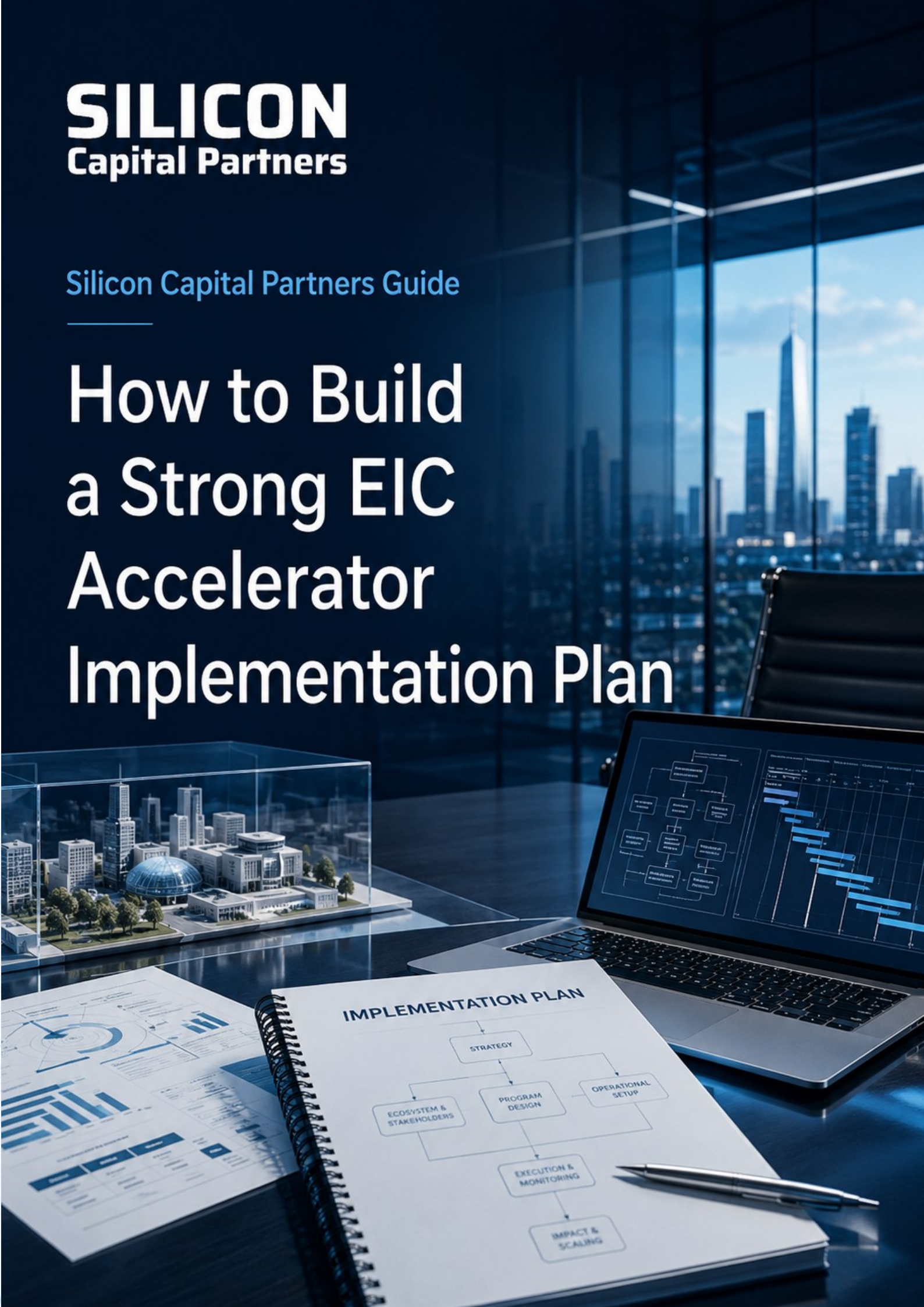
SILICON
Capital Partners

Silicon Capital Partners Guide

# How to Build a Strong EIC Accelerator Implementation Plan



## Intellectual Property Notice

# Introduction

At Silicon Capital Partners, we treat the Implementation Plan as one of the most revealing parts of the EIC Accelerator Full Proposal. It is where a company stops being a promising story and becomes an executable project. Many strong deep-tech companies underperform here — not because their technology is weak, but because their plan is too vague, too broad, too unstructured, or too disconnected from the real path to TRL 8 and market readiness.

This guide is written for CEOs who understand their company, their product, and their market, but who may be developing an EIC Accelerator Implementation Plan for the first time. It explains what the EIC is asking for, why evaluators care about this section so much, and how to build a plan that reads as credible, fundable, and execution-driven.

The guide also introduces the logic of lump sum budgeting, because even though the detailed budget is covered in later chapters of the full guide, the Implementation Plan and the lump sum logic are already deeply connected at this stage.

> *The Implementation Plan is not a narrative section. It is an operational proof that your company understands how to execute.*

## How this guide is structured

| Part | Content |
|---|---|
| Part I — Chapters 1–19 | The Implementation Plan: structure, components, writing logic, and evaluator patterns |
| Part II — Chapters 20–35 | The Lump Sum Budget: workbook logic, cost categories, and evaluation risks |
| Part III — Chapters 36–38 | Alignment: coherence matrix, consistency review, and evaluator perspective |
| Part IV — Chapters 39–42 | Practical tools: client input packages, checklists, and red team review |

# PART I — THE IMPLEMENTATION PLAN

Chapter 1

# How to Use This Guide

*Audience, scope, and prerequisites*

## Who this guide is for

This guide is for founders, CEOs, and leadership teams preparing a Full Proposal for the EIC Accelerator. It is especially useful for companies that already know their technology and market well, but need help turning that knowledge into a formal, evaluator-friendly execution plan. It is also useful for technical leaders and project managers who need to understand what the CEO must provide before drafting can begin.

## What this guide covers and what it does not

This guide explains the role, structure, and expectations of the Implementation Plan at Full Proposal stage. It also explains the strategic logic of lump sum budgeting insofar as it affects how you design your work packages, tasks, milestones, and resource narrative. It does not replace legal advice, regulatory strategy, or a full financial model.

