Low-Code vs .High-Code

Explore the key differences between low-code development and traditional high-coding. Understand the strengths and limitations of each approach to make an informed decision for your next project.

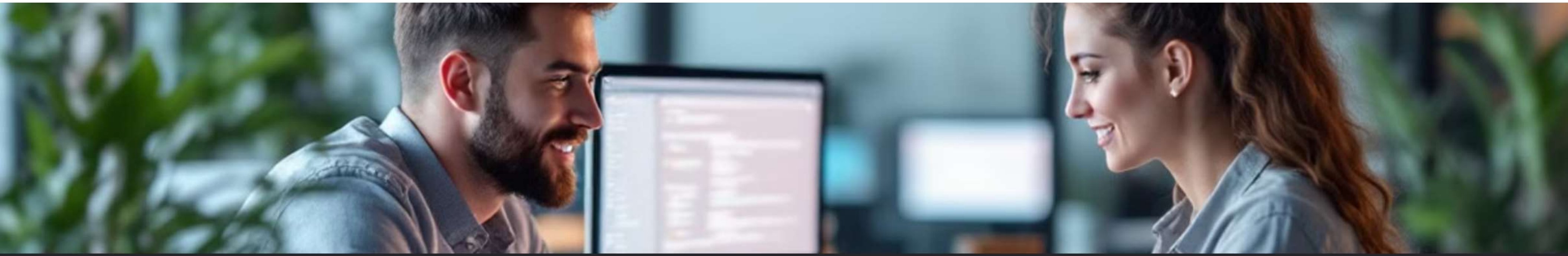
S by Sven Noomen

Sven Noomen

Microsoft composable architecture specialist | Dynamics, AI, Power Platform & Azure | creative, open minded, convincing, enthusiastic | Inspirational speaker | Loving nerdiness and the outdoors 🧑‍🚀 | Happy dad 👦 👶

CTO at InBluu

What do you know about low-code ? Any experiences?



