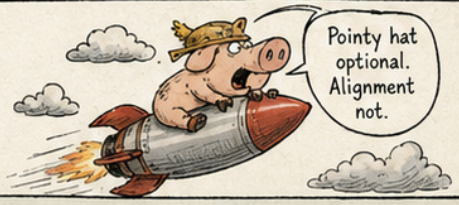
One Team. One Plan. Many Snacks.

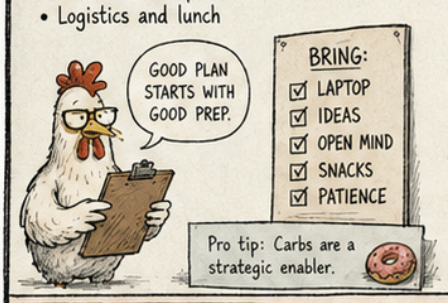
PI PLANNING

ALIGN TODAY. DELIVER TOMORROW.



1 PREPARE (BEFORE THE BIG DAY)

- Set the stage
- Pre-reads and pre-work
- Logistics and lunch



2 TEAM BREAKOUTS

Teams build their plan and identify dependencies.



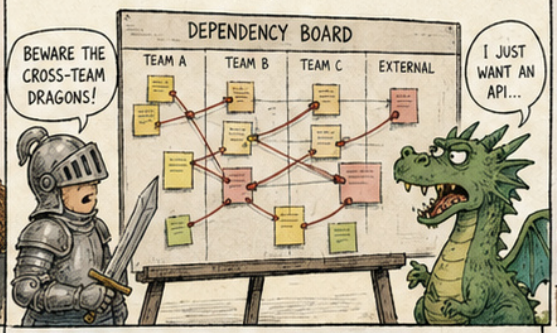
3 PLAN REVIEW

Teams present. Everyone learns.
Questions welcome. Surprises not.



4 IDENTIFY & RESOLVE DEPENDENCIES

Surface them. Own them. Slay them.



5 CONFIRM PLAN

Confidence check. Risks called out.
Plan committed.



6 ROAM & PROBLEM SOLVE

Leaders and experts roam to help unblock and strengthen plans.



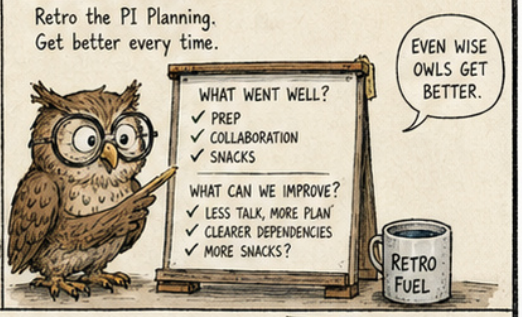
7 FINAL PLAN & NEXT STEPS

Publish the plan. Communicate widely.
Then... celebrate!



8 INSPECT & ADAPT

Retro the PI Planning.
Get better every time.



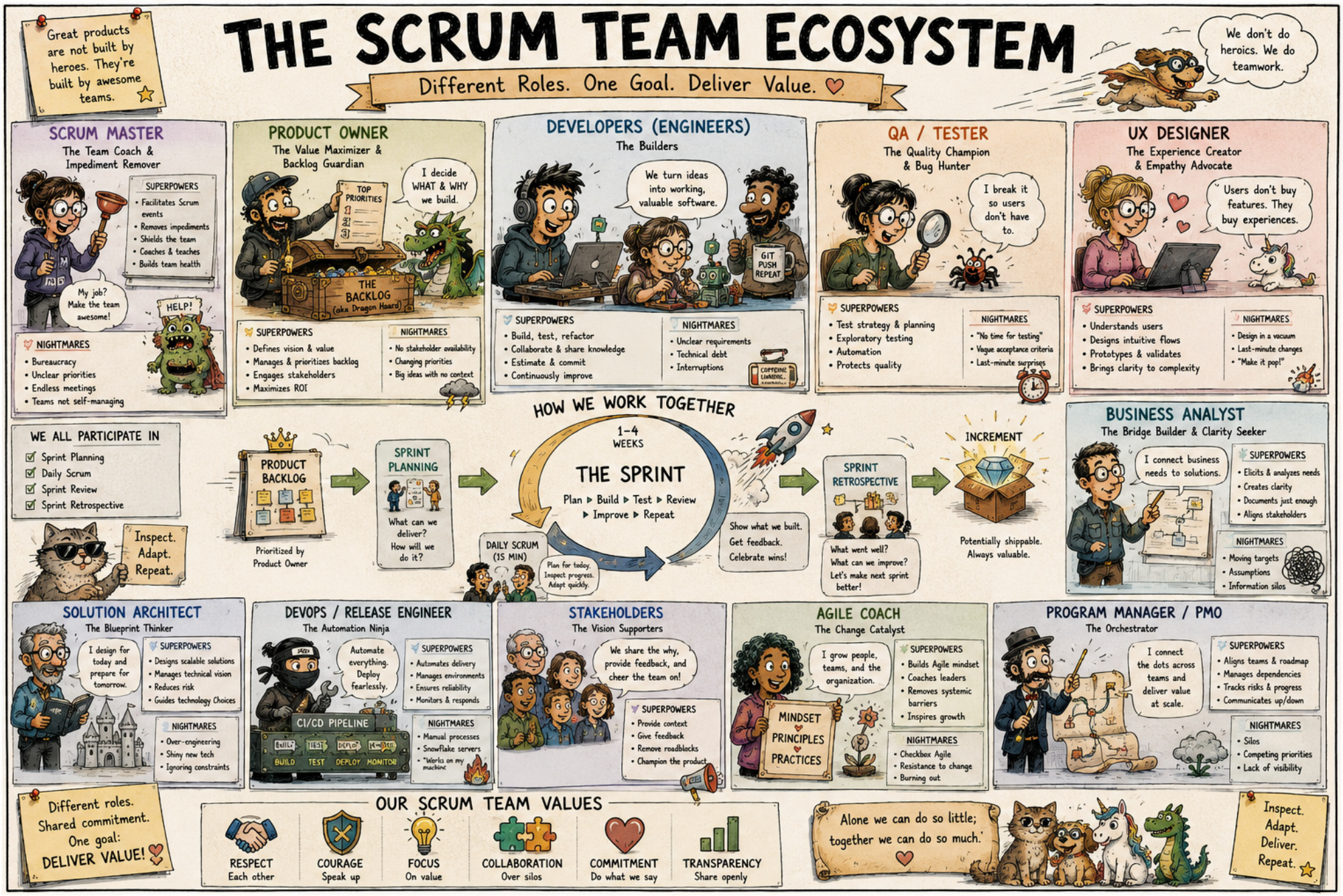
PI PLANNING SUCCESS FACTORS

- ☑ PREPARE WELL
- ☑ ENGAGE ALL
- ☑ STAY FOCUSED
- ☑ SOLVE TOGETHER
- ☑ COMMIT & DELIVER ☆

REMEMBER:



It's not about the perfect plan. It's about the better plan—together. ♥


Great products are not built by heroes. They're built by awesome teams.
THE SCRUM TEAM ECOSYSTEM
Different Roles. One Goal. Deliver Value.
We don't do heroics. We do teamwork.
SCRUM MASTER
The Team Coach & Impediment Remover
My job? Make the team awesome!
HELP!
SUPERPOWERS
• Facilitates Scrum events
• Removes impediments
• Shields the team
• Coaches & teaches
• Builds team health
NIGHTMARES
• Bureaucracy
• Unclear priorities
• Endless meetings
• Teams not self-managing
PRODUCT OWNER
The Value Maximizer & Backlog Guardian
TOP PRIORITIES
1
2
3
I decide WHAT & WHY we build.
THE BACKLOG (aka Dragon Hoard)
SUPERPOWERS
• Defines vision & value
• Manages & prioritizes backlog
• Engages stakeholders
• Maximizes ROI
NIGHTMARES
• No stakeholder availability
• Changing priorities
• Big ideas with no context
DEVELOPERS (ENGINEERS)
The Builders
We turn ideas into working, valuable software.
GIT PUSH REPEAT
SUPERPOWERS
• Build, test, refactor
• Collaborate & share knowledge
• Estimate & commit
• Continuously improve
NIGHTMARES
• Unclear requirements
• Technical debt
• Interruptions
QA / TESTER
The Quality Champion & Bug Hunter
I break it so users don't have to.
SUPERPOWERS
• Test strategy & planning
• Exploratory testing
• Automation
• Protects quality
NIGHTMARES
• "No time for testing"
• Vague acceptance criteria
• Last-minute surprises
UX DESIGNER
The Experience Creator & Empathy Advocate
Users don't buy features. They buy experiences.
SUPERPOWERS
• Understands users
• Designs intuitive flows
• Prototypes & validates
• Brings clarity to complexity
NIGHTMARES
• Design in a vacuum
• Last-minute changes
• "Make it pop!"
WE ALL PARTICIPATE IN
Sprint Planning
Daily Scrum
Sprint Review
Sprint Retrospective
Inspect. Adapt. Repeat.
HOW WE WORK TOGETHER
PRODUCT BACKLOG
Prioritized by Product Owner
SPRINT PLANNING
What can we deliver? How will we do it?
1–4 WEEKS
THE SPRINT
Plan ▸ Build ▸ Test ▸ Review ▸ Improve ▸ Repeat
DAILY SCRUM (15 MIN)
Plan for today. Inspect progress. Adapt quickly.
Show what we built. Get feedback. Celebrate wins!
SPRINT RETROSPECTIVE
What went well? What can we improve? Let's make next sprint better!
INCREMENT
Potentially shippable. Always valuable.
BUSINESS ANALYST
The Bridge Builder & Clarity Seeker
I connect business needs to solutions.
SUPERPOWERS
• Elicits & analyzes needs
• Creates clarity
• Documents just enough
• Aligns stakeholders
NIGHTMARES
• Moving targets
• Assumptions
• Information silos
SOLUTION ARCHITECT
The Blueprint Thinker
I design for today and prepare for tomorrow.
SUPERPOWERS
• Designs scalable solutions
• Manages technical vision
• Reduces risk
• Guides technology choices
NIGHTMARES
• Over-engineering
• Shiny new tech
• Ignoring constraints
DEVOPS / RELEASE ENGINEER
The Automation Ninja
Automate everything. Deploy fearlessly.
CI/CD PIPELINE
BUILD
TEST
DEPLOY
MONITOR
SUPERPOWERS
• Automates delivery
• Manages environments
• Ensures reliability
• Monitors & responds
NIGHTMARES
• Manual processes
• Snowflake servers
• "Works on my machine"
STAKEHOLDERS
The Vision Supporters
We share the why, provide feedback, and cheer the team on!
SUPERPOWERS
• Provide context
• Give feedback
• Remove roadblocks
• Champion the product
AGILE COACH
The Change Catalyst
I grow people, teams, and the organization.
MINDSET
PRINCIPLES
PRACTICES
SUPERPOWERS
• Builds Agile mindset
• Coaches leaders
• Removes systemic barriers
• Inspires growth
NIGHTMARES
• Checkbox Agile
• Resistance to change
• Burning out
PROGRAM MANAGER / PMO
The Orchestrator
I connect the dots across teams and deliver value at scale.
SUPERPOWERS
• Aligns teams & roadmap
• Manages dependencies
• Tracks risks & progress
• Communicates up/down
NIGHTMARES
• Silos
• Competing priorities
• Lack of visibility
Different roles. Shared commitment. One goal: DELIVER VALUE!
OUR SCRUM TEAM VALUES
RESPECT Each other
COURAGE Speak up
FOCUS On value
COLLABORATION Over silos
COMMITMENT Do what we say
TRANSPARENCY Share openly
Alone we can do so little; together we can do so much.
Inspect. Adapt. Deliver. Repeat.