## How to read this guide depending on where you are

| Your situation | Focus on | Why |
|---|---|---|
| Still shaping the project | Chapters 2–6 | Core architecture and EIC expectations |
| Draft already exists | Chapters 7–15 | Strengthen execution credibility |
| Close to submission | Chapters 18–19 | Red team the full plan |
| Working on the budget | Chapters 3–4 | Implementation and lump sum relationship |

## What you must be able to answer before drafting

- ✓ What exactly is the starting technical maturity?
- ✓ What exactly will be achieved by the end of the grant phase?
- ✓ What are the main uncertainties that must be resolved?
- ✓ What evidence must be generated for market, regulatory, and industrial progress?
- ✓ Who will do the work?

> *If you cannot answer those five questions precisely, the Implementation Plan will quickly become generic.*

Chapter 2

# What the Implementation Plan Really Is

*EIC expectations, evaluator logic, and framing risks*

## What the EIC expects

The EIC is not asking for a company narrative repeated in another format. It is asking for a structured demonstration that the company knows how to move from its current maturity level to a more advanced and commercially meaningful state.

The official annex requires: a work plan narrative, a timing view, lists of work packages, detailed work package descriptions, deliverables, milestones, risks, subcontracting, purchase costs, and other cost-related justifications where applicable.

## What role it plays inside the Full Proposal

In practice, the Implementation Plan carries a large share of the credibility burden of the proposal. The earlier chapters may explain why the opportunity is important. The Implementation Plan must show that the company can actually execute the path it claims.

> *Evaluators often reward ambition when the plan is precise, but punish even promising technologies when the execution logic is vague.*

## Description vs. proof of execution

| Project description | Implementation Plan |
|---|---|
| Tells the reader what the company wants to do | Tells the evaluator how the company will do it |
| Strategic language and vision | Operational clarity with sequence and evidence |
| 'Why it matters' | 'How it will happen' |

## What evaluators really read in this section

Evaluators look for coherence, sequence, risk awareness, and sufficiency of detail. They ask: do the work packages make sense? Are the tasks real? Do the milestones prove anything? Are risks taken seriously? Do the resources match the work?

## The most common framing mistakes

- ✓ Writing the section too late — treating it as a form-filling exercise
- ✓ Repeating marketing language instead of operational content
- ✓ Using work packages that are too broad or too vague

- ✓ Describing activity without showing sequence or dependencies
- ✓ Listing milestones that are really just dates
- ✓ Leaving commercial and regulatory steps outside the plan
- ✓ Assuming the evaluator will fill in missing logic

Chapter 3

# What the Lump Sum Really Is

*Logic, misconceptions, and evaluator perspective*

## What lump sum means in EIC Accelerator

In EIC Accelerator, the grant component follows the Horizon Europe lump sum logic. The proposal must include a detailed cost estimate, but grant management later is based on completion of work packages rather than reporting actual costs.

> *The shift is not that budget becomes irrelevant. The detailed estimates are used ex ante to define the lump sum, while reporting focuses on whether the work was completed.*

## What lump sum does NOT mean

- ✓ It does not mean budgeting can be vague
- ✓ It does not mean cost categories no longer matter
- ✓ It does not mean any cost can be classified anywhere
- ✓ It does not mean evaluators stop checking reasonableness

## The difference between evaluator logic and accounting logic

In EIC, the budget is first and foremost an execution and evaluation tool. It must help the evaluator understand whether the planned resources are coherent with the planned work. National cofunding bodies and auditors may read the same project with a more accounting-oriented logic. Both matter, but they are not the same.

## Why budget still matters even without actual cost reporting

Because the quality of the budget is a signal of execution maturity. A company that cannot translate work into resources often also struggles to execute in a disciplined way. Strong budgets make strong implementation sections more believable; weak budgets raise doubts even before the grant agreement stage.

Chapter 4

# How the Implementation Plan and Lump Sum Fit Together

*The inseparable relationship*

## Why these two pieces must not be written separately

The Implementation Plan and the lump sum budget are two views of the same project. One explains the logic of work. The other explains the logic of resources. If different people build them independently without a common structure, contradictions appear immediately.

| When this breaks | Symptom | Evaluator reaction |
|---|---|---|
| Weak technical plan | Budget may look numerically acceptable but evaluator cannot see if costs are justified | Doubts deliverability of large personnel and service blocks |
| Weak budget | Technically convincing plan loses credibility if cost distribution doesn't match work | Suspects proposal was not built from execution logic |
| Written in silos | Contradictions between documents: tasks exist in plan but not in budget | Immediate inconsistency flag |

> *The central rule: money must follow work. Every major budget line must have a visible execution reason.*

> *The reverse rule: work must be reflected in money. If a task is strategically critical but barely resourced, that is a red flag.*

Chapter 5

# The Overall Architecture of the Plan

*Sequence, structure, and grant vs. investment*

## What a good work plan looks like

A good work plan has a clear sequence, well-bounded work packages, concrete tasks, measurable milestones, useful deliverables, visible team ownership, and risks tied to real mitigation. It reads like a structured route from the current state to the next value-creating state.

## The logical sequence of the project

| Phase | Typical content |
|---|---|
| Technical development / refinement | Core R&D, prototyping, algorithm work, system integration |
| Validation and integration | Performance testing, relevant environment validation, benchmarking |
| Regulatory and industrial preparation | Certification pathways, quality systems, scale-up readiness |
| Market readiness | Pilot completion, commercial packaging, route to market confirmation |

## How to separate the four work streams

Development, validation, regulation, and market work can interact — but they should not collapse into one vague technical work package. If those four blocks are mixed without structure, the evaluator struggles to understand progress.

## Grant component vs. investment component

Where both components are present, the grant phase should cover the de-risking and maturation work up to the relevant pre-commercialisation stage, while the investment logic should address later scale, industrial deployment, or market rollout. The handoff must be explicit.

## Designing a plan a non-specialist evaluator can follow

Most evaluators are experienced, but not all will be deep specialists in your niche. Avoid overcompressed jargon. Explain what a task actually achieves. Show why each milestone matters. Translate complexity into operational clarity.

