ENGINEERING & R&D · GUIDE

# The skills matrix *for engineering teams*

Engineering and R&D rarely fail for lack of clever people; they fail when a project needs a discipline the team is thin in, mechanical, electrical, software, systems, and nobody saw it coming. A skills matrix maps capability across every discipline against what the work actually requires, so a lead can see at a glance whether the team is genuinely equipped for the project in hand, and where it is not.

**Dr Alex J. Martin-Smith**
CMGR · MBA · LLM · DBA

**Reading time** 12 min · **Method** Upleashed 0 to 5 capability framework · **Updated** May 2026

THE SHORT ANSWER

An engineering skills matrix maps the team against the disciplines the work spans, mechanical, electrical, software and firmware, systems integration, test and validation, plus project and domain skills, scored on a clear scale, and reads each against the level the project requires. Use it to see whether the team is equipped for the work in hand and where it is short. In short: **it shows, discipline by discipline, whether your engineering team is genuinely on par to tackle the project, and exactly where capability falls below what the work needs.**

KEY TAKEAWAYS

- **Engineering work is multidisciplinary.** Modern projects span mechanical, electrical, software and systems; a gap in any one can stall the whole thing.
- **Map capability against what the work requires.** The question is not "are we good?" but "are we strong enough in each discipline for this project?".
- **Depth and integration both matter.** R&D needs deep specialists and people who can integrate across disciplines; the matrix shows both.
- **Surface the discipline gaps early.** Seeing a thin discipline before a project starts lets you develop, hire or partner in time.
- **Protect against concentration.** Critical expertise often sits with one engineer; the matrix flags where to build depth.

START HERE

# The question is *"strong enough for this?"*

Engineering and R&D teams are judged by whether they can deliver the project in front of them, and modern projects are stubbornly multidisciplinary. A skills matrix reframes the capability question from a vague "is the team any good?" to the one that actually matters: **is the team strong enough, in each discipline the work touches, to deliver this project?** That is a question about capability against requirement, and the matrix is built to answer it.

## Map the disciplines the work spans

An engineering matrix maps the team against the **disciplines** a project draws on: mechanical, electrical, software and firmware, systems integration, test and validation, plus the project-management and domain knowledge that tie them together. Each is its own competency, scored on a clear scale, because a team strong in mechanical design can still be dangerously thin in systems integration. Breaking engineering into its real disciplines is what lets a lead see where the team's strength genuinely lies, and where it does not.

## Read capability against the requirement

The insight comes from reading each discipline's capability **against the level the work requires**. A discipline can look fine in isolation and still fall short of what a demanding project needs, while another may comfortably exceed a modest requirement. Plotting current capability against the required level,

discipline by discipline, shows immediately whether the team is on par to tackle the project, and turns "I think we can do this" into a defensible, evidenced judgement.

### Depth and integration

R&D has a particular shape of need: it requires both **deep specialists**, who push a single discipline forward, and **integrators**, who can work across disciplines to make a whole system function. A good engineering matrix captures both: depth within each discipline, and the cross-disciplinary capability that turns separate competences into a working product. Innovation tends to happen at the intersections, so seeing where the team has integration strength, and where it is siloed, matters as much as raw depth.

---

WHY IT MATTERS NOW

## A thin discipline *stalls the project*

In engineering, a capability gap in one discipline does not just slow that part, it can block the whole project, because the disciplines depend on each other. Seeing discipline strength against requirement is how a lead avoids committing to work the team cannot yet deliver.

**8%**

GARTNER, 2024

of organisations have reliable workforce skills data, so most engineering leads judge readiness on gut feel.

**39%**

WEF, 2025

of workers' core skills are expected to change by 2030, as engineering tools, methods and domains evolve fast.

**63%**

WEF, 2025

of employers call skills gaps the biggest barrier to change; in R&D they surface as projects that stall or slip.

The danger in engineering is interdependence. A project that is strong on mechanical and electrical but thin on systems integration does not run at three-quarters speed, it stalls at integration, because that is where the disciplines must come together. And because deep expertise tends to concentrate in individuals, a critical discipline can rest on one engineer whose absence halts progress. A skills matrix counters both by making **discipline capability against requirement visible**: where the team meets

what the work needs, where it falls short, and where strength is dangerously concentrated. Seeing this before committing to a project lets a lead develop, hire or partner to fill the gap, rather than discovering it at the integration phase when it is most expensive to fix.

WHAT IT REVEALS

## Four things an engineering matrix reveals

Read against what the work requires, an engineering skills matrix reveals four things a project lead needs to know before committing. Each turns capability from a hunch into evidence.

REVEALS 01

### Project readiness

By comparing each discipline's capability to what the project needs, the matrix shows whether the team is genuinely equipped to deliver, before the commitment is made.