THE AGILE COACH POSTER

I don't have all the answers. I ask better questions.

GROWING PEOPLE. EMPOWERING TEAMS. TRANSFORMING ORGANIZATIONS.

Agile is not what I do. It's what I help others become.



WHO I AM

- Trusted Advisor
- Change Catalyst
- System Thinker
- Teacher & Mentor
- Listener & Learner
- Challenger (with care)
- Culture Builder
- Empathy Driven
- Outcome Obsessed

MY SUPERPOWER?

I help people and organizations learn, adapt and grow.

MINDSET → BEHAVIOR → RESULTS

People first. Then method. Always purpose.

MY MINDSET

- Growth Mindset
- Curious & Open
- Non-Judgmental
- Empathetic
- Systems Thinker
- Adaptive & Flexible
- Patient & Persistent
- Data Informed
- Always Learning

WHAT I DO

- Assess current state
- Facilitate learning & workshops
- Coach individuals & teams
- Improve ways of working
- Remove impediments
- Build self-organizing teams
- Strengthen Agile mindset
- Guide transformations
- Measure & amplify impact

WHAT I FOCUS ON

PEOPLE	TEAMS	ORGANIZATION	PRACTICES	CULTURE	OUTCOMES
Help individuals and teams grow skills, confidence and ownership.	Enable collaboration, trust, and high performance.	Foster alignment, flow of value, and agility at scale.	Promote simple, effective Agile practices.	Nurture an Agile mindset and psychological safety.	Drive meaningful results and continuous value.

I don't implement Agile. I help YOU make Agile real.

I WORK WITH

- Teams (all kinds)
- Scrum Masters
- Product Owners
- Leaders & Managers
- Stakeholders
- HR / People Teams
- PMO & Portfolio Teams
- Communities of Practice
- The entire organization

At any level. In any role.

COACHING STANCES I USE

- Partner – side by side
- Teacher – when needed
- Mentor – sharing experience
- Facilitator – create space
- Observer – see the system
- Challenger – question assumptions
- Supporter – encourage & empower



HOW I COACH (THE COACHING FLOW)

1. CONNECT	2. CLARIFY	3. EXPLORE	4. EXPERIMENT	5. REFLECT	6. GROW
Build trust and understand context, people and purpose.	Define goals, opportunities and success together.	Ask powerful questions. Uncover insights.	Encourage small experiments. Safe to try.	Inspect, learn and adapt.	Sustain progress. Build capability. Repeat.

I ask. You discover. We grow.

TOOLS & TECHNIQUES IN MY TOOLBOX

Powerful Questions · Active Listening · Coaching Models (GROW, TGROW, CLEAR, etc.) · Retrospectives That Matter · Liberating Structures · Systems Thinking · Metrics that Drive Behavior (Flow, Value, etc.)

No silver bullet. Right tool. Right context.

AGILE VALUES I NURTURE

- Individuals & interactions over processes & tools
- Working software over comprehensive docs
- Customer collaboration over contract negotiation
- Responding to change over following a plan

Values over rules. Principles over prescriptions.

A DAY IN THE LIFE (EXAMPLE)

8:30	Reflect, plan, connect
9:30	Team coaching / workshop
11:00	Stakeholder alignment
1:00	Observe / partner with teams
2:30	Remove impediments
3:30	Design learning / session
4:30	Check progress & learn
5:00	Capture insights & plan tomorrow

ONE STEP BETTER

Every day is different. Impact is the same.

WHAT MAKES ME SUCCESSFUL

- Trust & credibility
- Strong relationships
- Clear communication
- Deep empathy
- Strategic perspective
- Business acumen
- Continuous learning
- Patience & persistence
- Focus on value & impact



SIGNS OF A HEALTHY AGILE CULTURE

- People take ownership
- Teams collaborate & trust
- Transparency is normal
- Learning is continuous
- Feedback is welcomed
- Adaptation is fast
- Focus is on outcomes
- Happiness is visible



CHALLENGES I FACE

- Resistance to change
- Siloed thinking
- Misaligned incentives
- Overloaded priorities
- Unrealistic expectations
- Lack of trust
- Impatience for results



I don't avoid challenges. I help people face them.

WHAT I DO WHEN IT'S TOUGH

- Listen more
- Stay curious
- Reframe the mindset
- Start small
- Celebrate wins
- Keep showing up

Progress not Perfection

Small steps. Consistent actions. Big impact.

COACHING IS NOT...

- ✗ Telling people what to do
- ✗ Doing the work for them
- ✗ Being an expert in everything
- ✗ Fixing people
- ✗ One-size-fits-all solutions
- ✗ A quick workshop

It's a partnership, not a prescription.

KPIs I CARE ABOUT

- Team engagement
- Velocity & flow
- Lead time improvements
- Quality & value delivered
- People growth
- Customer satisfaction
- Business outcomes

Measure what matters. Improve what you measure.

I LIGHT THE WAY. YOU WALK THE PATH. TOGETHER WE GROW.

AGILE IS A JOURNEY. I'M HERE TO HELP YOU ENJOY THE RIDE.

BETTER PEOPLE. BETTER TEAMS. BETTER BUSINESS. THAT'S THE AGILE PROMISE.

BE CURIOUS. BE KIND. BE THE CHANGE.

COACH TODAY. EMPOWER FOREVER.

THE PRODUCT OWNER POSTER

Vision Keeper. Value Maximizer. Backlog Boss.

I'm not here to do the work. I'm here to make sure we build the RIGHT work.

The best products solve real problems and deliver value!

WHO IS THE PRODUCT OWNER?

The Product Owner is responsible for maximizing the value of the product and managing the work of the development team.

- Represents stakeholders & customers
- Owns the Product Backlog
- Defines what to build & why
- Prioritizes for value
- Ensures transparency
- Accepts or rejects work



WHAT I AM NOT

- Not the Project Manager
- Not the Team Manager
- Not a Task Assigner
- Not a Micromanager
- Not a Technical Expert
- Not a "Feature Factory"
- Not Always Available
- Not Afraid to Say NO!

NO!

MY SUPERPOWERS

- Vision & Strategy
- Empathy for Customers
- Prioritization (Say NO!)
- Market & Domain Knowledge
- Communication & Influence
- Data-Informed Decisions
- Adaptability
- Decisive (when it matters!)

I SUCCEED WHEN...

- Customers love the product
- Business goals are achieved
- The team is empowered
- Value is delivered frequently
- We learn and adapt continuously
- We build the right thing!

MY CORE RESPONSIBILITIES

UNDERSTAND THE PROBLEM	DEFINE THE VISION & STRATEGY	MANAGE THE PRODUCT BACKLOG	PRIORITIZE FOR MAXIMUM VALUE	COLLABORATE CONSTANTLY	VALIDATE & ACCEPT WORK
Research, listen, observe. Know the users, market, business goals & pain points.	Create and communicate the product vision, goals and roadmap.	Create, refine, groom and maintain a healthy backlog.	Order the backlog to maximize value, reduce risk, and meet goals.	Work with team, stakeholders, users and leadership.	Ensure work meets the Definition of Done. Accept or reject.

THE PRODUCT BACKLOG

A living, breathing, prioritized list of everything we might build.

- Features
- User Stories
- Enhancements
- Bugs
- Spikes / Research
- Technical Debt
- Enablers

Ordered by value, not by effort or who screams the loudest.

BACKLOG REFINEMENT (AKA GROOMING)

ADD	CLARIFY	BREAK DOWN	ESTIMATE (IF NEEDED)	PRIORITIZE
Capture ideas from anywhere.	Add details, acceptance criteria.	Split into smaller, valuable pieces.	Understand size, complexity, uncertainty.	Order based on value & impact.