Chapter 6

# Work Packages

*Design, scope, effort allocation*

## What a work package is and what it is not

| A work package IS | A work package is NOT |
|---|---|
| A coherent block of work with a clear objective | A topic bucket or thematic area |
| A scope with visible outputs that can be monitored | A department or organisational unit |
| Defined well enough that completion can be judged | A general area of interest |

## How many work packages you should have

There is no magic number. Too few usually means they are too broad. Too many usually means artificial fragmentation. Under lump sum logic, long-duration work packages may be split along reporting periods — but this should not lead to many small artificial packages.

## Typical work package types in EIC

- ✓ Technical development and core innovation
- ✓ Validation and testing in a relevant environment
- ✓ Regulatory and quality preparation
- ✓ Pre-commercial or market readiness work
- ✓ Project management

## How to allocate effort between work packages

The concentration of effort should reflect where the real technical and execution challenge sits. If 80% of the work is genuinely in one core package, that is acceptable — but the narrative must be correspondingly detailed and the resource logic must support that concentration.

## Warning signs of a badly designed work package

- ✓ Broad title that could mean almost anything
- ✓ Vague objective with no visible output
- ✓ Only one or two oversized tasks with no internal structure

- ✓ No clear completion logic
- ✓ Mismatch between the claimed importance and the effort allocated
- ✓ Core R&D, industrialisation, validation, and market entry all in one block

Chapter 7

# Tasks

*Decomposition, quantification, and execution ownership*

## How to break each work package into tasks

Each task should describe a real step in execution. If a work package is the destination, tasks are the route. Good tasks are specific, bounded, and tied to one clear purpose.

## The five questions every task must answer

- ✓ What exactly is done?
- ✓ Why is it needed?
- ✓ Who does it?
- ✓ What output does it create?
- ✓ Which budget lines support it?

## What level of detail evaluators expect

The official template asks applicants to break each work package into tasks, describe the activities, quantify them where relevant, indicate effort distribution as percentages, and explain who will do the work. This section is expected to be specific.

## How to quantify activities inside each task

Quantification depends on the sector: number of protocols, tests, datasets, design iterations, pilot runs, regulatory dossiers, user interviews, validation campaigns, or integration events. The goal is not to count everything — it is to show that the task has real content.

## How to avoid black box tasks

A black box task sounds important but says almost nothing. Phrases such as 'optimise system architecture', 'improve AI engine', 'prepare for market', or 'enhance performance' are not enough. They must be unpacked into visible actions.

> *A key rule: the riskiest, most resource-intensive task should also be the best explained.*

Chapter 8

# Timeline and Sequence

*Gantt logic, dependencies, and common timing red flags*

## How to build a credible timeline

A credible timeline reflects actual technical, industrial, and regulatory reality. It should not be optimistic for the sake of optics. Evaluators prefer a believable path to an inflated one.

## How to use the Gantt correctly

A Gantt chart is useful only if it reflects the real sequence of work. It must not be decorative. It should show overlap where overlap is realistic, and dependencies where dependencies are real.

## Hidden delays that often break credibility

- ✓ Hiring lag — key profiles may take 3–6 months to recruit
- ✓ Procurement lead times for specialist equipment
- ✓ Pilot access and site permissions
- ✓ Validation cycles requiring multiple iterations
- ✓ Certification timelines driven by notified bodies or regulatory windows
- ✓ Partner dependencies with their own schedules
- ✓ Field seasonality in agricultural, environmental, or infrastructure sectors

## Common timing red flags in evaluation

| Red flag | Evaluator reaction |
|---|---|
| Late critical milestones (all at the end) | Cannot steer the project if something goes wrong |
| Compressed validation windows | Suggests insufficient time to produce credible evidence |
| Major commercialisation claims with almost no pre-commercial time | Looks aspirational rather than planned |
| Contradictory dates across sections | Signals weak document control |

Chapter 9

# Deliverables

*What to include, what to avoid, and naming quality*

## What a deliverable is in EIC

A deliverable is a concrete output used for monitoring the project. It is not a slogan, not a milestone, and not a general ambition.

## Include vs. avoid

| Include these deliverables | Avoid these deliverables |
|---|---|
| Documents that prove work was completed | Deliverables that add paperwork but no monitoring value |
| Outputs that demonstrate validated results | Excessive numbers that reduce clarity |
| Required project management or compliance artefacts | Deliverables scheduled too early or too late to be meaningful |

## Mandatory early deliverables

> *For grant work packages, the official annex expects a Data Management Plan and a dissemination and exploitation deliverable including communication activities within the first six months. These must appear clearly and not be forgotten.*

## Public vs. Sensitive classification

Not everything should be marked sensitive. Use sensitive classification only when there is a real reason. Overusing it can make the plan harder to monitor and can create the impression that transparency is being avoided.

## How to name deliverables well

| Good deliverable name | Weak deliverable name |
|---|---|
| Validated performance report — 100h endurance test | Testing report |
| Regulatory pre-submission dossier — CE pathway | Compliance document |
| Signed pilot acceptance certificate — Site A | Pilot output |

Chapter 10

# Milestones

*Proof points, stage gates, and means of verification*

## What a milestone is and what it is not

| A milestone IS | A milestone is NOT |
|---|---|
| A decision point that shows the project has reached a meaningful stage | Just a date on a calendar |
| A moment of validated progress that closes uncertainty | A deliverable submission |
| A stage gate that enables the next phase | A restatement of a work package objective |

## How to formulate milestones that prove real progress

- ✓ Proves that a validation target was achieved
- ✓ Confirms a regulatory package has been accepted
- ✓ Demonstrates a prototype reaching a defined performance level
- ✓ Verifies a pilot is complete
- ✓ Passes a commercial gate

> *The official template explicitly asks for means of verification and how the milestone contributes to the work package objective — including achievement of a specific TRL level where relevant. This is one of the most important fields in the whole plan.*

## Common milestone mistakes in evaluations

- ✓ Vague milestone wording with no acceptance criteria
- ✓ Missing or weak means of verification
- ✓ Milestones scheduled too late to guide execution
- ✓ Milestones that do not actually prove the maturity claimed
- ✓ Milestones that are really just deliverables in disguise

Chapter 11

# Means of Verification

*Evidence types and quality standards*

## What kind of evidence convinces evaluators

Evaluators trust evidence that is specific, external where possible, and proportionate to the claim being made. A milestone claiming validated performance needs stronger proof than one claiming a management protocol was created.

