        Container(
                          padding: const EdgeInsets.symmetric(
                            horizontal: 8,
                            vertical: 2,
                          ),
                          decoration: BoxDecoration(
                            color: const Color(0xFFA5DCFF),
                            borderRadius: BorderRadius.circular(4),
                          ),
                          child: Text(
                            item.txId,
                            textAlign: TextAlign.center,
                            style: const TextStyle(
                              color: Color(0xFF323335),
                              fontSize: 9,
                              fontWeight: FontWeight.w700,
                              height: 1.78,
                              letterSpacing: 1,
                            ),
                          ),
                        ),
                        const SizedBox(width: 8),
                        // Amount
                        Text(
                          _currencyFormat.format(item.jumlah),
                          style: const TextStyle(
                            color: Color(0xFF858586),
                            fontSize: 12,
                            fontWeight: FontWeight.w500,
                            letterSpacing: -0.12,
                          ),
                        ),
                      ],
                    ),
                  ],
                ),
              ),
              // Date
              Column(
                crossAxisAlignment: CrossAxisAlignment.end,
                children: [
                  Text(
                    _formatDate(item.tanggal),
                    textAlign: TextAlign.right,
                    style: const TextStyle(
                      color: Color(0xFF858586),
                      fontSize: 9,
                      fontWeight: FontWeight.w500,
                      letterSpacing: -0.09,
                    ),
                  ),
                  Text(
                    _formatYear(item.tanggal),
                    textAlign: TextAlign.right,
                    style: const TextStyle(
                      color: Color(0xFF858586),
                      fontSize: 9,
                      fontWeight: FontWeight.w500,
                      letterSpacing: -0.09,
                    ),
                  ),
                ],
              ),
            ],
          ),
        ),
      ),
    );
  }

  String _formatDate(String dateString) {
    try {
      final parts = dateString.split(' ');
      if (parts.length >= 2) {
        return '${parts[0]} ${parts[1]}';
      }
      return dateString;
    } catch (e) {
      return dateString;
    }
  }

  String _formatYear(String dateString) {
    try {
      final parts = dateString.split(' ');
      if (parts.length >= 3) {
        return parts[2];
      }
      return DateTime.now().year.toString();
    } catch (e) {
      return DateTime.now().year.toString();
    }
  }
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\payment\presentation\pages\transaction_detail_page.dart

```
import 'package:flutter/material.dart';
import 'package:flutter/services.dart';
import 'package:flutter_bloc/flutter_bloc.dart';
import 'package:intl/intl.dart';
import 'package:wismon_keuangan/features/payment/domain/entities/payment.dart';
import 'package:wismon_keuangan/features/payment/presentation/bloc/payment_bloc.dart';
import 'package:wismon_keuangan/features/payment/presentation/bloc/payment_event.dart';
import 'package:wismon_keuangan/features/payment/presentation/bloc/payment_state.dart';

class TransactionDetailPage extends StatefulWidget {
  final String transactionId;

  const TransactionDetailPage({super.key, required this.transactionId});

  @override
  State<TransactionDetailPage> createState() => _TransactionDetailPageState();
}

class _TransactionDetailPageState extends State<TransactionDetailPage> {
```

```
@override
void initState() {
  super.initState();
  context.read<PaymentBloc>().add(
    LoadTransactionDetailEvent(transactionId: widget.transactionId),
  );
}

@override
Widget build(BuildContext context) {
  return Scaffold(
    backgroundColor: Colors.grey[50],
    appBar: AppBar(
      title: const Text(
        'Detail Transaksi',
        style: TextStyle(fontWeight: FontWeight.bold),
      ),
      backgroundColor: Colors.white,
      foregroundColor: Colors.grey[800],
      elevation: 1,
    ),
    body: BlocConsumer<PaymentBloc, PaymentState>(
      listener: (context, state) {
        if (state is PaymentError) {
          ScaffoldMessenger.of(context).showSnackBar(
            SnackBar(
              content: Text(state.message),
              backgroundColor: Colors.red,
            ),
          );
        }
      },
      builder: (context, state) {
        if (state is PaymentLoading) {
          return const Center(child: CircularProgressIndicator());
        }

        if (state is TransactionDetailLoaded) {
          return _buildTransactionDetail(context, state.transactionDetail);
        }
        // Use the current state if it contains the detail, e.g. after a refresh
        if (state is PaymentHistoryLoaded &&
            state.transactionDetail != null) {
          return _buildTransactionDetail(context, state.transactionDetail!);
        }

        return _buildErrorState(context);
      },
    ),
  );
}

Widget _buildTransactionDetail(
  BuildContext context,
  TransactionDetail transactionDetail,
) {
  final currencyFormat = NumberFormat.currency(
    locale: 'id_ID',
    symbol: 'Rp ',
    decimalDigits: 0,
  );
  Color statusColor = Colors.green;
  if (transactionDetail.status.toUpperCase() == 'LUNAS') {
    statusColor = Colors.green;
  } else {
    statusColor = Colors.orange;
  }

  return SingleChildScrollView(
    padding: const EdgeInsets.all(16),
    child: Column(
      children: [
        // Status & Amount Summary
        Card(
          elevation: 2,
          shape: RoundedRectangleBorder(
            borderRadius: BorderRadius.circular(12),
          ),
          child: Padding(
            padding: const EdgeInsets.all(24),
            child: Column(
              children: [
                Text(
                  'Pembayaran Berhasil',
                  style: TextStyle(fontSize: 16, color: Colors.grey[600]),
                ),
                const SizedBox(height: 8),
                Text(
                  currencyFormat.format(transactionDetail.jumlah),
                  style: TextStyle(
                    fontSize: 36,
                    fontWeight: FontWeight.bold,
                    color: statusColor,
                  ),
                ),
                const SizedBox(height: 8),
                Container(
                  padding: const EdgeInsets.symmetric(
                    horizontal: 16,
                    vertical: 6,
                  ),
                  decoration: BoxDecoration(
                    color: statusColor.withOpacity(0.1),
                    borderRadius: BorderRadius.circular(20),
                  ),
                  child: Text(
                    transactionDetail.status,
                    style: TextStyle(
                      color: statusColor,
                      fontWeight: FontWeight.bold,
                      fontSize: 14,
                    ),
                  ),
                ),
              ],
            ),
          ),
        ),
        const SizedBox(height: 16),

        // Rincian Pembayaran
        _buildDetailCard(
          context,
          title: 'Rincian Pembayaran',
          children: [
            _buildDetailRow('Jenis Pembayaran', transactionDetail.type),
            _buildDetailRow(
              'Tanggal Transaksi',
              transactionDetail.tanggalFull,
            ),
            _buildDetailRow('Metode Pembayaran', transactionDetail.method),
            _buildCopyableDetailRow(
              context,
              'Nomor Transaksi',
              transactionDetail.txId,
            ),
          ],
        ),
        const SizedBox(height: 16),

        // Rincian Tagihan
        if (transactionDetail.paymentBreakdown.isNotEmpty) ...[
          _buildDetailCard(
            context,
            title: 'Rincian Tagihan',
            children: [
              ...transactionDetail.paymentBreakdown.entries.map(
                (entry) => _buildDetailRow(
                  entry.key,
                  currencyFormat.format(entry.value),
                ),
              ),
              const Divider(height: 24),
              _buildDetailRow(
                'Total Dibayar',
                currencyFormat.format(transactionDetail.jumlah),
                isTotal: true,
              ),
            ],
          ),
          const SizedBox(height: 16),
        ],

        // Informasi Mahasiswa
        _buildDetailCard(
          context,
          title: 'Informasi Mahasiswa',
          children: [
            _buildDetailRow('Nama Mahasiswa', transactionDetail.studentName),
            _buildDetailRow('NIM', transactionDetail.studentNim),
            _buildDetailRow('Program Studi', transactionDetail.studentProdi),
          ],
        ),
        const SizedBox(height: 24),
      ],
    ),
  );
}

Widget _buildDetailCard(
  BuildContext context, {
  required String title,
  required List<Widget> children,
}) {
  return Card(
    elevation: 2,
    shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(12)),
    child: Padding(
      padding: const EdgeInsets.all(20),
      child: Column(
        crossAxisAlignment: CrossAxisAlignment.start,
        children: [
          Text(
            title,
            style: Theme.of(context).textTheme.titleLarge¿.copyWith(
              fontWeight: FontWeight.bold,
              color: Colors.grey[800],
            ),
          ),
          const SizedBox(height: 16),
          ...children,
        ],
      ),
    ),
  );
}

Widget _buildDetailRow(String label, String value, {bool isTotal = false}) {
  return Padding(
    padding: const EdgeInsets.symmetric(vertical: 6),
    child: Row(
      crossAxisAlignment: CrossAxisAlignment.start,
      children: [
        Expanded(
          flex: 2,
          child: Text(
            label,
            style: TextStyle(
              color: Colors.grey[600],
              fontSize: 14,
              fontWeight: isTotal ¿ FontWeight.bold : FontWeight.normal,
            ),
          ),
        ),
        Expanded(
          flex: 3,
          child: Text(
            value,
            style: TextStyle(
              fontWeight: isTotal ¿ FontWeight.bold : FontWeight.w600,
              fontSize: isTotal ¿ 16 : 14,
              color: isTotal ¿ Colors.black : Colors.grey[700],
            ),
            textAlign: TextAlign.end,
          ),
        ),
      ],
    ),
  );
}

Widget _buildCopyableDetailRow(
  BuildContext context,
  String label,
  String value,
) {
  return Padding(
    padding: const EdgeInsets.symmetric(vertical: 6),
    child: Row(
      crossAxisAlignment: CrossAxisAlignment.center,
      children: [
        Expanded(
          flex: 2,
          child: Text(
            label,
            style: TextStyle(color: Colors.grey[600], fontSize: 14),
          ),
        ),
        Expanded(
          flex: 3,
          child: Row(
            mainAxisAlignment: MainAxisAlignment.end,
            children: [
              Flexible(
                child: Text(
                  value,
                  style: TextStyle(
                    fontWeight: FontWeight.w600,
                    fontSize: 14,
                    color: Colors.blue[600],
                  ),
                  textAlign: TextAlign.end,
                ),
              ),
              const SizedBox(width: 8),
              InkWell(
                onTap: () => _copyToClipboard(context, value),
                borderRadius: BorderRadius.circular(4),
                child: Padding(
                  padding: const EdgeInsets.all(4),
                  child: Icon(Icons.copy, size: 16, color: Colors.blue[600]),
                ),
              ),
            ],
          ),
        ),
      ],
    ),
  );
}

Widget _buildErrorState(BuildContext context) {
  return Center(
    child: Column(
      mainAxisAlignment: MainAxisAlignment.center,
      children: [
        Icon(Icons.error_outline, size: 64, color: Colors.grey[400]),
        const SizedBox(height: 16),
        Text(
          'Transaction not found',
          style: TextStyle(
            fontSize: 18,
            fontWeight: FontWeight.bold,
            color: Colors.grey[600],
          ),
        ),
        const SizedBox(height: 8),
        Text(
          'The transaction details could not be loaded',
          style: TextStyle(color: Colors.grey[500], fontSize: 14),
        ),
        const SizedBox(height: 24),
        ElevatedButton(
          onPressed: () {
            context.read<PaymentBloc>().add(
              LoadTransactionDetailEvent(transactionId: widget.transactionId),
            );
          },
          child: const Text('Try Again'),
        ),
      ],
    ),
  );
}

void _copyToClipboard(BuildContext context, String text) {
  Clipboard.setData(ClipboardData(text: text));
  _showToast(context, 'Nomor transaksi disalin!');
}
```

```
  void _showToast(BuildContext context, String message) {
    ScaffoldMessenger.of(context).showSnackBar(
      SnackBar(
        content: Text(message),
        backgroundColor: Colors.grey[800],
        duration: const Duration(seconds: 2),
        behavior: SnackBarBehavior.floating,
        shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(8)),
      ),
    );
  }
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\payment\presentation\pages\wismon_page.dart

```
import 'package:flutter/material.dart';
import 'package:flutter_bloc/flutter_bloc.dart';
import 'package:wismon_keuangan/core/di/injection_container.dart' as di;
import 'package:wismon_keuangan/features/payment/domain/entities/payment.dart';
import 'package:wismon_keuangan/features/payment/presentation/bloc/payment_bloc.dart';
import 'package:wismon_keuangan/features/payment/presentation/bloc/payment_event.dart';
import 'package:wismon_keuangan/features/payment/presentation/bloc/payment_state.dart';
import 'package:wismon_keuangan/features/payment/presentation/pages/transaction_detail_page.dart';
import 'package:wismon_keuangan/features/payment/presentation/components/components.dart';

class WismonPage extends StatefulWidget {
  const WismonPage({super.key});

  @override
  State<WismonPage> createState() => _WismonPageState();
}

class _WismonPageState extends State<WismonPage> with RouteAware {
  List<PaymentHistoryItem>¿ _historyItems;
  List<PaymentHistoryItem> _filteredHistoryItems = [];
  PaymentSummary¿ _paymentSummary;

  // TODO: Fetch from API - These will be replaced by actual API data
  final List<Map<String, String>> _paymentTypes = [
    {'kode': 'all', 'nama': 'Semua Jenis Pembayaran'},
    {'kode': 'SPP', 'nama': 'SPP'},
    {'kode': 'SWP', 'nama': 'SWP'},
    {
      'kode': 'Pendaftaran Mahasiswa Baru',
      'nama': 'Pendaftaran Mahasiswa Baru',
    },
    {'kode': 'Praktek Rumah Sakit', 'nama': 'Praktek Rumah Sakit'},
    {'kode': 'Seragam', 'nama': 'Seragam'},
    {'kode': 'Wisuda', 'nama': 'Wisuda'},
    {'kode': 'KTI dan Wisuda', 'nama': 'KTI dan Wisuda'},
  ];
  String _selectedPaymentTypeCode = 'all';

  @override
  void initState() {
    super.initState();
    _loadPaymentData();
  }
```

```
@override
void didChangeDependencies() {
  super.didChangeDependencies();
  final ModalRoute¿ currentRoute = ModalRoute.of(context);
  if (currentRoute is PageRoute) {
    di.sl<RouteObserver<PageRoute>>().subscribe(this, currentRoute);
  }
}

@override
void dispose() {
  di.sl<RouteObserver<PageRoute>>().unsubscribe(this);
  super.dispose();
}

@override
void didPopNext() {
  context.read<PaymentBloc>().add(const RefreshPaymentDataEvent());
  context.read<PaymentBloc>().add(const LoadPaymentSummaryEvent());
}

void _loadPaymentData() {
  // Clear existing data
  if (mounted) {
    setState(() {
      _historyItems = null;
      _paymentSummary = null;
    });
  }
  context.read<PaymentBloc>().add(const LoadPaymentHistoryEvent());
  context.read<PaymentBloc>().add(const LoadPaymentSummaryEvent());
}

void _filterHistoryItems() {
  if (_historyItems == null) {
    _filteredHistoryItems = [];
    return;
  }
  if (_selectedPaymentTypeCode == 'all') {
    _filteredHistoryItems = _historyItems!;
  } else {
    _filteredHistoryItems = _historyItems!
        .where((item) => item.type == _selectedPaymentTypeCode)
        .toList();
  }
}

@override
Widget build(BuildContext context) {
  return Scaffold(
    backgroundColor: const Color(0xFFFAFAFA),
    body: Column(
      children: [
        _buildHeader(context),
        Expanded(
          child: BlocListener<PaymentBloc, PaymentState>(
            listener: (context, state) {
              if (state is PaymentError) {
                if (mounted) {
                  ScaffoldMessenger.of(context).showSnackBar(
                    SnackBar(
                      content: Text(state.message),
                      backgroundColor: Colors.red,
                    ),
                  );
                }
              } else if (state is PaymentHistoryLoaded) {
                if (mounted) {
                  setState(() {
                    _historyItems = state.historyItems;
                    _filterHistoryItems();
                  });
                }
              } else if (state is PaymentSummaryLoaded) {
                if (mounted) {
                  setState(() {
                    _paymentSummary = state.summary;
                  });
                }
              }
            },
            child: BlocBuilder<PaymentBloc, PaymentState>(
              buildWhen: (previous, current) {
                // Only rebuild on specific state changes
                return current is PaymentLoading ||
                    current is PaymentError ||
                    (_historyItems != null && _paymentSummary != null);
              },
              builder: (context, state) {
                if (state is PaymentLoading && _historyItems == null) {
                  return const Center(child: CircularProgressIndicator());
                }

                if (_historyItems != null) {
                  return _buildPaymentContent(context);
                }

                return _buildEmptyState(context);
              },
            ),
          ),
        ),
      ],
    ),
  );
}

Widget _buildHeader(BuildContext context) {
  return Container(
    width: double.infinity,
    decoration: const BoxDecoration(
      color: Color(0xFF135EA2),
      borderRadius: BorderRadius.only(
        bottomLeft: Radius.circular(16),
        bottomRight: Radius.circular(16),
      ),
    ),
    child: Column(
      children: [
        // Status bar height
        SizedBox(height: MediaQuery.of(context).padding.top),
        // Header content
        Padding(
          padding: const EdgeInsets.only(
            top: 12,
            left: 16,
            right: 16,
            bottom: 16,
          ),
          child: Row(
            mainAxisAlignment: MainAxisAlignment.spaceBetween,
            children: [
              // Back button
              Container(
                width: 40,
                height: 40,
                decoration: BoxDecoration(
                  color: const Color(0xFFFAFAFA),
                  borderRadius: BorderRadius.circular(8),
                  boxShadow: const [
                    BoxShadow(
                      color: Color(0x0C000000),
                      blurRadius: 10,
                      offset: Offset(0, 4),
                      spreadRadius: 0,
                    ),
                  ],
                ),
                child: IconButton(
                  onPressed: () => Navigator.of(context).pop(),
                  icon: const Icon(
                    Icons.arrow_back,
                    color: Color(0xFF135EA2),
                    size: 20,
                  ),
                  padding: EdgeInsets.zero,
                ),
              ),
              // Title
              const Text(
                'Biaya Kuliah',
                textAlign: TextAlign.center,
                style: TextStyle(
                  color: Color(0xFFFAFAFA),
                  fontSize: 18,
                  fontWeight: FontWeight.w800,
                  letterSpacing: -0.18,
                ),
              ),
              // Right placeholder (to balance the layout)
              const SizedBox(width: 40, height: 40),
            ],
          ),
        ),
      ],
    ),
  );
}

Widget _buildPaymentContent(BuildContext context) {
  return RefreshIndicator(
    onRefresh: () async {
      context.read<PaymentBloc>().add(const RefreshPaymentDataEvent());
      context.read<PaymentBloc>().add(const LoadPaymentSummaryEvent());
    },
    child: CustomScrollView(
      physics: const AlwaysScrollableScrollPhysics(),
      slivers: [
        SliverPadding(
          padding: const EdgeInsets.fromLTRB(16, 20, 16, 0),
          sliver: SliverToBoxAdapter(
            child: Column(
              crossAxisAlignment: CrossAxisAlignment.start,
              children: [
                _buildFilterSection(context),
                const SizedBox(height: 20),
                if (_paymentSummary != null) ...[
                  _buildPaymentSummary(context, _paymentSummary!),
                  const SizedBox(height: 20),
                ],
                Container(height: 1, color: const Color(0xFFE7E7E7)),
                const SizedBox(height: 16),
              ],
            ),
          ),
        ),
        if (_filteredHistoryItems.isNotEmpty)
          SliverPadding(
            padding: const EdgeInsets.symmetric(horizontal: 16),
            sliver: SliverList(
              delegate: SliverChildBuilderDelegate((context, index) {
                if (index >= _filteredHistoryItems.length) return null;
                return Padding(
                  padding: EdgeInsets.only(
                    bottom: index < _filteredHistoryItems.length - 1 ¿ 8 : 20,
                  ),
                  child: _buildTransactionCard(
                    context,
                    _filteredHistoryItems[index],
                  ),
                );
              }, childCount: _filteredHistoryItems.length),
            ),
          )
        else
          SliverPadding(
            padding: const EdgeInsets.symmetric(horizontal: 16),
            sliver: SliverToBoxAdapter(child: _buildEmptyHistory(context)),
          ),
      ],
    ),
  );
}

