🛠️ DAOUI Reports - Developer Guide

🧾 Overview

DAOUI Reports is a visual report designer and viewer built using **Delphi FireMonkey (FMX)**. It allows users to design printable reports using a visual layout interface with support for headers, footers, details, zoom, pagination, and PDF export.

📁 Project Structure

Unit Name	Description
`DesignU.pas`	Main unit for report design. Handles adding visual components to the canvas.
`PreviewU.pas`	Handles rendering and previewing the designed report before print/export.
`DataU.pas`	Manages connections and datasets for dynamic data binding.
`Main.pas`	Main form for switching between design/preview modes.

🕰️ Report Structure

Each report page contains the following sections:

- **Header (**: Static top section, repeated on every page.
- **Detail ()**: The core repeating section, dynamically created for each record.
- **Footer ()**: Bottom section for summaries, page numbers, etc.

Each section is implemented using `TRectangle` and holds child controls like `TLabel`, `TImage`, or `TText`.

🧱 Core Components

🕰️ `ZoneDetail`, `ZoneHeader`, `ZoneFooter`

Containers that represent each section. All rectangles are added here.

🕰️ `AddText`, `AddImage`

Functions that add labels or image boxes inside a zone (header/detail/footer).

🕰️ `ClerDetail`, `ClerReport`

Functions to clear existing visual content from the zones.

🕰️ `DESINGFACT`

Function that dynamically builds the full report layout by calculating how many pages are needed and cloning visual elements.

🕰️ `ComboZOMChangeTracking`

Handles zoom changes and resizes the preview page proportionally.

📖Image Rendering and Zooming

When rendering images in `PreviewU`, the zoom is applied using this formula:

```
hh := Round((v * 1754) / 150);
ww := Round((v * 1240) / 150);
```

This resizes the A4 page size proportionally based on the zoom percentage `v`.

Each preview image is aligned using:

```
TImage.Align := TAlignLayout.Bottom;
TImage.Margins.Top := 15;
```

This ensures each page image stacks vertically with spacing.

🧹 Memory and Cleanup

- `DisposeOf` is used for all components created at runtime.
- All clearing functions (`ClerDetail`, `FormDestroy`, etc.) carefully release both `Children` and `Components`.
- `Application.ProcessMessages` is used in loops to ensure the UI remains responsive during bulk deletion.

📦Data Binding

Each control can optionally be linked to a dataset field:

- A `TagString` or `Hint` can be used to store the field name.
- At runtime, the detail section clones are updated with live data for each record.

📄 PDF Export and Printing

• Each page (as `TImage`) is rendered and converted to a printable format.
• Pages can be saved to PDF using a third-party library or Windows print services.

🔗 Features Summary

• Visual layout design with drag-and-drop.
• Dynamic detail record rendering.
• Zoom support (25%–200%).
• Print and PDF export.
• Multi-page layout and auto pagination.

🕐 Roadmap Suggestions

• Add undo/redo support.
• Implement snapping and alignment guides.
• Enable field binding via UI dropdown.
• Save/load full report designs as JSON or XML.
• Introduce themes or styling templates.

💻 Developer Notes

• Target Platforms: Windows, Android (FMX cross-platform support)
• Required Delphi Version: Delphi 10.4+ recommended
• Open Source License: MIT (suggested)
• Author: Daoui Mahmoud (`@devlop0` on GitHub)

🌡 Support

If you enjoy using or contributing to DAOUI Reports, you can support development at: **https://coff.ee/daouimahmoud**

Feel free to suggest improvements or contribute on GitHub!