| Evidence type | Examples of strong evidence | Examples of weak evidence |
|---|---|---|
| Technical | Validated test reports, benchmark results, independent technical assessments | Internal claims, generic 'optimization' outputs |
| Clinical | Approved trial milestones, patient recruitment data, hospital validation records | Informal safety observations |
| Industrial | Pilot acceptance reports, quality audits, process capability data | 'Manufacturing feasibility confirmed' |
| Regulatory | Notified body interactions, audit outputs, certification milestones | 'CE process started' |
| Commercial | Signed partner agreements, validated willingness to pay, structured pilot conversions | Letters of intent without commitment |

## How to write verification without becoming vague

> *State what will exist, who will validate it, how it will be checked, and why it proves the milestone has been achieved.*

Chapter 12

# Risks and Mitigation

*Risk taxonomy and how to write operational mitigation*

## What counts as a critical risk

The official annex defines a critical risk as a plausible event or issue that could have a strong negative impact on the ability of the project to achieve its objectives. It is not just any problem — it is something that could materially derail execution.

| Risk category | Typical examples | Common omission |
|---|---|---|
| Technical | Performance, robustness, scale-up feasibility, data quality | Core algorithm risks often understated |
| Validation / TRL | Failure to reproduce results, insufficient user acceptance | Claimed TRL not backed by relevant environment testing |
| Regulatory | Delays in certification, quality system gaps, changing requirements | Often treated as a post-project issue |
| Commercial | Slow adoption, low willingness to pay, partner disengagement | Treated as 'assumed solved' |
| Team / hiring | Recruitment delays, founder overload, missing capability | Hiring lag rarely modelled |
| Financial / supply | Budget overrun, supplier instability, component shortages | Underestimated co-financing risk |

## How to write operational mitigation

Mitigation must describe what you will actually do: add a fallback route, identify a partner alternative, add an earlier test, add a buffer, add a recruitment backup, or change the sequence.

> *Phrases such as 'monitor closely', 'remain flexible', or 'adapt as needed' are not enough unless followed by a concrete action mechanism.*

Chapter 13

# Human Resources and Execution

*Person months, team structure, and hiring logic*

## How to assign person months credibly

Person months must reflect the true difficulty and intensity of the work. They should not be reverse-engineered merely to fit a budget target.

## Key HR questions evaluators ask

- ✓ Do the person months by work package reflect the real effort of the work?
- ✓ Is the coverage complete? — technical, validation, regulatory, commercial, and management
- ✓ Are missing competencies identified and linked to a realistic hiring plan?
- ✓ Does the leadership commitment look realistic?
- ✓ If founders are heavily committed to the project, who covers daily operations?

## What a strong hiring plan contains

| Required element | Weak version |
|---|---|
| Named profiles with specific skills | 'We plan to expand the team' |
| Timing linked to when the work actually needs the role | All hires listed without timing |
| Contingency if recruitment slips | No fallback mentioned |
| Gender balance commitment with operational detail | A rhetorical sentence |

> *Common trap: assigning 100% of the CEO to the project without explaining who covers the rest of the business. If unresolved, this weakens credibility.*

Chapter 14

# Partners, Subcontractors, and Third Parties

*When they add value and when they add risk*

## When a partner strengthens the plan

A partner strengthens the plan when its role is specific, necessary, and timely. A named partner with a clear contribution creates credibility. A logo without a role does not.

| Partner role that adds credibility | Partner role that adds risk |
|---|---|
| Specific task contribution with defined output | Heavy dependency with no fallback |
| Named role tied to a work package milestone | Partner carries the real delivery burden |
| Complementary capability the company lacks | Core innovation task pushed to external party |

## What should NOT be subcontracted

> *The official annex is explicit: core tasks of the project should not be subcontracted. The scientific, technological, and strategic core of the innovation must remain with the beneficiary.*

## How to describe the role of each partner

- ✓ What they do in the project
- ✓ Why they matter for the specific objective
- ✓ What output they provide
- ✓ How their contribution affects the success of the project
- ✓ What happens if they delay — and what the fallback is

Chapter 15

# TRL and Technical Progress

*Maturity claims, evidence, and how evaluators check them*

## What TRL means in EIC practice

In EIC Accelerator, TRL is not a slogan. It is a maturity claim that must be backed by validation evidence in a relevant environment and by a credible path to the next level.

## How to prove the starting point

You need to show what has already been built, tested, validated, and learned. This is especially important when the innovation contains several components and only some have reached the stated maturity.

## How to avoid unsupported TRL claims

| Typical TRL mistake | Why it fails | What to do instead |
|---|---|---|
| Claiming TRL 6 when only peripheral modules are tested | Core innovation remains less proven than stated | Separate core component TRL from system TRL |
| Confusing proof of concept with relevant environment testing | Lab conditions ≠ relevant environment | Specify the environment and why it is relevant |
| TRL mentioned only in WP headers | Not supported by task/milestone logic | Connect each TRL progression to a specific task and milestone |

> *What kind of validation supports maturity progression: relevant environment validation, benchmark performance, independent assessment, integration testing, user confirmation, and reproducibility.*

Chapter 16

# Regulation, Certification, and Compliance

*When and how to integrate regulatory work*

## When you need a dedicated regulatory work package

You need a visible regulatory work package when certification, compliance, or formal evidence generation is critical to reaching the next value point.

## How to treat regulatory pathways as execution tracks

CE, FDA, ISO, and equivalent pathways should be treated as execution tracks with tasks, responsibilities, timing, dependencies, and outputs — not as one-sentence promises at the end of the plan.