Sven Noomen

1 The beginnings

Business analist, frustrated by lack of tools to do mock up’s

2 Under Water

Programming inside ERP and custom coding

3 New beginnings

Discover Low Code, training 100+ partners

2018 : Co-founded multiple low-code companies

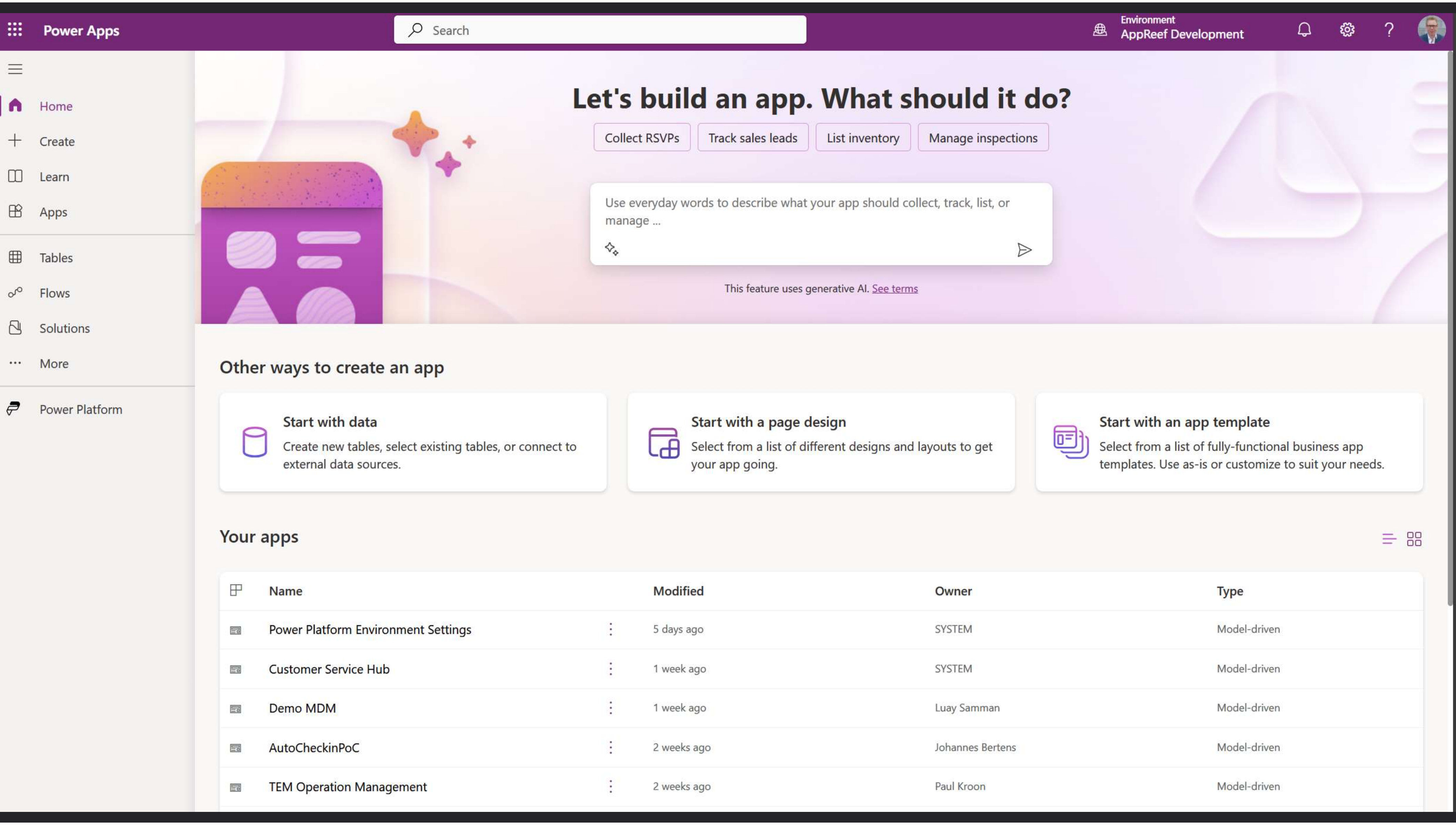
Comparing Low-Code to Custom Coding

Traditional Custom Coding

Requires extensive technical expertise, is time-consuming, and often results in complex, rigid applications that are difficult to maintain and update.

Low-Code Development

Leverages visual tools and pre-built components, enabling faster development, greater flexibility, and increased collaboration between technical and non-technical teams.



Let's build an app. What should it do?

Use everyday words to describe what your app should collect, track, list, or manage ...

This feature uses generative AI. See terms

Other ways to create an app

Start with data

Create new tables, select existing tables, or connect to external data sources.

Start with a page design

Select from a list of different designs and layouts to get your app going.

Start with an app template

Select from a list of fully-functional business app templates. Use as-is or customize to suit your needs.

Your apps

Name	Modified	Owner	Type
Power Platform Environment Settings	5 days ago	SYSTEM	Model-driven
Customer Service Hub	1 week ago	SYSTEM	Model-driven
Demo MDM	1 week ago	Luay Samman	Model-driven
AutoCheckinPoC	2 weeks ago	Johannes Bertens	Model-driven
TEM Operation Management	2 weeks ago	Paul Kroon	Model-driven

Ideal Use Cases for Low-Code

Process Automation

Low-code excels at streamlining and automating business processes, such as workflow management, customer onboarding, and data processing.

Rapid Prototyping and Mobile Apps

Low-code platforms enable quick iteration and experimentation, allowing teams to validate ideas and gather user feedback more efficiently.

Enterprise Integration

Low-code tools simplify the integration of custom applications with existing enterprise systems, promoting seamless data flow and operational efficiency.
Logging / event log etc. comes automatic

Citizen Development

By empowering non-technical employees to build applications, low-code fosters a culture of innovation and problem-solving within the organization.

Strong Opinions on Low Code

Requirements Analysis

Still needed, if you don’t know what to build, to tools don’t matter.
Yes, it is easer to change later in the project.