Widget _buildFilterSection(BuildContext context) {
  return Column(
    crossAxisAlignment: CrossAxisAlignment.start,
    children: [
      const Text(
        'Jenis Pembayaran',
        style: TextStyle(
          color: Color(0xFF1C1D1F),
          fontSize: 14,
          fontWeight: FontWeight.w500,
          letterSpacing: -0.14,
        ),
      ),
      const SizedBox(height: 8),
      DropdownButtonFormField<String>(
        value: _selectedPaymentTypeCode,
        onChanged: (String¿ newValue) {
          if (newValue != null && mounted) {
            setState(() {
              _selectedPaymentTypeCode = newValue;
              _filterHistoryItems();
            });
          }
        },
        items: _paymentTypes.map<DropdownMenuItem<String>>((
          Map<String, String> type,
        ) {
          return DropdownMenuItem<String>(
            value: type['kode'],
            child: Text(
              type['nama']!,
              style: const TextStyle(
                color: Color(0xFF545556),
                fontSize: 14,
                fontWeight: FontWeight.w500,
                letterSpacing: -0.14,
              ),
            ),
          );
        }).toList(),
        decoration: const InputDecoration(
          filled: true,
          fillColor: Color(0xFFFAFAFA),
          contentPadding: EdgeInsets.symmetric(horizontal: 12, vertical: 8),
          border: OutlineInputBorder(
            borderRadius: BorderRadius.all(Radius.circular(8)),
            borderSide: BorderSide(color: Color(0xFFE7E7E7), width: 1),
          ),
          enabledBorder: OutlineInputBorder(
            borderRadius: BorderRadius.all(Radius.circular(8)),
            borderSide: BorderSide(color: Color(0xFFE7E7E7), width: 1),
          ),
        ),
      ),
    ],
  );
}

Widget _buildPaymentSummary(BuildContext context, PaymentSummary summary) {
  // Map database payment types to user-friendly names
  const paymentTypeMapping = {
    'SPP': 'SPP',
    'SWP': 'SWP',
    'Pendaftaran Mahasiswa Baru': 'Pendaftaran Mahasiswa Baru',
    'Praktek Rumah Sakit': 'Praktek Rumah Sakit',
    'Seragam': 'Seragam',
    'Wisuda': 'Wisuda',
    'KTI dan Wisuda': 'KTI dan Wisuda',
  };

  // Filter summary to only show what's in the mapping and has a value > 0
  final filteredBreakdown = summary.breakdown.entries
      .where(
        (entry) =>
            paymentTypeMapping.containsKey(entry.key) && entry.value > 0,
      )
      .toList();

  if (filteredBreakdown.isEmpty) {
    return const SizedBox.shrink();
  }

  final List<Widget> cardRows = [];
  for (int i = 0; i < filteredBreakdown.length; i += 2) {
    final item1 = filteredBreakdown[i];
    final card1 = Expanded(
      child: PaymentSummaryCard(
        title: paymentTypeMapping[item1.key]!,
        amount: item1.value,
      ),
    );

    final rowChildren = <Widget>[card1];

    if (i + 1 < filteredBreakdown.length) {
      final item2 = filteredBreakdown[i + 1];
      final card2 = Expanded(
        child: PaymentSummaryCard(
          title: paymentTypeMapping[item2.key]!,
          amount: item2.value,
        ),
      );
      rowChildren.addAll([const SizedBox(width: 12), card2]);
    }

    cardRows.add(Row(children: rowChildren));
    if (i + 2 < filteredBreakdown.length) {
      cardRows.add(const SizedBox(height: 12));
    }
  }

  return Column(mainAxisSize: MainAxisSize.min, children: cardRows);
}

Widget _buildTransactionCard(BuildContext context, PaymentHistoryItem item) {
  return TransactionCard(
    item: item,
    onTap: () {
      Navigator.of(context).push(
        MaterialPageRoute(
          builder: (_) => TransactionDetailPage(transactionId: item.id),
        ),
      );
    },
  );
}

Widget _buildEmptyHistory(BuildContext context) {
  return Container(
    width: double.infinity,
    padding: const EdgeInsets.all(40),
    decoration: BoxDecoration(
      color: const Color(0xFFFAFAFA),
      borderRadius: BorderRadius.circular(8),
      boxShadow: const [
        BoxShadow(
          color: Color(0x0C000000),
          blurRadius: 10,
          offset: Offset(0, 4),
          spreadRadius: 0,
        ),
      ],
    ),
    child: Column(
      children: [
        Icon(Icons.receipt_long_outlined, size: 64, color: Colors.grey[400]),
        const SizedBox(height: 16),
        Text(
          'No Payment History',
          style: TextStyle(
            fontSize: 18,
            fontWeight: FontWeight.bold,
            color: Colors.grey[600],
          ),
        ),
        const SizedBox(height: 8),
        Text(
          'You don\'t have any payment records yet',
          style: TextStyle(color: Colors.grey[500], fontSize: 14),
          textAlign: TextAlign.center,
        ),
      ],
      ),
    );
  }

  Widget _buildEmptyState(BuildContext context) {
    return Center(
      child: Column(
        mainAxisAlignment: MainAxisAlignment.center,
        children: [
          Icon(Icons.error_outline, size: 64, color: Colors.grey[400]),
          const SizedBox(height: 16),
          Text(
            'Unable to load payment data',
            style: TextStyle(
              fontSize: 18,
              fontWeight: FontWeight.bold,
              color: Colors.grey[600],
            ),
          ),
          const SizedBox(height: 8),
          Text(
            'Please try again later',
            style: TextStyle(color: Colors.grey[500], fontSize: 14),
          ),
          const SizedBox(height: 24),
          ElevatedButton(
            onPressed: () => _loadPaymentData(),
            child: const Text('Try Again'),
          ),
        ],
      ),
    );
  }
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\lib\features\profile\presentation\pages

\profile_page.dart

```
import 'package:flutter/material.dart';
import 'package:flutter_bloc/flutter_bloc.dart';
import 'package:wismon_keuangan/features/auth/presentation/bloc/auth_bloc.dart';
import 'package:wismon_keuangan/features/auth/presentation/bloc/auth_state.dart';
import 'package:wismon_keuangan/features/auth/presentation/bloc/auth_event.dart';
import 'package:flutter_svg/flutter_svg.dart';

class ProfilePage extends StatelessWidget {
  const ProfilePage({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.transparent,
      body: Stack(
        children: [
          // Correctly positioned logo background
          Positioned(
            left: 120,
            bottom: 1,
            child: RepaintBoundary(
              child: Opacity(
                opacity: 1,
                child: Transform.scale(
                  scale: 3,
                  child: SvgPicture.asset(
                    'assets/LogoWHNProfil.svg',
                    width: 300,
                    height: 185,
                    fit: BoxFit.contain,
                  ),
                ),
              ),
            ),
          ),
          SafeArea(
            child: Column(
              children: [
                _buildHeader(context),
                Expanded(
                  child: BlocBuilder<AuthBloc, AuthState>(
                    buildWhen: (previous, current) {
                      // Only rebuild when auth state changes
                      return previous.runtimeType != current.runtimeType;
                    },
                    builder: (context, state) {
                      if (state is AuthAuthenticated) {
                        return _buildProfileContent(context, state.user);
                      }
                      if (state is AuthError) {
                        return _buildErrorState(context, state.message);
                      }
                      return const Center(
                        child: CircularProgressIndicator(
                          valueColor: AlwaysStoppedAnimation<Color>(
                            Color(0xFF135EA2),
                          ),
                        ),
                      );
                    },
                  ),
                ),
              ],
            ),
          ),
        ],
      ),
    );
  }

  Widget _buildHeader(BuildContext context) {
    return RepaintBoundary(
      child: Container(
        padding: const EdgeInsets.symmetric(horizontal: 16, vertical: 12),
        child: Row(
          mainAxisAlignment: MainAxisAlignment.spaceBetween,
          children: [
            const Text(
              'Profil',
              style: TextStyle(
                fontSize: 24,
                fontWeight: FontWeight.w800,
                color: Color(0xFF121212),
                letterSpacing: -0.24,
              ),
            ),
            Container(
              width: 40,
              height: 40,
              decoration: BoxDecoration(
                color: Colors.white,
                borderRadius: BorderRadius.circular(8),
                boxShadow: const [
                  BoxShadow(
                    color: Color(0x0D000000),
                    blurRadius: 10,
                    offset: Offset(0, 4),
                  ),
                ],
              ),
              child: IconButton(
                onPressed: () {
                  _showSettingsBottomSheet(context);
                },
                icon: const Icon(
                  Icons.settings,
                  color: Color(0xFF121212),
                  size: 20,
                ),
                padding: EdgeInsets.zero,
              ),
            ),
          ],
        ),
      ),
    );
  }

  void _showSettingsBottomSheet(BuildContext context) {
    showModalBottomSheet(
      context: context,
      shape: const RoundedRectangleBorder(
        borderRadius: BorderRadius.vertical(top: Radius.circular(20)),
      ),
      builder: (BuildContext sheetContext) {
        return Container(
          padding: const EdgeInsets.all(20),
          child: Column(
            mainAxisSize: MainAxisSize.min,
            children: [
              const Text(
                'Pengaturan Profil',
                style: TextStyle(fontSize: 18, fontWeight: FontWeight.bold),
              ),
              const SizedBox(height: 20),
              const Text(
                'Kelola profil dan pengaturan akun Anda di sini.',
                textAlign: TextAlign.center,
                style: TextStyle(fontSize: 14, color: Colors.grey),
              ),
              const SizedBox(height: 20),
              ElevatedButton(
                onPressed: () {
                  Navigator.pop(sheetContext);
                  // TODO: Implement profile edit functionality
                  ScaffoldMessenger.of(context).showSnackBar(
                    const SnackBar(
                      content: Text('Fitur edit profil akan segera tersedia'),
                    ),
                  );
                },
                style: ElevatedButton.styleFrom(
                  backgroundColor: const Color(0xFF207BB5),
                  padding: const EdgeInsets.symmetric(vertical: 12),
                  shape: RoundedRectangleBorder(
                    borderRadius: BorderRadius.circular(8),
                  ),
                  minimumSize: const Size(double.infinity, 50),
                ),
                child: const Text(
                  'Edit Profil',
                  style: TextStyle(color: Colors.white, fontSize: 16),
                ),
              ),
              const SizedBox(height: 10),
              TextButton(
                onPressed: () => Navigator.pop(sheetContext),
                child: const Text(
                  'Tutup',
                  style: TextStyle(color: Colors.grey),
                ),
              ),
            ],
          ),
        );
      },
    );
  }

  Widget _buildErrorState(BuildContext context, String message) {
    return RepaintBoundary(
      child: Center(
        child: Padding(
          padding: const EdgeInsets.all(16),
          child: Column(
            mainAxisAlignment: MainAxisAlignment.center,
            children: [
              Icon(Icons.error_outline, size: 64, color: Colors.grey[400]),
              const SizedBox(height: 16),
              Text(
                'Gagal memuat profil',
                style: TextStyle(
                  fontSize: 18,
                  fontWeight: FontWeight.w600,
                  color: Colors.grey[700],
                ),
              ),
              const SizedBox(height: 8),
              Text(
                message,
                textAlign: TextAlign.center,
                style: TextStyle(fontSize: 14, color: Colors.grey[600]),
              ),
              const SizedBox(height: 24),
              ElevatedButton(
                onPressed: () {
                  // Retry loading profile
                  context.read<AuthBloc>().add(const CheckAuthStatusEvent());
                },
                style: ElevatedButton.styleFrom(
                  backgroundColor: const Color(0xFF207BB5),
                  padding: const EdgeInsets.symmetric(
                    horizontal: 24,
                    vertical: 12,
                  ),
                  shape: RoundedRectangleBorder(
                    borderRadius: BorderRadius.circular(8),
                  ),
                ),
                child: const Text(
                  'Coba Lagi',
                  style: TextStyle(color: Colors.white, fontSize: 16),
                ),
              ),
            ],
          ),
        ),
      ),
    );
  }

  Widget _buildProfileContent(BuildContext context, dynamic user) {
    // Add null safety checks
    if (user == null) {
      return _buildErrorState(context, 'Data profil tidak tersedia');
    }

    return RepaintBoundary(
      child: SingleChildScrollView(
        padding: const EdgeInsets.symmetric(horizontal: 16, vertical: 8),
        physics: const AlwaysScrollableScrollPhysics(),
        child: Column(
          children: [
            _buildProfileCard(context, user),
            const SizedBox(height: 20),
            _buildInfoGrid(user),
          ],
        ),
      ),
    );
  }

  Widget _buildProfileCard(BuildContext context, dynamic user) {
    // Safe access to user properties with fallbacks
    final userName = _safeGetString(user.namam) ¿¿ 'Nama tidak tersedia';
    final birthPlace = _safeGetString(user.tplahir) ¿¿ '';
    final registrationDate = _safeGetString(user.tgdaftar) ¿¿ '';

    String locationInfo = '';
    if (birthPlace.isNotEmpty && registrationDate.isNotEmpty) {
      locationInfo = '$birthPlace ¿ Terdaftar $registrationDate';
    } else if (birthPlace.isNotEmpty) {
      locationInfo = 'Tempat Lahir: $birthPlace';
    } else if (registrationDate.isNotEmpty) {
      locationInfo = 'Terdaftar: $registrationDate';
    } else {
      locationInfo = 'Informasi tidak tersedia';
    }

    return Container(
      padding: const EdgeInsets.all(8),
      decoration: BoxDecoration(
        color: Colors.white,
        borderRadius: BorderRadius.circular(8),
        border: Border.all(color: const Color(0xFFE7E7E7)),
        boxShadow: const [
          BoxShadow(
            color: Color(0x05000000),
            blurRadius: 8,
            offset: Offset(0, 2),
          ),
        ],
      ),
      child: Row(
        children: [
          Container(
            width: 40,
            height: 40,
            decoration: BoxDecoration(
              color: const Color(0xFF207BB5),
              borderRadius: BorderRadius.circular(4),
            ),
            child: const Icon(Icons.person, color: Colors.white, size: 32),
          ),
          const SizedBox(width: 8),
          Expanded(
            child: Column(
              crossAxisAlignment: CrossAxisAlignment.start,
              children: [
                Text(
                  userName,
                  style: const TextStyle(
                    fontSize: 14,
                    fontWeight: FontWeight.w500,
                    color: Color(0xFF121212),
                  ),
                  maxLines: 2,
                  overflow: TextOverflow.ellipsis,
                ),
                const SizedBox(height: 2),
                Text(
                  locationInfo,
                  style: const TextStyle(
                    fontSize: 12,
                    fontWeight: FontWeight.w400,
                    color: Color(0xFF545556),
                  ),
                  maxLines: 1,
                  overflow: TextOverflow.ellipsis,
                ),
              ],
            ),
          ),
        ],
      ),
    );
  }

  Widget _buildInfoGrid(dynamic user) {
    return Column(
      children: [
        Row(
          children: [
            Expanded(
              child: _buildInfoBox('NRM', _safeGetString(user.nrm) ¿¿ '-'),
            ),
          ],
        ),
        const SizedBox(height: 8),
        _buildInfoBox('NIM', _safeGetString(user.nim) ¿¿ 'Belum tersedia'),
        const SizedBox(height: 8),
        _buildInfoBox('Email', _safeGetString(user.email) ¿¿ 'Belum diatur'),
        const SizedBox(height: 8),
        _buildInfoBox(
          'No. Telepon',
          _safeGetString(user.phone) ¿¿ 'Belum diatur',
        ),
        const SizedBox(height: 8),
        _buildInfoBox(
          'Tempat Lahir',
          _safeGetString(user.tplahir) ¿¿ 'Belum diatur',
        ),
        const SizedBox(height: 8),
        _buildInfoBox(
          'Tanggal Daftar',
          _safeGetString(user.tgdaftar) ¿¿ 'Belum tersedia',
        ),
      ],
    );
  }

  Widget _buildInfoBox(String title, String value) {
    return Container(
      padding: const EdgeInsets.all(8),
      decoration: BoxDecoration(
        color: const Color(0xFFFBFBFB),
        borderRadius: BorderRadius.circular(8),
        border: Border.all(color: const Color(0xFFE7E7E7)),
      ),
      child: Column(
        crossAxisAlignment: CrossAxisAlignment.start,
        children: [
          Text(
            title,
            style: const TextStyle(
              fontSize: 12,
              fontWeight: FontWeight.w500,
              color: Color(0xFF1C1D1F),
            ),
          ),
          const Divider(height: 8, color: Color(0xFFE7E7E7)),
          Text(
            value,
            style: const TextStyle(
              fontSize: 18,
              fontWeight: FontWeight.w500,
              color: Color(0xFF1C1D1F),
            ),
            maxLines: 2,
            overflow: TextOverflow.ellipsis,
          ),
        ],
      ),
    );
  }

  // Helper method to safely get string values
  String¿ _safeGetString(dynamic value) {
    if (value == null) return null;
    if (value is String) return value.isEmpty ¿ null : value;
    return value.toString().isEmpty ¿ null : value.toString();
  }
}
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\linux\.gitignore

```
flutter/ephemeral
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\linux\CMakeLists.txt

```
# Project-level configuration.
cmake_minimum_required(VERSION 3.13)
project(runner LANGUAGES CXX)

# The name of the executable created for the application. Change this to change
# the on-disk name of your application.
set(BINARY_NAME "wismon_keuangan")
# The unique GTK application identifier for this application. See:
# https://wiki.gnome.org/HowDoI/ChooseApplicationID
set(APPLICATION_ID "com.example.wismon_keuangan")

# Explicitly opt in to modern CMake behaviors to avoid warnings with recent
# versions of CMake.
cmake_policy(SET CMP0063 NEW)

# Load bundled libraries from the lib/ directory relative to the binary.
set(CMAKE_INSTALL_RPATH "$ORIGIN/lib")

# Root filesystem for cross-building.
if(FLUTTER_TARGET_PLATFORM_SYSROOT)
  set(CMAKE_SYSROOT ${FLUTTER_TARGET_PLATFORM_SYSROOT})
  set(CMAKE_FIND_ROOT_PATH ${CMAKE_SYSROOT})
  set(CMAKE_FIND_ROOT_PATH_MODE_PROGRAM NEVER)
  set(CMAKE_FIND_ROOT_PATH_MODE_PACKAGE ONLY)
  set(CMAKE_FIND_ROOT_PATH_MODE_LIBRARY ONLY)
  set(CMAKE_FIND_ROOT_PATH_MODE_INCLUDE ONLY)
endif()

# Define build configuration options.
if(NOT CMAKE_BUILD_TYPE AND NOT CMAKE_CONFIGURATION_TYPES)
  set(CMAKE_BUILD_TYPE "Debug" CACHE
    STRING "Flutter build mode" FORCE)
  set_property(CACHE CMAKE_BUILD_TYPE PROPERTY STRINGS
    "Debug" "Profile" "Release")
endif()

# Compilation settings that should be applied to most targets.
#
# Be cautious about adding new options here, as plugins use this function by
# default. In most cases, you should add new options to specific targets instead
# of modifying this function.
function(APPLY_STANDARD_SETTINGS TARGET)
  target_compile_features(${TARGET} PUBLIC cxx_std_14)
  target_compile_options(${TARGET} PRIVATE -Wall -Werror)
  target_compile_options(${TARGET} PRIVATE "$<$<NOT:$<CONFIG:Debug>>:-O3>")
  target_compile_definitions(${TARGET} PRIVATE "$<$<NOT:$<CONFIG:Debug>>:NDEBUG>")
endfunction()

# Flutter library and tool build rules.
set(FLUTTER_MANAGED_DIR "${CMAKE_CURRENT_SOURCE_DIR}/flutter")
add_subdirectory(${FLUTTER_MANAGED_DIR})

# System-level dependencies.
find_package(PkgConfig REQUIRED)
pkg_check_modules(GTK REQUIRED IMPORTED_TARGET gtk+-3.0)

# Application build; see runner/CMakeLists.txt.
add_subdirectory("runner")

# Run the Flutter tool portions of the build. This must not be removed.
add_dependencies(${BINARY_NAME} flutter_assemble)

# Only the install-generated bundle's copy of the executable will launch
# correctly, since the resources must in the right relative locations. To avoid
# people trying to run the unbundled copy, put it in a subdirectory instead of
# the default top-level location.
set_target_properties(${BINARY_NAME}
  PROPERTIES
  RUNTIME_OUTPUT_DIRECTORY "${CMAKE_BINARY_DIR}/intermediates_do_not_run"
)
```

```
# Generated plugin build rules, which manage building the plugins and adding
# them to the application.
include(flutter/generated_plugins.cmake)


# === Installation ===
# By default, "installing" just makes a relocatable bundle in the build
# directory.
set(BUILD_BUNDLE_DIR "${PROJECT_BINARY_DIR}/bundle")
if(CMAKE_INSTALL_PREFIX_INITIALIZED_TO_DEFAULT)
  set(CMAKE_INSTALL_PREFIX "${BUILD_BUNDLE_DIR}" CACHE PATH "..." FORCE)
endif()

# Start with a clean build bundle directory every time.
install(CODE "
  file(REMOVE_RECURSE \"${BUILD_BUNDLE_DIR}/\")
  " COMPONENT Runtime)

set(INSTALL_BUNDLE_DATA_DIR "${CMAKE_INSTALL_PREFIX}/data")
set(INSTALL_BUNDLE_LIB_DIR "${CMAKE_INSTALL_PREFIX}/lib")

install(TARGETS ${BINARY_NAME} RUNTIME DESTINATION "${CMAKE_INSTALL_PREFIX}"
  COMPONENT Runtime)

install(FILES "${FLUTTER_ICU_DATA_FILE}" DESTINATION "${INSTALL_BUNDLE_DATA_DIR}"
  COMPONENT Runtime)

install(FILES "${FLUTTER_LIBRARY}" DESTINATION "${INSTALL_BUNDLE_LIB_DIR}"
  COMPONENT Runtime)

foreach(bundled_library ${PLUGIN_BUNDLED_LIBRARIES})
  install(FILES "${bundled_library}"
    DESTINATION "${INSTALL_BUNDLE_LIB_DIR}"
    COMPONENT Runtime)
endforeach(bundled_library)

# Copy the native assets provided by the build.dart from all packages.
set(NATIVE_ASSETS_DIR "${PROJECT_BUILD_DIR}native_assets/linux/")
install(DIRECTORY "${NATIVE_ASSETS_DIR}"
   DESTINATION "${INSTALL_BUNDLE_LIB_DIR}"
   COMPONENT Runtime)

# Fully re-copy the assets directory on each build to avoid having stale files
# from a previous install.
set(FLUTTER_ASSET_DIR_NAME "flutter_assets")
install(CODE "
  file(REMOVE_RECURSE \"${INSTALL_BUNDLE_DATA_DIR}/${FLUTTER_ASSET_DIR_NAME}\")
  " COMPONENT Runtime)
install(DIRECTORY "${PROJECT_BUILD_DIR}/${FLUTTER_ASSET_DIR_NAME}"
  DESTINATION "${INSTALL_BUNDLE_DATA_DIR}" COMPONENT Runtime)

# Install the AOT library on non-Debug builds only.
if(NOT CMAKE_BUILD_TYPE MATCHES "Debug")
  install(FILES "${AOT_LIBRARY}" DESTINATION "${INSTALL_BUNDLE_LIB_DIR}"
    COMPONENT Runtime)
endif()
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\linux\flutter\CMakeLists.txt