## Common mistakes in compliance planning

- ✓ Missing regulatory sequencing — treating certification as a post-project step
- ✓ Unclear ownership — no named person responsible for the regulatory pathway
- ✓ Underbudgeted quality work — missing personnel and service costs for QMS preparation
- ✓ Claiming market readiness before certification logic supports it
- ✓ No integration with the project timeline or milestones

Chapter 17

# Pre-Commercialisation and Market Readiness

*From technical readiness to practical adoption*

## What should be included before launch

Pre-commercialisation includes the work needed to move from technical readiness toward practical adoption: pilot design, pricing validation, key partner onboarding, commercial packaging, and early route-to-market preparation.

| Pre-commercial work that belongs in the plan | What to avoid |
|---|---|
| Structured pilot with acceptance criteria | A generic 'commercialisation WP' with only high-level phrases |
| Customer validation generating specific market evidence | Claiming market launch without showing the pre-commercial path |
| Strategic alliances tied to specific execution tasks | A logo or name without a role |
| Market access tasks: account development, distributor prep | Postponing all market evidence to after the grant phase |

> *Commercial evidence generation means generating the evidence the market needs to adopt: ROI logic, pilot outcomes, compliance proof, workflow validation, or reference customer data.*

Chapter 18

# How to Write an Implementation Plan That Sounds Like Execution

*Language, tone, and the five tests*

## Language that works

Use precise, action-based language. Describe what will be done, by whom, for what purpose, and what proof will exist when it is finished.

## Language that creates suspicion

Evaluators become suspicious when they read abstract, inflated, or repetitive language. They lose trust when the proposal sounds more like brand positioning than project control.

| Language that works | Language that creates suspicion |
|---|---|
| 'We will conduct 48-hour endurance tests under ISO 9227 conditions to validate corrosion resistance at TRL 6' | 'We will further develop and optimise our innovative solution' |
| 'Regulatory consultant will prepare CE pre-submission package for MDR Article 62' | 'We will pursue regulatory approvals as needed' |
| 'CEO (0.6 FTE) leads WP3; CTO (0.8 FTE) leads WP1-2' | 'Our experienced team will manage the project' |

## How to turn claims into actions

> *Whenever you make an important claim, ask immediately: what activity produces this outcome? What milestone proves it? What resource enables it? What risk threatens it?*

## Write for evaluators, not for internal morale

The Implementation Plan is not the place to inspire your team. It is the place to help an external evaluator believe that your team can execute. Remove any sentence that sounds positive but contains no execution content.

Chapter 19

# Lessons Learned from Evaluations

*Patterns that lead to GO and NO GO*

## What evaluators penalise most often

- ✓ Lack of detail in task descriptions
- ✓ Weak task decomposition inside work packages
- ✓ Overbroad work packages that cover too many different objectives
- ✓ Underexplained core technical work package
- ✓ Vague milestones without means of verification
- ✓ Generic risks that apply to any startup
- ✓ Budget logic that does not match the work

## What tends to convince evaluators

- ✓ Clear structure with visible sequence
- ✓ Strong means of verification for every key milestone
- ✓ Realistic timing that integrates hiring, regulation, and validation
- ✓ Coherent resource allocation — money following work
- ✓ Explicit role ownership for every critical stream
- ✓ A credible path from TRL 6 toward TRL 8

| GO leaning proposal | NO GO leaning proposal |
|---|---|
| Structured path with logical sequence | Evaluator must fill in missing execution logic |
| Meaningful milestones with clear verification | Milestones are dates or deliverable restatements |
| Proportionate resources aligned to work | Budget disconnected from the work plan |
| Strong evidence logic | Technically ambitious but operationally vague |

# PART II — THE LUMP SUM BUDGET

Chapter 20

# What the Lump Sum Budget Document Actually Is

*Workbook structure, mandatory sheets, and formal requirements*

## What the detailed lump sum budget workbook is

For EIC Accelerator grant proposals using project-based lump sum logic, the company must submit a detailed budget table showing realistic cost estimates by beneficiary, by work package, and by cost category. Uploading the file is mandatory — proposal submission is blocked if the workbook is missing.

## What sheets it contains

| Sheet | Purpose |
|---|---|
| BE list | Define the beneficiary and affiliated entities |
| WP list | Define the work packages (must match the proposal) |
| BE1 sheet | Enter costs per work package and cost category |
| Depreciation costs sheet | Calculate equipment costs charged via depreciation (required if equipment is budgeted this way) |

> *The structure is fixed and must not be changed. The workbook, the Implementation Plan Annex, and the Part A requested grant amount must tell the same story.*

## What is calculated automatically — and what is not

The workbook calculates totals, indirect costs, and summary views automatically. However, the result of the depreciation tool is NOT automatically transferred into the BE1 budget sheet. The applicant must move the charged depreciation cost manually into category C.2 Equipment under the relevant work package.

## Formal mistakes that weaken or block the proposal

- ✓ Not uploading the workbook
- ✓ Changing the workbook structure
- ✓ Using a file format the submission system will not accept
- ✓ Entity structure inconsistent with the proposal
- ✓ WP list inconsistent with the Implementation Plan
- ✓ Using cost categories that are not allowed

Chapter 21

# Principles for Building the Lump Sum Budget

*Eligibility, reasonableness, and the right build order*

> *A common misunderstanding: lump sum changes the eligibility logic of costs. It does not. Cost estimations must be an approximation of actual costs and must follow the same eligibility logic as actual cost grants.*

## The central test: reasonable, non-excessive, and sufficient

The evaluator does not ask whether your budget is high or low in absolute terms. The evaluator asks whether it is reasonable, non-excessive, and sufficient for the activities proposed. A budget can fail by being inflated, but it can also fail by being implausibly light — underbudgeting suggests the company does not understand what the work actually requires.

## The correct build order

| Step | Action |
|---|---|
| 1. Define the work | Work packages, tasks, milestones, outputs |
| 2. Identify the resources | Who does what, with what equipment, with what external support |
| 3. Quantify the resources | Translate into person months, cost categories, and amounts in the workbook |

> *If you start from a top-down target grant figure and reverse-engineer the work plan to fit, the proposal almost always becomes incoherent. Strong budgets are built from execution logic, not from target grant logic.*

## Signs of cosmetic budgeting

- ✓ Evenly distributed numbers across unrelated work packages
- ✓ Round figures that do not correspond to any cost reality
- ✓ Major spend with no visible task behind it
- ✓ Cost structure clearly disconnected from the Implementation Plan

Chapter 22

# Beneficiaries, Affiliates, and Administrative Structure

*BE list, WP list, and consistency requirements*

## The BE list

The BE list defines the beneficiary and any affiliated entities. The workbook requires the same entities as in the main proposal, in the same order. For each entity, the country must be selected correctly. Mistakes here create structural inconsistencies throughout the budget.

## The WP list

The WP list must correspond to the work packages listed in the main proposal, in the same order. The workbook is not designed to invent a second structure — it is designed to mirror the execution structure already presented in Part B.

