Is your debt out of control?

Find out before it’s too late – Are you taking more risks than you think?

Jan Matus (v 2.0)

Hi!

My name is Jan, and I'm a professional IT leader and founder of janmatus.com.

With great satisfaction, I help project managers and engineers find a common language, whether its:

- managing technical debt
- software architecture definition
- processes and technical management
- and more...

Experience

Founder
janmatus.com · Self-employed
Dec 2024 - Present · 4 mos
Remote

Technology Consulting | Training | Thought Leadership
...
...see more

Content Creation, Technical Consulting and +2 skills

Senior Principal System Architect
Mavenir · Full-time
Apr 2022 - Present · 3 yrs

Global Telecom Solutions | $10M+ O-RU Development
...
...see more

System Architecture, Product Development and +3 skills

Radio Architect
Xilinx · Full-time
Jun 2021 - Apr 2022 · 11 mos
Dresden i okolice

Productising Next-Gen Radio Platform
...
...see more

Architecture Reviews, Leadership and +3 skills

Nokia
7 yrs 7 mos

System Architect
May 2017 - May 2021 · 4 yrs 1 mo

Architecting High-Reliability and High-Reusability Embedded Radio Platforms
...
...see more

Team Leadership, Problem Solving and +6 skills

Product Owner
Dec 2015 - May 2017 · 1 yr 6 mos

Driving Product Development & Engineering Execution
...
...see more

Product Management, Agile Methodologies and +3 skills

Engineer, Embedded Software Development
Nov 2013 - Dec 2015 · 2 yrs 2 mos
Wrocław, woj. dolnośląskie, Polska

Developing High-Performance Embedded Radio Systems
...
...see more

Embedded Systems, C++ Development and +3 skills

REC Global
1 yr 10 mos
Wrocław, woj. dolnośląskie, Polska

Values

Clarity, Collaboration, Efficiency & Practicality

I’m proud to provide

5-star experience*

*Based on the previous webinar satisfaction rating 2025.

Goal

Be aware of the risk you're taking.

The “Surprise”

Technical debt is like a silent killer—everything seems fine until suddenly, nothing works.

- Unexpected crash of critical system
- Missed deadlines & Project delays
- Incomplete or broken features
- Security breaches
- Customer complaints and lost revenue