INVEST IN GOOD STORIES

- **I** Independent – Stand alone
- **N** Negotiable – Open to discussion
- **V** Valuable – Delivers value
- **E** Estimable – Can be sized
- **S** Small – Small enough to deliver
- **T** Testable – Acceptance is clear



STAKEHOLDERS & COMMUNICATION

- **Users & Customers** → Understand needs, validate solutions.
- **Business & Leadership** → Align on goals, report outcomes.
- **Development Team** → Collaborate daily, clarify & decide.
- **Other Teams** → Remove roadblocks, share & align.

PRIORITIZATION FRAMEWORKS I USE

VALUE (Impact)	RISK (What if we don't do it?)	EFFORT (Cost)	CONFIDENCE (How sure are we?)	DEPENDENCIES (What comes first?)	STRATEGIC FIT (Does it move the needle?)

It's not about what's urgent. It's about what's important.

KPW I WORK WITH TEEAM (Outcomes over Output)

Sprint Planning	Daily Scrum	During the Sprint	Sprint Review	Sprint Retrospective
Share context, goals and priorities.	Answer questions. Remove blockers.	Available for clarification. Adjust as needed.	Celebrate! Gather feedback. Learn & adapt.	Reflect and improve together.

A DAY IN THE LIFE OF A PO

8:30	Check messages, stakeholder updates
9:00	Refine backlog & review feedback
10:30	Stakeholder meeting
11:30	Work with team / clarify stories
12:30	Lunch (and maybe more backlog 🙂)
1:30	Analyze data & user insights
2:30	Prioritize & adjust roadmap
3:30	Remove blockers / make decisions
4:30	Review progress & plan tomorrow

PO FUEL

Every day is different. Every decision creates impact.

CHALLENGES I FACE

Too many opinions! · Changing priorities! · Not enough information! · Saying NO (is hard!) · Limited time, endless work!

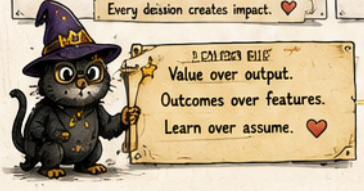
NO!

I don't have all the answers. I ask the right questions.

SIGNS OF PRODUCT OWNER SUCCESS

- Team building the right things
- problems?
- **CUSTOMER SATISFACTION** Are users happy?
- **BUSINESS OUTCOMES** Are we meeting our business goals?
- **BACKLOG HEALTH** Is the backlog well-groomed, prioritized & understood?

Happy users. Happy business. Happy team!



I DON'T BUILD THE PRODUCT.
I HELP THE TEAM BUILD THE RIGHT PRODUCT.
TOGETHER, WE CREATE VALUE!



REMEMBER

- Inspect.
- Adapt.
- Deliver.
- Repeat.

Not a Manager. Not a Boss. A Servant Leader & Agile Wizard!

THE AGILE WIZARD POSTER

SCRUM MASTER: SERVANT LEADER. TEAM COACH. IMPEDIMENT DESTROYER.

My magic doesn't create the work... It helps the team make it awesome!

WHO IS THE SCRUM MASTER?

The Scrum Master is a servant leader who helps the Scrum Team and the organization understand and adopt Agile values, principles, and practices.

I don't do the work for the team. I empower the team to do their best work.





MY MISSION

Create the conditions for my team to do their best work and deliver value continuously.

WHAT I AM NOT

- ✗ Not the Project Manager
- ✗ Not the Team Manager
- ✗ Not the Task Assigner
- ✗ Not the Decision Maker
- ✗ Not the Hero
- ✗ Not above the Team

I serve the team. I don't lead it.

MY SUPERPOWERS

- Facilitate amazing ceremonies
- Remove impediments (POOF!)
- Coach the team & organization
- Protect the team from distractions
- Help the team self-organize
- Promote continuous improvement
- Build a culture of trust & safety
- Champion Agile values

POOF!

WHAT I DO EVERY DAY



CEREMONIES I FACILITATE

Sprint Planning	Plan the work together
Daily Scrum	Inspect progress. Adjust the plan.
Sprint Review	Show value. Get feedback.
Sprint Retrospective	Reflect. Decide. Improve.

IMPEDIMENTS I DESTROY

I fight anything that slows the team down!



Spot it early. Raise it fast. Remove it together.

KEY RESPONSIBILITIES

- ☑ Ensure Scrum is understood & adopted
- ☑ Help the team be effective
- ☑ Remove impediments
- ☑ Protect the team's focus
- ☑ Facilitate Scrum events
- ☑ Coach the team in self-organization
- ☑ Collaborate with Product Owner
- ☑ Coach the organization on Agile
- ☑ Promote continuous improvement

KPIs I CARE ABOUT

- Team Happiness
- Sprint Predictability
- Impediments Resolved
- Team Velocity Trend
- Retrospective Actions Completed
- Escaped Defects
- Cycle Time Improvement
- Deployment Frequency

Healthy Team = High Performance

TOOLS IN MY WIZARD BELT

- Jira / Boards
- Confluence / Docs
- Miro / Whiteboards
- Slack / Teams
- Burndown / Burnup Charts
- Retrospective Tools
- Feedback & Surveys

A DAY IN THE LIFE

8:30	Check-in & remove overnight blockers
9:00	Daily Scrum: energize the team
9:30	Work on impediments & coach team
11:00	Stakeholder sync / align
12:00	Lunch (maybe!)
1:00	Facilitate / prepare upcoming events
2:00	Team 1:1s / observe & support
3:30	Retrospective follow-up actions
4:30	Improve something small today

Little by little... A lot becomes a LOT!

WHEN I KNOW I'M DOING GOOD

- Team is self-organizing
- Conversations are open & healthy
- Impediments are rare & small
- Predictability is improving
- Team delivers value consistently
- Stakeholders are happy
- Team learns & grows together

My success = My team's success!

COMMON MISCONCEPTIONS



SIGNS OF A HEALTHY TEAM

- Psychological safety is high
- People speak up & share ideas
- Team owns the work
- Feedback is frequent
- Mistakes are learning opportunities
- We celebrate small wins
- Everyone helps each other

SIGNS OF AGILE CHAOS

- Lots of blockers & delays
- Unclear priorities
- Blame & politics
- Meetings everywhere
- Team is tired & frustrated
- No time to improve
- Agile is just a joke now





Be a servant leader. Empower the team. Trust the process. Magic happens.

I DON'T HAVE ALL THE ANSWERS. BUT I HELP THE TEAM FIND THEM. TOGETHER, WE MAKE THE MAGIC HAPPEN!



Inspect. Adapt. Deliver. Repeat.

We don't just manage plans. We drive outcomes, remove roadblocks, and make it all come together.

THE PROGRAM MANAGER / PMO POSTER

ALIGNING STRATEGY. DRIVING DELIVERY. MAXIMIZING IMPACT.

We bring structure to complexity and calm to chaos.



WHO I AM

I'm the glue that connects strategy to execution. I enable teams, remove roadblocks, manage risks, and ensure we deliver maximum value sustainably.

- Strategic Partner
- Plan & Prioritize
- Align & Communicate
- Track & Report
- Remove Roadblocks
- Optimize & Improve

MY MINDSET

- Strategic Thinker
- Outcome Focused
- Data Driven
- Adaptive & Flexible
- Collaborative Leader
- Influence without Authority
- Problem Solver
- Continuous Improver
- Calm Under Pressure

WHAT I DO

- Define & manage programs
- Plan, coordinate & integrate
- Manage scope, schedule, budget
- Identify & manage risks/issues
- Track performance & report
- Facilitate decision making
- Manage dependencies
- Drive benefits realization
- Improve processes & governance
- Support teams to succeed

THE PROGRAM MANAGEMENT LIFECYCLE

1. INITIATE	2. PLAN	3. EXECUTE	4. MONITOR	5. CONTROL	6. CLOSE
Define purpose, objectives & success criteria.	Develop roadmap, plans, resources & budgets.	Coordinate work, manage resources, remove blockers.	Track progress, manage risks, ensure alignment.	Take corrective actions, adjust plans as needed.	Deliver results, capture learnings, measure value.

COMMUNICATION • COLLABORATION • TRANSPARENCY • ACCOUNTABILITY • CONTINUOUS IMPROVEMENT

PMO VALUE WE DELIVER

- Aligned strategic investments
- Predictable delivery
- Optimized resources
- Managed risks
- Increased efficiency
- Standardized processes
- Better visibility & reporting
- Higher quality outcomes
- Sustainable benefits realization

KEY STAKEHOLDERS I WORK WITH

- Executives / Leadership
- Portfolio Managers
- Project Managers
- Product Owners
- Teams / Squad Leads
- Finance / Operations
- Customers / Users
- Vendors / Partners



MY FOCUS AREAS

STRATEGY ALIGNMENT	GOVERNANCE	PERFORMANCE MANAGEMENT	RESOURCE MANAGEMENT	RISK & ISSUE MANAGEMENT	COMMUNICATION & CHANGE
Ensure programs support business strategy & goals.	Establish policies, standards & compliance.	Measure what matters. Improve what counts.	Right people. Right skills. Right time.	See it early. Solve it fast. Reduce impact.	Communicate clearly. Drive change effectively.

"A good plan today is better than a perfect plan tomorrow." ♥

What gets measured, gets managed. ♥

"Plans are nothing. Planning is everything." ♥

REPORTING & VISIBILITY

- Program Dashboards
- Status Reports
- Executive Summaries
- Risk & Issue Logs
- Roadmaps
- Portfolios / Benefits Reports
- KPIs & Metrics



If you can't see it, you can't manage it.