## The pre-submission consistency check

- ✓ Same entities in same order
- ✓ Same work packages with same titles
- ✓ Same resource logic across documents
- ✓ Same requested grant logic in Part A
- ✓ Countries entered consistently for all entities

## Common administrative errors

- ✓ Missing affiliated entities
- ✓ Misordered entities vs. main proposal
- ✓ Work packages renamed in the workbook
- ✓ Cost categories populated in the wrong beneficiary section
- ✓ Forgetting that associated partners do not complete the detailed lump sum budget table

Chapter 23

# Personnel Costs

*Categories, person months, and suspicious patterns*

## How personnel costs are entered

In the BE1 sheet, personnel costs are entered under section A. The workbook treats the number of items as person months and the cost per item as the average monthly cost for the people concerned. It is asking for a realistic and coherent estimate of effort and average monthly cost by staff category and work package.

| Personnel category | Notes on correct use |
|---|---|
| Senior scientists / researchers | Use for profiles driving core technical or regulatory work |
| Junior scientists / researchers | Use for supporting technical profiles, data work, testing |
| Technical personnel | Use for engineers, lab technicians, industrial specialists |
| Administrative personnel | Use for project management, finance, reporting roles |
| SME owners / natural person beneficiaries | Pre-filled cost per item; enter number of items only |

## What makes personnel costs look suspicious

- ✓ Clearly too high for the geography and seniority profile without explanation
- ✓ Disconnected from the person month allocation in the work packages
- ✓ Senior profiles dominating without justification
- ✓ Small team asked to deliver unrealistic total effort

> *Personnel costs above dashboard values can be accepted if justified in the proposal or comments tab. Typical valid reasons: strong seniority, scarce skills, or market rates in a high-cost geography. Explain it before the evaluator has to guess.*

Chapter 24

# Subcontracting

*What can be externalized and how to justify it*

> *The official annex is explicit: core tasks of the project should not be subcontracted. The scientific, technological, and strategic core of the innovation must remain with the beneficiary.*

## The four questions every subcontracting entry must answer

- ✓ What exactly is being outsourced?
- ✓ Why is it needed for the project?
- ✓ Why does it not replace a core task?
- ✓ Why is the chosen supplier economically justified (best value for money)?

## Warning signs for evaluators

| Warning sign | Evaluator interpretation |
|---|---|
| Very large subcontracting block vs. internal personnel effort | Company may not be the real driver of the project |
| Generic scope without acceptance criteria | Complementarity cannot be verified |
| External work appears to carry the real delivery burden | Substitution rather than complementarity |
| Misclassified as purchase cost or vice versa | Budget control is weak |

Chapter 25

# Purchase Costs

*Travel, equipment, goods, and the 15% threshold*

| Cost category | Key rules |
|---|---|
| C.1 Travel and subsistence | Must support real project needs; challenged when high and not justified by the work plan |
| C.2 Equipment | Usually entered via the depreciation sheet; distinguishes equipment, infrastructure, and other assets |
| C.3 Other goods, works and services | Consumables, dissemination, publication fees, support services; must not become a catch-all |
| Remaining purchase costs | Items below the 15% of personnel costs threshold |

> *The 15% threshold: items exceeding 15% of personnel costs must be listed and justified individually. Always start with the largest and most consequential items — these drive the evaluator's overall budget impression.*

Chapter 26

# Equipment

*Classification, depreciation, and common mistakes*

Equipment belongs under C.2 in the workbook but is not a synonym for any technical resource. The workbook distinguishes between equipment, infrastructure, and other assets.

## What should NOT go into equipment

- ✓ Pure services or subscription-based software
- ✓ Ordinary dissemination spend, workshops
- ✓ IP database access or connectivity subscriptions
- ✓ Generic legal or support work
- ✓ Anything that behaves like a service, licence, or access right rather than an asset

## Typical equipment mistakes

- ✓ Treating all technical costs as equipment
- ✓ Leaving the depreciation tab blank next to a visible equipment block
- ✓ Failing to show project-specific use percentage
- ✓ Not carrying the depreciation result into BE1

Chapter 27

# Depreciation

*Calculation logic and the manual transfer step*

| Depreciation field | What it means |
|---|---|
| Purchase cost | Gross acquisition price — the basis for calculation |
| % used for the project | Share of asset use truly attributable to the action |
| % of useful life in the project | Project duration as a share of total depreciation period (months of project use ÷ depreciation period × 100) |
| Charged depreciation cost | The output — must be manually entered into BE1 under C.2 Equipment |

> *Critical: the charged depreciation cost is NOT automatically transferred to BE1. You must enter it manually in category C.2 Equipment under the appropriate work package. Missing this step breaks the internal logic of the budget.*

Chapter 28

# Rental, Leasing, and Full Equipment Costs

*When standard depreciation does not apply*

| Treatment type | When it applies | What to do |
|---|---|---|
| Rental / leasing | Project uses rented or leased equipment during the grant period | Enter usage cost under C.2 without the depreciation sheet; explain in comments tab |
| Full equipment cost | Topic conditions explicitly allow full capitalised costs | Enter full purchase cost under C.2; note that this is topic-specific and must be explicitly allowed |

> *Whenever you use rental, leasing, or full equipment cost treatment, explain it in the comments tab. The workbook guidance explicitly recommends doing so.*

Chapter 29

# Other Goods, Works, and Services

*Scope, typical items, and when it becomes a problem*

This category captures direct purchase costs that are not travel and not equipment: consumables, meeting and seminar services, dissemination services, publication fees, software licences, cloud access, testing, and QA support.

## How to avoid making this a catch-all

When this category is large, the evaluator needs to understand the major drivers. Break it down and tie each significant item to a specific execution need: validation, compliance, industrialisation, or commercialisation.

