

Make it scale:

Guiding Principles for Low-Code Modeling Excellence

Eric Tieniber / Technical Lead, East / Mendix

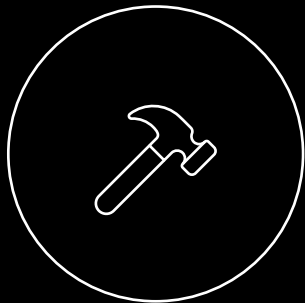
Kilian Croese / Low Code Solution Architect / Conclusion



Kilian Croese

Low Code Solution Architect, Conclusion

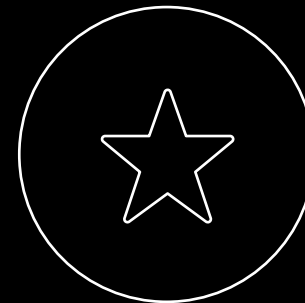
 /in/kiliancroese/



Developing with
Mendix since 2011



Involved in
50+ projects



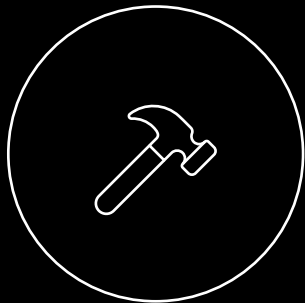
Most Valuable
Professional
since 2015



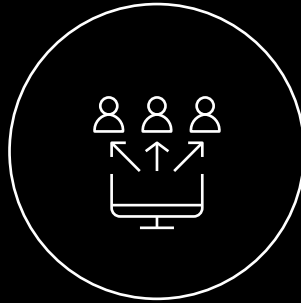
Eric Tieniber

Technical Lead, East, Mendix

 /in/tieniber/



Developing with
Mendix since 2014



Involved with 53
Mendix clients



5.7K App Store
Downloads

Agenda: Organization



Project



Module



Logic

**Organize your project into modules,
each with a clear purpose**

“

**A developer is constantly
focused on features and
processes.”**

Module Categories



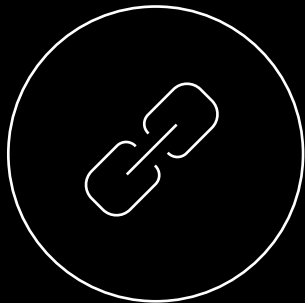
Process or Feature



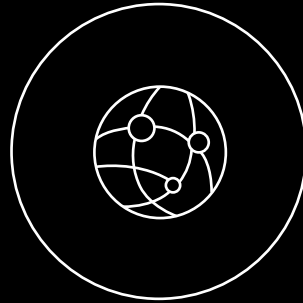
Administration



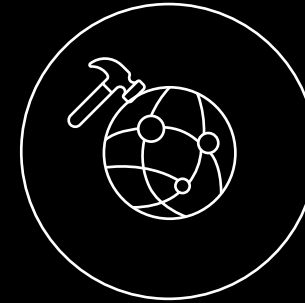
Transformation



Integration



App Store

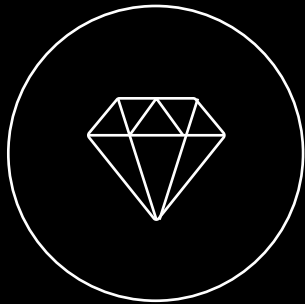


App Store
Customization

Demo

Group module content by function over type

Folder Structure

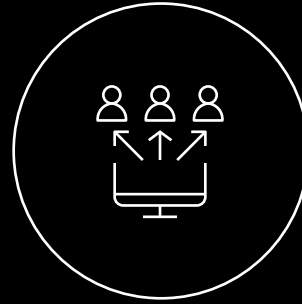


By Function

A folder per major function in a module.

Subfolders for smaller functions, each containing potentially many types of application objects.

Requires thought but enables rapid project navigation and refactoring.



By Type

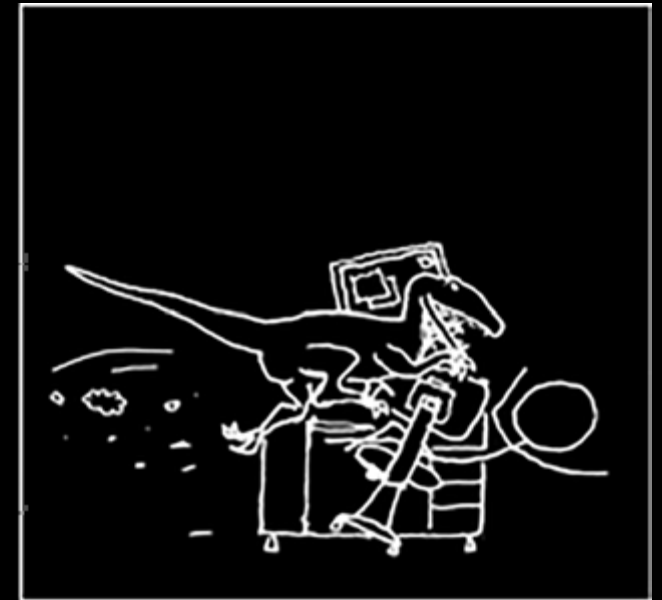
A folder for pages, another for microflows.

Easy to do, but ultimately useless for a fellow developer.

Demo

Design high-level microflows that read like a story

<https://www.xkcd.com/292/>



“At all times and under all circumstances, the programmer must keep the program within his intellectual grasp.”

– T. Harbron

“...I now suggest that we confine ourselves to the design and implementation of intellectually manageable programs.”

– E. Dijkstra

Demo

Wrap up



Project



Module



Logic

Q & A