TOOLS I USE

- Jira / Azure DevOps
- Microsoft Project
- Smartsheet
- Monday.com
- Confluence
- Power BI / Tableau
- Excel (because... yes)



A DAY IN THE LIFE

8:30	10:00	11:30	1:00	2:30	4:00	5:00
Morning check-in & priority review	Stakeholder updates	Risk / Issue review	Plan / Roadmap work	Team syncs & unblocks	Reporting & follow-ups	Plan tomorrow. Learn today.



Every day is different. Impact is the same. ♥

KPIs I CARE ABOUT

- On-Time Delivery %
- Scope Change Rate
- Budget Variance %
- Resource Utilization
- Risk Exposure / Burn Down
- Benefits Realization
- Stakeholder Satisfaction



CHALLENGES I FACE

- Changing priorities
- Unclear requirements
- Resource constraints
- Dependencies outside control
- Scope creep
- Communication gaps
- Resistance to change



WHAT MAKES ME SUCCESSFUL

- Clear communication
- Strong relationships
- Strategic thinking
- Problem solving
- Adaptability
- Attention to detail
- Resilience & persistence
- Passion for delivery



SIGNS OF A HIGH-PERFORMING PROGRAM

- Clear goals & aligned stakeholders
- Healthy & empowered teams
- Effective communication
- Proactive risk management
- Data-driven decisions
- Continuous improvement
- Delivering real business value



REMEMBER
People don't care how much you know until they know how much you care. ♥

WE DON'T JUST MANAGE WORK.
We lead. We enable. We deliver value.

GREAT PROGRAMS DON'T HAPPEN BY CHANCE.
THEY HAPPEN BY DESIGN. ♥

KEEP IT SIMPLE. KEEP IT REAL. KEEP DELIVERING.

PMO: MAKING COMPLEX FEEL POSSIBLE.

Agile Truth #42: We meet so we can deliver.

SCRUM CEREMONIES

BECAUSE CHAOS NEEDS A CALENDAR



1 SPRINT PLANNING

Where optimism meets wild guessing.



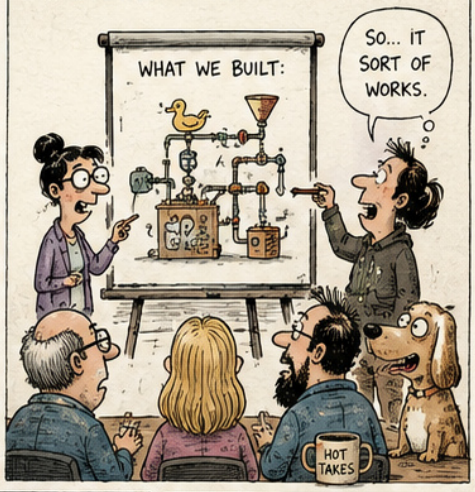
2 DAILY SCRUM

A 15-minute stand-up for updates, blockers, and existential dread.



3 SPRINT REVIEW

Show and tell for stakeholders and curious animals.



4 SPRINT RETROSPECTIVE

Where we make things better... eventually.



5 BACKLOG REFINEMENT

Turning chaos into... slightly less chaotic chaos.



BONUS CEREMONY: THE IMPROMPTU CORNER

Because agile is real life.



REMEMBER:



THESE CEREMONIES GIVE US RHYTHM

+



THE TEAM GIVES US HEART

+



THE PRODUCT GIVES US PURPOSE

=



DELIVER VALUE. REPEAT.

Agile isn't a destination. It's a series of awkward but well-intentioned steps.

I don't just document requirements. I uncover problems, connect dots, and help deliver value!

THE BUSINESS ANALYST POSTER

Bridging Business Needs and Solutions. Adding Clarity. Delivering Value.

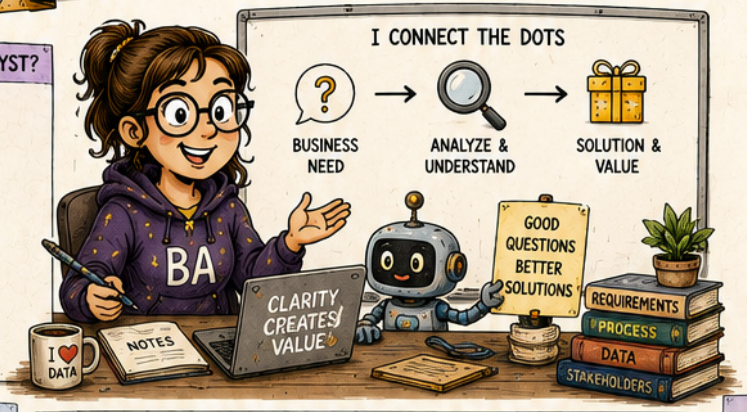
Translate confusion into clarity!

WHO IS THE BUSINESS ANALYST?

I'm the bridge between business and technology. I listen, analyze, challenge, and translate needs into clear requirements and solutions that create real value.

- Understand the business
- Elicit and analyze requirements
- Document and validate solutions
- Collaborate across teams
- Drive better decisions
- Add value every day!

I CONNECT THE DOTS



MY MINDSET

- Curious
- Empathetic
- Analytical
- Detail Oriented
- Critical Thinker
- Problem Solver
- Collaborative
- Adaptable
- Outcome Focused
- Value Driven

WHAT I DO

- Understand business goals & strategy
- Elicit requirements
- Analyze & model processes
- Document requirements
- Validate with stakeholders
- Identify gaps & risks
- Recommend solutions
- Support testing & UAT
- Measure value & success

MY WORK: FROM DISCOVER TO DELIVERY

DISCOVER	ELICIT	ANALYZE	DOCUMENT	VALIDATE	DELIVER VALUE
Explore the current state, problems, and opportunities	Ask the right questions. Listen actively. Dig deeper.	Break things down. Find patterns. Identify root causes.	Create clear, concise, and traceable requirements.	Confirm we're solving the right problem the right way.	Support the team. Measure impact. Improve continuously.

Good analysis today. Great solutions tomorrow.

KEY DELIVERABLES

- Business Requirements
- Functional Requirements
- User Stories
- Process Maps (As-Is / To-Be)
- Use Cases / Scenarios
- Data Models & Definitions
- Wireframes / Mockups
- Reports / Dashboards
- RTM (Traceability Matrix)
- BRD / SRS / Vision Docs

Clear docs. Happy teams. Successful projects!

MY SUPERPOWERS

- Asking awesome questions
- Seeing the bigger picture
- Turning chaos into clarity
- Connecting people & ideas
- Simplifying complex things
- Finding gaps others miss
- Making data meaningful
- Driving decisions
- Keeping the team aligned



REQUIREMENT TYPES I WORK WITH

BUSINESS REQUIREMENTS	STAKEHOLDER REQUIREMENTS	SOLUTION REQUIREMENTS	NON-FUNCTIONAL REQUIREMENTS	CONSTRAINTS & CONDITIONS
Why are we doing this?	What do stakeholders need?	What should the solution do?	How well should the solution perform?	What are the limits or assumptions?

MODELING & ANALYSIS TOOLS I USE

Process Diagrams (BPMN) · Use Case Diagrams · Data Models (ERD) · User Stories & Epics · Wireframes & Prototypes · Decision Tables

I COLLABORATE WITH

- **Product Owner** — Align on vision, value & priorities
- **Developers / Engineers** — Clarify requirements & solution
- **QA / Testers** — Ensure testability & quality
- **UX / Designers** — Understand user needs & flows
- **DevOps / Ops** — Ensure operational readiness
- **Stakeholders & Users** — Validate needs & get feedback

Alone we go fast. Together we go far!

A DAY IN THE LIFE OF A BA

8:30	Check messages & priorities
9:00	Stakeholder sync
10:00	Elicit requirements / Interviews
11:00	Analyze & model
12:00	Lunch & learn (stay curious!)
1:00	Document & refine requirements
2:30	Work with dev / clarify questions
3:30	Review designs / prototypes
4:30	Update docs / RTM
5:00	Plan for tomorrow

MORE COFFEE PLS

STAKEHOLDER COMMUNICATION

I LISTEN	I ASK	I EXPLAIN	I ALIGN
Understand needs and concerns	Ask smart questions	Communicate clearly	Get agreement & buy-in

Clear communication prevents expensive assumptions.

KPIs I CARE ABOUT

- Requirement Accuracy
- Stakeholder Satisfaction
- Requirement Stability
- Defect Leakage (UAT / Prod)
- On-time Delivery of Analysis
- Rework Reduction
- Value Delivered

TOOLS I USE

- Jira / Azure DevOps
- Confluence / Notion
- Visio / Draw.io / Lucidchart
- Excel / Google Sheets
- Power BI / Tableau
- Figma / Balsamiq
- SQL / Data Tools

CHALLENGES I FACE

- Unclear or changing requirements
- Too many stakeholders (all with opinions!)
- Scope creep
- Conflicting priorities
- Lack of documentation
- Tight deadlines
- Assumptions & lack of clarity



WHAT MAKES ME SUCCESSFUL

- Curiosity
- Strong communication
- Business acumen
- Analytical thinking
- Empathy
- Attention to detail
- Adaptability
- Critical thinking
- Time management
- Facilitation skills