```
# This file controls Flutter-level build steps. It should not be edited.
cmake_minimum_required(VERSION 3.10)

set(EPHEMERAL_DIR "${CMAKE_CURRENT_SOURCE_DIR}/ephemeral")
```

```
# Configuration provided via flutter tool.
include(${EPHEMERAL_DIR}/generated_config.cmake)

# TODO: Move the rest of this into files in ephemeral. See
# https://github.com/flutter/flutter/issues/57146.

# Serves the same purpose as list(TRANSFORM ... PREPEND ...),
# which isn't available in 3.10.
function(list_prepend LIST_NAME PREFIX)
    set(NEW_LIST "")
    foreach(element ${${LIST_NAME}})
        list(APPEND NEW_LIST "${PREFIX}${element}")
    endforeach(element)
    set(${LIST_NAME} "${NEW_LIST}" PARENT_SCOPE)
endfunction()

# === Flutter Library ===
# System-level dependencies.
find_package(PkgConfig REQUIRED)
pkg_check_modules(GTK REQUIRED IMPORTED_TARGET gtk+-3.0)
pkg_check_modules(GLIB REQUIRED IMPORTED_TARGET glib-2.0)
pkg_check_modules(GIO REQUIRED IMPORTED_TARGET gio-2.0)

set(FLUTTER_LIBRARY "${EPHEMERAL_DIR}/libflutter_linux_gtk.so")

# Published to parent scope for install step.
set(FLUTTER_LIBRARY ${FLUTTER_LIBRARY} PARENT_SCOPE)
set(FLUTTER_ICU_DATA_FILE "${EPHEMERAL_DIR}/icudtl.dat" PARENT_SCOPE)
set(PROJECT_BUILD_DIR "${PROJECT_DIR}/build/" PARENT_SCOPE)
set(AOT_LIBRARY "${PROJECT_DIR}/build/lib/libapp.so" PARENT_SCOPE)

list(APPEND FLUTTER_LIBRARY_HEADERS
  "fl_basic_message_channel.h"
  "fl_binary_codec.h"
  "fl_binary_messenger.h"
  "fl_dart_project.h"
  "fl_engine.h"
  "fl_json_message_codec.h"
  "fl_json_method_codec.h"
  "fl_message_codec.h"
  "fl_method_call.h"
  "fl_method_channel.h"
  "fl_method_codec.h"
  "fl_method_response.h"
  "fl_plugin_registrar.h"
  "fl_plugin_registry.h"
  "fl_standard_message_codec.h"
  "fl_standard_method_codec.h"
  "fl_string_codec.h"
  "fl_value.h"
  "fl_view.h"
  "flutter_linux.h"
)
list_prepend(FLUTTER_LIBRARY_HEADERS "${EPHEMERAL_DIR}/flutter_linux/")
add_library(flutter INTERFACE)
target_include_directories(flutter INTERFACE
  "${EPHEMERAL_DIR}"
)
target_link_libraries(flutter INTERFACE "${FLUTTER_LIBRARY}")
target_link_libraries(flutter INTERFACE
  PkgConfig::GTK
  PkgConfig::GLIB
  PkgConfig::GIO
)
add_dependencies(flutter flutter_assemble)
```

```
# === Flutter tool backend ===
# _phony_ is a non-existent file to force this command to run every time,
# since currently there's no way to get a full input/output list from the
# flutter tool.
add_custom_command(
  OUTPUT ${FLUTTER_LIBRARY} ${FLUTTER_LIBRARY_HEADERS}
    ${CMAKE_CURRENT_BINARY_DIR}/_phony_
  COMMAND ${CMAKE_COMMAND} -E env
    ${FLUTTER_TOOL_ENVIRONMENT}
    "${FLUTTER_ROOT}/packages/flutter_tools/bin/tool_backend.sh"
      ${FLUTTER_TARGET_PLATFORM} ${CMAKE_BUILD_TYPE}
  VERBATIM
)
add_custom_target(flutter_assemble DEPENDS
  "${FLUTTER_LIBRARY}"
  ${FLUTTER_LIBRARY_HEADERS}
)
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\linux\flutter\generated_plugins.cmake

```
#
# Generated file, do not edit.
#

list(APPEND FLUTTER_PLUGIN_LIST
)

list(APPEND FLUTTER_FFI_PLUGIN_LIST
)

set(PLUGIN_BUNDLED_LIBRARIES)

foreach(plugin ${FLUTTER_PLUGIN_LIST})
  add_subdirectory(flutter/ephemeral/.plugin_symlinks/${plugin}/linux plugins/${plugin})
  target_link_libraries(${BINARY_NAME} PRIVATE ${plugin}_plugin)
  list(APPEND PLUGIN_BUNDLED_LIBRARIES $<TARGET_FILE:${plugin}_plugin>)
  list(APPEND PLUGIN_BUNDLED_LIBRARIES ${${plugin}_bundled_libraries})
endforeach(plugin)

foreach(ffi_plugin ${FLUTTER_FFI_PLUGIN_LIST})
  add_subdirectory(flutter/ephemeral/.plugin_symlinks/${ffi_plugin}/linux plugins/${ffi_plugin})
  list(APPEND PLUGIN_BUNDLED_LIBRARIES ${${ffi_plugin}_bundled_libraries})
endforeach(ffi_plugin)
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\linux\flutter\generated_plugin_registrant.cc

```
//
//  Generated file. Do not edit.
//

// clang-format off

#include "generated_plugin_registrant.h"
```

```
void fl_register_plugins(FlPluginRegistry* registry) {
}
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\linux\flutter\generated_plugin_registrant.h

```
//
//  Generated file. Do not edit.
//

// clang-format off

#ifndef GENERATED_PLUGIN_REGISTRANT_
#define GENERATED_PLUGIN_REGISTRANT_

#include <flutter_linux/flutter_linux.h>

// Registers Flutter plugins.
void fl_register_plugins(FlPluginRegistry* registry);

#endif  // GENERATED_PLUGIN_REGISTRANT_
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\linux\runner\CMakeLists.txt

```
cmake_minimum_required(VERSION 3.13)
project(runner LANGUAGES CXX)

# Define the application target. To change its name, change BINARY_NAME in the
# top-level CMakeLists.txt, not the value here, or `flutter run` will no longer
# work.
#
# Any new source files that you add to the application should be added here.
add_executable(${BINARY_NAME}
  "main.cc"
  "my_application.cc"
  "${FLUTTER_MANAGED_DIR}/generated_plugin_registrant.cc"
)

# Apply the standard set of build settings. This can be removed for applications
# that need different build settings.
apply_standard_settings(${BINARY_NAME})

# Add preprocessor definitions for the application ID.
add_definitions(-DAPPLICATION_ID="${APPLICATION_ID}")

# Add dependency libraries. Add any application-specific dependencies here.
target_link_libraries(${BINARY_NAME} PRIVATE flutter)
target_link_libraries(${BINARY_NAME} PRIVATE PkgConfig::GTK)

target_include_directories(${BINARY_NAME} PRIVATE "${CMAKE_SOURCE_DIR}")
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\linux\runner\main.cc

```
#include "my_application.h"

int main(int argc, char** argv) {
  g_autoptr(MyApplication) app = my_application_new();
  return g_application_run(G_APPLICATION(app), argc, argv);
}
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\linux\runner\my_application.cc

```
#include "my_application.h"

#include <flutter_linux/flutter_linux.h>
#ifdef GDK_WINDOWING_X11
#include <gdk/gdkx.h>
#endif

#include "flutter/generated_plugin_registrant.h"

struct _MyApplication {
  GtkApplication parent_instance;
  char** dart_entrypoint_arguments;
};

G_DEFINE_TYPE(MyApplication, my_application, GTK_TYPE_APPLICATION)

// Implements GApplication::activate.
static void my_application_activate(GApplication* application) {
  MyApplication* self = MY_APPLICATION(application);
  GtkWindow* window =
      GTK_WINDOW(gtk_application_window_new(GTK_APPLICATION(application)));

  // Use a header bar when running in GNOME as this is the common style used
  // by applications and is the setup most users will be using (e.g. Ubuntu
  // desktop).
  // If running on X and not using GNOME then just use a traditional title bar
  // in case the window manager does more exotic layout, e.g. tiling.
  // If running on Wayland assume the header bar will work (may need changing
  // if future cases occur).
  gboolean use_header_bar = TRUE;
#ifdef GDK_WINDOWING_X11
  GdkScreen* screen = gtk_window_get_screen(window);
  if (GDK_IS_X11_SCREEN(screen)) {
    const gchar* wm_name = gdk_x11_screen_get_window_manager_name(screen);
    if (g_strcmp0(wm_name, "GNOME Shell") != 0) {
      use_header_bar = FALSE;
    }
  }
#endif
  if (use_header_bar) {
    GtkHeaderBar* header_bar = GTK_HEADER_BAR(gtk_header_bar_new());
    gtk_widget_show(GTK_WIDGET(header_bar));
    gtk_header_bar_set_title(header_bar, "wismon_keuangan");
    gtk_header_bar_set_show_close_button(header_bar, TRUE);
    gtk_window_set_titlebar(window, GTK_WIDGET(header_bar));
  } else {
    gtk_window_set_title(window, "wismon_keuangan");
  }

  gtk_window_set_default_size(window, 1280, 720);
  gtk_widget_show(GTK_WIDGET(window));

  g_autoptr(FlDartProject) project = fl_dart_project_new();
  fl_dart_project_set_dart_entrypoint_arguments(project, self->dart_entrypoint_arguments);

  FlView* view = fl_view_new(project);
  gtk_widget_show(GTK_WIDGET(view));
  gtk_container_add(GTK_CONTAINER(window), GTK_WIDGET(view));
  fl_register_plugins(FL_PLUGIN_REGISTRY(view));

  gtk_widget_grab_focus(GTK_WIDGET(view));
}

// Implements GApplication::local_command_line.
static gboolean my_application_local_command_line(GApplication* application, gchar*** arguments, int*
exit_status) {
  MyApplication* self = MY_APPLICATION(application);
  // Strip out the first argument as it is the binary name.
  self->dart_entrypoint_arguments = g_strdupv(*arguments + 1);

  g_autoptr(GError) error = nullptr;
  if (!g_application_register(application, nullptr, &error)) {
     g_warning("Failed to register: %s", error->message);
     *exit_status = 1;
     return TRUE;
  }

  g_application_activate(application);
  *exit_status = 0;

  return TRUE;
}

// Implements GApplication::startup.
static void my_application_startup(GApplication* application) {
  //MyApplication* self = MY_APPLICATION(object);

  // Perform any actions required at application startup.

  G_APPLICATION_CLASS(my_application_parent_class)->startup(application);
}

// Implements GApplication::shutdown.
static void my_application_shutdown(GApplication* application) {
  //MyApplication* self = MY_APPLICATION(object);

  // Perform any actions required at application shutdown.

  G_APPLICATION_CLASS(my_application_parent_class)->shutdown(application);
}

// Implements GObject::dispose.
static void my_application_dispose(GObject* object) {
  MyApplication* self = MY_APPLICATION(object);
  g_clear_pointer(&self->dart_entrypoint_arguments, g_strfreev);
  G_OBJECT_CLASS(my_application_parent_class)->dispose(object);
}

static void my_application_class_init(MyApplicationClass* klass) {
  G_APPLICATION_CLASS(klass)->activate = my_application_activate;
  G_APPLICATION_CLASS(klass)->local_command_line = my_application_local_command_line;
  G_APPLICATION_CLASS(klass)->startup = my_application_startup;
  G_APPLICATION_CLASS(klass)->shutdown = my_application_shutdown;
  G_OBJECT_CLASS(klass)->dispose = my_application_dispose;
}

static void my_application_init(MyApplication* self) {}

MyApplication* my_application_new() {
  // Set the program name to the application ID, which helps various systems
  // like GTK and desktop environments map this running application to its
  // corresponding .desktop file. This ensures better integration by allowing
  // the application to be recognized beyond its binary name.
  g_set_prgname(APPLICATION_ID);

  return MY_APPLICATION(g_object_new(my_application_get_type(),
                                     "application-id", APPLICATION_ID,
                                     "flags", G_APPLICATION_NON_UNIQUE,
                                     nullptr));
}
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\linux\runner\my_application.h

```
#ifndef FLUTTER_MY_APPLICATION_H_
#define FLUTTER_MY_APPLICATION_H_

#include <gtk/gtk.h>

G_DECLARE_FINAL_TYPE(MyApplication, my_application, MY, APPLICATION,
                     GtkApplication)

/**
 * my_application_new:
 *
 * Creates a new Flutter-based application.
 *
 * Returns: a new #MyApplication.
 */
MyApplication* my_application_new();

#endif  // FLUTTER_MY_APPLICATION_H_
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\macos\.gitignore

```
# Flutter-related
**/Flutter/ephemeral/
**/Pods/

# Xcode-related
**/dgph
**/xcuserdata/
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\macos\Flutter\Flutter-Debug.xcconfig

```
#include "ephemeral/Flutter-Generated.xcconfig"
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\macos\Flutter\Flutter-Release.xcconfig

```
#include "ephemeral/Flutter-Generated.xcconfig"
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\macos\Flutter\GeneratedPluginRegistrant

.swift

```
//
//  Generated file. Do not edit.
//

import FlutterMacOS
import Foundation

import path_provider_foundation
import shared_preferences_foundation

func RegisterGeneratedPlugins(registry: FlutterPluginRegistry) {
  PathProviderPlugin.register(with: registry.registrar(forPlugin: "PathProviderPlugin"))
  SharedPreferencesPlugin.register(with: registry.registrar(forPlugin: "SharedPreferencesPlugin"))
}
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\macos\Runner\AppDelegate.swift

```
import Cocoa
import FlutterMacOS

@main
class AppDelegate: FlutterAppDelegate {
  override func applicationShouldTerminateAfterLastWindowClosed(_ sender: NSApplication) -> Bool {
    return true
  }

  override func applicationSupportsSecureRestorableState(_ app: NSApplication) -> Bool {
    return true
  }
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\macos\Runner\DebugProfile.entitlements

```
<¿xml version="1.0" encoding="UTF-8"¿>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
 <key>com.apple.security.app-sandbox</key>
 <true/>
 <key>com.apple.security.cs.allow-jit</key>
 <true/>
 <key>com.apple.security.network.server</key>
 <true/>
</dict>
</plist>
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\macos\Runner\Info.plist

```
<¿xml version="1.0" encoding="UTF-8"¿>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
 <key>CFBundleDevelopmentRegion</key>
 <string>$(DEVELOPMENT_LANGUAGE)</string>
 <key>CFBundleExecutable</key>
 <string>$(EXECUTABLE_NAME)</string>
 <key>CFBundleIconFile</key>
 <string></string>
 <key>CFBundleIdentifier</key>
 <string>$(PRODUCT_BUNDLE_IDENTIFIER)</string>
 <key>CFBundleInfoDictionaryVersion</key>
 <string>6.0</string>
 <key>CFBundleName</key>
 <string>$(PRODUCT_NAME)</string>
 <key>CFBundlePackageType</key>
 <string>APPL</string>
 <key>CFBundleShortVersionString</key>
 <string>$(FLUTTER_BUILD_NAME)</string>
 <key>CFBundleVersion</key>
 <string>$(FLUTTER_BUILD_NUMBER)</string>
 <key>LSMinimumSystemVersion</key>
 <string>$(MACOSX_DEPLOYMENT_TARGET)</string>
 <key>NSHumanReadableCopyright</key>
 <string>$(PRODUCT_COPYRIGHT)</string>
 <key>NSMainNibFile</key>
 <string>MainMenu</string>
 <key>NSPrincipalClass</key>
 <string>NSApplication</string>
</dict>
</plist>
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\macos\Runner\MainFlutterWindow.swift

```
import Cocoa
import FlutterMacOS

class MainFlutterWindow: NSWindow {
  override func awakeFromNib() {
    let flutterViewController = FlutterViewController()
    let windowFrame = self.frame
    self.contentViewController = flutterViewController
    self.setFrame(windowFrame, display: true)

    RegisterGeneratedPlugins(registry: flutterViewController)

    super.awakeFromNib()
  }
}
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\macos\Runner\Release.entitlements

```
<¿xml version="1.0" encoding="UTF-8"¿>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
 <key>com.apple.security.app-sandbox</key>
 <true/>
</dict>
</plist>
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\macos\Runner\Assets.xcassets\AppIcon.ap

piconset\Contents.json

```
{
  "images" : [
    {
      "size" : "16x16",
      "idiom" : "mac",
      "filename" : "app_icon_16.png",
      "scale" : "1x"
    },
    {
      "size" : "16x16",
      "idiom" : "mac",
      "filename" : "app_icon_32.png",
      "scale" : "2x"
    },
    {
      "size" : "32x32",
      "idiom" : "mac",
      "filename" : "app_icon_32.png",
      "scale" : "1x"
    },
    {
      "size" : "32x32",
      "idiom" : "mac",
      "filename" : "app_icon_64.png",
      "scale" : "2x"
    },
    {
      "size" : "128x128",
      "idiom" : "mac",
      "filename" : "app_icon_128.png",
      "scale" : "1x"
    },
    {
      "size" : "128x128",
      "idiom" : "mac",
      "filename" : "app_icon_256.png",
      "scale" : "2x"
    },
    {
      "size" : "256x256",
      "idiom" : "mac",
      "filename" : "app_icon_256.png",
      "scale" : "1x"
    },
    {
      "size" : "256x256",
      "idiom" : "mac",
      "filename" : "app_icon_512.png",
      "scale" : "2x"
    },
    {
      "size" : "512x512",
      "idiom" : "mac",
      "filename" : "app_icon_512.png",
      "scale" : "1x"
    },
    {
      "size" : "512x512",
      "idiom" : "mac",
      "filename" : "app_icon_1024.png",
      "scale" : "2x"
    }
  ],
  "info" : {
    "version" : 1,
    "author" : "xcode"
  }
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\macos\Runner\Base.lproj\MainMenu.xib