REVEALS 02

### The limiting discipline

It pinpoints the discipline most below requirement, the one likely to stall the project, so effort goes where the constraint actually is.

REVEALS 03

### Integration capability

It shows whether the team has the cross-disciplinary strength to make a whole system work, not just depth in isolated specialisms.

REVEALS 04

### Concentration risk

It flags critical disciplines resting on a single expert, so depth can be built before that person's absence halts the work.

The common thread is matching **capability to the demands of the work**. An engineering team is not strong or weak in the abstract; it is strong or weak relative to what a given project requires across each discipline. The matrix is the instrument that makes that relative judgement precise, so a lead can commit to projects the team can deliver, shore up the limiting discipline before it bites, and build the depth and integration that turn a group of specialists into a team that ships working systems.

THE SCALE BEHIND THE SCORES

# The 0 to 5 capability framework

An engineering matrix needs a scale that distinguishes someone learning a discipline from someone who can carry it on a demanding project. This framework, developed by Dr Alex J. Martin-Smith, draws that line at Level 3, works the discipline unsupervised to standard, with Level 4 plus the deep experts who push it forward and mentor others.

0

### Not required for the role EXCLUDED

The discipline is not part of this engineer's role, for example RF design for a mechanical engineer. Excluded from their score, not counted as a gap.

1

### In training / Junior WEIGHTING 25%

Learning the discipline, works under supervision. Up to 75% of the way there. Valuable developing capability, but not yet someone to rely on for project-critical work.

2

### Developing WEIGHTING 50%

More than 75% trained; handles routine work in the discipline alone, but novel or complex problems still need a senior engineer's review.

3

### Capable WEIGHTING 75% · CARRIES THE WORK

Fully capable, works the discipline unsupervised to standard across the normal range of problems. The level at which an engineer genuinely carries that discipline on a project.

4

### Specialist / Technical lead WEIGHTING 100%

Deep expertise; solves the hardest problems, sets technical direction in the discipline and mentors others. Your principal engineers and technical leads.

5

### Strategic ownership / Authority WEIGHTING 100%

Recognised authority who shapes architecture, standards and the technical strategy across projects. The purple flag marks your chief engineers and R&D leads.

### Aggregate the discipline, compare to requirement

Score each engineer 0 to 5 per discipline, and the weightings, Level 1 = 25% up to 3 = 75% and 4 and 5 = 100%, let you express the team's capability in each discipline as a percentage. Set against the level the project requires, also as a percentage, the comparison is immediate: meets it, falls short, or comfortably exceeds. Reading current against required, discipline by discipline, is what tells a lead whether the team is ready for the work.

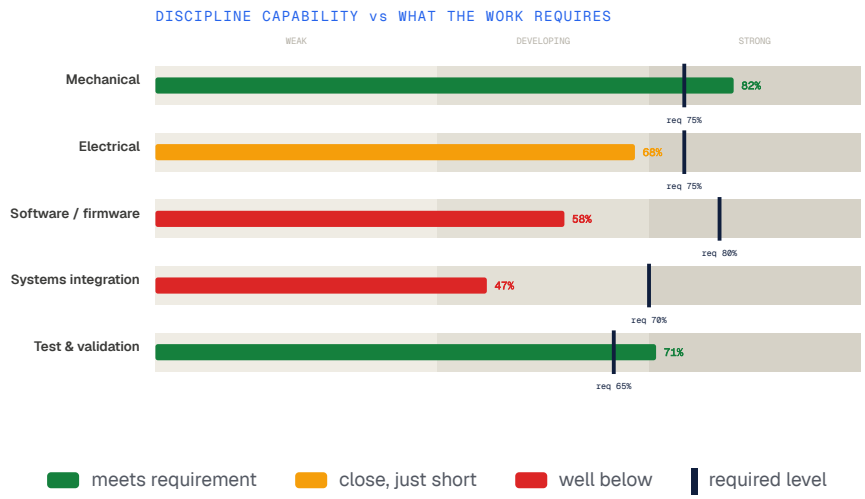
**A worked example.** Why one discipline decides readiness:

```
Project needs  systems integration at 70%; team is at 47%  →
the limiting discipline
mechanical 82% vs need 75%, electrical 68% vs 75%  →  broadly
fine
verdict  →  fix integration first — strength elsewhere will
not compensate.
```

SEE THE READINESS

## Each discipline against *what the work needs*

Here is the team's capability by discipline, each shown as a bar against the level the project requires, the dark tick. The shaded bands give a sense of weak, developing and strong. Where the bar clears the tick, the team is ready; where it falls short, that discipline is the risk. The picture says, at a glance, whether the team is on par for the project.