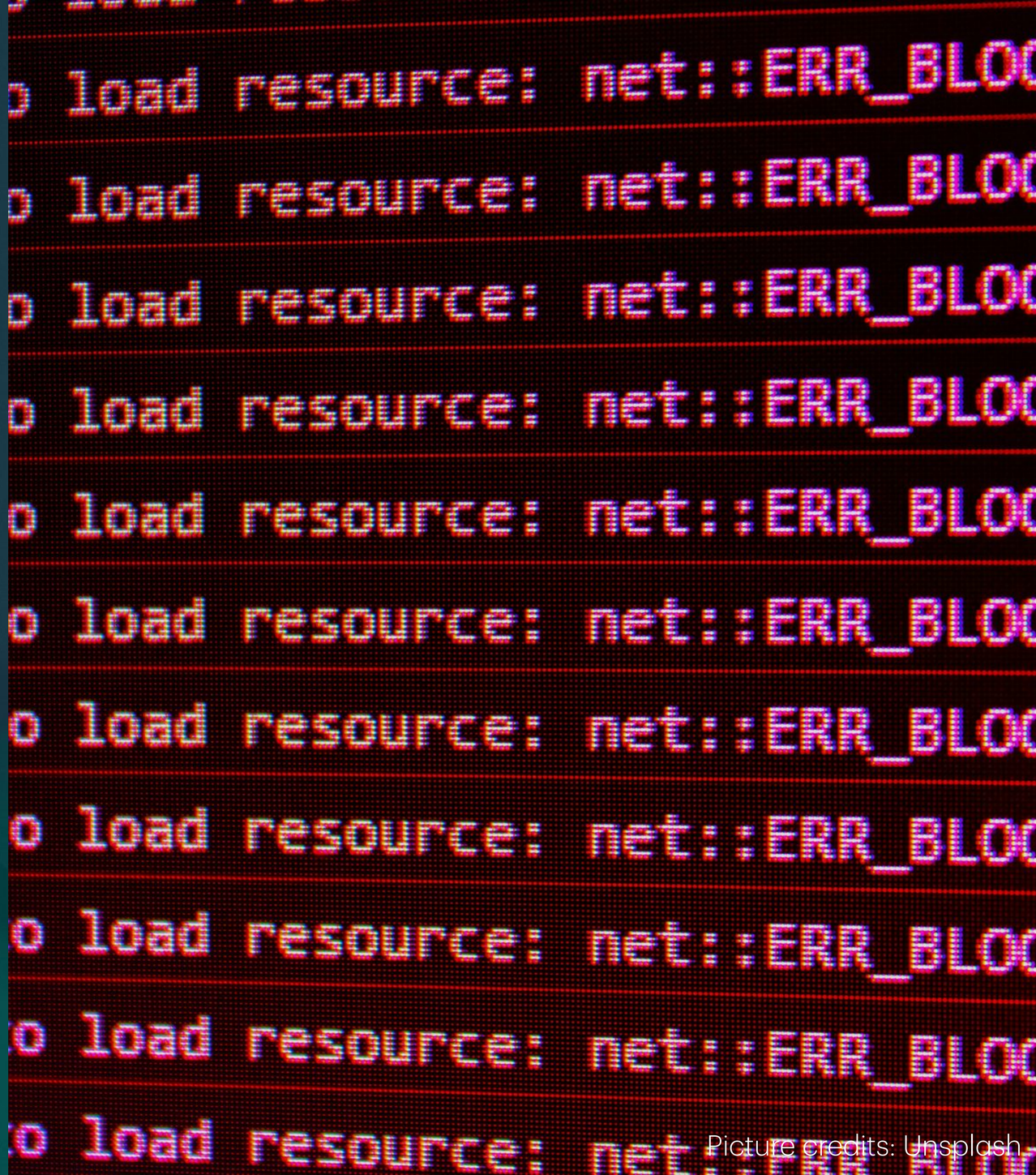
Picture credits: Unsplash

Have you ever wondered why it happens? 

The True Cost

You'll pay... continously.

- Slower development cycles
- Longer bug fixes and regressions
- Developer burnout & attrition
- More customer complaints & lost opportunities