> *A large 'other goods, works and services' block with no decomposition is one of the most common triggers for evaluator concern.*

Chapter 30

# Indirect Costs and Part A Alignment

*How they are calculated and when Part A may differ*

| Item | Rule |
|---|---|
| Indirect costs calculation | 25% of direct personnel (A) plus direct purchase costs (C) — calculated automatically in the workbook |
| Part A = workbook maximum (standard case) | When there is no operating grant and no own resources or income to consider |
| Part A < workbook maximum (operating grant case) | Indirect costs must be subtracted because they are not eligible when an operating grant exists |
| Part A < workbook maximum (own resources case) | Requested amount adjusted downward to maintain a balanced budget |

> *Always compare the Part A requested amount with the workbook maximum and ask: if they differ, is the reason explicit and documented? If not, the proposal is not ready.*

Chapter 31

# The Comments Tab

*The most underused credibility tool in the workbook*

## When to use it

Use the comments tab whenever the budget includes something that is correct but not self-evident. The workbook itself explicitly suggests this for high personnel costs, unusual equipment treatments, and any discrepancy between the workbook amount and Part A.

| What to explain in comments | Why it matters |
|---|---|
| High personnel costs — seniority or market rate above dashboard | Prevents automatic suspicion; a short explanation can preserve a defensible cost line |
| Equipment via depreciation | Makes the depreciation logic visible, especially for large equipment blocks |
| Equipment via rental or full cost | Required by the workbook guidance; avoids misinterpretation |
| Part A ≠ workbook maximum | Without explanation, the difference looks like an error |

> *The best comments are not defensive — they are preventive. They explain the few lines most likely to attract scrutiny and save the evaluator time.*

Chapter 32

# Lessons Learned from Budget Evaluations

*What gets cut, what gets questioned, and what builds trust*

## What evaluators question most often

- ✓ High personnel costs without explanation in comments or narrative
- ✓ Vague subcontracting blocks with no scope or acceptance criteria
- ✓ Major service blocks weakly justified and not linked to deliverables
- ✓ Equipment with missing or blank depreciation logic
- ✓ Budget that does not visibly support the Implementation Plan work packages

## What evaluators cut most often

- ✓ Weakly justified personnel rates above dashboard values
- ✓ Meeting and seminar services that look excessive relative to the work
- ✓ Equipment where project use or depreciation is not substantiated
- ✓ Commercial or support services not clearly linked to deliverables
- ✓ Inflated travel and dissemination costs

> *The official Horizon guidance is explicit: clearly overestimated or underestimated costs can reduce the score and can lead to a lower requested amount or reallocations across work packages. Do not game the number — build the budget from the work, then defend it clearly.*

# PART III — ALIGNMENT: IMPLEMENTATION PLAN & BUDGET

Chapter 33

# The Total Coherence Matrix

*Six dimensions of alignment*

By the time a company reaches Full Proposal stage, the main challenge is no longer writing isolated sections well. The challenge is making the full proposal behave like one project.

| Coherence dimension | What the evaluator checks |
|---|---|
| WP by WP | Each work package reads as a complete execution block with objective, scope, tasks, milestones, and cost structure proportionate to its importance |
| Task by task | Each task answers the five questions: what, why, who, what output, what budget support |
| PM by PM | Person months match the narrative complexity, task volume, and strategic importance of each work package |
| Deliverable by deliverable | Each deliverable is linked to a task, timed correctly, and has monitoring value |
| Milestone by milestone | Each milestone closes real uncertainty, has a strong means of verification, and is not pushed too late |
| Cost by cost | Every material cost line has an execution reason: it supports a task, enables validation, or removes a risk |

> *A misconception: because the EIC grant is lump sum, detailed coherence matters less. In reality, coherence matters more — the evaluator is explicitly instructed to assess whether the cost estimations are reasonable and whether the proposed resources allow completion of the activities as described.*

Chapter 34

# How to Review Consistency Before Submission

*The five-step review sequence*

| Review step | What to check |
|---|---|
| 1. Part B internal logic | Work packages coherent, sequence visible, tasks real, milestones verifiable |
| 2. Workbook vs. Part B | Same WP names, same order, same entities, same cost category treatment |
| 3. Part A vs. workbook | Requested amount matches or differs for a documented reason |
| 4. Hiring plan vs. execution | Missing competencies are named, timed, and linked to work package delivery |
| 5. De-risking logic vs. cost allocation | Budget is allocated where the real uncertainties sit — regulatory, technical, or commercial |

> *Under lump sum logic, the proposal is largely evaluated as-is. The company should not rely on grant preparation to repair structural inconsistencies that should have been resolved before submission.*

Chapter 35

# How Evaluators Read the Plan and Budget Together

*What builds trust and what destroys it*

## What evaluators are trying to verify

- ✓ Is the project structured well enough to reach the claimed maturity?
- ✓ Are the milestones real and monitorable?
- ✓ Are resources sufficient and not excessive?
- ✓ Is the budget split aligned with the work?
- ✓ Does the company understand its own execution risks?
- ✓ Can this team actually do what it says it will do?

## What creates distrust

Distrust begins with inconsistency. Generic tasks. Repeated task descriptions. A blank depreciation sheet. Subcontracting that cannot be traced. Work packages whose objectives do not match their tasks. Person months that do not match the budget. Milestones pushed to the end when they should guide the project earlier.

| What creates a GO impression | What creates a NO GO impression |
|---|---|
| Coherent before impressive | Strong technology narrative with thin execution logic |
| Milestones that close real uncertainty | Milestones that only mark the passage of time |
| Budget clearly following the work | Budget that looks assembled rather than designed |
| Risks identified with operational mitigation | Generic risks copied from a template |
| Clear de-risking logic visible in resource allocation | Budget overloaded in familiar categories, underfunded in bottlenecks |

# PART IV — PRACTICAL TOOLS FOR CLIENTS

Chapter 36

# What We Need From the Client Before Drafting Starts

*Six input packages*

The quality of the Implementation Plan is determined long before anyone starts writing. Most weak plans are not weak because the writer lacks skill — they are weak because the company never assembled the right raw material in the first place.

| Input package | Key content required | Why it matters |
|---|---|---|
| Technical package | Current TRL evidence, what has been built and tested, core innovation vs. enabling elements, open uncertainties, what must be achieved during the grant phase | Enables real work package design and task decomposition |
| Validation package | Internal and external tests, pilot results, benchmarking, field trials, signed partner or customer feedback, evidence categories by validation type | Milestones and means of verification must emerge from the actual validation logic |
| Regulatory package | Formal route, current status, critical path, remaining steps, external advisors or bodies, realistic regulatory timeline | Regulatory work belongs inside the plan and the lump sum from the beginning |
| Market package | Target segment, buyer/user, route to market, proof of demand, pricing logic, adoption barriers, next market evidence needed | Pre-commercial WP must generate real market evidence, not just a commercial slide |
| Team package | Execution map with roles, missing competencies, hiring plan, external advisors, partner contributions, CEO/CTO time allocation | Work plan and budget must reflect real execution capacity |
| Financial package | Personnel cost assumptions, major cost drivers, subcontracting logic, equipment treatment, co-funding constraints, Part A logic | Budget must be built from the work, not from a target grant figure |