```
<¿xml version="1.0" encoding="UTF-8"¿>
<document type="com.apple.InterfaceBuilder3.Cocoa.XIB" version="3.0" toolsVersion="14490.70"
targetRuntime="MacOSX.Cocoa" propertyAccessControl="none" useAutolayout="YES"
customObjectInstantitationMethod="direct">
    <dependencies>
        <deployment identifier="macosx"/>
        <plugIn identifier="com.apple.InterfaceBuilder.CocoaPlugin" version="14490.70"/>
        <capability name="documents saved in the Xcode 8 format" minToolsVersion="8.0"/>
    </dependencies>
    <objects>
        <customObject id="-2" userLabel="File's Owner" customClass="NSApplication">
            <connections>
                <outlet property="delegate" destination="Voe-Tx-rLC" id="GzC-gU-4Uq"/>
            </connections>
        </customObject>
        <customObject id="-1" userLabel="First Responder" customClass="FirstResponder"/>
        <customObject id="-3" userLabel="Application" customClass="NSObject"/>
                           <customObject id="Voe-Tx-rLC" customClass="AppDelegate" customModule="Runner"
customModuleProvider="target">
            <connections>
                <outlet property="applicationMenu" destination="uQy-DD-JDr" id="XBo-yE-nKs"/>
                <outlet property="mainFlutterWindow" destination="QvC-M9-y7g" id="gIp-Ho-8D9"/>
            </connections>
        </customObject>
        <customObject id="YLy-65-1bz" customClass="NSFontManager"/>
        <menu title="Main Menu" systemMenu="main" id="AYu-sK-qS6">
            <items>
                <menuItem title="APP_NAME" id="1Xt-HY-uBw">
                    <modifierMask key="keyEquivalentModifierMask"/>
                    <menu key="submenu" title="APP_NAME" systemMenu="apple" id="uQy-DD-JDr">
                        <items>
                            <menuItem title="About APP_NAME" id="5kV-Vb-QxS">
                                <modifierMask key="keyEquivalentModifierMask"/>
                                <connections>
                                                  <action selector="orderFrontStandardAboutPanel:" target="-1"
id="Exp-CZ-Vem"/>
                                </connections>
                            </menuItem>
                            <menuItem isSeparatorItem="YES" id="VOq-y0-SEH"/>
                            <menuItem title="Preferences¿" keyEquivalent="," id="BOF-NM-1cW"/>
                            <menuItem isSeparatorItem="YES" id="wFC-TO-SCJ"/>
                            <menuItem title="Services" id="NMo-om-nkz">
                                <modifierMask key="keyEquivalentModifierMask"/>
                                <menu key="submenu" title="Services" systemMenu="services" id="hz9-B4-Xy5"/>
                            </menuItem>
                            <menuItem isSeparatorItem="YES" id="4je-JR-u6R"/>
                            <menuItem title="Hide APP_NAME" keyEquivalent="h" id="Olw-nP-bQN">
                                <connections>
                                    <action selector="hide:" target="-1" id="PnN-Uc-m68"/>
                                </connections>
                            </menuItem>
                            <menuItem title="Hide Others" keyEquivalent="h" id="Vdr-fp-XzO">
                                <modifierMask key="keyEquivalentModifierMask" option="YES" command="YES"/>
                                <connections>
                                    <action selector="hideOtherApplications:" target="-1" id="VT4-aY-XCT"/>
                </connections>
            </menuItem>
            <menuItem title="Show All" id="Kd2-mp-pUS">
                <modifierMask key="keyEquivalentModifierMask"/>
                <connections>
                    <action selector="unhideAllApplications:" target="-1" id="Dhg-Le-xox"/>
                </connections>
            </menuItem>
            <menuItem isSeparatorItem="YES" id="kCx-OE-vgT"/>
            <menuItem title="Quit APP_NAME" keyEquivalent="q" id="4sb-4s-VLi">
                <connections>
                    <action selector="terminate:" target="-1" id="Te7-pn-YzF"/>
                </connections>
            </menuItem>
        </items>
    </menu>
</menuItem>
<menuItem title="Edit" id="5QF-Oa-p0T">
    <modifierMask key="keyEquivalentModifierMask"/>
    <menu key="submenu" title="Edit" id="W48-6f-4Dl">
        <items>
            <menuItem title="Undo" keyEquivalent="z" id="dRJ-4n-Yzg">
                <connections>
                    <action selector="undo:" target="-1" id="M6e-cu-g7V"/>
                </connections>
            </menuItem>
            <menuItem title="Redo" keyEquivalent="Z" id="6dh-zS-Vam">
                <connections>
                    <action selector="redo:" target="-1" id="oIA-Rs-6OD"/>
                </connections>
            </menuItem>
            <menuItem isSeparatorItem="YES" id="WRV-NI-Exz"/>
            <menuItem title="Cut" keyEquivalent="x" id="uRl-iY-unG">
                <connections>
                    <action selector="cut:" target="-1" id="YJe-68-I9s"/>
                </connections>
            </menuItem>
            <menuItem title="Copy" keyEquivalent="c" id="x3v-GG-iWU">
                <connections>
                    <action selector="copy:" target="-1" id="G1f-GL-Joy"/>
                </connections>
            </menuItem>
            <menuItem title="Paste" keyEquivalent="v" id="gVA-U4-sdL">
                <connections>
                    <action selector="paste:" target="-1" id="UvS-8e-Qdg"/>
                </connections>
            </menuItem>
            <menuItem title="Paste and Match Style" keyEquivalent="V" id="WeT-3V-zwk">
                <modifierMask key="keyEquivalentModifierMask" option="YES" command="YES"/>
                <connections>
                    <action selector="pasteAsPlainText:" target="-1" id="cEh-KX-wJQ"/>
                </connections>
            </menuItem>
            <menuItem title="Delete" id="pa3-QI-u2k">
                <modifierMask key="keyEquivalentModifierMask"/>
                <connections>
                    <action selector="delete:" target="-1" id="0Mk-Ml-PaM"/>
                </connections>
            </menuItem>
            <menuItem title="Select All" keyEquivalent="a" id="Ruw-6m-B2m">
                <connections>
                    <action selector="selectAll:" target="-1" id="VNm-Mi-diN"/>
                </connections>
            </menuItem>
            <menuItem isSeparatorItem="YES" id="uyl-h8-XO2"/>
            <menuItem title="Find" id="4EN-yA-p0u">
```

```
                                <modifierMask key="keyEquivalentModifierMask"/>
                                <menu key="submenu" title="Find" id="1b7-l0-nxx">
                                    <items>
                                        <menuItem title="Find¿" tag="1" keyEquivalent="f" id="Xz5-n4-O0W">
                                            <connections>
                                                         <action selector="performFindPanelAction:" target="-1"
id="cD7-Qs-BN4"/>
                                            </connections>
                                        </menuItem>
                                                <menuItem title="Find and Replace¿" tag="12" keyEquivalent="f"
id="YEy-JH-Tfz">
                                                    <modifierMask key="keyEquivalentModifierMask" option="YES"
command="YES"/>
                                            <connections>
                                                         <action selector="performFindPanelAction:" target="-1"
id="WD3-Gg-5AJ"/>
                                            </connections>
                                        </menuItem>
                                        <menuItem title="Find Next" tag="2" keyEquivalent="g" id="q09-fT-Sye">
                                            <connections>
                                                         <action selector="performFindPanelAction:" target="-1"
id="NDo-RZ-v9R"/>
                                            </connections>
                                        </menuItem>
                                                     <menuItem title="Find Previous" tag="3" keyEquivalent="G"
id="OwM-mh-QMV">
                                            <connections>
                                                         <action selector="performFindPanelAction:" target="-1"
id="HOh-sY-3ay"/>
                                            </connections>
                                        </menuItem>
                                             <menuItem title="Use Selection for Find" tag="7" keyEquivalent="e"
id="buJ-ug-pKt">
                                            <connections>
                                                         <action selector="performFindPanelAction:" target="-1"
id="U76-nv-p5D"/>
                                            </connections>
                                        </menuItem>
                                        <menuItem title="Jump to Selection" keyEquivalent="j" id="S0p-oC-mLd">
                                            <connections>
                                                   <action selector="centerSelectionInVisibleArea:" target="-1"
id="IOG-6D-g5B"/>
                                            </connections>
                                        </menuItem>
                                    </items>
                                </menu>
                            </menuItem>
                            <menuItem title="Spelling and Grammar" id="Dv1-io-Yv7">
                                <modifierMask key="keyEquivalentModifierMask"/>
                                <menu key="submenu" title="Spelling" id="3IN-sU-3Bg">
                                    <items>
                                                 <menuItem title="Show Spelling and Grammar" keyEquivalent=":"
id="HFo-cy-zxI">
                                            <connections>
                                                                 <action selector="showGuessPanel:" target="-1"
id="vFj-Ks-hy3"/>
                                            </connections>
                                        </menuItem>
                                        <menuItem title="Check Document Now" keyEquivalent=";" id="hz2-CU-CR7">
                                            <connections>
                                                <action selector="checkSpelling:" target="-1" id="fz7-VC-reM"/>
                                            </connections>
                                        </menuItem>
                                        <menuItem isSeparatorItem="YES" id="bNw-od-mp5"/>
                                        <menuItem title="Check Spelling While Typing" id="rbD-Rh-wIN">
                                            <modifierMask key="keyEquivalentModifierMask"/>
                                            <connections>
                                                  <action selector="toggleContinuousSpellChecking:" target="-1"
id="7w6-Qz-0kB"/>
                                            </connections>
                                        </menuItem>
                                        <menuItem title="Check Grammar With Spelling" id="mK6-2p-4JG">
                                            <modifierMask key="keyEquivalentModifierMask"/>
                                            <connections>
                                                          <action selector="toggleGrammarChecking:" target="-1"
id="muD-Qn-j4w"/>
                                            </connections>
                                        </menuItem>
                                        <menuItem title="Correct Spelling Automatically" id="78Y-hA-62v">
                                            <modifierMask key="keyEquivalentModifierMask"/>
                                            <connections>
                                                          <action selector="toggleAutomaticSpellingCorrection:"
target="-1" id="2lM-Qi-WAP"/>
                                            </connections>
                                        </menuItem>
                                    </items>
                                </menu>
                            </menuItem>
                            <menuItem title="Substitutions" id="9ic-FL-obx">
                                <modifierMask key="keyEquivalentModifierMask"/>
                                <menu key="submenu" title="Substitutions" id="FeM-D8-WVr">
                                    <items>
                                        <menuItem title="Show Substitutions" id="z6F-FW-3nz">
                                            <modifierMask key="keyEquivalentModifierMask"/>
                                            <connections>
                                                   <action selector="orderFrontSubstitutionsPanel:" target="-1"
id="oku-mr-iSq"/>
                                            </connections>
                                        </menuItem>
                                        <menuItem isSeparatorItem="YES" id="gPx-C9-uUO"/>
                                        <menuItem title="Smart Copy/Paste" id="9yt-4B-nSM">
                                            <modifierMask key="keyEquivalentModifierMask"/>
                                            <connections>
                                                        <action selector="toggleSmartInsertDelete:" target="-1"
id="3IJ-Se-DZD"/>
                                            </connections>
                                        </menuItem>
                                        <menuItem title="Smart Quotes" id="hQb-2v-fYv">
                                            <modifierMask key="keyEquivalentModifierMask"/>
                                            <connections>
                                                           <action selector="toggleAutomaticQuoteSubstitution:"
target="-1" id="ptq-xd-QOA"/>
                                            </connections>
                                        </menuItem>
                                        <menuItem title="Smart Dashes" id="rgM-f4-ycn">
                                            <modifierMask key="keyEquivalentModifierMask"/>
                                            <connections>
                                                <action selector="toggleAutomaticDashSubstitution:" target="-1"
id="oCt-pO-9gS"/>
                                            </connections>
                                        </menuItem>
                                        <menuItem title="Smart Links" id="cwL-P1-jid">
                                            <modifierMask key="keyEquivalentModifierMask"/>
                                            <connections>
                                                   <action selector="toggleAutomaticLinkDetection:" target="-1"
id="Gip-E3-Fov"/>
                                            </connections>
                                        </menuItem>
                                        <menuItem title="Data Detectors" id="tRr-pd-1PS">
                                            <modifierMask key="keyEquivalentModifierMask"/>
                                            <connections>
                                                   <action selector="toggleAutomaticDataDetection:" target="-1"
```

```
id="R1I-Nq-Kbl"/>
                                            </connections>
                                        </menuItem>
                                        <menuItem title="Text Replacement" id="HFQ-gK-NFA">
                                            <modifierMask key="keyEquivalentModifierMask"/>
                                            <connections>
                                                <action selector="toggleAutomaticTextReplacement:" target="-1"
id="DvP-Fe-Py6"/>
                                            </connections>
                                        </menuItem>
                                    </items>
                                </menu>
                            </menuItem>
                            <menuItem title="Transformations" id="2oI-Rn-ZJC">
                                <modifierMask key="keyEquivalentModifierMask"/>
                                <menu key="submenu" title="Transformations" id="c8a-y6-VQd">
                                    <items>
                                        <menuItem title="Make Upper Case" id="vmV-6d-7jI">
                                            <modifierMask key="keyEquivalentModifierMask"/>
                                            <connections>
                                                <action selector="uppercaseWord:" target="-1" id="sPh-Tk-edu"/>
                                            </connections>
                                        </menuItem>
                                        <menuItem title="Make Lower Case" id="d9M-CD-aMd">
                                            <modifierMask key="keyEquivalentModifierMask"/>
                                            <connections>
                                                <action selector="lowercaseWord:" target="-1" id="iUZ-b5-hil"/>
                                            </connections>
                                        </menuItem>
                                        <menuItem title="Capitalize" id="UEZ-Bs-lqG">
                                            <modifierMask key="keyEquivalentModifierMask"/>
                                            <connections>
                                                <action selector="capitalizeWord:" target="-1"
id="26H-TL-nsh"/>
                                            </connections>
                                        </menuItem>
                                    </items>
                                </menu>
                            </menuItem>
                            <menuItem title="Speech" id="xrE-MZ-jX0">
                                <modifierMask key="keyEquivalentModifierMask"/>
                                <menu key="submenu" title="Speech" id="3rS-ZA-NoH">
                                    <items>
                                        <menuItem title="Start Speaking" id="Ynk-f8-cLZ">
                                            <modifierMask key="keyEquivalentModifierMask"/>
                                            <connections>
                                                <action selector="startSpeaking:" target="-1" id="654-Ng-kyl"/>
                                            </connections>
                                        </menuItem>
                                        <menuItem title="Stop Speaking" id="Oyz-dy-DGm">
                                            <modifierMask key="keyEquivalentModifierMask"/>
                                            <connections>
                                                <action selector="stopSpeaking:" target="-1" id="dX8-6p-jy9"/>
                                            </connections>
                                        </menuItem>
                                    </items>
                                </menu>
                            </menuItem>
                        </items>
                    </menu>
                </menuItem>
                <menuItem title="View" id="H8h-7b-M4v">
                    <modifierMask key="keyEquivalentModifierMask"/>
                    <menu key="submenu" title="View" id="HyV-fh-RgO">
                        <items>
                            <menuItem title="Enter Full Screen" keyEquivalent="f" id="4J7-dP-txa">
```

```
                                <modifierMask key="keyEquivalentModifierMask" control="YES" command="YES"/>
                                <connections>
                                    <action selector="toggleFullScreen:" target="-1" id="dU3-MA-1Rq"/>
                                </connections>
                            </menuItem>
                        </items>
                    </menu>
                </menuItem>
                <menuItem title="Window" id="aUF-d1-5bR">
                    <modifierMask key="keyEquivalentModifierMask"/>
                    <menu key="submenu" title="Window" systemMenu="window" id="Td7-aD-5lo">
                        <items>
                            <menuItem title="Minimize" keyEquivalent="m" id="OY7-WF-poV">
                                <connections>
                                    <action selector="performMiniaturize:" target="-1" id="VwT-WD-YPe"/>
                                </connections>
                            </menuItem>
                            <menuItem title="Zoom" id="R4o-n2-Eq4">
                                <modifierMask key="keyEquivalentModifierMask"/>
                                <connections>
                                    <action selector="performZoom:" target="-1" id="DIl-cC-cCs"/>
                                </connections>
                            </menuItem>
                            <menuItem isSeparatorItem="YES" id="eu3-7i-yIM"/>
                            <menuItem title="Bring All to Front" id="LE2-aR-0XJ">
                                <modifierMask key="keyEquivalentModifierMask"/>
                                <connections>
                                    <action selector="arrangeInFront:" target="-1" id="DRN-fu-gQh"/>
                                </connections>
                            </menuItem>
                        </items>
                    </menu>
                </menuItem>
                <menuItem title="Help" id="EPT-qC-fAb">
                    <modifierMask key="keyEquivalentModifierMask"/>
                    <menu key="submenu" title="Help" systemMenu="help" id="rJ0-wn-3NY"/>
                </menuItem>
            </items>
            <point key="canvasLocation" x="142" y="-258"/>
        </menu>
        <window title="APP_NAME" allowsToolTipsWhenApplicationIsInactive="NO" autorecalculatesKeyViewLoop="NO" releasedWhenClosed="NO" animationBehavior="default" id="QvC-M9-y7g" customClass="MainFlutterWindow" customModule="Runner" customModuleProvider="target">
            <windowStyleMask key="styleMask" titled="YES" closable="YES" miniaturizable="YES" resizable="YES"/>
            <rect key="contentRect" x="335" y="390" width="800" height="600"/>
            <rect key="screenRect" x="0.0" y="0.0" width="2560" height="1577"/>
            <view key="contentView" wantsLayer="YES" id="EiT-Mj-1SZ">
                <rect key="frame" x="0.0" y="0.0" width="800" height="600"/>
                <autoresizingMask key="autoresizingMask"/>
            </view>
        </window>
    </objects>
</document>
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\macos\Runner\Configs\AppInfo.xcconfig

```
// Application-level settings for the Runner target.
//
// This may be replaced with something auto-generated from metadata (e.g., pubspec.yaml) in the
// future. If not, the values below would default to using the project name when this becomes a
// 'flutter create' template.
// The application's name. By default this is also the title of the Flutter window.
PRODUCT_NAME = wismon_keuangan

// The application's bundle identifier
PRODUCT_BUNDLE_IDENTIFIER = com.example.wismonKeuangan

// The copyright displayed in application information
PRODUCT_COPYRIGHT = Copyright © 2025 com.example. All rights reserved.
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\macos\Runner\Configs\Debug.xcconfig

```
#include "../../Flutter/Flutter-Debug.xcconfig"
#include "Warnings.xcconfig"
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\macos\Runner\Configs\Release.xcconfig

```
#include "../../Flutter/Flutter-Release.xcconfig"
#include "Warnings.xcconfig"
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\macos\Runner\Configs\Warnings.xcconfig