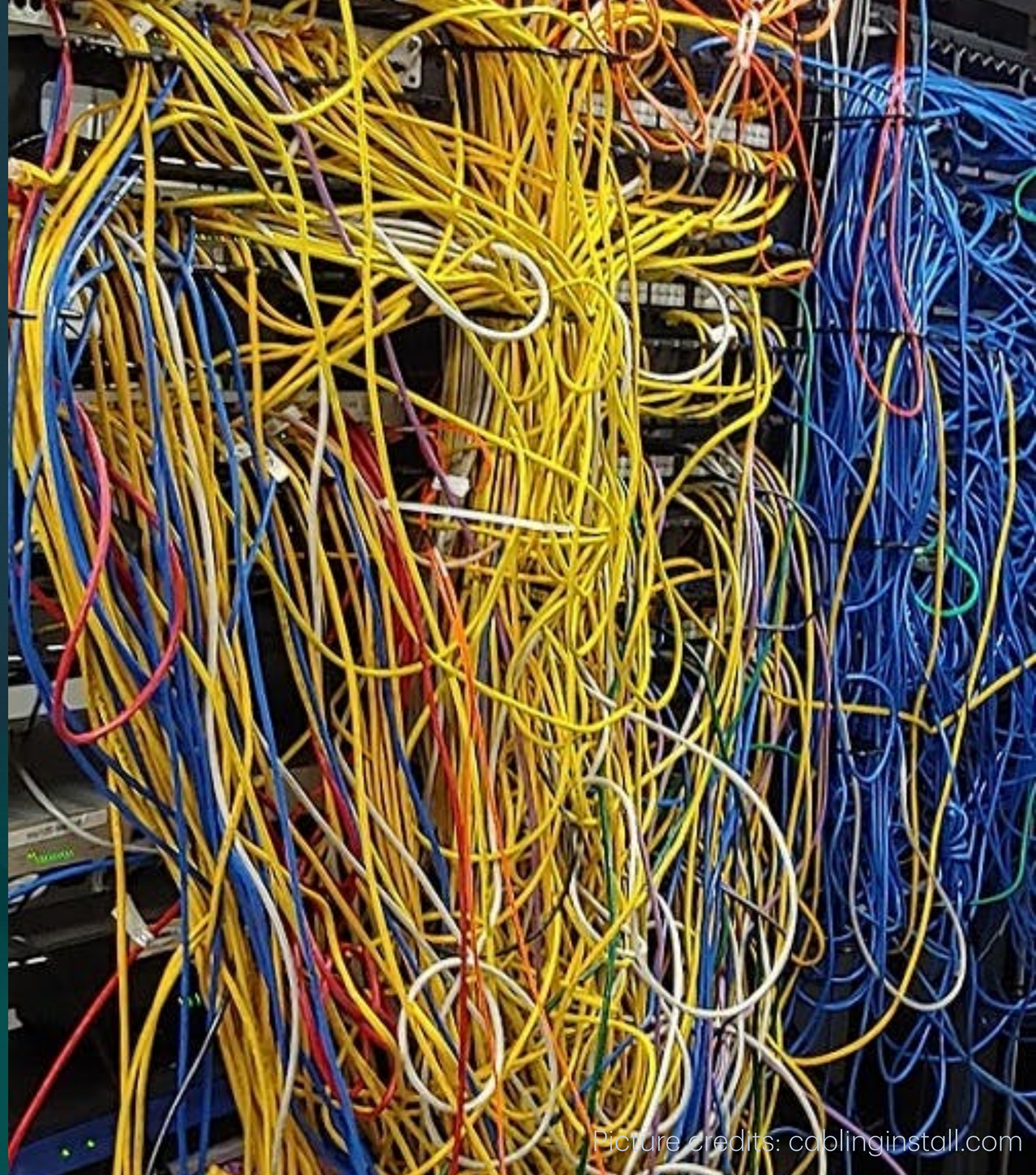


Picture credits: Unsplash

It starts with small things.

IT accumulates even as you watch this presentation.

- Business Pressures
- Lack of long-term planning
- Fear of changing the legacy systems.



Picture credits: cablinginstall.com

What is Technical Debt?

Shortcuts.

- Trade-offs that are made **for short-term speed** that create long-term challenges.
- Analogy: The "credit card" of software development.

Picture credits: Unsplash

Is all debt bad? 🤨

Let me know in the chat...

Intentional vs Unintentional

- What exactly is technical debt?
- When taking on debt is positive?
- Do you know that there are nearly 10 types of technical debt?

Picture credits: Unsplash

5 Biggest Myths

🚨 Myth #1: "Technical Debt is Just Bad Code.”

🚨 Myth #2: "If It Works, It’s Not Technical Debt."

🚨 Myth #3: "We’ll Fix It Later."

🚨 Myth #4: "Technical Debt is Always a Mistake.”

🚨 Myth #5: "Engineers Own Technical Debt.”

Picture credits: Unsplash

⚡ Checklist - how prepared are you?

You get one point for each confirmation.

Q1: We're tracking in backlog, code, specification & documentation.

Q2: Tech debt in my project has an owner.

Q3: There is a process defining when we take debt.

Q4: We're periodically scheduling refactoring in my project.

Share your score!
Extended checklist available in the bonus materials!

When to Overhaul?

- The default answer from the Engineering team is “No.”
- Expanding a system takes as much time as building from scratch
- Bug fixes take forever
- Fixing one thing breaks the other
- Significant productivity loss

Picture credits: Unsplash

Look out!

Just refactoring will not fix it!

You'll be back at square one without fixing:

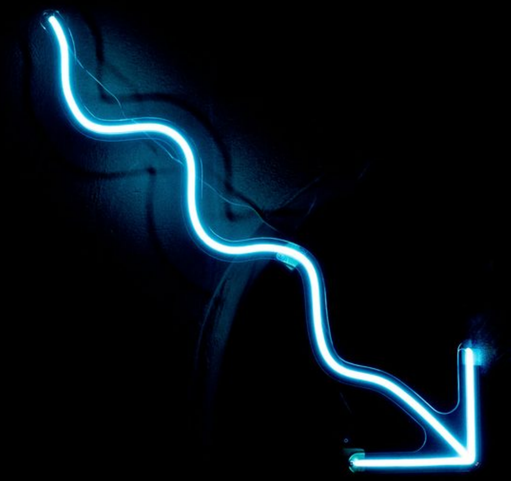
- Processes to monitor and consistently evaluate the debt
- Conscious debt taking
- Key Performance Indicators
- Ownership & Communication

Picture credits: Unsplash

Strategy to Reduce Debt

The hardest part is to start

- Audit/Track & Evaluate
- Prioritize debt - not all debt is worth settling
- Balance new features, bug fixes, and refactoring
- Educate yourself and others



Picture credits: Unsplash

Policies and Processes

A team without a tech debt policy is like a credit card without a spending limit.

- Define **when** and **why** to take on tech debt and **who** needs to approve it.
- Monitor KPIs to track tech debt's impact on defects & stability

Need a step-by-step guide for setting up tech debt policies?

Picture credits: Unsplash

? Feedback time!

What have you learned today?

Summary and Key Takeaway

- Tech debt is unavoidable but can be managed.
- Consistent tracking, prioritization, and structured reduction are key.
- Align technical and business goals for sustainable growth.

Picture credits: Unsplash

Thank you!