Methodology

Scrum
WaterScrum
Waterfall

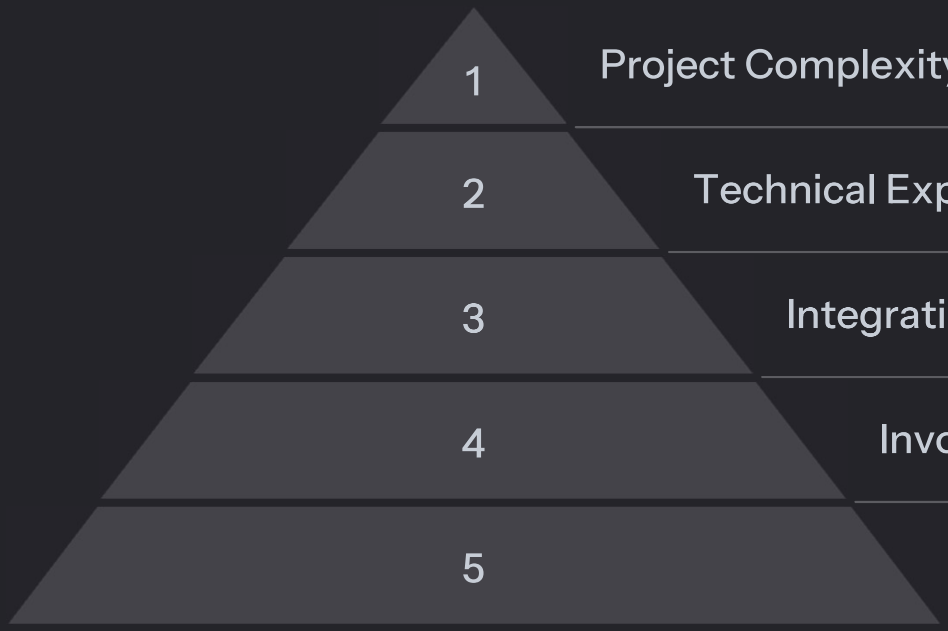
You need to know your basics

Database type, SharePoint List, Excel – When?
System-to-System Integrations, Event based, Serv. Bus
Streaming data, Push, Pull (API, file based)
UX / UI!

Speed

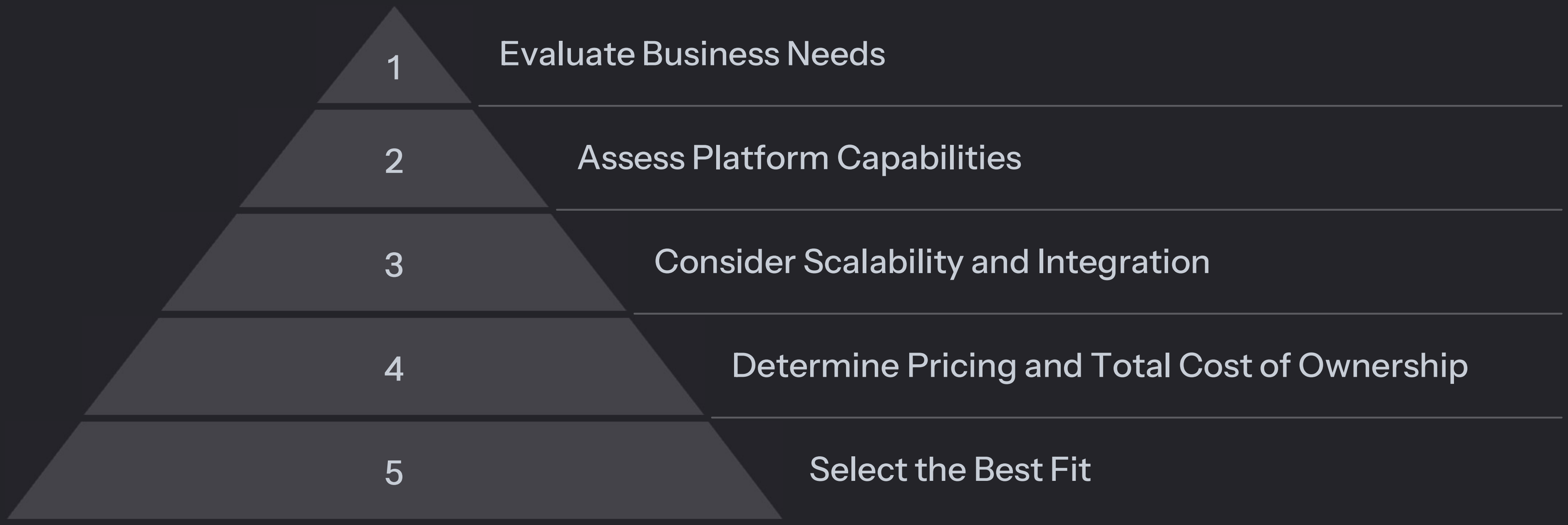
First PoC / MVPs come fast, getting to production and doing pipelines can still be painfull
Speed versus perceived value.
(Hamer and know where to hit)