```
WARNING_CFLAGS = -Wall -Wconditional-uninitialized -Wnullable-to-nonnull-conversion
-Wmissing-method-return-type -Woverlength-strings
GCC_WARN_UNDECLARED_SELECTOR = YES
CLANG_UNDEFINED_BEHAVIOR_SANITIZER_NULLABILITY = YES
CLANG_WARN_UNGUARDED_AVAILABILITY = YES_AGGRESSIVE
CLANG_WARN__DUPLICATE_METHOD_MATCH = YES
CLANG_WARN_PRAGMA_PACK = YES
CLANG_WARN_STRICT_PROTOTYPES = YES
CLANG_WARN_COMMA = YES
GCC_WARN_STRICT_SELECTOR_MATCH = YES
CLANG_WARN_OBJC_REPEATED_USE_OF_WEAK = YES
CLANG_WARN_OBJC_IMPLICIT_RETAIN_SELF = YES
GCC_WARN_SHADOW = YES
CLANG_WARN_UNREACHABLE_CODE = YES
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\macos\Runner.xcodeproj\project.pbxproj

```
// !$*UTF8*$!
{
 archiveVersion = 1;
 classes = {
 };
 objectVersion = 54;
 objects = {

/* Begin PBXAggregateTarget section */
  33CC111A2044C6BA0003C045 /* Flutter Assemble */ = {
   isa = PBXAggregateTarget;
   buildConfigurationList = 33CC111B2044C6BA0003C045 /* Build configuration list for PBXAggregateTarget
"Flutter Assemble" */;
   buildPhases = (
    33CC111E2044C6BF0003C045 /* ShellScript */,
   );
   dependencies = (
   );
   name = "Flutter Assemble";
   productName = FLX;
  };
/* End PBXAggregateTarget section */

/* Begin PBXBuildFile section */
  331C80D8294CF71000263BE5 /* RunnerTests.swift in Sources */ = {isa = PBXBuildFile; fileRef =
331C80D7294CF71000263BE5 /* RunnerTests.swift */; };
  335BBD1B22A9A15E00E9071D /* GeneratedPluginRegistrant.swift in Sources */ = {isa = PBXBuildFile; fileRef =
335BBD1A22A9A15E00E9071D /* GeneratedPluginRegistrant.swift */; };
  33CC10F12044A3C60003C045 /* AppDelegate.swift in Sources */ = {isa = PBXBuildFile; fileRef =
33CC10F02044A3C60003C045 /* AppDelegate.swift */; };
  33CC10F32044A3C60003C045 /* Assets.xcassets in Resources */ = {isa = PBXBuildFile; fileRef =
33CC10F22044A3C60003C045 /* Assets.xcassets */; };
  33CC10F62044A3C60003C045 /* MainMenu.xib in Resources */ = {isa = PBXBuildFile; fileRef =
33CC10F42044A3C60003C045 /* MainMenu.xib */; };
  33CC11132044BFA00003C045 /* MainFlutterWindow.swift in Sources */ = {isa = PBXBuildFile; fileRef =
33CC11122044BFA00003C045 /* MainFlutterWindow.swift */; };
/* End PBXBuildFile section */

/* Begin PBXContainerItemProxy section */
  331C80D9294CF71000263BE5 /* PBXContainerItemProxy */ = {
   isa = PBXContainerItemProxy;
   containerPortal = 33CC10E52044A3C60003C045 /* Project object */;
   proxyType = 1;
   remoteGlobalIDString = 33CC10EC2044A3C60003C045;
   remoteInfo = Runner;
  };
  33CC111F2044C79F0003C045 /* PBXContainerItemProxy */ = {
   isa = PBXContainerItemProxy;
   containerPortal = 33CC10E52044A3C60003C045 /* Project object */;
   proxyType = 1;
   remoteGlobalIDString = 33CC111A2044C6BA0003C045;
   remoteInfo = FLX;
  };
/* End PBXContainerItemProxy section */

/* Begin PBXCopyFilesBuildPhase section */
  33CC110E2044A8840003C045 /* Bundle Framework */ = {
   isa = PBXCopyFilesBuildPhase;
   buildActionMask = 2147483647;
   dstPath = "";
   dstSubfolderSpec = 10;
   files = (
   );
   name = "Bundle Framework";
   runOnlyForDeploymentPostprocessing = 0;
  };
/* End PBXCopyFilesBuildPhase section */

/* Begin PBXFileReference section */
  331C80D5294CF71000263BE5 /* RunnerTests.xctest */ = {isa = PBXFileReference; explicitFileType =
wrapper.cfbundle; includeInIndex = 0; path = RunnerTests.xctest; sourceTree = BUILT_PRODUCTS_DIR; };
  331C80D7294CF71000263BE5 /* RunnerTests.swift */ = {isa = PBXFileReference; lastKnownFileType =
sourcecode.swift; path = RunnerTests.swift; sourceTree = "<group>"; };
  333000ED22D3DE5D00554162 /* Warnings.xcconfig */ = {isa = PBXFileReference; lastKnownFileType =
text.xcconfig; path = Warnings.xcconfig; sourceTree = "<group>"; };
  335BBD1A22A9A15E00E9071D /* GeneratedPluginRegistrant.swift */ = {isa = PBXFileReference; fileEncoding = 4;
lastKnownFileType = sourcecode.swift; path = GeneratedPluginRegistrant.swift; sourceTree = "<group>"; };
```

```
  33CC10ED2044A3C60003C045 /* wismon_keuangan.app */ = {isa = PBXFileReference; explicitFileType = wrapper.application; includeInIndex = 0; path = "wismon_keuangan.app"; sourceTree = BUILT_PRODUCTS_DIR; };
  33CC10F02044A3C60003C045 /* AppDelegate.swift */ = {isa = PBXFileReference; lastKnownFileType = sourcecode.swift; path = AppDelegate.swift; sourceTree = "<group>"; };
  33CC10F22044A3C60003C045 /* Assets.xcassets */ = {isa = PBXFileReference; lastKnownFileType = folder.assetcatalog; name = Assets.xcassets; path = Runner/Assets.xcassets; sourceTree = "<group>"; };
  33CC10F52044A3C60003C045 /* Base */ = {isa = PBXFileReference; lastKnownFileType = file.xib; name = Base; path = Base.lproj/MainMenu.xib; sourceTree = "<group>"; };
  33CC10F72044A3C60003C045 /* Info.plist */ = {isa = PBXFileReference; lastKnownFileType = text.plist.xml; name = Info.plist; path = Runner/Info.plist; sourceTree = "<group>"; };
  33CC11122044BFA00003C045 /* MainFlutterWindow.swift */ = {isa = PBXFileReference; lastKnownFileType = sourcecode.swift; path = MainFlutterWindow.swift; sourceTree = "<group>"; };
  33CEB47222A05771004F2AC0 /* Flutter-Debug.xcconfig */ = {isa = PBXFileReference; lastKnownFileType = text.xcconfig; path = "Flutter-Debug.xcconfig"; sourceTree = "<group>"; };
  33CEB47422A05771004F2AC0 /* Flutter-Release.xcconfig */ = {isa = PBXFileReference; lastKnownFileType = text.xcconfig; path = "Flutter-Release.xcconfig"; sourceTree = "<group>"; };
  33CEB47722A0578A004F2AC0 /* Flutter-Generated.xcconfig */ = {isa = PBXFileReference; lastKnownFileType = text.xcconfig; name = "Flutter-Generated.xcconfig"; path = "ephemeral/Flutter-Generated.xcconfig"; sourceTree = "<group>"; };
  33E51913231747F40026EE4D /* DebugProfile.entitlements */ = {isa = PBXFileReference; lastKnownFileType = text.plist.entitlements; path = DebugProfile.entitlements; sourceTree = "<group>"; };
  33E51914231749380026EE4D /* Release.entitlements */ = {isa = PBXFileReference; fileEncoding = 4; lastKnownFileType = text.plist.entitlements; path = Release.entitlements; sourceTree = "<group>"; };
  33E5194F232828860026EE4D /* AppInfo.xcconfig */ = {isa = PBXFileReference; lastKnownFileType = text.xcconfig; path = AppInfo.xcconfig; sourceTree = "<group>"; };
  7AFA3C8E1D35360C0083082E /* Release.xcconfig */ = {isa = PBXFileReference; lastKnownFileType = text.xcconfig; path = Release.xcconfig; sourceTree = "<group>"; };
  9740EEB21CF90195004384FC /* Debug.xcconfig */ = {isa = PBXFileReference; fileEncoding = 4; lastKnownFileType = text.xcconfig; path = Debug.xcconfig; sourceTree = "<group>"; };
/* End PBXFileReference section */

/* Begin PBXFrameworksBuildPhase section */
  331C80D2294CF70F00263BE5 /* Frameworks */ = {
   isa = PBXFrameworksBuildPhase;
   buildActionMask = 2147483647;
   files = (
   );
   runOnlyForDeploymentPostprocessing = 0;
  };
  33CC10EA2044A3C60003C045 /* Frameworks */ = {
   isa = PBXFrameworksBuildPhase;
   buildActionMask = 2147483647;
   files = (
   );
   runOnlyForDeploymentPostprocessing = 0;
  };
/* End PBXFrameworksBuildPhase section */

/* Begin PBXGroup section */
  331C80D6294CF71000263BE5 /* RunnerTests */ = {
   isa = PBXGroup;
   children = (
    331C80D7294CF71000263BE5 /* RunnerTests.swift */,
   );
   path = RunnerTests;
   sourceTree = "<group>";
  };
  33BA886A226E78AF003329D5 /* Configs */ = {
   isa = PBXGroup;
   children = (
    33E5194F232828860026EE4D /* AppInfo.xcconfig */,
    9740EEB21CF90195004384FC /* Debug.xcconfig */,
    7AFA3C8E1D35360C0083082E /* Release.xcconfig */,
    333000ED22D3DE5D00554162 /* Warnings.xcconfig */,
   );
   path = Configs;
   sourceTree = "<group>";
  };
  33CC10E42044A3C60003C045 = {
   isa = PBXGroup;
   children = (
    33FAB671232836740065AC1E /* Runner */,
    33CEB47122A05771004F2AC0 /* Flutter */,
    331C80D6294CF71000263BE5 /* RunnerTests */,
    33CC10EE2044A3C60003C045 /* Products */,
    D73912EC22F37F3D000D13A0 /* Frameworks */,
   );
   sourceTree = "<group>";
  };
  33CC10EE2044A3C60003C045 /* Products */ = {
   isa = PBXGroup;
   children = (
    33CC10ED2044A3C60003C045 /* wismon_keuangan.app */,
    331C80D5294CF71000263BE5 /* RunnerTests.xctest */,
   );
   name = Products;
   sourceTree = "<group>";
  };
  33CC11242044D66E0003C045 /* Resources */ = {
   isa = PBXGroup;
   children = (
    33CC10F22044A3C60003C045 /* Assets.xcassets */,
    33CC10F42044A3C60003C045 /* MainMenu.xib */,
    33CC10F72044A3C60003C045 /* Info.plist */,
   );
   name = Resources;
   path = ..;
   sourceTree = "<group>";
  };
  33CEB47122A05771004F2AC0 /* Flutter */ = {
   isa = PBXGroup;
   children = (
    335BBD1A22A9A15E00E9071D /* GeneratedPluginRegistrant.swift */,
    33CEB47222A05771004F2AC0 /* Flutter-Debug.xcconfig */,
    33CEB47422A05771004F2AC0 /* Flutter-Release.xcconfig */,
    33CEB47722A0578A004F2AC0 /* Flutter-Generated.xcconfig */,
   );
   path = Flutter;
   sourceTree = "<group>";
  };
  33FAB671232836740065AC1E /* Runner */ = {
   isa = PBXGroup;
   children = (
    33CC10F02044A3C60003C045 /* AppDelegate.swift */,
    33CC11122044BFA00003C045 /* MainFlutterWindow.swift */,
    33E51913231747F40026EE4D /* DebugProfile.entitlements */,
    33E51914231749380026EE4D /* Release.entitlements */,
    33CC11242044D66E0003C045 /* Resources */,
    33BA886A226E78AF003329D5 /* Configs */,
   );
   path = Runner;
   sourceTree = "<group>";
  };
  D73912EC22F37F3D000D13A0 /* Frameworks */ = {
   isa = PBXGroup;
   children = (
   );
   name = Frameworks;
   sourceTree = "<group>";
  };
/* End PBXGroup section */
/* Begin PBXNativeTarget section */
  331C80D4294CF70F00263BE5 /* RunnerTests */ = {
   isa = PBXNativeTarget;
   buildConfigurationList = 331C80DE294CF71000263BE5 /* Build configuration list for PBXNativeTarget "RunnerTests" */;
   buildPhases = (
    331C80D1294CF70F00263BE5 /* Sources */,
    331C80D2294CF70F00263BE5 /* Frameworks */,
    331C80D3294CF70F00263BE5 /* Resources */,
   );
   buildRules = (
   );
   dependencies = (
    331C80DA294CF71000263BE5 /* PBXTargetDependency */,
   );
   name = RunnerTests;
   productName = RunnerTests;
   productReference = 331C80D5294CF71000263BE5 /* RunnerTests.xctest */;
   productType = "com.apple.product-type.bundle.unit-test";
  };
  33CC10EC2044A3C60003C045 /* Runner */ = {
   isa = PBXNativeTarget;
   buildConfigurationList = 33CC10FB2044A3C60003C045 /* Build configuration list for PBXNativeTarget "Runner" */;
   buildPhases = (
    33CC10E92044A3C60003C045 /* Sources */,
    33CC10EA2044A3C60003C045 /* Frameworks */,
    33CC10EB2044A3C60003C045 /* Resources */,
    33CC110E2044A8840003C045 /* Bundle Framework */,
    3399D490228B24CF009A79C7 /* ShellScript */,
   );
   buildRules = (
   );
   dependencies = (
    33CC11202044C79F0003C045 /* PBXTargetDependency */,
   );
   name = Runner;
   productName = Runner;
   productReference = 33CC10ED2044A3C60003C045 /* wismon_keuangan.app */;
   productType = "com.apple.product-type.application";
  };
/* End PBXNativeTarget section */

/* Begin PBXProject section */
  33CC10E52044A3C60003C045 /* Project object */ = {
   isa = PBXProject;
   attributes = {
    BuildIndependentTargetsInParallel = YES;
    LastSwiftUpdateCheck = 0920;
    LastUpgradeCheck = 1510;
    ORGANIZATIONNAME = "";
    TargetAttributes = {
     331C80D4294CF70F00263BE5 = {
      CreatedOnToolsVersion = 14.0;
      TestTargetID = 33CC10EC2044A3C60003C045;
     };
     33CC10EC2044A3C60003C045 = {
      CreatedOnToolsVersion = 9.2;
      LastSwiftMigration = 1100;
      ProvisioningStyle = Automatic;
      SystemCapabilities = {
       com.apple.Sandbox = {
        enabled = 1;
       };
      };
     };
     33CC111A2044C6BA0003C045 = {
      CreatedOnToolsVersion = 9.2;
      ProvisioningStyle = Manual;
     };
    };
   };
   buildConfigurationList = 33CC10E82044A3C60003C045 /* Build configuration list for PBXProject "Runner" */;
   compatibilityVersion = "Xcode 9.3";
   developmentRegion = en;
   hasScannedForEncodings = 0;
   knownRegions = (
    en,
    Base,
   );
   mainGroup = 33CC10E42044A3C60003C045;
   productRefGroup = 33CC10EE2044A3C60003C045 /* Products */;
   projectDirPath = "";
   projectRoot = "";
   targets = (
    33CC10EC2044A3C60003C045 /* Runner */,
    331C80D4294CF70F00263BE5 /* RunnerTests */,
    33CC111A2044C6BA0003C045 /* Flutter Assemble */,
   );
  };
/* End PBXProject section */

/* Begin PBXResourcesBuildPhase section */
  331C80D3294CF70F00263BE5 /* Resources */ = {
   isa = PBXResourcesBuildPhase;
   buildActionMask = 2147483647;
   files = (
   );
   runOnlyForDeploymentPostprocessing = 0;
  };
  33CC10EB2044A3C60003C045 /* Resources */ = {
   isa = PBXResourcesBuildPhase;
   buildActionMask = 2147483647;
   files = (
    33CC10F32044A3C60003C045 /* Assets.xcassets in Resources */,
    33CC10F62044A3C60003C045 /* MainMenu.xib in Resources */,
   );
   runOnlyForDeploymentPostprocessing = 0;
  };
/* End PBXResourcesBuildPhase section */

/* Begin PBXShellScriptBuildPhase section */
  3399D490228B24CF009A79C7 /* ShellScript */ = {
   isa = PBXShellScriptBuildPhase;
   alwaysOutOfDate = 1;
   buildActionMask = 2147483647;
   files = (
   );
   inputFileListPaths = (
   );
   inputPaths = (
   );
   outputFileListPaths = (
   );
   outputPaths = (
   );
   runOnlyForDeploymentPostprocessing = 0;
   shellPath = /bin/sh;
   shellScript = "echo \"$PRODUCT_NAME.app\" > \"$PROJECT_DIR\"/Flutter/ephemeral/.app_filename &&
\"$FLUTTER_ROOT\"/packages/flutter_tools/bin/macos_assemble.sh embed\n";
  };
  33CC111E2044C6BF0003C045 /* ShellScript */ = {
   isa = PBXShellScriptBuildPhase;
   buildActionMask = 2147483647;
   files = (
   );
   inputFileListPaths = (
    Flutter/ephemeral/FlutterInputs.xcfilelist,
   );
   inputPaths = (
    Flutter/ephemeral/tripwire,
   );
   outputFileListPaths = (
    Flutter/ephemeral/FlutterOutputs.xcfilelist,
   );
   outputPaths = (
   );
   runOnlyForDeploymentPostprocessing = 0;
   shellPath = /bin/sh;
   shellScript = "\"$FLUTTER_ROOT\"/packages/flutter_tools/bin/macos_assemble.sh && touch
Flutter/ephemeral/tripwire";
  };
/* End PBXShellScriptBuildPhase section */

/* Begin PBXSourcesBuildPhase section */
  331C80D1294CF70F00263BE5 /* Sources */ = {
   isa = PBXSourcesBuildPhase;
   buildActionMask = 2147483647;
   files = (
    331C80D8294CF71000263BE5 /* RunnerTests.swift in Sources */,
   );
   runOnlyForDeploymentPostprocessing = 0;
  };
  33CC10E92044A3C60003C045 /* Sources */ = {
   isa = PBXSourcesBuildPhase;
   buildActionMask = 2147483647;
   files = (
    33CC11132044BFA00003C045 /* MainFlutterWindow.swift in Sources */,
    33CC10F12044A3C60003C045 /* AppDelegate.swift in Sources */,
    335BBD1B22A9A15E00E9071D /* GeneratedPluginRegistrant.swift in Sources */,
   );
   runOnlyForDeploymentPostprocessing = 0;
  };
/* End PBXSourcesBuildPhase section */

/* Begin PBXTargetDependency section */
  331C80DA294CF71000263BE5 /* PBXTargetDependency */ = {
   isa = PBXTargetDependency;
   target = 33CC10EC2044A3C60003C045 /* Runner */;
   targetProxy = 331C80D9294CF71000263BE5 /* PBXContainerItemProxy */;
  };
  33CC11202044C79F0003C045 /* PBXTargetDependency */ = {
   isa = PBXTargetDependency;
   target = 33CC111A2044C6BA0003C045 /* Flutter Assemble */;
   targetProxy = 33CC111F2044C79F0003C045 /* PBXContainerItemProxy */;
  };
/* End PBXTargetDependency section */

/* Begin PBXVariantGroup section */
  33CC10F42044A3C60003C045 /* MainMenu.xib */ = {
   isa = PBXVariantGroup;
   children = (
    33CC10F52044A3C60003C045 /* Base */,
   );
   name = MainMenu.xib;
   path = Runner;
   sourceTree = "<group>";
  };
```

```
/* End PBXVariantGroup section */

/* Begin XCBuildConfiguration section */
  331C80DB294CF71000263BE5 /* Debug */ = {
   isa = XCBuildConfiguration;
   buildSettings = {
    BUNDLE_LOADER = "$(TEST_HOST)";
    CURRENT_PROJECT_VERSION = 1;
    GENERATE_INFOPLIST_FILE = YES;
    MARKETING_VERSION = 1.0;
    PRODUCT_BUNDLE_IDENTIFIER = com.example.wismonKeuangan.RunnerTests;
    PRODUCT_NAME = "$(TARGET_NAME)";
    SWIFT_VERSION = 5.0;
    TEST_HOST = "$(BUILT_PRODUCTS_DIR)/wismon_keuangan.app/$(BUNDLE_EXECUTABLE_FOLDER_PATH)/wismon_keuangan";
   };
   name = Debug;
  };
  331C80DC294CF71000263BE5 /* Release */ = {
   isa = XCBuildConfiguration;
   buildSettings = {
    BUNDLE_LOADER = "$(TEST_HOST)";
    CURRENT_PROJECT_VERSION = 1;
    GENERATE_INFOPLIST_FILE = YES;
    MARKETING_VERSION = 1.0;
    PRODUCT_BUNDLE_IDENTIFIER = com.example.wismonKeuangan.RunnerTests;
    PRODUCT_NAME = "$(TARGET_NAME)";
    SWIFT_VERSION = 5.0;
    TEST_HOST = "$(BUILT_PRODUCTS_DIR)/wismon_keuangan.app/$(BUNDLE_EXECUTABLE_FOLDER_PATH)/wismon_keuangan";
   };
   name = Release;
  };
  331C80DD294CF71000263BE5 /* Profile */ = {
   isa = XCBuildConfiguration;
   buildSettings = {
    BUNDLE_LOADER = "$(TEST_HOST)";
    CURRENT_PROJECT_VERSION = 1;
    GENERATE_INFOPLIST_FILE = YES;
    MARKETING_VERSION = 1.0;
    PRODUCT_BUNDLE_IDENTIFIER = com.example.wismonKeuangan.RunnerTests;
    PRODUCT_NAME = "$(TARGET_NAME)";
    SWIFT_VERSION = 5.0;
    TEST_HOST = "$(BUILT_PRODUCTS_DIR)/wismon_keuangan.app/$(BUNDLE_EXECUTABLE_FOLDER_PATH)/wismon_keuangan";
   };
   name = Profile;
  };
  338D0CE9231458BD00FA5F75 /* Profile */ = {
   isa = XCBuildConfiguration;
   baseConfigurationReference = 7AFA3C8E1D35360C0083082E /* Release.xcconfig */;
   buildSettings = {
    ALWAYS_SEARCH_USER_PATHS = NO;
    ASSETCATALOG_COMPILER_GENERATE_SWIFT_ASSET_SYMBOL_EXTENSIONS = YES;
    CLANG_ANALYZER_NONNULL = YES;
    CLANG_ANALYZER_NUMBER_OBJECT_CONVERSION = YES_AGGRESSIVE;
    CLANG_CXX_LANGUAGE_STANDARD = "gnu++14";
    CLANG_CXX_LIBRARY = "libc++";
    CLANG_ENABLE_MODULES = YES;
    CLANG_ENABLE_OBJC_ARC = YES;
    CLANG_WARN_BLOCK_CAPTURE_AUTORELEASING = YES;
    CLANG_WARN_BOOL_CONVERSION = YES;
    CLANG_WARN_CONSTANT_CONVERSION = YES;
    CLANG_WARN_DEPRECATED_OBJC_IMPLEMENTATIONS = YES;
    CLANG_WARN_DIRECT_OBJC_ISA_USAGE = YES_ERROR;
    CLANG_WARN_DOCUMENTATION_COMMENTS = YES;
    CLANG_WARN_EMPTY_BODY = YES;
    CLANG_WARN_ENUM_CONVERSION = YES;
    CLANG_WARN_INFINITE_RECURSION = YES;
    CLANG_WARN_INT_CONVERSION = YES;
    CLANG_WARN_NON_LITERAL_NULL_CONVERSION = YES;
    CLANG_WARN_OBJC_LITERAL_CONVERSION = YES;
    CLANG_WARN_OBJC_ROOT_CLASS = YES_ERROR;
    CLANG_WARN_RANGE_LOOP_ANALYSIS = YES;
    CLANG_WARN_SUSPICIOUS_MOVE = YES;
    CODE_SIGN_IDENTITY = "-";
    COPY_PHASE_STRIP = NO;
    DEAD_CODE_STRIPPING = YES;
    DEBUG_INFORMATION_FORMAT = "dwarf-with-dsym";
    ENABLE_NS_ASSERTIONS = NO;
    ENABLE_STRICT_OBJC_MSGSEND = YES;
    ENABLE_USER_SCRIPT_SANDBOXING = NO;
    GCC_C_LANGUAGE_STANDARD = gnu11;
    GCC_NO_COMMON_BLOCKS = YES;
    GCC_WARN_64_TO_32_BIT_CONVERSION = YES;
    GCC_WARN_ABOUT_RETURN_TYPE = YES_ERROR;
    GCC_WARN_UNINITIALIZED_AUTOS = YES_AGGRESSIVE;
    GCC_WARN_UNUSED_FUNCTION = YES;
    GCC_WARN_UNUSED_VARIABLE = YES;
    MACOSX_DEPLOYMENT_TARGET = 10.14;
    MTL_ENABLE_DEBUG_INFO = NO;
    SDKROOT = macosx;
    SWIFT_COMPILATION_MODE = wholemodule;
    SWIFT_OPTIMIZATION_LEVEL = "-O";
   };
   name = Profile;
  };
  338D0CEA231458BD00FA5F75 /* Profile */ = {
   isa = XCBuildConfiguration;
   baseConfigurationReference = 33E5194F232828860026EE4D /* AppInfo.xcconfig */;
   buildSettings = {
    ASSETCATALOG_COMPILER_APPICON_NAME = AppIcon;
    CLANG_ENABLE_MODULES = YES;
    CODE_SIGN_ENTITLEMENTS = Runner/DebugProfile.entitlements;
    CODE_SIGN_STYLE = Automatic;
    COMBINE_HIDPI_IMAGES = YES;
    INFOPLIST_FILE = Runner/Info.plist;
    LD_RUNPATH_SEARCH_PATHS = (
     "$(inherited)",
     "@executable_path/../Frameworks",
    );
    PROVISIONING_PROFILE_SPECIFIER = "";
    SWIFT_VERSION = 5.0;
   };
   name = Profile;
  };
  338D0CEB231458BD00FA5F75 /* Profile */ = {
   isa = XCBuildConfiguration;
   buildSettings = {
    CODE_SIGN_STYLE = Manual;
    PRODUCT_NAME = "$(TARGET_NAME)";
   };
   name = Profile;
  };
  33CC10F92044A3C60003C045 /* Debug */ = {
   isa = XCBuildConfiguration;
   baseConfigurationReference = 9740EEB21CF90195004384FC /* Debug.xcconfig */;
   buildSettings = {
    ALWAYS_SEARCH_USER_PATHS = NO;
    ASSETCATALOG_COMPILER_GENERATE_SWIFT_ASSET_SYMBOL_EXTENSIONS = YES;
    CLANG_ANALYZER_NONNULL = YES;
    CLANG_ANALYZER_NUMBER_OBJECT_CONVERSION = YES_AGGRESSIVE;
    CLANG_CXX_LANGUAGE_STANDARD = "gnu++14";
    CLANG_CXX_LIBRARY = "libc++";
    CLANG_ENABLE_MODULES = YES;
    CLANG_ENABLE_OBJC_ARC = YES;
    CLANG_WARN_BLOCK_CAPTURE_AUTORELEASING = YES;
    CLANG_WARN_BOOL_CONVERSION = YES;
    CLANG_WARN_CONSTANT_CONVERSION = YES;
    CLANG_WARN_DEPRECATED_OBJC_IMPLEMENTATIONS = YES;
    CLANG_WARN_DIRECT_OBJC_ISA_USAGE = YES_ERROR;
    CLANG_WARN_DOCUMENTATION_COMMENTS = YES;
    CLANG_WARN_EMPTY_BODY = YES;
    CLANG_WARN_ENUM_CONVERSION = YES;
    CLANG_WARN_INFINITE_RECURSION = YES;
    CLANG_WARN_INT_CONVERSION = YES;
    CLANG_WARN_NON_LITERAL_NULL_CONVERSION = YES;
    CLANG_WARN_OBJC_LITERAL_CONVERSION = YES;
    CLANG_WARN_OBJC_ROOT_CLASS = YES_ERROR;
    CLANG_WARN_RANGE_LOOP_ANALYSIS = YES;
    CLANG_WARN_SUSPICIOUS_MOVE = YES;
    CODE_SIGN_IDENTITY = "-";
    COPY_PHASE_STRIP = NO;
    DEAD_CODE_STRIPPING = YES;
    DEBUG_INFORMATION_FORMAT = dwarf;
    ENABLE_STRICT_OBJC_MSGSEND = YES;
    ENABLE_TESTABILITY = YES;
    ENABLE_USER_SCRIPT_SANDBOXING = NO;
    GCC_C_LANGUAGE_STANDARD = gnu11;
    GCC_DYNAMIC_NO_PIC = NO;
    GCC_NO_COMMON_BLOCKS = YES;
    GCC_OPTIMIZATION_LEVEL = 0;
    GCC_PREPROCESSOR_DEFINITIONS = (
     "DEBUG=1",
     "$(inherited)",
    );
    GCC_WARN_64_TO_32_BIT_CONVERSION = YES;
    GCC_WARN_ABOUT_RETURN_TYPE = YES_ERROR;
    GCC_WARN_UNINITIALIZED_AUTOS = YES_AGGRESSIVE;
    GCC_WARN_UNUSED_FUNCTION = YES;
    GCC_WARN_UNUSED_VARIABLE = YES;
    MACOSX_DEPLOYMENT_TARGET = 10.14;
    MTL_ENABLE_DEBUG_INFO = YES;
    ONLY_ACTIVE_ARCH = YES;
    SDKROOT = macosx;
    SWIFT_ACTIVE_COMPILATION_CONDITIONS = DEBUG;
    SWIFT_OPTIMIZATION_LEVEL = "-Onone";
   };
   name = Debug;
  };
  33CC10FA2044A3C60003C045 /* Release */ = {
   isa = XCBuildConfiguration;
   baseConfigurationReference = 7AFA3C8E1D35360C0083082E /* Release.xcconfig */;
   buildSettings = {
    ALWAYS_SEARCH_USER_PATHS = NO;
    ASSETCATALOG_COMPILER_GENERATE_SWIFT_ASSET_SYMBOL_EXTENSIONS = YES;
    CLANG_ANALYZER_NONNULL = YES;
    CLANG_ANALYZER_NUMBER_OBJECT_CONVERSION = YES_AGGRESSIVE;
    CLANG_CXX_LANGUAGE_STANDARD = "gnu++14";
    CLANG_CXX_LIBRARY = "libc++";
    CLANG_ENABLE_MODULES = YES;
    CLANG_ENABLE_OBJC_ARC = YES;
    CLANG_WARN_BLOCK_CAPTURE_AUTORELEASING = YES;
    CLANG_WARN_BOOL_CONVERSION = YES;
    CLANG_WARN_CONSTANT_CONVERSION = YES;
    CLANG_WARN_DEPRECATED_OBJC_IMPLEMENTATIONS = YES;
    CLANG_WARN_DIRECT_OBJC_ISA_USAGE = YES_ERROR;
    CLANG_WARN_DOCUMENTATION_COMMENTS = YES;
    CLANG_WARN_EMPTY_BODY = YES;
    CLANG_WARN_ENUM_CONVERSION = YES;
    CLANG_WARN_INFINITE_RECURSION = YES;
    CLANG_WARN_INT_CONVERSION = YES;
    CLANG_WARN_NON_LITERAL_NULL_CONVERSION = YES;
    CLANG_WARN_OBJC_LITERAL_CONVERSION = YES;
    CLANG_WARN_OBJC_ROOT_CLASS = YES_ERROR;
    CLANG_WARN_RANGE_LOOP_ANALYSIS = YES;
    CLANG_WARN_SUSPICIOUS_MOVE = YES;
    CODE_SIGN_IDENTITY = "-";
    COPY_PHASE_STRIP = NO;
    DEAD_CODE_STRIPPING = YES;
    DEBUG_INFORMATION_FORMAT = "dwarf-with-dsym";
    ENABLE_NS_ASSERTIONS = NO;
    ENABLE_STRICT_OBJC_MSGSEND = YES;
    ENABLE_USER_SCRIPT_SANDBOXING = NO;
    GCC_C_LANGUAGE_STANDARD = gnu11;
    GCC_NO_COMMON_BLOCKS = YES;
    GCC_WARN_64_TO_32_BIT_CONVERSION = YES;
    GCC_WARN_ABOUT_RETURN_TYPE = YES_ERROR;
    GCC_WARN_UNINITIALIZED_AUTOS = YES_AGGRESSIVE;
    GCC_WARN_UNUSED_FUNCTION = YES;
    GCC_WARN_UNUSED_VARIABLE = YES;
    MACOSX_DEPLOYMENT_TARGET = 10.14;
    MTL_ENABLE_DEBUG_INFO = NO;
    SDKROOT = macosx;
    SWIFT_COMPILATION_MODE = wholemodule;
    SWIFT_OPTIMIZATION_LEVEL = "-O";
   };
   name = Release;
  };
  33CC10FC2044A3C60003C045 /* Debug */ = {
   isa = XCBuildConfiguration;
   baseConfigurationReference = 33E5194F232828860026EE4D /* AppInfo.xcconfig */;
   buildSettings = {
    ASSETCATALOG_COMPILER_APPICON_NAME = AppIcon;
    CLANG_ENABLE_MODULES = YES;
    CODE_SIGN_ENTITLEMENTS = Runner/DebugProfile.entitlements;
    CODE_SIGN_STYLE = Automatic;
    COMBINE_HIDPI_IMAGES = YES;
    INFOPLIST_FILE = Runner/Info.plist;
    LD_RUNPATH_SEARCH_PATHS = (
     "$(inherited)",
     "@executable_path/../Frameworks",
    );
    PROVISIONING_PROFILE_SPECIFIER = "";
    SWIFT_OPTIMIZATION_LEVEL = "-Onone";
    SWIFT_VERSION = 5.0;
   };
   name = Debug;
  };
  33CC10FD2044A3C60003C045 /* Release */ = {
   isa = XCBuildConfiguration;
   baseConfigurationReference = 33E5194F232828860026EE4D /* AppInfo.xcconfig */;
   buildSettings = {
    ASSETCATALOG_COMPILER_APPICON_NAME = AppIcon;
    CLANG_ENABLE_MODULES = YES;
    CODE_SIGN_ENTITLEMENTS = Runner/Release.entitlements;
    CODE_SIGN_STYLE = Automatic;
    COMBINE_HIDPI_IMAGES = YES;
    INFOPLIST_FILE = Runner/Info.plist;
    LD_RUNPATH_SEARCH_PATHS = (
     "$(inherited)",
     "@executable_path/../Frameworks",
    );
    PROVISIONING_PROFILE_SPECIFIER = "";
    SWIFT_VERSION = 5.0;
   };
   name = Release;
```

```
  };
  33CC111C2044C6BA0003C045 /* Debug */ = {
   isa = XCBuildConfiguration;
   buildSettings = {
    CODE_SIGN_STYLE = Manual;
    PRODUCT_NAME = "$(TARGET_NAME)";
   };
   name = Debug;
  };
  33CC111D2044C6BA0003C045 /* Release */ = {
   isa = XCBuildConfiguration;
   buildSettings = {
    CODE_SIGN_STYLE = Automatic;
    PRODUCT_NAME = "$(TARGET_NAME)";
   };
   name = Release;
  };
/* End XCBuildConfiguration section */

/* Begin XCConfigurationList section */
  331C80DE294CF71000263BE5 /* Build configuration list for PBXNativeTarget "RunnerTests" */ = {
   isa = XCConfigurationList;
   buildConfigurations = (
    331C80DB294CF71000263BE5 /* Debug */,
    331C80DC294CF71000263BE5 /* Release */,
    331C80DD294CF71000263BE5 /* Profile */,
   );
   defaultConfigurationIsVisible = 0;
   defaultConfigurationName = Release;
  };
  33CC10E82044A3C60003C045 /* Build configuration list for PBXProject "Runner" */ = {
   isa = XCConfigurationList;
   buildConfigurations = (
    33CC10F92044A3C60003C045 /* Debug */,
    33CC10FA2044A3C60003C045 /* Release */,
    338D0CE9231458BD00FA5F75 /* Profile */,
   );
   defaultConfigurationIsVisible = 0;
   defaultConfigurationName = Release;
  };
  33CC10FB2044A3C60003C045 /* Build configuration list for PBXNativeTarget "Runner" */ = {
   isa = XCConfigurationList;
   buildConfigurations = (
    33CC10FC2044A3C60003C045 /* Debug */,
    33CC10FD2044A3C60003C045 /* Release */,
    338D0CEA231458BD00FA5F75 /* Profile */,
   );
   defaultConfigurationIsVisible = 0;
   defaultConfigurationName = Release;
  };
  33CC111B2044C6BA0003C045 /* Build configuration list for PBXAggregateTarget "Flutter Assemble" */ = {
   isa = XCConfigurationList;
   buildConfigurations = (
    33CC111C2044C6BA0003C045 /* Debug */,
    33CC111D2044C6BA0003C045 /* Release */,
    338D0CEB231458BD00FA5F75 /* Profile */,
   );
   defaultConfigurationIsVisible = 0;
   defaultConfigurationName = Release;
  };
/* End XCConfigurationList section */
 };
 rootObject = 33CC10E52044A3C60003C045 /* Project object */;
}
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\macos\Runner.xcodeproj\project.xcworkspace\xcshareddata\IDEWorkspaceChecks.plist