SIGNS OF A GREAT BA

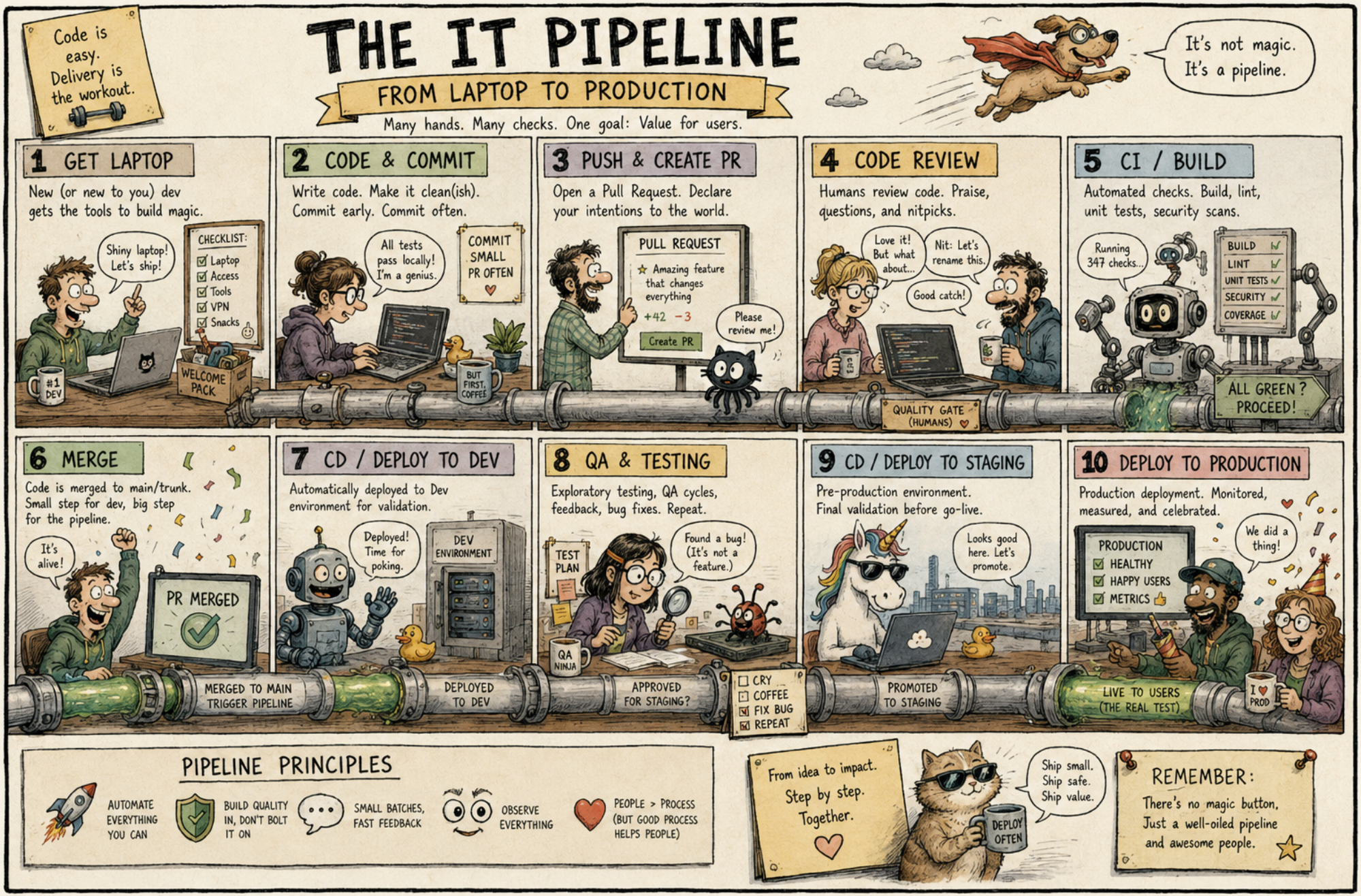
- People trust my insights
- Stakeholders seek my advice
- My docs are clear & useful
- The team delivers the right thing
- Users love the solution
- I help create business impact



I DON'T HAVE ALL THE ANSWERS. BUT I ASK THE RIGHT QUESTIONS.

I BRIDGE THE GAP BETWEEN BUSINESS NEEDS AND TECHNOLOGY SOLUTIONS TO DELIVER REAL VALUE.

UNDERSTAND. ANALYZE. ALIGN. DELIVER VALUE. REPEAT.


Code is easy. Delivery is the workout.
THE IT PIPELINE
FROM LAPTOP TO PRODUCTION
Many hands. Many checks. One goal: Value for users.
It's not magic. It's a pipeline.
1 GET LAPTOP
New (or new to you) dev gets the tools to build magic.
Shiny laptop! Let's ship!
CHECKLIST:
Laptop
Access
Tools
VPN
Snacks
#1 DEV
WELCOME PACK
2 CODE & COMMIT
Write code. Make it clean(ish). Commit early. Commit often.
All tests pass locally! I'm a genius.
COMMIT SMALL PR OFTEN
BUT FIRST, COFFEE
3 PUSH & CREATE PR
Open a Pull Request. Declare your intentions to the world.
PULL REQUEST
Amazing feature that changes everything
+42 −3
Create PR
Please review me!
4 CODE REVIEW
Humans review code. Praise, questions, and nitpicks.
Love it! But what about...
Nit: Let's rename this.
Good catch!
QUALITY GATE (HUMANS)
5 CI / BUILD
Automated checks. Build, lint, unit tests, security scans.
Running 347 checks...
BUILD
LINT
UNIT TESTS
SECURITY
COVERAGE
ALL GREEN? PROCEED!
6 MERGE
Code is merged to main/trunk. Small step for dev, big step for the pipeline.
It's alive!
PR MERGED
MERGED TO MAIN TRIGGER PIPELINE
7 CD / DEPLOY TO DEV
Automatically deployed to Dev environment for validation.
Deployed! Time for poking.
DEV ENVIRONMENT
DEPLOYED TO DEV
8 QA & TESTING
Exploratory testing, QA cycles, feedback, bug fixes. Repeat.
TEST PLAN
Found a bug! (It's not a feature.)
QA NINJA
APPROVED FOR STAGING?
CRY
COFFEE
FIX BUG
REPEAT
9 CD / DEPLOY TO STAGING
Pre-production environment. Final validation before go-live.
Looks good here. Let's promote.
PROMOTED TO STAGING
10 DEPLOY TO PRODUCTION
Production deployment. Monitored, measured, and celebrated.
PRODUCTION
HEALTHY
HAPPY USERS
METRICS
We did a thing!
LIVE TO USERS (THE REAL TEST)
I ♥ PROD
PIPELINE PRINCIPLES
AUTOMATE EVERYTHING YOU CAN
BUILD QUALITY IN, DON'T BOLT IT ON
SMALL BATCHES, FAST FEEDBACK
OBSERVE EVERYTHING
PEOPLE > PROCESS (BUT GOOD PROCESS HELPS PEOPLE)
From idea to impact. Step by step. Together.
Ship small. Ship safe. Ship value.
DEPLOY OFTEN
REMEMBER:
There's no magic button, Just a well-oiled pipeline and awesome people.

THE DEVELOPERS / ENGINEERS POSTER

We turn ideas into working software people actually use and love.

We don't just write code. We build solutions. We solve problems. We ship value.

Code is art. Teamwork is magic. Coffee is fuel.



WHO WE ARE

We are problem solvers, critical thinkers, builders, learners, and collaborators. We own the craft of engineering and take pride in delivering high-quality software.





OUR MINDSET

- [x] Take ownership
- [x] Keep learning
- [x] Embrace change
- [x] Focus on impact
- [x] Be kind & respectful
- [x] Communicate openly
- [x] Raise the bar
- [x] Leave it better

WHAT WE DO

- Understand problems & requirements
- Break down work into smaller pieces
- Design & write clean, maintainable code
- Build, test, and integrate
- Review code & give feedback
- Fix bugs & improve performance
- Collaborate & share knowledge
- Continuously improve

HOW WE WORK (AGILE + TEAMWORK)

PLAN	BUILD	TEST	REVIEW	DEPLOY	LEARN
We break down stories, estimate, and plan together.	We code, pair, ask questions, and help each other.	We test early and often. Quality is everyone's job.	We review code, share feedback, and learn together.	We integrate continuously and deploy frequently.	We reflect, experiment, and get better every day.

INSPECT & ADAPT
We inspect our work and our process, then adapt and improve.

TYPES OF WORK WE DO

- New Features
- Bug Fixes
- Refactoring
- Technical Debt
- Spikes / Research
- Performance Tuning
- Automation
- Support & Maintenance

OUR SUPERPOWERS

- Problem Solving
- Logic & Critical Thinking
- Attention to Detail
- Creativity
- Resilience
- Collaboration
- Continuous Learning
- Turning Coffee into Code



TECHNICAL EXCELLENCE

CODE QUALITY	TESTING	ARCHITECTURE	DEVOPS & CI/CD	SECURITY
Clean	Unit Tests	Scalable	Version Control	Secure Code
Readable	Integration Tests	Reliable	CI Builds	Validate Input
Maintainable	Automated Tests	Secure	Automated Tests	Least Privilege
Testable	Manual Testing	Maintainable	Continuous Delivery	Data Protection
Efficient	Test Early	SOLID Design	Monitor & Iterate	Stay Updated

TOOLS WE USE

(Our Utility Belt)

- IDE: VS Code, IntelliJ, etc.
- Languages: Java, JS, Python, C#, Go, etc.
- Frameworks & Libraries
- Git & Version Control
- CI/CD: Jenkins, GitHub Actions, GitLab CI, Azure DevOps
- Testing: JUnit, NUnit, PyTest, Selenium, Cypress, etc.
- APIs, Postman, Swagger
- Containers: Docker, Kubernetes
- Cloud: AWS, Azure, GCP
- Monitoring: Datadog, New Relic

KPIs WE CARE ABOUT

- Sprint Commitments Met
- Code Quality / Defect Rate
- Lead Time for Changes
- Deployment Frequency
- Test Coverage
- Escaped Defects
- Cycle Time
- Velocity (Team)

A DAY IN THE LIFE

9:00	Daily Standup
9:30	Deep Work (Coding)
11:00	Pairing / Collaboration
12:00	Lunch & Recharge
1:00	Code, Test, Build
3:00	Code Review
4:00	Fix, Improve, Refactor
5:00	Learn / Spike / Plan

Every day is different.
Every day we create impact.