Is low-code a possibility for my project?



1. Project Complexity (yes, low code can be an option!
2. Technical Expertise
3. Integration Requirements
4. Involvement by the business (citizen development)
5. Customization Needs (some things can’t be done in low-code)

Choosing the Right Low-Code Platform



1 Evaluate Business Needs

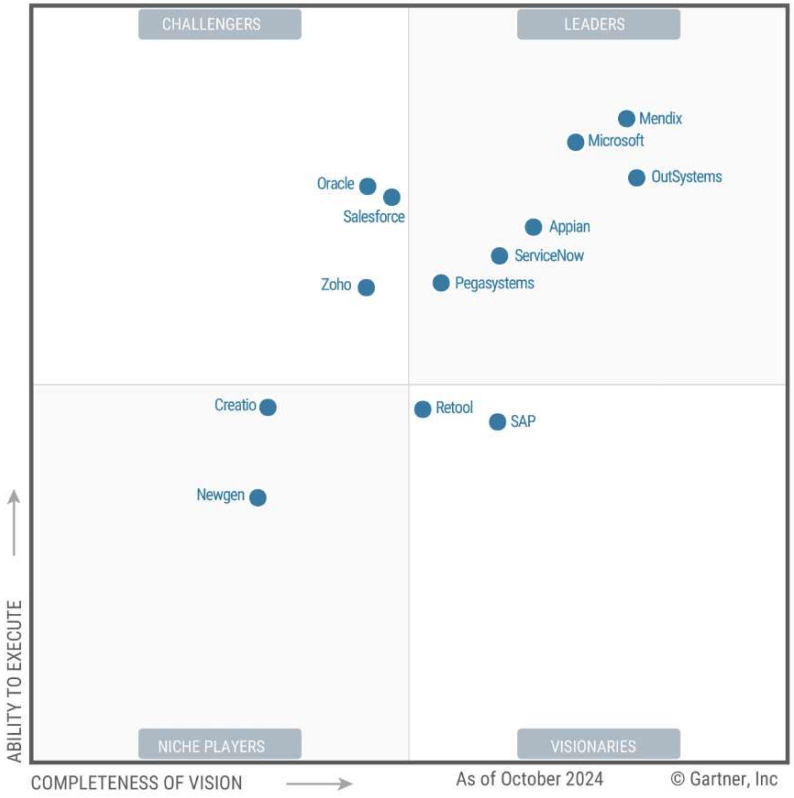
2 Assess Platform Capabilities

3 Consider Scalability and Integration

4 Determine Pricing and Total Cost of Ownership

5 Select the Best Fit

Figure 1: Magic Quadrant for Enterprise Low-Code Application Platforms



As of October 2024

© Gartner, Inc

Gartner

Considerations Low-Code Platforms

IP Protection limited

Typically deployed on customer environment.

Forget the basics

Governance, pipelines, user-rights, adoption, training, maintenance and support etc.

SaaS: Verwerkersovereenkomst

Roadmap

Your platform will have a fast-moving foundation layer.

Training

Keeping up to date is harder

Key Takeaways

1

Low-code development offers rapid prototyping and deployment, enabling faster time-to-market and empowering non-technical users.

2

Don’t ignore it, or your customer will ignore you

3

START, what are you waiting for?

The future is fusion