47%

**systems integration is the limiting discipline**, well below the 70% the project needs; fix it first

*Illustrative team on the Upleashed 0 to 5 framework. Each bar is a discipline's capability; the tick is the level the project requires.*

WHAT THE ENGINEERING LEAD READS HERE

- **Systems integration is the constraint.** At 47% against a 70% requirement, it is well short and shown red. Because integration is where the disciplines meet, this is where the project will stall, so it is the first and biggest priority.
- **Electrical is close, but watch it.** At 68% against 75%, it is just short (amber). A focused effort or one capable hire closes it; not a crisis, but not ready to ignore either.
- **Mechanical and test clear the bar.** Both exceed what the work requires (green). Strength here is welcome, but it cannot compensate for the integration gap, the project is only as ready as its weakest discipline.
- **Software needs a plan.** Below its demanding 80% requirement, firmware and software capability needs developing or hiring before the project depends on it, not during.

READY-TO-USE EXAMPLES

# Example disciplines to map for an engineering team

An engineering matrix should map the disciplines your projects span, plus the project and domain skills that tie them together. Here are ready-

to-adapt categories, a starting point to tailor to your field.

| Category | Examples to map (the columns) | Watch out for |
|---|---|---|
| **Core disciplines** | Mechanical, electrical / electronics, software and firmware, systems | A discipline that looks fine overall but is thin for a demanding project |
| **Integration & architecture** | Systems integration, architecture, requirements, interfaces | Plenty of depth but no one who can integrate across the disciplines |
| **Verification** | Test and validation, simulation, quality, compliance to standards | Under-mapping verification until a project hits qualification |
| **Domain & tools** | Domain knowledge, key CAD/EDA/PLM tools, relevant standards | Assuming tool skill equals engineering judgement in the discipline |
| **Project & collaboration** | Project management, problem-solving, communication, documentation | Mapping only technical depth and missing what delivers projects |

Map the disciplines your projects actually draw on, scored so Level 3 means an engineer can carry that discipline on the work unsupervised, and always read capability against the level a given project requires. Capture both depth within disciplines and the integration capability that joins them, since innovation happens at the intersections. As always, map the disciplines that matter for your field, keep the scores current as engineers grow and tools change, and use the matrix when scoping projects, not after committing to them.

## AVOID THESE

# Six mistakes on an engineering matrix

MISTAKE 01

### Judging capability in the abstract

"Are we good?" is the wrong question. Read each discipline against what the specific project requires.

MISTAKE 02

### Averaging across disciplines

A healthy overall average can hide a fatal gap. The team is only as ready as its weakest needed discipline.

MISTAKE 03

### Ignoring integration

Depth in silos is not enough. Map the cross-disciplinary capability that makes a whole system work.

MISTAKE 04

### Concentrated expertise

A critical discipline on one engineer is a stall risk. Build depth behind your specialists.

MISTAKE 05

### Mapping tools, not judgement

Knowing the CAD package is not engineering capability. Score the discipline, not just the toolset.

MISTAKE 06

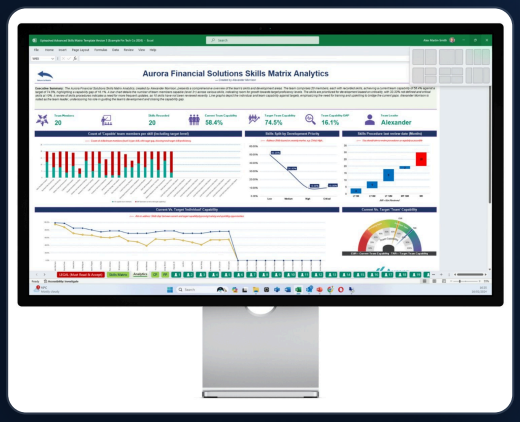
### Checking readiness too late

Discovering a gap mid-project is costly. Read the matrix when scoping, before you commit to the work.