GOOD CODE IS...

- Correct
- Clear
- Consistent
- Complete
- Cohesive
- Tested
- Well Documented
- Built for Change



COLLABORATION IS OUR FORCE MULTIPLIER

Pair Programming | Mob Sessions | Code Reviews | Knowledge Sharing | Help & Support



We win together. We grow together.

CHALLENGES WE FACE

- Unclear Requirements
- Changing Priorities
- Technical Debt
- Legacy Systems
- Too Many Interruptions
- Tight Deadlines
- Context Switching
- Underestimation



ANTI-PATTERNS WE AVOID

- Overengineering
- Copy-Paste Code
- Skipping Tests
- Poor Communication
- Blaming Others
- Working in Silos
- Ignoring Tech Debt
- Hero Culture



WHAT MAKES US HAPPY

- Solving hard problems
- Seeing users benefit
- Clean, elegant code
- Ship something valuable
- Learning new things
- Amazing teammates
- Positive impact



REMEMBER

- We don't just write code.
- We build solutions.
- We are engineers.
- We are creators.
- We are a team.
- Together, we can build anything!



CODE IS HOW WE EXPRESS OUR LOGIC.
BUT PURPOSE IS WHY WE SHOW UP EVERY DAY.
WE BUILD SOFTWARE. WE BUILD TRUST.
WE BUILD THE FUTURE.



WORDS WE LIVE BY

```
// Stay curious
// Keep it simple
// Make it work
// Make it better
// Leave it better than you found it
```

FINAL SPELL

May our code be clean.
May our builds be green.
May our users be happy.
May our team be strong.
And may we always keep learning and delivering value!

We don't just find bugs.
We prevent them.
We improve quality.
We protect users.

THE QA / TESTER POSTER

QUALITY CHAMPIONS. BUG HUNTERS. USER ADVOCATES.

A bug today, a disaster tomorrow.

WHO WE ARE

We are QA / Testers.
We care about quality and user experience. We work with the team to build reliable, secure, and high-quality software.

- Quality Advocates
- Detail Detectives
- Risk Identifiers
- Problem Preventers
- Process Improvers

QUALITY IS EVERYONE'S RESPONSIBILITY. WE HELP EVERYONE DELIVER IT.



OUR MINDSET

- Curious
- Critical Thinkers
- Quality Driven
- User Focused
- Collaborative
- Continuous Learners
- Detail Oriented
- Risk Aware
- Always Improving

WHAT WE DO

- Understand requirements
- Analyze & identify risks
- Design test strategy & plans
- Create test cases
- Execute tests
- Log defects with clarity
- Verify fixes
- Do regression testing
- Ensure release readiness
- Improve processes & quality

OUR TESTING APPROACH

UNDERSTAND	PLAN	DESIGN	EXECUTE	REPORT	VERIFY	IMPROVE
Understand the feature, users, and risks.	Plan test strategy, scope and scenarios.	Design test cases & test data.	Execute tests (manual & automated).	Report defects with clear steps to reproduce.	Verify fixes and prevent regressions.	Learn & improve quality and processes.

Quality is not an act, it's a habit.

TYPES OF TESTING WE DO

- Functional Testing
- Regression Testing
- Integration Testing
- Smoke Testing
- Sanity Testing
- Exploratory Testing
- Performance Testing
- Security Testing
- Usability Testing
- Compatibility Testing
- UAT (User Acceptance Testing)

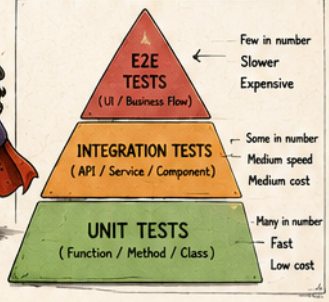
OUR SUPERPOWERS

- Sharp Eyes
- Curious Mind
- Break Things (Safely)
- Find What Others Miss
- Think Like a User
- Risk Radar
- Patience + Persistence
- Love for Details



Bugs don't hide from us. We find them.

TESTING PYRAMID



DEFECT LIFE CYCLE



Clear defects. Clear communication. Strong quality.

THE BUG REPORT ANATOMY

- Clear Title
- Environment
- Steps to Reproduce
- Expected Result
- Actual Result
- Attachments / Screenshots
- Severity & Priority

A good bug report is a bug half fixed!

TOOLS WE USE

- Test Management: Jira, TestRail, Zephyr
- Automation: Selenium, Playwright, Cypress
- API Testing: Postman, Rest Assured
- Performance: JMeter, LoadRunner, Gatling
- Mobile Testing: Appium, BrowserStack
- Defect Tracking: Jira, Azure DevOps
- CI/CD Integration: Jenkins, GitHub Actions

GOOD TEST CASES ARE...

- Clear & easy to understand
- Focused on user behavior
- Reusable & maintainable
- Have clear expected results
- Independent & atomic
- Cover happy paths & edge cases
- Traceable to requirements

Good tests lead to confident releases!

A DAY IN THE LIFE

9:00	Daily Stand-up
9:30	Review / Analyze Requirements
10:30	Test Design / Test Planning
11:30	Test Execution
1:00	Lunch & Learn (or break!)
2:00	Defect Triage / Reporting
3:00	Re-test / Regression Testing
4:30	Collaborate & Improve
5:00	Wrap up & Plan Tomorrow

TEST REPEAT COFFEE

KPIs WE CARE ABOUT

- Defect Detection Rate
- Defect Leakage (Escaped Defects)
- Test Coverage
- Requirement Coverage
- Test Automation %
- Defect Closure Rate
- Release Readiness
- Customer Satisfaction

CHALLENGES WE FACE

- Ambiguous Requirements
- Last Minute Changes
- Tight Deadlines
- Environment Issues
- Too Many Interruptions
- Underestimated Testing
- Unclear Priorities

So many bugs... So little time!

BUGS DON'T SLEEP

HOW WE ADD VALUE

- Prevent bugs, not just find them
- Improve product quality & reliability
- Enhance user experience
- Reduce rework & production issues
- Build confidence in every release
- Protect the brand & the users

QA

QUALITY IS OUR PROMISE TO USERS.

SIGNS OF A HEALTHY QUALITY CULTURE

- Quality is a shared goal
- Testing starts early, not late
- Open communication
- Safe to raise issues
- Continuous learning
- Automation where it adds value
- Celebrate quality wins!

Teamwork makes quality unbeatable!

WE TEST THE PRESENT...

WE PREVENT THE FUTURE...

WE DELIVER CONFIDENCE.

Happy Users. Happy Business. That's Quality.

WE ARE THE GUARDIANS OF QUALITY!

QA

THE SOLUTION ARCHITECT POSTER

I don't just draw diagrams. I design solutions that solve real problems.

DESIGN TODAY. SCALE TOMORROW. DELIVER VALUE.

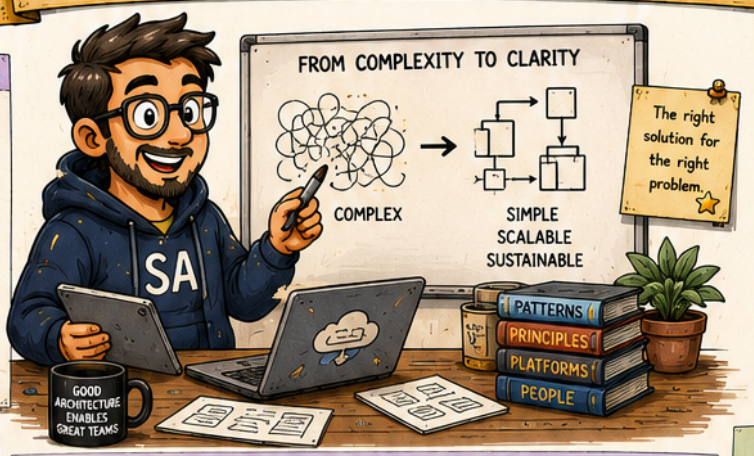
Architecture is about the important stuff.

WHO AM I?

I'm the Solution Architect.

I design the blueprint for successful software solutions. I align business needs, technology, and teams to build solutions that are secure, scalable, reliable, and cost-effective.

- Business Aligned
- Technology Agnostic
- Big Picture Thinker
- Detail Aware
- Risk Anticipator
- Value Driven



The right solution for the right problem.

MY MINDSET

- Think Big, Start Small
- Outcome Focused
- Curious & Analytical
- Pragmatic not Dogmatic
- Collaborative Leader
- Adaptive & Forward Thinking
- Always Learning

WHAT I DO

- Understand business goals & constraints
- Elicit & analyze requirements
- Design solution architecture
- Define NFRs (Quality Attributes)
- Evaluate technology options
- Manage technical risks
- Create architecture artifacts
- Guide implementation
- Support delivery & operations
- Continuously improve solutions

HOW I WORK (ARCHITECTURE LIFECYCLE)

1. DISCOVER	2. ANALYZE	3. DESIGN	4. VALIDATE	5. PLAN	6. DELIVER
Understand the problem, context, constraints and opportunities.	Model requirements, scenarios, data and user journeys.	Create solution architecture that meets functional & quality needs.	Assess options, prototype, and validate with stakeholders.	Define roadmap, sequencing, and implementation strategy.	Guide teams, resolve issues, ensure success, and iterate.

PRINCIPLES I FOLLOW

Simplicity · Modularity · Scalability · Security · Reliability · Performance · Cost Effective

WHAT I DELIVER

- Solution Architecture
- Architecture Diagrams
- Technology Recommendations
- Integration Patterns
- Data Architecture
- Security Architecture
- NFRs & Trade-off Analysis
- Roadmaps & Guidelines
- Decision Records (ADRs)
- POCs & Prototypes

Clarity today. Confidence tomorrow.