```
<¿xml version="1.0" encoding="UTF-8"¿>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
 <key>IDEDidComputeMac32BitWarning</key>
 <true/>
</dict>
</plist>
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\macos\Runner.xcodeproj\xcshareddata\xcschemes\Runner.xcscheme

```
<¿xml version="1.0" encoding="UTF-8"¿>
<Scheme
   LastUpgradeVersion = "1510"
   version = "1.3">
   <BuildAction
      parallelizeBuildables = "YES"
      buildImplicitDependencies = "YES">
      <BuildActionEntries>
         <BuildActionEntry
            buildForTesting = "YES"
            buildForRunning = "YES"
            buildForProfiling = "YES"
            buildForArchiving = "YES"
            buildForAnalyzing = "YES">
            <BuildableReference
               BuildableIdentifier = "primary"
               BlueprintIdentifier = "33CC10EC2044A3C60003C045"
               BuildableName = "wismon_keuangan.app"
               BlueprintName = "Runner"
               ReferencedContainer = "container:Runner.xcodeproj">
            </BuildableReference>
         </BuildActionEntry>
      </BuildActionEntries>
   </BuildAction>
   <TestAction
      buildConfiguration = "Debug"
      selectedDebuggerIdentifier = "Xcode.DebuggerFoundation.Debugger.LLDB"
      selectedLauncherIdentifier = "Xcode.DebuggerFoundation.Launcher.LLDB"
      shouldUseLaunchSchemeArgsEnv = "YES">
      <MacroExpansion>
         <BuildableReference
            BuildableIdentifier = "primary"
            BlueprintIdentifier = "33CC10EC2044A3C60003C045"
            BuildableName = "wismon_keuangan.app"
            BlueprintName = "Runner"
            ReferencedContainer = "container:Runner.xcodeproj">
         </BuildableReference>
      </MacroExpansion>
      <Testables>
         <TestableReference
            skipped = "NO"
            parallelizable = "YES">
            <BuildableReference
               BuildableIdentifier = "primary"
               BlueprintIdentifier = "331C80D4294CF70F00263BE5"
               BuildableName = "RunnerTests.xctest"
               BlueprintName = "RunnerTests"
               ReferencedContainer = "container:Runner.xcodeproj">
            </BuildableReference>
         </TestableReference>
      </Testables>
   </TestAction>
   <LaunchAction
      buildConfiguration = "Debug"
      selectedDebuggerIdentifier = "Xcode.DebuggerFoundation.Debugger.LLDB"
      selectedLauncherIdentifier = "Xcode.DebuggerFoundation.Launcher.LLDB"
      launchStyle = "0"
      useCustomWorkingDirectory = "NO"
      ignoresPersistentStateOnLaunch = "NO"
      debugDocumentVersioning = "YES"
      debugServiceExtension = "internal"
      enableGPUValidationMode = "1"
      allowLocationSimulation = "YES">
      <BuildableProductRunnable
         runnableDebuggingMode = "0">
         <BuildableReference
            BuildableIdentifier = "primary"
            BlueprintIdentifier = "33CC10EC2044A3C60003C045"
            BuildableName = "wismon_keuangan.app"
            BlueprintName = "Runner"
            ReferencedContainer = "container:Runner.xcodeproj">
         </BuildableReference>
      </BuildableProductRunnable>
   </LaunchAction>
   <ProfileAction
      buildConfiguration = "Profile"
      shouldUseLaunchSchemeArgsEnv = "YES"
      savedToolIdentifier = ""
      useCustomWorkingDirectory = "NO"
      debugDocumentVersioning = "YES">
      <BuildableProductRunnable
         runnableDebuggingMode = "0">
         <BuildableReference
            BuildableIdentifier = "primary"
            BlueprintIdentifier = "33CC10EC2044A3C60003C045"
            BuildableName = "wismon_keuangan.app"
            BlueprintName = "Runner"
            ReferencedContainer = "container:Runner.xcodeproj">
         </BuildableReference>
      </BuildableProductRunnable>
   </ProfileAction>
   <AnalyzeAction
      buildConfiguration = "Debug">
   </AnalyzeAction>
   <ArchiveAction
      buildConfiguration = "Release"
      revealArchiveInOrganizer = "YES">
   </ArchiveAction>
</Scheme>
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\macos\Runner.xcworkspace\contents.xcwor

kspacedata

```
<¿xml version="1.0" encoding="UTF-8"¿>
<Workspace
   version = "1.0">
   <FileRef
      location = "group:Runner.xcodeproj">
   </FileRef>
</Workspace>
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\macos\Runner.xcworkspace\xcshareddata\IDEWorkspaceChecks.plist

```
<¿xml version="1.0" encoding="UTF-8"¿>
<!DOCTYPE plist PUBLIC "-//Apple//DTD PLIST 1.0//EN" "http://www.apple.com/DTDs/PropertyList-1.0.dtd">
<plist version="1.0">
<dict>
 <key>IDEDidComputeMac32BitWarning</key>
 <true/>
</dict>
</plist>
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\macos\RunnerTests\RunnerTests.swift

```
import Cocoa
import FlutterMacOS
import XCTest

class RunnerTests: XCTestCase {

  func testExample() {
    // If you add code to the Runner application, consider adding tests here.
    // See https://developer.apple.com/documentation/xctest for more information about using XCTest.
  }

}
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\test\widget_test.dart

```
// This is a basic Flutter widget test.
//
// To perform an interaction with a widget in your test, use the WidgetTester
// utility in the flutter_test package. For example, you can send tap and scroll
// gestures. You can also use WidgetTester to find child widgets in the widget
// tree, read text, and verify that the values of widget properties are correct.

import 'package:flutter/material.dart';
import 'package:flutter_test/flutter_test.dart';

import 'package:wismon_keuangan/main.dart';

void main() {
  testWidgets('Counter increments smoke test', (WidgetTester tester) async {
    // Build our app and trigger a frame.
    await tester.pumpWidget(const MyApp());

    // Verify that our counter starts at 0.
    expect(find.text('0'), findsOneWidget);
    expect(find.text('1'), findsNothing);

    // Tap the '+' icon and trigger a frame.
    await tester.tap(find.byIcon(Icons.add));
    await tester.pump();

    // Verify that our counter has incremented.
    expect(find.text('0'), findsNothing);
    expect(find.text('1'), findsOneWidget);
  });
}
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\web\index.html

```
<!DOCTYPE html>
<html>
<head>
  <!--
    If you are serving your web app in a path other than the root, change the
    href value below to reflect the base path you are serving from.

    The path provided below has to start and end with a slash "/" in order for
    it to work correctly.