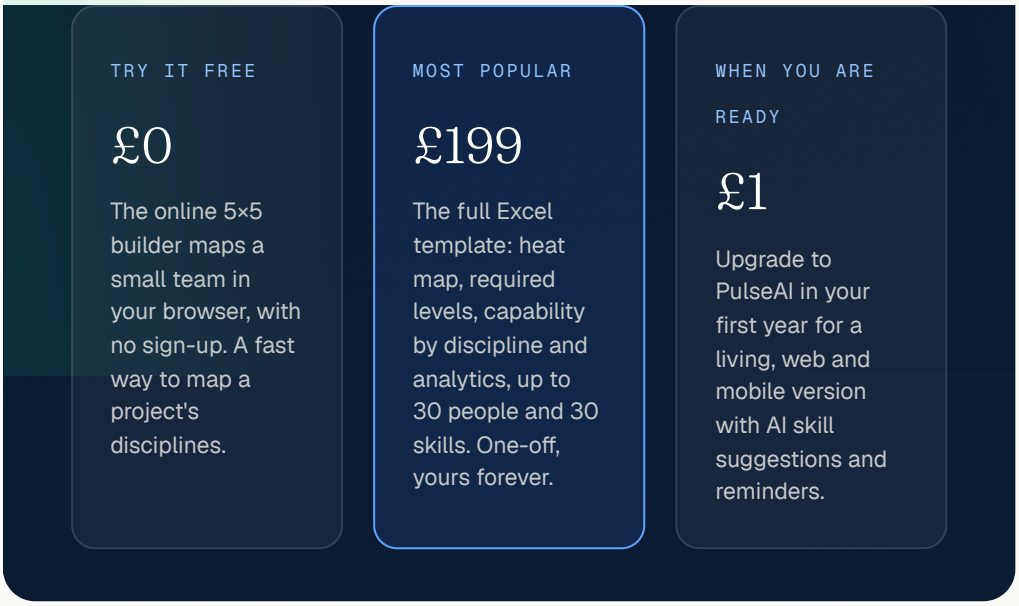
FROM GUT FEEL TO PROJECT READINESS

# The method is free. A ready-made matrix just makes discipline readiness *obvious.*

Everything here works in a blank spreadsheet, and that is a fine place to start. A purpose-built template just makes the engineering view effortless: score engineers on the 0 to 5 scale across the disciplines, set the level each project requires, and the capability per discipline calculates itself against the requirement, so the limiting discipline, the concentration risks and the integration gaps stand out, before you commit to a project rather than during it.




*The Advanced Excel Skills Matrix shows capability by discipline against required levels, the basis for reading project readiness and spotting the limiting discipline, all on the same 0 to 5 framework used throughout this guide.*




COMMON QUESTIONS

# Quick *answers*

### Q What is a skills matrix for an engineering team?

It is a grid mapping engineers against the disciplines the work spans, mechanical, electrical, software and firmware, systems integration, test and validation, plus project and domain skills, with a level in each cell. Read against what each project requires, it shows whether the team is equipped for the work and where it falls short.

### Q Why map by discipline?

Because engineering work is multidisciplinary and the disciplines depend on each other. A project strong on mechanical and electrical but thin on systems integration will stall where they must come together. Mapping each discipline separately, and reading it against requirement, is the only way to see whether the team is genuinely ready for the work.

### Q What does reading capability against requirement mean?

It means comparing each discipline's current capability to the level the specific project needs, rather than judging the team in the abstract. A discipline can look fine yet fall short of a demanding project, or comfortably exceed a modest one. The comparison shows immediately which disciplines are ready and which are the risk.

### Q How does it handle depth versus integration?

A good engineering matrix captures both: depth within each discipline, the specialists who push it forward, and the cross-disciplinary integration capability that makes a whole system work. Innovation tends to happen at the intersections, so seeing where the team can integrate, not just where it is deep, matters as much as raw specialist strength.

### Q How do I find the discipline that will stall a project?

Compare each discipline's capability to its required level and look for the biggest shortfall, especially in integration. Because the disciplines are interdependent, the team is only as ready as its weakest needed discipline, so the one furthest below requirement is the limiting constraint and the first place to invest, before the project depends on it.

### Q Does this work for R&D as well as delivery engineering?

Yes. R&D has the same multidisciplinary shape and an even greater need for both deep specialists and integrators. The matrix helps an R&D lead see whether the team has the depth to push each discipline and the integration capability to turn that into working innovation, and where concentrated expertise needs a backup built behind it.

## ABOUT THE AUTHOR

### Dr Alex J. Martin-Smith

CMGR · MBA · LLM · DBA

Alex is the creator of the Upleashed capability framework that powers Skills Matrix Template, the award-winning Excel skills matrix. A Chartered Manager with an MBA, an LLM and a doctorate in business administration, he has spent more than two decades helping operations, HR and quality teams turn capability from a gut feel into something they can measure, manage and prove.

Connect on LinkedIn: linkedin.com/in/alexmartinsmith

Dr Alex J. Martin-Smith

## SOURCES


Gartner. (2024). *Talent management research: Workforce skills data.* Gartner.

Martin-Smith, A. J. (n.d.). *The 0 to 5 capability framework.* Upleashed Limited. https://upleashed.com/capability-framework/

World Economic Forum. (2025). *The future of jobs report 2025.* World Economic Forum.


# Know you are ready *before you commit.*

You now have the engineering method. The quickest way to start is to list the disciplines your next project spans, score the team in each, and set the level the work requires. The discipline that falls furthest below the line is exactly where to develop, hire or partner, before the project depends on it.

Try the free 5×5 builder →

Get the template, £199

Award-winning method · 148,000+ teams · instant download · single-team licence

---

**Skills Matrix Template** — the award-winning Excel skills matrix by Upleashed. skillsmatrixtemplate.com

Powered by Upleashed Limited · upleashed.com