THE HATS I WEAR



Different hat. Same goal: Build the right solution.

THE 5+1 ARCHITECTURAL VIEWS

LOGICAL VIEW	DEVELOPMENT VIEW	PROCESS VIEW
What the system does.	How it's built.	How it runs.
DEPLOYMENT VIEW	**DATA VIEW**	**+ SCENARIO VIEW**
Where it runs.	What data it uses.	Why & how it's used.

Different views. One solution.

I PARTNER WITH

- Product Owner — Align on vision & value
- Developers / Engineers — Make it real
- QA / Testers — Ensure quality
- DevOps / SRE — Run it reliably
- Security Team — Build it secure
- Business Stakeholders — Deliver what matters

Great solutions are a team sport.

TOOLS I USE

- Modeling: diagrams.net, Lucidchart, Structurizr
- Documentation: Confluence, SharePoint, Notion
- Cloud: AWS, Azure, GCP
- IaC: Terraform, CloudFormation, Bicep
- API: Postman, Swagger / OpenAPI
- Collaboration: Miro, Mural, Jira

ARCHITECTURAL ARTIFACTS

- Context Diagram
- C4 Model
- Component Diagram
- Sequence Diagram
- Data Flow Diagram
- ER Diagram
- API Contracts
- Decision Records (ADR)
- Architecture Principles
- Architecture Roadmap

QUALITIES I DESIGN FOR (NFRs)

- Performance
- Scalability
- Availability
- Reliability
- Security
- Maintainability
- Usability
- Observability
- Portability
- Cost Efficiency

A DAY IN THE LIFE

9:00	10:00	11:30	1:00	2:30	4:00	5:00
Sync & align with team	Analyze & model	Design & evaluate options	Stakeholder workshop	Review & refine	Guide teams & unblock	Learn & improve

No two days are the same. That's what I love about it!

CHALLENGES I FACE

- Changing requirements
- Balancing trade-offs
- Legacy system constraints
- Too many options
- Keeping it simple
- Stakeholder alignment
- Technology evolution
- Scope creep
- Time & budget pressure
- Communication gaps

Architects don't predict the future. We prepare for it.

WHAT SUCCESS LOOKS LIKE

- Business goals achieved
- Teams empowered
- Systems reliable & scalable
- Users happy
- Value delivered continuously
- Stakeholders confident

Good architecture is invisible. Great architecture is unforgettable.

REMEMBER
Architecture is not about technology. It's about people, problems and possibilities.

I DON'T BUILD SYSTEMS. I DESIGN SOLUTIONS THAT EMPOWER PEOPLE TO DO AMAZING THINGS.

The best architecture enables change, not fear.

We build the highways so great software can reach users safely and fast!

THE DEVOPS / RELEASE ENGINEER POSTER

AUTOMATE EVERYTHING. DEPLOY ANYWHERE. DELIVER VALUE.

Code is committed. Pipelines are happy. Users are happy. I sleep (sometimes).

WHO I AM

I bridge development and operations. I automate, integrate, deploy, monitor, and improve everything so software delivery is fast, reliable, and safe.

- Automation Enthusiast
- Problem Solver
- System Thinker
- Reliability Champion
- Security Advocate
- Collaboration Driver
- Always Learning



MY MINDSET

- Automation First
- You build it, you run it
- Measure everything
- Fail fast, learn faster
- Security is everyone's responsibility
- Small changes, big impact
- Continuous improvement

WHAT I DO

- CI/CD Pipeline Management
- Infrastructure as Code (IaC)
- Environment Management
- Automated Testing & Quality Gates
- Release Management
- Monitoring & Alerting
- Security & Compliance
- Incident Response
- Performance Optimization
- Documentation & Runbooks

THE SOFTWARE DELIVERY LIFECYCLE (MY WORLD)

PLAN	CODE	BUILD	TEST	DEPLOY	OPERATE	FEEDBACK
Collaborate on requirements, architecture & solutions	Source control, branching, code reviews	Compile, package, create artifacts	Automated tests, quality gates, security scans	Release to environments automatically and consistently	Run, monitor, scale, and ensure reliability	Observe, learn, improve, and iterate

CONTINUOUS INTEGRATION · CONTINUOUS DELIVERY · CONTINUOUS IMPROVEMENT ∞

I WORK WITH

- Developers / Engineers
- QA / Testers
- Product Owner
- Scrum Master
- Business Analysts
- Security Team
- Operations Team
- Support / SRE
- Stakeholders

Together we deliver value!

PIPELINE AS CODE (SAMPLE)

```
stages:
  - build
  - test
  - scan
  - deploy

build:
  script: mvn clean package

test:
  script: mvn test

deploy:
  script: kubectl apply -f k8s/
```

Automated. Versioned. Repeatable.

MY TOOLBELT (SOME OF MY FAVORITE TOOLS)

CI / CD	CONTAINERS & ORCHESTRATION	INFRASTRUCTURE AS CODE	CLOUD PLATFORMS	MONITORING & LOGGING	COLLABORATION
Jenkins	docker	Terraform	aws	Prometheus	Slack
GitHub Actions	kubernetes	AWS CloudFormation	Azure	Grafana	Jira
GitLab CI	HELM	ANSIBLE	Google Cloud	ELK Stack	Confluence

Ship small. Ship often. Ship safely.



ENVIRONMENTS I CARE FOR

DEV	TEST	STAGING	PRODUCTION
Build & Experiment	Validate & Verify	Pre-Production Validation	Live Users Real Impact

Same process. Same pipeline. Different configurations.

RELEASE STRATEGIES

- Blue/Green Deployments
- Canary Releases
- Rolling Updates
- Feature Flags
- Revert Ready

MONITOR. ALERT. REACT.

Metrics · Logs · Traces · Alerts

Observe everything. Alert on what matters. React fast. Learn and improve.

INFRASTRUCTURE AS CODE

- Reusable
- Versioned
- Consistent
- Reviewable
- Automated
- Auditable

No clicks. Just code.

SECURITY IS BUILT IN

- Scan early, scan often
- Secrets management
- Least privilege access
- Patch & update
- Compliance as code
- Everyone's responsibility

Secure by default, not by luck.

KPIS I CARE ABOUT

- Deployment Frequency
- Lead Time for Changes
- Change Failure Rate
- Mean Time to Restore
- System Availability
- Customer Satisfaction

Measure. Improve. Repeat.

A DAY IN THE LIFE

8:30	9:00	10:30	12:00	2:00	3:30	5:00
Check alerts & system health	Pipeline runs & builds	Collaborate & solve problems	Automate & improve	Deploy with confidence	Monitor & optimize	Document & wrap up

Every day is different. Impact is the same.

CHALLENGES I FACE

- Flaky environments
- Broken builds
- Dependency hell
- Last-minute changes
- People & process gaps
- Too many manual steps

But every challenge is an opportunity to make it better!

WHAT MAKES IT AWESOME

- Automate boring stuff
- Solve complex problems
- See real-world impact
- Work with amazing people
- Ship value to users
- Always learning

We build the foundation for awesome software!

I DON'T DEPLOY CODE. I DELIVER CONFIDENCE.

GREAT SOFTWARE DOESN'T SHIP ITSELF. I HELP IT GET THERE SAFELY.

AUTOMATE TODAY. ELEVATE TOMORROW.

Remember: Small changes Big impact Every day!

Metrics are not about punishing teams. They are about improving systems.

PROJECT SUCCESS ISN'T LUCK. IT'S MEASURED.

Measure to understand, not to punish!

KPIs THAT HELP TEAMS DELIVER VALUE

The right metrics. The right conversations. Better decisions. Better outcomes.

1. DELIVERY KPIs

Are we delivering work predictably and on time?

- **Sprint Velocity** — Story points completed per sprint.
- **Sprint Predictability** — Planned vs completed work. (Planned: 80 pts, Delivered: 72 pts, Predictability: 90%)
- **Burndown Chart** — Remaining work during the sprint.
- **Burnup Chart** — Completed scope vs total scope.
- **Release Progress** — % complete of release or milestone. (75%)

Visibility builds confidence!

2. FLOW & PROCESS KPIs

How efficiently is work moving?

- **Cycle Time** — Time from "work started" to "work completed".
- **Lead Time** — Time from "request created" to "delivered".
- **Throughput** — Number of work items completed in a timeframe.
- **Work in Progress (WIP)** — How much work is active at a given time.
- **Aging Work** — How long items stay open.
- **Blocked Items** — Count and duration of blocked stories/tasks.

Reduce WIP. Increase flow. Delight customers!

3. QUALITY KPIs

Are we building stable, reliable solutions?

- **Defect Density** — Bugs per feature / release / story point.
- **Escaped Defects** — Bugs found in production.
- **Reopen Rate** — % of tickets reopened after "done".
- **Test Coverage** — % of code / process covered by tests.
- **Production Incidents** — Number and severity (Sev 1, 2, 3).
- **Mean Time to Recovery (MTTR)** — How long it takes to restore service after failure.

Quality isn't expensive. Failures are.

4. TEAM HEALTH KPIs

Is the team healthy and sustainable?

- **Team Happiness / Satisfaction** — Survey: morale, burnout, confidence.
- **Retro Action Completion** — Are improvement actions actually implemented?
- **Capacity Utilization** — How overloaded is the team?
- **Unplanned Work %** — Measures interruptions / firefighting.
- **Ceremony Effectiveness** — Are meetings useful, on time, and outcome-driven?

Healthy teams build great things!

5. BUSINESS VALUE KPIs

Is the project creating value?