    For more details:
    * https://developer.mozilla.org/en-US/docs/Web/HTML/Element/base

    This is a placeholder for base href that will be replaced by the value of
    the `--base-href` argument provided to `flutter build`.
  -->
  <base href="$FLUTTER_BASE_HREF">

  <meta charset="UTF-8">
  <meta content="IE=Edge" http-equiv="X-UA-Compatible">
  <meta name="description" content="A new Flutter project.">

  <!-- iOS meta tags & icons -->
  <meta name="mobile-web-app-capable" content="yes">
  <meta name="apple-mobile-web-app-status-bar-style" content="black">
  <meta name="apple-mobile-web-app-title" content="wismon_keuangan">
  <link rel="apple-touch-icon" href="icons/Icon-192.png">

  <!-- Favicon -->
  <link rel="icon" type="image/png" href="favicon.png"/>

  <title>wismon_keuangan</title>
  <link rel="manifest" href="manifest.json">
</head>
<body>
  <script src="flutter_bootstrap.js" async></script>
</body>
</html>
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\web\manifest.json

```
{
    "name": "wismon_keuangan",
    "short_name": "wismon_keuangan",
    "start_url": ".",
    "display": "standalone",
    "background_color": "#0175C2",
    "theme_color": "#0175C2",
    "description": "A new Flutter project.",
    "orientation": "portrait-primary",
    "prefer_related_applications": false,
    "icons": [
        {
            "src": "icons/Icon-192.png",
            "sizes": "192x192",
            "type": "image/png"
        },
        {
            "src": "icons/Icon-512.png",
            "sizes": "512x512",
            "type": "image/png"
        },
        {
            "src": "icons/Icon-maskable-192.png",
            "sizes": "192x192",
            "type": "image/png",
            "purpose": "maskable"
        },
        {
            "src": "icons/Icon-maskable-512.png",
            "sizes": "512x512",
            "type": "image/png",
            "purpose": "maskable"
        }
    ]
}
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\windows\.gitignore

```
flutter/ephemeral/

# Visual Studio user-specific files.
*.suo
*.user
*.userosscache
*.sln.docstates

# Visual Studio build-related files.
x64/
x86/

# Visual Studio cache files
# files ending in .cache can be ignored
*.[Cc]ache
# but keep track of directories ending in .cache
!*.[Cc]ache/
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\windows\CMakeLists.txt

```
# Project-level configuration.
cmake_minimum_required(VERSION 3.14)
project(wismon_keuangan LANGUAGES CXX)

# The name of the executable created for the application. Change this to change
# the on-disk name of your application.
set(BINARY_NAME "wismon_keuangan")

# Explicitly opt in to modern CMake behaviors to avoid warnings with recent
# versions of CMake.
cmake_policy(VERSION 3.14...3.25)

# Define build configuration option.
get_property(IS_MULTICONFIG GLOBAL PROPERTY GENERATOR_IS_MULTI_CONFIG)
if(IS_MULTICONFIG)
  set(CMAKE_CONFIGURATION_TYPES "Debug;Profile;Release"
    CACHE STRING "" FORCE)
else()
  if(NOT CMAKE_BUILD_TYPE AND NOT CMAKE_CONFIGURATION_TYPES)
    set(CMAKE_BUILD_TYPE "Debug" CACHE
      STRING "Flutter build mode" FORCE)
    set_property(CACHE CMAKE_BUILD_TYPE PROPERTY STRINGS
      "Debug" "Profile" "Release")
  endif()
endif()
# Define settings for the Profile build mode.
set(CMAKE_EXE_LINKER_FLAGS_PROFILE "${CMAKE_EXE_LINKER_FLAGS_RELEASE}")
set(CMAKE_SHARED_LINKER_FLAGS_PROFILE "${CMAKE_SHARED_LINKER_FLAGS_RELEASE}")
set(CMAKE_C_FLAGS_PROFILE "${CMAKE_C_FLAGS_RELEASE}")
set(CMAKE_CXX_FLAGS_PROFILE "${CMAKE_CXX_FLAGS_RELEASE}")

# Use Unicode for all projects.
add_definitions(-DUNICODE -D_UNICODE)

# Compilation settings that should be applied to most targets.
#
# Be cautious about adding new options here, as plugins use this function by
# default. In most cases, you should add new options to specific targets instead
# of modifying this function.
function(APPLY_STANDARD_SETTINGS TARGET)
  target_compile_features(${TARGET} PUBLIC cxx_std_17)
  target_compile_options(${TARGET} PRIVATE /W4 /WX /wd"4100")
  target_compile_options(${TARGET} PRIVATE /EHsc)
  target_compile_definitions(${TARGET} PRIVATE "_HAS_EXCEPTIONS=0")
  target_compile_definitions(${TARGET} PRIVATE "$<$<CONFIG:Debug>:_DEBUG>")
endfunction()

# Flutter library and tool build rules.
set(FLUTTER_MANAGED_DIR "${CMAKE_CURRENT_SOURCE_DIR}/flutter")
add_subdirectory(${FLUTTER_MANAGED_DIR})

# Application build; see runner/CMakeLists.txt.
add_subdirectory("runner")


# Generated plugin build rules, which manage building the plugins and adding
# them to the application.
include(flutter/generated_plugins.cmake)


# === Installation ===
# Support files are copied into place next to the executable, so that it can
# run in place. This is done instead of making a separate bundle (as on Linux)
# so that building and running from within Visual Studio will work.
set(BUILD_BUNDLE_DIR "$<TARGET_FILE_DIR:${BINARY_NAME}>")
# Make the "install" step default, as it's required to run.
set(CMAKE_VS_INCLUDE_INSTALL_TO_DEFAULT_BUILD 1)
if(CMAKE_INSTALL_PREFIX_INITIALIZED_TO_DEFAULT)
  set(CMAKE_INSTALL_PREFIX "${BUILD_BUNDLE_DIR}" CACHE PATH "..." FORCE)
endif()

set(INSTALL_BUNDLE_DATA_DIR "${CMAKE_INSTALL_PREFIX}/data")
set(INSTALL_BUNDLE_LIB_DIR "${CMAKE_INSTALL_PREFIX}")

install(TARGETS ${BINARY_NAME} RUNTIME DESTINATION "${CMAKE_INSTALL_PREFIX}"
  COMPONENT Runtime)

install(FILES "${FLUTTER_ICU_DATA_FILE}" DESTINATION "${INSTALL_BUNDLE_DATA_DIR}"
  COMPONENT Runtime)

install(FILES "${FLUTTER_LIBRARY}" DESTINATION "${INSTALL_BUNDLE_LIB_DIR}"
  COMPONENT Runtime)

if(PLUGIN_BUNDLED_LIBRARIES)
  install(FILES "${PLUGIN_BUNDLED_LIBRARIES}"
    DESTINATION "${INSTALL_BUNDLE_LIB_DIR}"
    COMPONENT Runtime)
endif()

# Copy the native assets provided by the build.dart from all packages.
set(NATIVE_ASSETS_DIR "${PROJECT_BUILD_DIR}native_assets/windows/")
install(DIRECTORY "${NATIVE_ASSETS_DIR}"
   DESTINATION "${INSTALL_BUNDLE_LIB_DIR}"
   COMPONENT Runtime)

# Fully re-copy the assets directory on each build to avoid having stale files
# from a previous install.
set(FLUTTER_ASSET_DIR_NAME "flutter_assets")
install(CODE "
  file(REMOVE_RECURSE \"${INSTALL_BUNDLE_DATA_DIR}/${FLUTTER_ASSET_DIR_NAME}\")
  " COMPONENT Runtime)
install(DIRECTORY "${PROJECT_BUILD_DIR}/${FLUTTER_ASSET_DIR_NAME}"
  DESTINATION "${INSTALL_BUNDLE_DATA_DIR}" COMPONENT Runtime)

# Install the AOT library on non-Debug builds only.
install(FILES "${AOT_LIBRARY}" DESTINATION "${INSTALL_BUNDLE_DATA_DIR}"
  CONFIGURATIONS Profile;Release
  COMPONENT Runtime)
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\windows\flutter\CMakeLists.txt

```
# This file controls Flutter-level build steps. It should not be edited.
cmake_minimum_required(VERSION 3.14)

set(EPHEMERAL_DIR "${CMAKE_CURRENT_SOURCE_DIR}/ephemeral")

# Configuration provided via flutter tool.
include(${EPHEMERAL_DIR}/generated_config.cmake)

# TODO: Move the rest of this into files in ephemeral. See
# https://github.com/flutter/flutter/issues/57146.
set(WRAPPER_ROOT "${EPHEMERAL_DIR}/cpp_client_wrapper")

# Set fallback configurations for older versions of the flutter tool.
if (NOT DEFINED FLUTTER_TARGET_PLATFORM)
  set(FLUTTER_TARGET_PLATFORM "windows-x64")
endif()

# === Flutter Library ===
set(FLUTTER_LIBRARY "${EPHEMERAL_DIR}/flutter_windows.dll")

# Published to parent scope for install step.
set(FLUTTER_LIBRARY ${FLUTTER_LIBRARY} PARENT_SCOPE)
set(FLUTTER_ICU_DATA_FILE "${EPHEMERAL_DIR}/icudtl.dat" PARENT_SCOPE)
set(PROJECT_BUILD_DIR "${PROJECT_DIR}/build/" PARENT_SCOPE)
set(AOT_LIBRARY "${PROJECT_DIR}/build/windows/app.so" PARENT_SCOPE)

list(APPEND FLUTTER_LIBRARY_HEADERS
  "flutter_export.h"
  "flutter_windows.h"
  "flutter_messenger.h"
  "flutter_plugin_registrar.h"
  "flutter_texture_registrar.h"
)
list(TRANSFORM FLUTTER_LIBRARY_HEADERS PREPEND "${EPHEMERAL_DIR}/")
add_library(flutter INTERFACE)
target_include_directories(flutter INTERFACE
  "${EPHEMERAL_DIR}"
)
target_link_libraries(flutter INTERFACE "${FLUTTER_LIBRARY}.lib")
add_dependencies(flutter flutter_assemble)

# === Wrapper ===
list(APPEND CPP_WRAPPER_SOURCES_CORE
  "core_implementations.cc"
  "standard_codec.cc"
)
list(TRANSFORM CPP_WRAPPER_SOURCES_CORE PREPEND "${WRAPPER_ROOT}/")
list(APPEND CPP_WRAPPER_SOURCES_PLUGIN
  "plugin_registrar.cc"
)
list(TRANSFORM CPP_WRAPPER_SOURCES_PLUGIN PREPEND "${WRAPPER_ROOT}/")
list(APPEND CPP_WRAPPER_SOURCES_APP
  "flutter_engine.cc"
  "flutter_view_controller.cc"
)
list(TRANSFORM CPP_WRAPPER_SOURCES_APP PREPEND "${WRAPPER_ROOT}/")

# Wrapper sources needed for a plugin.
add_library(flutter_wrapper_plugin STATIC
  ${CPP_WRAPPER_SOURCES_CORE}
  ${CPP_WRAPPER_SOURCES_PLUGIN}
)
apply_standard_settings(flutter_wrapper_plugin)
set_target_properties(flutter_wrapper_plugin PROPERTIES
  POSITION_INDEPENDENT_CODE ON)
set_target_properties(flutter_wrapper_plugin PROPERTIES
  CXX_VISIBILITY_PRESET hidden)
target_link_libraries(flutter_wrapper_plugin PUBLIC flutter)
target_include_directories(flutter_wrapper_plugin PUBLIC
  "${WRAPPER_ROOT}/include"
)
add_dependencies(flutter_wrapper_plugin flutter_assemble)

# Wrapper sources needed for the runner.
add_library(flutter_wrapper_app STATIC
  ${CPP_WRAPPER_SOURCES_CORE}
  ${CPP_WRAPPER_SOURCES_APP}
)
apply_standard_settings(flutter_wrapper_app)
target_link_libraries(flutter_wrapper_app PUBLIC flutter)
target_include_directories(flutter_wrapper_app PUBLIC
  "${WRAPPER_ROOT}/include"
)
add_dependencies(flutter_wrapper_app flutter_assemble)

# === Flutter tool backend ===
# _phony_ is a non-existent file to force this command to run every time,
# since currently there's no way to get a full input/output list from the
# flutter tool.
set(PHONY_OUTPUT "${CMAKE_CURRENT_BINARY_DIR}/_phony_")
set_source_files_properties("${PHONY_OUTPUT}" PROPERTIES SYMBOLIC TRUE)
add_custom_command(
  OUTPUT ${FLUTTER_LIBRARY} ${FLUTTER_LIBRARY_HEADERS}
    ${CPP_WRAPPER_SOURCES_CORE} ${CPP_WRAPPER_SOURCES_PLUGIN}
    ${CPP_WRAPPER_SOURCES_APP}
    ${PHONY_OUTPUT}
  COMMAND ${CMAKE_COMMAND} -E env
    ${FLUTTER_TOOL_ENVIRONMENT}
    "${FLUTTER_ROOT}/packages/flutter_tools/bin/tool_backend.bat"
      ${FLUTTER_TARGET_PLATFORM} $<CONFIG>
  VERBATIM
)
add_custom_target(flutter_assemble DEPENDS
  "${FLUTTER_LIBRARY}"
  ${FLUTTER_LIBRARY_HEADERS}
  ${CPP_WRAPPER_SOURCES_CORE}
  ${CPP_WRAPPER_SOURCES_PLUGIN}
  ${CPP_WRAPPER_SOURCES_APP}
)
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\windows\flutter\generated_plugins.cmake

```
#
# Generated file, do not edit.
#

list(APPEND FLUTTER_PLUGIN_LIST
)

list(APPEND FLUTTER_FFI_PLUGIN_LIST
)

set(PLUGIN_BUNDLED_LIBRARIES)

foreach(plugin ${FLUTTER_PLUGIN_LIST})
  add_subdirectory(flutter/ephemeral/.plugin_symlinks/${plugin}/windows plugins/${plugin})
  target_link_libraries(${BINARY_NAME} PRIVATE ${plugin}_plugin)
  list(APPEND PLUGIN_BUNDLED_LIBRARIES $<TARGET_FILE:${plugin}_plugin>)
  list(APPEND PLUGIN_BUNDLED_LIBRARIES ${${plugin}_bundled_libraries})
endforeach(plugin)

foreach(ffi_plugin ${FLUTTER_FFI_PLUGIN_LIST})
  add_subdirectory(flutter/ephemeral/.plugin_symlinks/${ffi_plugin}/windows plugins/${ffi_plugin})
  list(APPEND PLUGIN_BUNDLED_LIBRARIES ${${ffi_plugin}_bundled_libraries})
endforeach(ffi_plugin)
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\windows\flutter\generated_plugin_registrant.cc

```
//
//  Generated file. Do not edit.
//

// clang-format off

#include "generated_plugin_registrant.h"


void RegisterPlugins(flutter::PluginRegistry* registry) {
}
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\windows\flutter\generated_plugin_registrant.h

```
//
//  Generated file. Do not edit.
//

// clang-format off

#ifndef GENERATED_PLUGIN_REGISTRANT_
#define GENERATED_PLUGIN_REGISTRANT_

#include <flutter/plugin_registry.h>

// Registers Flutter plugins.
void RegisterPlugins(flutter::PluginRegistry* registry);

#endif  // GENERATED_PLUGIN_REGISTRANT_
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\windows\runner\CMakeLists.txt

```
cmake_minimum_required(VERSION 3.14)
project(runner LANGUAGES CXX)

# Define the application target. To change its name, change BINARY_NAME in the
# top-level CMakeLists.txt, not the value here, or `flutter run` will no longer
# work.
#
# Any new source files that you add to the application should be added here.
add_executable(${BINARY_NAME} WIN32
  "flutter_window.cpp"
  "main.cpp"
  "utils.cpp"
  "win32_window.cpp"
  "${FLUTTER_MANAGED_DIR}/generated_plugin_registrant.cc"
  "Runner.rc"
  "runner.exe.manifest"
)

# Apply the standard set of build settings. This can be removed for applications
# that need different build settings.
apply_standard_settings(${BINARY_NAME})

# Add preprocessor definitions for the build version.
target_compile_definitions(${BINARY_NAME} PRIVATE "FLUTTER_VERSION=\"${FLUTTER_VERSION}\"")
target_compile_definitions(${BINARY_NAME} PRIVATE "FLUTTER_VERSION_MAJOR=${FLUTTER_VERSION_MAJOR}")
target_compile_definitions(${BINARY_NAME} PRIVATE "FLUTTER_VERSION_MINOR=${FLUTTER_VERSION_MINOR}")
target_compile_definitions(${BINARY_NAME} PRIVATE "FLUTTER_VERSION_PATCH=${FLUTTER_VERSION_PATCH}")
target_compile_definitions(${BINARY_NAME} PRIVATE "FLUTTER_VERSION_BUILD=${FLUTTER_VERSION_BUILD}")

# Disable Windows macros that collide with C++ standard library functions.
target_compile_definitions(${BINARY_NAME} PRIVATE "NOMINMAX")

# Add dependency libraries and include directories. Add any application-specific
# dependencies here.
target_link_libraries(${BINARY_NAME} PRIVATE flutter flutter_wrapper_app)
target_link_libraries(${BINARY_NAME} PRIVATE "dwmapi.lib")
target_include_directories(${BINARY_NAME} PRIVATE "${CMAKE_SOURCE_DIR}")
```

```
# Run the Flutter tool portions of the build. This must not be removed.
add_dependencies(${BINARY_NAME} flutter_assemble)
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\windows\runner\flutter_window.cpp

```
#include "flutter_window.h"

#include <optional>

#include "flutter/generated_plugin_registrant.h"

FlutterWindow::FlutterWindow(const flutter::DartProject& project)
    : project_(project) {}

FlutterWindow::~FlutterWindow() {}

bool FlutterWindow::OnCreate() {
  if (!Win32Window::OnCreate()) {
    return false;
  }

  RECT frame = GetClientArea();

  // The size here must match the window dimensions to avoid unnecessary surface
  // creation / destruction in the startup path.
  flutter_controller_ = std::make_unique<flutter::FlutterViewController>(
      frame.right - frame.left, frame.bottom - frame.top, project_);
  // Ensure that basic setup of the controller was successful.
  if (!flutter_controller_->engine() || !flutter_controller_->view()) {
    return false;
  }
  RegisterPlugins(flutter_controller_->engine());
  SetChildContent(flutter_controller_->view()->GetNativeWindow());

  flutter_controller_->engine()->SetNextFrameCallback([&]() {
    this->Show();
  });

  // Flutter can complete the first frame before the "show window" callback is
  // registered. The following call ensures a frame is pending to ensure the
  // window is shown. It is a no-op if the first frame hasn't completed yet.
  flutter_controller_->ForceRedraw();

  return true;
}

void FlutterWindow::OnDestroy() {
  if (flutter_controller_) {
    flutter_controller_ = nullptr;
  }

  Win32Window::OnDestroy();
}

LRESULT
FlutterWindow::MessageHandler(HWND hwnd, UINT const message,
                              WPARAM const wparam,
                              LPARAM const lparam) noexcept {
  // Give Flutter, including plugins, an opportunity to handle window messages.
  if (flutter_controller_) {
    std::optional<LRESULT> result =
        flutter_controller_->HandleTopLevelWindowProc(hwnd, message, wparam,
                                                      lparam);
    if (result) {
      return *result;
    }
  }

  switch (message) {
    case WM_FONTCHANGE:
      flutter_controller_->engine()->ReloadSystemFonts();
      break;
  }

  return Win32Window::MessageHandler(hwnd, message, wparam, lparam);
}
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\windows\runner\flutter_window.h

```
#ifndef RUNNER_FLUTTER_WINDOW_H_
#define RUNNER_FLUTTER_WINDOW_H_

#include <flutter/dart_project.h>
#include <flutter/flutter_view_controller.h>

#include <memory>

#include "win32_window.h"

// A window that does nothing but host a Flutter view.
class FlutterWindow : public Win32Window {
 public:
  // Creates a new FlutterWindow hosting a Flutter view running |project|.
  explicit FlutterWindow(const flutter::DartProject& project);
  virtual ~FlutterWindow();

 protected:
  // Win32Window:
  bool OnCreate() override;
  void OnDestroy() override;
  LRESULT MessageHandler(HWND window, UINT const message, WPARAM const wparam,
                         LPARAM const lparam) noexcept override;

 private:
  // The project to run.
  flutter::DartProject project_;

  // The Flutter instance hosted by this window.
  std::unique_ptr<flutter::FlutterViewController> flutter_controller_;
};

#endif  // RUNNER_FLUTTER_WINDOW_H_
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\windows\runner\main.cpp

```
#include <flutter/dart_project.h>
#include <flutter/flutter_view_controller.h>
#include <windows.h>

#include "flutter_window.h"
#include "utils.h"

int APIENTRY wWinMain(_In_ HINSTANCE instance, _In_opt_ HINSTANCE prev,
                      _In_ wchar_t *command_line, _In_ int show_command) {
  // Attach to console when present (e.g., 'flutter run') or create a
  // new console when running with a debugger.
  if (!::AttachConsole(ATTACH_PARENT_PROCESS) && ::IsDebuggerPresent()) {
    CreateAndAttachConsole();
  }

  // Initialize COM, so that it is available for use in the library and/or
  // plugins.
  ::CoInitializeEx(nullptr, COINIT_APARTMENTTHREADED);

  flutter::DartProject project(L"data");

  std::vector<std::string> command_line_arguments =
      GetCommandLineArguments();

  project.set_dart_entrypoint_arguments(std::move(command_line_arguments));

  FlutterWindow window(project);
  Win32Window::Point origin(10, 10);
  Win32Window::Size size(1280, 720);
  if (!window.Create(L"wismon_keuangan", origin, size)) {
    return EXIT_FAILURE;
  }
  window.SetQuitOnClose(true);

  ::MSG msg;
  while (::GetMessage(&msg, nullptr, 0, 0)) {
    ::TranslateMessage(&msg);
    ::DispatchMessage(&msg);
  }

  ::CoUninitialize();
  return EXIT_SUCCESS;
}
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\windows\runner\resource.h