- **ROI / Business Value Delivered** — Revenue, savings, efficiency gains.
- **Feature Adoption** — Are users actually using the features?
- **Customer Satisfaction (CSAT)** — Survey-based customer satisfaction.
- **Net Promoter Score (NPS)** — How likely are users to recommend us?
- **Time to Market** — How quickly value reaches users.
- **Benefits Realization** — Did promised business outcomes happen?

Deliver value. Make impact. Repeat.

6. FINANCIAL KPIs

Are we good stewards of resources?

- **Budget vs Actual** — Spend tracking.
- **Forecast Accuracy** — How accurate financial forecasting is.
- **Resource Utilization** — How effectively staff are allocated.
- **Cost Variance** — Difference between planned vs actual cost.
- **Earned Value Metrics (EVM)** — PMO metrics like:
  - CPI (Cost Performance Index)
  - SPI (Schedule Performance Index)

Watch the numbers. Tell the story.

7. DEPENDENCY & RISK KPIs

What could get in the way?

- **Open Risks** — Count and severity (High, Medium, Low).
- **Dependency Tracking** — Cross-team and external dependencies.
- **RAID Metrics** — Track Risks, Assumptions, Issues, Dependencies.
- **Escalation Volume** — How many issues require leadership intervention.

Shine a light on risks early.

8. ROLE-SPECIFIC FOCUS

SCRUM MASTER

- Sprint predictability
- Velocity trends
- Blocked items
- Carryover work
- Retro action completion
- Team dynamics & morale
- Process improvement

I remove impediments, not roadblocks!

PROGRAM MANAGER

- Cross-team dependencies
- Milestone alignment
- Roadmap delivery
- Budget & resource health
- Release readiness
- Business outcomes

PROJECT MANAGER

- Scope, schedule, budget
- Risk & issue management
- Stakeholder communication
- Change management
- Delivery confidence

9. EXECUTIVE DASHBOARD KPIs

What leaders care about most.

OVERALL STATUS	TOP RISKS	DELIVERY CONFIDENCE	BUSINESS IMPACT	KEY DECISIONS NEEDED
Scope, Schedule, Budget, Risks	What keeps us up at night?	Will we hit our targets?	What value was delivered?	What do you need to decide?

Data with purrpose.

Data with purrpose.

THE GOLDEN RULE

KPIs are like a compass, not a hammer.

Use metrics to:

- ☑ Gain visibility
- ☑ Forecast outcomes
- ☑ Identify bottlenecks
- ☑ Improve systems

- ✗ Not to punish people
- ✗ Not to compare teams unfairly
- ✗ Not to drive the wrong behavior

A GREAT KPI IS...

- ☑ Actionable
- ☑ Understandable
- ☑ Timely
- ☑ Relevant
- ☑ Connected to outcomes

When in doubt, ask: "Will this metric help us deliver more value?"

BETTER METRICS + BETTER CONVERSATIONS = BETTER OUTCOMES

INSPECT → ADAPT → IMPROVE → DELIVER VALUE → REPEAT

Metrics tell you WHAT is happening. People tell you WHY.

KPIs are sensors on the system, not weapons against the team.

PROJECT KPIs — THAT MATTER —

MEASURE WHAT MATTERS. IMPROVE WHAT YOU MEASURE. DELIVER VALUE.

GOOD DATA
BETTER INSIGHTS
SMARTER DECISIONS
HAPPIER TEAMS
GREATER IMPACT!

ROLES THAT USE THESE KPIs

PROJECT MANAGER — Focus on delivery, scope, time, budget, risks.

PROGRAM MANAGER — Focus on alignment, dependencies, roadmap, outcomes.

SCRUM MASTER — Focus on team health, flow, impediments, continuous improvement.

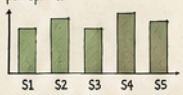
AGILE COACH / DELIVERY LEAD — Focus on agility, coaching, capability, organizational health.

ENGINEERING / PRODUCT LEADER — Focus on quality, value, adoption, customer impact.

1 DELIVERY KPIs

Are we delivering work predictably and on time?

SPRINT VELOCITY
Story points completed per sprint.



SPRINT PREDICTABILITY
Planned vs. Completed

Planned	80 pts
Completed	72 pts
Predictability	90%

BURNDOWN CHART
Work remaining in the sprint.



BURNUP CHART
Track progress when scope changes.



RELEASE PROGRESS
Overall progress toward release goal.

72%

Features: 18 / 25

MILESTONE ACHIEVEMENT
Are key milestones hit?

- ✓ UAT Complete
- ✓ Security Review
- ☐ Go-Live
- ☐ Post-Implementation

2 FLOW & PROCESS KPIs

How efficiently is work moving?

CYCLE TIME
Time from "work started" to "work completed".

5.2 days

LEAD TIME
Time from request to delivered.

12.7 days

THROUGHPUT
Work items completed in a time period.

45 items / week

WORK IN PROGRESS (WIP)
How much work is active at one time.

12 items

AGING WORK
How long items stay open.

> 30 days	8
> 15 days	12
> 7 days	10

BLOCKED ITEMS
Count and duration of blocked work.

7 blocked

3 QUALITY KPIs

Are we building stable, reliable solutions?

DEFECT DENSITY
Bugs per 100 story points.

3.6

↓ lower is better

ESCAPED DEFECTS
Bugs found in Production.

14 this release

REOPEN RATE
% of items reopened after done.

8%

TEST COVERAGE
% of code covered by automated tests.

78%

PRODUCTION INCIDENTS
Count by severity.

Sev 1	2
Sev 2	5
Sev 3	11

MEAN TIME TO RECOVERY
Average time to restore service after failure.

2.4 hours

4 AGILE TEAM HEALTH KPIs

Is the team healthy and sustainable?

TEAM HAPPINESS
(Example Survey Score)

7.8 /10

+ from last survey!

RETROSPECTIVE ACTIONS
Are improvement actions actually completed?

Completed 8 / 10

CAPACITY UTILIZATION
How loaded is the team?

78%

UNPLANNED WORK %
Interruptions & firefighting.

23%

CEREMONY EFFECTIVENESS
Are meetings useful and producing outcomes?

★★★★☆ (4.2 / 5)

"Shorter. Focused. Valuable."

CROSS-FUNCTIONAL COLLABORATION
How well is the team working together?

5 BUSINESS VALUE KPIs

Is the project creating value?

ROI / BUSINESS VALUE DELIVERED

$1.8M

Value Realized (2024 YTD)

FEATURE ADOPTION
% of users actively using the feature.

62%

CUSTOMER SATISFACTION (CSAT)

4.3 /5

NET PROMOTER SCORE (NPS)

+42

TIME TO MARKET
Time from idea to delivery.

45 days

BENEFITS REALIZATION
Did promised outcomes happen?

- ✓ Reduce time by 30%
- ✓ Save $1M annually
- ☐ Improve NPS to 50+

6 FINANCIAL KPIs

Are we managing budget and resources wisely?

BUDGET VS ACTUAL
Spend tracking.

$2.4M of $3.0M

FORECAST ACCURACY
How accurate are our forecasts?

92%

RESOURCE UTILIZATION
How effectively staff are allocated.

81%

COST VARIANCE
Planned vs. Actual.

-$120K (Underspent)

EARNED VALUE (TRADITIONAL PMO)

CPI (Cost Perf. Index)	1.05 ✓
SPI (Schedule Perf. Index)	0.95 ~

On track (ish)!

7 DEPENDENCY & RISK KPIs

What could derail us?

OPEN RISKS
Count by severity.

High	5
Medium	8
Low	6

DEPENDENCY TRACKING
Cross-team blockers.

Open	6
At Risk	3
Resolved	12

RAID SUMMARY
Risks, Assumptions, Issues, Dependencies

Risks	Assumptions	Issues	Dependencies
19	11	9	15

ESCALATION VOLUME
Issues escalated to leadership.

4 this week

8 SCRUM MASTER FOCUS KPIs

Helping the team, removing impediments, improving flow.

TEAM DELIVERY HEALTH
- Sprint Predictability
- Velocity Trends
- Carryover Work
- Blocked Items
- Retro Actions Completion

PROCESS IMPROVEMENT
- Cycle Time
- Reduced Bottlenecks
- Fewer Interruptions
- Better Estimation Accuracy

TEAM DYNAMICS
- Participation
- Collaboration
- Trust
- Morale

DELIVERY STABILITY
- Reduced Spillover
- Fewer Defects
- Improved Quality
- Sustainable Pace

9 EXECUTIVE DASHBOARD KPIs

High-level view. Clear. Actionable. Outcome focused.

OVERALL HEALTH

SCOPE	SCHEDULE	BUDGET	RISKS
✓	⚠	✓	⚠

TOP RISKS
1 Vendor Delay
2 Key Resource Availability
3 Security Compliance

DELIVERY CONFIDENCE
Can we hit our targets?

HIGH

VALUE DELIVERED (YTD)

$1.8M

Business Value Realized

KEY DECISIONS NEEDED
- Approve Budget for Phase 2
- Resource Decision
- Vendor Contract Approval

✓ HOW TO USE KPIs WELL

DO
- Use for visibility, not punishment
- Look at trends, not one-time snapshots
- Drive conversations and decisions
- Connect metrics to outcomes
- Improve the system, not blame people

DATA OVER DRAMA

✗ HOW NOT TO USE KPIs

DON'T
- Weaponize metrics
- Compare teams unfairly
- Encourage fake estimates
- Ignore context
- Focus only on output, not outcomes

Numbers without context create stress, not success.

REMEMBER ★

Great teams don't chase numbers.
They build value, learn continuously.
and use KPIs to guide the journey.

BETTER EVERY DAY.

THE GOLDEN RULE

If a KPI doesn't help improve the product, the process, or the customer experience... maybe you don't need it.

GOOD KPIs TELL YOU WHERE YOU ARE. GREAT TEAMS DECIDE WHERE TO GO.