```
//{{NO_DEPENDENCIES}}
// Microsoft Visual C++ generated include file.
// Used by Runner.rc
//
#define IDI_APP_ICON                    101

// Next default values for new objects
//
#ifdef APSTUDIO_INVOKED
#ifndef APSTUDIO_READONLY_SYMBOLS
#define _APS_NEXT_RESOURCE_VALUE        102
#define _APS_NEXT_COMMAND_VALUE         40001
#define _APS_NEXT_CONTROL_VALUE         1001
#define _APS_NEXT_SYMED_VALUE           101
#endif
#endif
```

File:

D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\windows\runner\runner.exe.manifest

```
<¿xml version="1.0" encoding="UTF-8" standalone="yes"¿>
<assembly xmlns="urn:schemas-microsoft-com:asm.v1" manifestVersion="1.0">
  <application xmlns="urn:schemas-microsoft-com:asm.v3">
    <windowsSettings>
      <dpiAwareness xmlns="http://schemas.microsoft.com/SMI/2016/WindowsSettings">PerMonitorV2</dpiAwareness>
    </windowsSettings>
  </application>
  <compatibility xmlns="urn:schemas-microsoft-com:compatibility.v1">
    <application>
      <!-- Windows 10 and Windows 11 -->
      <supportedOS Id="{8e0f7a12-bfb3-4fe8-b9a5-48fd50a15a9a}"/>
    </application>
  </compatibility>
</assembly>
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\windows\runner\Runner.rc

```
// Microsoft Visual C++ generated resource script.
//
#pragma code_page(65001)
#include "resource.h"

#define APSTUDIO_READONLY_SYMBOLS
/////////////////////////////////////////////////////////////////////////////
//
// Generated from the TEXTINCLUDE 2 resource.
//
#include "winres.h"

/////////////////////////////////////////////////////////////////////////////
#undef APSTUDIO_READONLY_SYMBOLS

/////////////////////////////////////////////////////////////////////////////
// English (United States) resources

#if !defined(AFX_RESOURCE_DLL) || defined(AFX_TARG_ENU)
LANGUAGE LANG_ENGLISH, SUBLANG_ENGLISH_US

#ifdef APSTUDIO_INVOKED
/////////////////////////////////////////////////////////////////////////////
//
// TEXTINCLUDE
//

1 TEXTINCLUDE
BEGIN
    "resource.h\0"
END

2 TEXTINCLUDE
BEGIN
    "#include ""winres.h""\r\n"
    "\0"
END

3 TEXTINCLUDE
BEGIN
    "\r\n"
    "\0"
END

#endif    // APSTUDIO_INVOKED


/////////////////////////////////////////////////////////////////////////////
//
// Icon
//

// Icon with lowest ID value placed first to ensure application icon
// remains consistent on all systems.
IDI_APP_ICON            ICON                    "resources\\app_icon.ico"


///////////////////////////////////////////////////////////////////////////////
//
// Version
//

#if  defined(FLUTTER_VERSION_MAJOR)  &&  defined(FLUTTER_VERSION_MINOR)  &&  defined(FLUTTER_VERSION_PATCH)  &&
defined(FLUTTER_VERSION_BUILD)
#define                                                                                       VERSION_AS_NUMBER
FLUTTER_VERSION_MAJOR,FLUTTER_VERSION_MINOR,FLUTTER_VERSION_PATCH,FLUTTER_VERSION_BUILD
#else
#define VERSION_AS_NUMBER 1,0,0,0
#endif

#if defined(FLUTTER_VERSION)
#define VERSION_AS_STRING FLUTTER_VERSION
#else
#define VERSION_AS_STRING "1.0.0"
#endif

VS_VERSION_INFO VERSIONINFO
 FILEVERSION VERSION_AS_NUMBER
 PRODUCTVERSION VERSION_AS_NUMBER
 FILEFLAGSMASK VS_FFI_FILEFLAGSMASK
#ifdef _DEBUG
 FILEFLAGS VS_FF_DEBUG
#else
 FILEFLAGS 0x0L
#endif
 FILEOS VOS__WINDOWS32
 FILETYPE VFT_APP
 FILESUBTYPE 0x0L
BEGIN
    BLOCK "StringFileInfo"
    BEGIN
        BLOCK "040904e4"
        BEGIN
            VALUE "CompanyName", "com.example" "\0"
            VALUE "FileDescription", "wismon_keuangan" "\0"
            VALUE "FileVersion", VERSION_AS_STRING "\0"
            VALUE "InternalName", "wismon_keuangan" "\0"
            VALUE "LegalCopyright", "Copyright (C) 2025 com.example. All rights reserved." "\0"
            VALUE "OriginalFilename", "wismon_keuangan.exe" "\0"
            VALUE "ProductName", "wismon_keuangan" "\0"
            VALUE "ProductVersion", VERSION_AS_STRING "\0"
        END
    END
    BLOCK "VarFileInfo"
    BEGIN
        VALUE "Translation", 0x409, 1252
    END
END

#endif    // English (United States) resources
///////////////////////////////////////////////////////////////////////////////



#ifndef APSTUDIO_INVOKED
///////////////////////////////////////////////////////////////////////////////
//
// Generated from the TEXTINCLUDE 3 resource.
//
```

```
////////////////////////////////////////////////////////////////////////////
#endif    // not APSTUDIO_INVOKED
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\windows\runner\utils.cpp

```
#include "utils.h"

#include <flutter_windows.h>
#include <io.h>
#include <stdio.h>
#include <windows.h>

#include <iostream>

void CreateAndAttachConsole() {
  if (::AllocConsole()) {
    FILE *unused;
    if (freopen_s(&unused, "CONOUT$", "w", stdout)) {
      _dup2(_fileno(stdout), 1);
    }
    if (freopen_s(&unused, "CONOUT$", "w", stderr)) {
      _dup2(_fileno(stdout), 2);
    }
    std::ios::sync_with_stdio();
    FlutterDesktopResyncOutputStreams();
  }
}

std::vector<std::string> GetCommandLineArguments() {
  // Convert the UTF-16 command line arguments to UTF-8 for the Engine to use.
  int argc;
  wchar_t** argv = ::CommandLineToArgvW(::GetCommandLineW(), &argc);
  if (argv == nullptr) {
    return std::vector<std::string>();
  }

  std::vector<std::string> command_line_arguments;

  // Skip the first argument as it's the binary name.
  for (int i = 1; i < argc; i++) {
    command_line_arguments.push_back(Utf8FromUtf16(argv[i]));
  }

  ::LocalFree(argv);

  return command_line_arguments;
}

std::string Utf8FromUtf16(const wchar_t* utf16_string) {
  if (utf16_string == nullptr) {
    return std::string();
  }
  unsigned int target_length = ::WideCharToMultiByte(
      CP_UTF8, WC_ERR_INVALID_CHARS, utf16_string,
      -1, nullptr, 0, nullptr, nullptr)
    -1; // remove the trailing null character
  int input_length = (int)wcslen(utf16_string);
  std::string utf8_string;
  if (target_length == 0 || target_length > utf8_string.max_size()) {
    return utf8_string;
  }
  utf8_string.resize(target_length);
  int converted_length = ::WideCharToMultiByte(
      CP_UTF8, WC_ERR_INVALID_CHARS, utf16_string,
      input_length, utf8_string.data(), target_length, nullptr, nullptr);
  if (converted_length == 0) {
    return std::string();
  }
  return utf8_string;
}
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\windows\runner\utils.h

```
#ifndef RUNNER_UTILS_H_
#define RUNNER_UTILS_H_

#include <string>
#include <vector>

// Creates a console for the process, and redirects stdout and stderr to
// it for both the runner and the Flutter library.
void CreateAndAttachConsole();

// Takes a null-terminated wchar_t* encoded in UTF-16 and returns a std::string
// encoded in UTF-8. Returns an empty std::string on failure.
std::string Utf8FromUtf16(const wchar_t* utf16_string);

// Gets the command line arguments passed in as a std::vector<std::string>,
// encoded in UTF-8. Returns an empty std::vector<std::string> on failure.
std::vector<std::string> GetCommandLineArguments();

#endif  // RUNNER_UTILS_H_
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\windows\runner\win32_window.cpp

```
#include "win32_window.h"

#include <dwmapi.h>
#include <flutter_windows.h>

#include "resource.h"

namespace {

/// Window attribute that enables dark mode window decorations.
///
/// Redefined in case the developer's machine has a Windows SDK older than
/// version 10.0.22000.0.
/// See: https://docs.microsoft.com/windows/win32/api/dwmapi/ne-dwmapi-dwmwindowattribute
#ifndef DWMWA_USE_IMMERSIVE_DARK_MODE
#define DWMWA_USE_IMMERSIVE_DARK_MODE 20
#endif

constexpr const wchar_t kWindowClassName[] = L"FLUTTER_RUNNER_WIN32_WINDOW";

/// Registry key for app theme preference.
///
/// A value of 0 indicates apps should use dark mode. A non-zero or missing
/// value indicates apps should use light mode.
constexpr const wchar_t kGetPreferredBrightnessRegKey[] =
  L"Software\\Microsoft\\Windows\\CurrentVersion\\Themes\\Personalize";
constexpr const wchar_t kGetPreferredBrightnessRegValue[] = L"AppsUseLightTheme";

// The number of Win32Window objects that currently exist.
static int g_active_window_count = 0;
```

```
using EnableNonClientDpiScaling = BOOL __stdcall(HWND hwnd);

// Scale helper to convert logical scaler values to physical using passed in
// scale factor
int Scale(int source, double scale_factor) {
  return static_cast<int>(source * scale_factor);
}

// Dynamically loads the |EnableNonClientDpiScaling| from the User32 module.
// This API is only needed for PerMonitor V1 awareness mode.
void EnableFullDpiSupportIfAvailable(HWND hwnd) {
  HMODULE user32_module = LoadLibraryA("User32.dll");
  if (!user32_module) {
    return;
  }
  auto enable_non_client_dpi_scaling =
      reinterpret_cast<EnableNonClientDpiScaling*>(
          GetProcAddress(user32_module, "EnableNonClientDpiScaling"));
  if (enable_non_client_dpi_scaling != nullptr) {
    enable_non_client_dpi_scaling(hwnd);
  }
  FreeLibrary(user32_module);
}

}  // namespace

// Manages the Win32Window's window class registration.
class WindowClassRegistrar {
 public:
  ~WindowClassRegistrar() = default;

  // Returns the singleton registrar instance.
  static WindowClassRegistrar* GetInstance() {
    if (!instance_) {
      instance_ = new WindowClassRegistrar();
    }
    return instance_;
  }

  // Returns the name of the window class, registering the class if it hasn't
  // previously been registered.
  const wchar_t* GetWindowClass();

  // Unregisters the window class. Should only be called if there are no
  // instances of the window.
  void UnregisterWindowClass();

 private:
  WindowClassRegistrar() = default;

  static WindowClassRegistrar* instance_;

  bool class_registered_ = false;
};

WindowClassRegistrar* WindowClassRegistrar::instance_ = nullptr;

const wchar_t* WindowClassRegistrar::GetWindowClass() {
  if (!class_registered_) {
    WNDCLASS window_class{};
    window_class.hCursor = LoadCursor(nullptr, IDC_ARROW);
    window_class.lpszClassName = kWindowClassName;
    window_class.style = CS_HREDRAW | CS_VREDRAW;
    window_class.cbClsExtra = 0;
    window_class.cbWndExtra = 0;
    window_class.hInstance = GetModuleHandle(nullptr);
    window_class.hIcon =
        LoadIcon(window_class.hInstance, MAKEINTRESOURCE(IDI_APP_ICON));
    window_class.hbrBackground = 0;
    window_class.lpszMenuName = nullptr;
    window_class.lpfnWndProc = Win32Window::WndProc;
    RegisterClass(&window_class);
    class_registered_ = true;
  }
  return kWindowClassName;
}

void WindowClassRegistrar::UnregisterWindowClass() {
  UnregisterClass(kWindowClassName, nullptr);
  class_registered_ = false;
}

Win32Window::Win32Window() {
  ++g_active_window_count;
}

Win32Window::~Win32Window() {
  --g_active_window_count;
  Destroy();
}

bool Win32Window::Create(const std::wstring& title,
                         const Point& origin,
                         const Size& size) {
  Destroy();

  const wchar_t* window_class =
      WindowClassRegistrar::GetInstance()->GetWindowClass();

  const POINT target_point = {static_cast<LONG>(origin.x),
                              static_cast<LONG>(origin.y)};
  HMONITOR monitor = MonitorFromPoint(target_point, MONITOR_DEFAULTTONEAREST);
  UINT dpi = FlutterDesktopGetDpiForMonitor(monitor);
  double scale_factor = dpi / 96.0;

  HWND window = CreateWindow(
      window_class, title.c_str(), WS_OVERLAPPEDWINDOW,
      Scale(origin.x, scale_factor), Scale(origin.y, scale_factor),
      Scale(size.width, scale_factor), Scale(size.height, scale_factor),
      nullptr, nullptr, GetModuleHandle(nullptr), this);

  if (!window) {
    return false;
  }

  UpdateTheme(window);

  return OnCreate();
}

bool Win32Window::Show() {
  return ShowWindow(window_handle_, SW_SHOWNORMAL);
}

// static
LRESULT CALLBACK Win32Window::WndProc(HWND const window,
                                      UINT const message,
                                      WPARAM const wparam,
                                      LPARAM const lparam) noexcept {
  if (message == WM_NCCREATE) {
    auto window_struct = reinterpret_cast<CREATESTRUCT*>(lparam);
    SetWindowLongPtr(window, GWLP_USERDATA,
                     reinterpret_cast<LONG_PTR>(window_struct->lpCreateParams));

    auto that = static_cast<Win32Window*>(window_struct->lpCreateParams);
    EnableFullDpiSupportIfAvailable(window);
    that->window_handle_ = window;
  } else if (Win32Window* that = GetThisFromHandle(window)) {
    return that->MessageHandler(window, message, wparam, lparam);
  }

  return DefWindowProc(window, message, wparam, lparam);
}

LRESULT
Win32Window::MessageHandler(HWND hwnd,
                            UINT const message,
                            WPARAM const wparam,
                            LPARAM const lparam) noexcept {
  switch (message) {
    case WM_DESTROY:
      window_handle_ = nullptr;
      Destroy();
      if (quit_on_close_) {
        PostQuitMessage(0);
      }
      return 0;

    case WM_DPICHANGED: {
      auto newRectSize = reinterpret_cast<RECT*>(lparam);
      LONG newWidth = newRectSize->right - newRectSize->left;
      LONG newHeight = newRectSize->bottom - newRectSize->top;

      SetWindowPos(hwnd, nullptr, newRectSize->left, newRectSize->top, newWidth,
                   newHeight, SWP_NOZORDER | SWP_NOACTIVATE);

      return 0;
    }
    case WM_SIZE: {
      RECT rect = GetClientArea();
      if (child_content_ != nullptr) {
        // Size and position the child window.
        MoveWindow(child_content_, rect.left, rect.top, rect.right - rect.left,
                   rect.bottom - rect.top, TRUE);
      }
      return 0;
    }

    case WM_ACTIVATE:
      if (child_content_ != nullptr) {
        SetFocus(child_content_);
      }
      return 0;

    case WM_DWMCOLORIZATIONCOLORCHANGED:
      UpdateTheme(hwnd);
      return 0;
  }

  return DefWindowProc(window_handle_, message, wparam, lparam);
}

void Win32Window::Destroy() {
  OnDestroy();

  if (window_handle_) {
    DestroyWindow(window_handle_);
    window_handle_ = nullptr;
  }
  if (g_active_window_count == 0) {
    WindowClassRegistrar::GetInstance()->UnregisterWindowClass();
  }
}

Win32Window* Win32Window::GetThisFromHandle(HWND const window) noexcept {
  return reinterpret_cast<Win32Window*>(
      GetWindowLongPtr(window, GWLP_USERDATA));
}

void Win32Window::SetChildContent(HWND content) {
  child_content_ = content;
  SetParent(content, window_handle_);
  RECT frame = GetClientArea();

  MoveWindow(content, frame.left, frame.top, frame.right - frame.left,
             frame.bottom - frame.top, true);

  SetFocus(child_content_);
}

RECT Win32Window::GetClientArea() {
  RECT frame;
  GetClientRect(window_handle_, &frame);
  return frame;
}

HWND Win32Window::GetHandle() {
  return window_handle_;
}

void Win32Window::SetQuitOnClose(bool quit_on_close) {
  quit_on_close_ = quit_on_close;
}

bool Win32Window::OnCreate() {
  // No-op; provided for subclasses.
  return true;
}

void Win32Window::OnDestroy() {
  // No-op; provided for subclasses.
}

void Win32Window::UpdateTheme(HWND const window) {
  DWORD light_mode;
  DWORD light_mode_size = sizeof(light_mode);
  LSTATUS result = RegGetValue(HKEY_CURRENT_USER, kGetPreferredBrightnessRegKey,
                               kGetPreferredBrightnessRegValue,
                               RRF_RT_REG_DWORD, nullptr, &light_mode,
                               &light_mode_size);

  if (result == ERROR_SUCCESS) {
    BOOL enable_dark_mode = light_mode == 0;
    DwmSetWindowAttribute(window, DWMWA_USE_IMMERSIVE_DARK_MODE,
                          &enable_dark_mode, sizeof(enable_dark_mode));
  }
}
```

File: D:\Android\sso_wis_wismon_nodejs\wismon_keuangan\windows\runner\win32_window.h

```
#ifndef RUNNER_WIN32_WINDOW_H_
```

```
#define RUNNER_WIN32_WINDOW_H_

#include <windows.h>

#include <functional>
#include <memory>
#include <string>

// A class abstraction for a high DPI-aware Win32 Window. Intended to be
// inherited from by classes that wish to specialize with custom
// rendering and input handling
class Win32Window {
 public:
  struct Point {
    unsigned int x;
    unsigned int y;
    Point(unsigned int x, unsigned int y) : x(x), y(y) {}
  };

  struct Size {
    unsigned int width;
    unsigned int height;
    Size(unsigned int width, unsigned int height)
        : width(width), height(height) {}
  };

  Win32Window();
  virtual ~Win32Window();

  // Creates a win32 window with |title| that is positioned and sized using
  // |origin| and |size|. New windows are created on the default monitor. Window
  // sizes are specified to the OS in physical pixels, hence to ensure a
  // consistent size this function will scale the inputted width and height as
  // as appropriate for the default monitor. The window is invisible until
  // |Show| is called. Returns true if the window was created successfully.
  bool Create(const std::wstring& title, const Point& origin, const Size& size);

  // Show the current window. Returns true if the window was successfully shown.
  bool Show();

  // Release OS resources associated with window.
  void Destroy();

  // Inserts |content| into the window tree.
  void SetChildContent(HWND content);

  // Returns the backing Window handle to enable clients to set icon and other
  // window properties. Returns nullptr if the window has been destroyed.
  HWND GetHandle();

  // If true, closing this window will quit the application.
  void SetQuitOnClose(bool quit_on_close);

  // Return a RECT representing the bounds of the current client area.
  RECT GetClientArea();

 protected:
  // Processes and route salient window messages for mouse handling,
  // size change and DPI. Delegates handling of these to member overloads that
  // inheriting classes can handle.
  virtual LRESULT MessageHandler(HWND window,
                                 UINT const message,
                                 WPARAM const wparam,
                                 LPARAM const lparam) noexcept;

  // Called when CreateAndShow is called, allowing subclass window-related
  // setup. Subclasses should return false if setup fails.
  virtual bool OnCreate();

  // Called when Destroy is called.
  virtual void OnDestroy();

 private:
  friend class WindowClassRegistrar;

  // OS callback called by message pump. Handles the WM_NCCREATE message which
  // is passed when the non-client area is being created and enables automatic
  // non-client DPI scaling so that the non-client area automatically
  // responds to changes in DPI. All other messages are handled by
  // MessageHandler.
  static LRESULT CALLBACK WndProc(HWND const window,
                                  UINT const message,
                                  WPARAM const wparam,
                                  LPARAM const lparam) noexcept;

  // Retrieves a class instance pointer for |window|
  static Win32Window* GetThisFromHandle(HWND const window) noexcept;

  // Update the window frame's theme to match the system theme.
  static void UpdateTheme(HWND const window);

  bool quit_on_close_ = false;

  // window handle for top level window.
  HWND window_handle_ = nullptr;

  // window handle for hosted content.
  HWND child_content_ = nullptr;
};

#endif  // RUNNER_WIN32_WINDOW_H_
```