> *Before drafting begins, the client should deliver all six packages. They do not need to be perfect — but they must be concrete enough to support a real implementation logic.*

Chapter 37

# Client Checklist: Implementation Plan

*Quality control tool for use on a first serious draft*

> *This checklist is for after a first serious draft exists. It is a quality control tool, not a writing template. Use it to test whether the content is strong enough to survive evaluator scrutiny.*

## Work packages

- ✓ Does each work package have one clear objective rather than several loosely connected ambitions?
- ✓ Can each work package be explained as a coherent execution block?
- ✓ Is the sequence between work packages visible and logical?
- ✓ Is the core technical work package the one that is most clearly described?
- ✓ Is there a work package that is suspiciously short on tasks or milestones despite being important?
- ✓ Are all four streams — development, validation, regulation, and market — visibly represented?

## Tasks

- ✓ Does each task describe a real step in execution — not an activity category?
- ✓ Are the most critical tasks the most detailed ones?
- ✓ Is the effort split across tasks proportionate to actual complexity?
- ✓ Are there any tasks that could be removed without losing monitoring value?
- ✓ Is the owner of each critical task clearly identifiable?

## Deliverables

- ✓ Are there too many deliverables?
- ✓ Are too many of them marked Sensitive without strong reason?
- ✓ Could an evaluator understand what success looks like from the deliverable list alone?
- ✓ Is the Data Management Plan included and due within the first 6 months?
- ✓ Is the dissemination and exploitation deliverable included within the first 6 months?

## Milestones

- ✓ Does each milestone prove progress rather than just passage of time?
- ✓ Does each milestone have a clear means of verification?
- ✓ Is the milestone tied to the objective of its work package?
- ✓ Are key milestones spread through the project rather than pushed to the end?
- ✓ Could a third party objectively decide whether the milestone has been reached?
- ✓ Does the milestone list show the path to the claimed TRL progression?

## Risks

- ✓ Are the risks specific to the project rather than generic to any startup?
- ✓ Are all major technical, validation, regulatory, commercial, and hiring risks visible?
- ✓ Are mitigation actions concrete and operational — not 'monitor closely'?
- ✓ Is the risk table connected to the structure of work packages and milestones?

## Roles and execution ownership

- ✓ Is every critical stream owned by a named role?
- ✓ Is it clear who leads technical work, validation, regulation, and market work?
- ✓ Are missing competencies acknowledged and linked to a hiring or partnering plan?
- ✓ If founders are heavily committed, is the rest of company management covered?

Chapter 38

# Client Checklist: Lump Sum Budget

*Budget quality control — use together with the Implementation Plan checklist*

> *Many budget weaknesses only become visible once the work plan and the budget are compared. Use this checklist together with the Implementation Plan checklist, not instead of it.*

## Personnel

- ✓ Do the person months by work package reflect the real effort of the work?
- ✓ Are staff categories used correctly?
- ✓ Are average monthly costs plausible for the geography and profile?
- ✓ If personnel costs are high, are they explained in the comments tab?
- ✓ Are there work packages with clearly too little or too much personnel effort?
- ✓ Does the hiring plan support the PM logic?

## Subcontracting

- ✓ Is each subcontracting line clearly defined with scope and acceptance criteria?
- ✓ Is it obvious why that work is externalised?
- ✓ Is it clear that the subcontracted work is not a core task?
- ✓ Is best value for money logic explained?
- ✓ Do the subcontracting lines trace between Part B and the workbook?

## Equipment and depreciation

- ✓ Is each equipment line really equipment and not a service, licence, or access right?
- ✓ Does the equipment support a visible work package task?
- ✓ Is the depreciation sheet completed where needed?
- ✓ Has the resulting charged depreciation cost been manually transferred into BE1?
- ✓ Is the treatment (depreciation, rental, or full cost) explained in the comments tab?

## Part A consistency

- ✓ Does the requested amount in Part A match the workbook maximum?

- ✓ If not, is there a documented reason (operating grant, own resources)?
- ✓ Do the work packages appear in the same order across all documents?
- ✓ Is the cost story consistent between Part A, Part B, and the workbook?
- ✓ Could a grant preparation reviewer understand the numbers immediately?

Chapter 39

# Red Team Review Before Submission

*Twenty questions on execution, twenty on budget, ten on perception*

## Why a red team review is essential

A final red team pass is one of the most valuable things a company can do before submission. The purpose is not to improve style. The purpose is to identify the reasons why an evaluator might say no. At Silicon Capital Partners, we recommend attacking the proposal from three angles: execution credibility, budget credibility, and evaluator perception.

## Twenty questions on execution credibility

- ✓ If the evaluator reads only the work packages, will the project still look coherent?
- ✓ Is the core technical work package the best described one?
- ✓ Does the plan show a believable route from the current maturity to the claimed end point?
- ✓ Is the TRL progression supported by tasks and milestones, not just by labels?
- ✓ Are any milestones' means of verification too weak for the claims made?
- ✓ Are critical risks real, specific, and linked to operational mitigation?
- ✓ Are technical, regulatory, and market adoption risks all visible?
- ✓ Is the hiring plan connected to the work packages?
- ✓ Do partner roles look concrete?
- ✓ Does the project remain executable if one key partner slips?
- ✓ Are timing dependencies realistic, including hiring lag?
- ✓ Is the project management logic visible?
- ✓ Does the plan separate development, validation, regulation, and market preparation clearly?
- ✓ Is the proposed outcome of the grant phase fully clear?
- ✓ Would a skeptical external reviewer believe the company can actually deliver this plan?

## Twenty questions on budget credibility

- ✓ Does the budget clearly follow the work?
- ✓ Are the highest budget blocks linked to the highest effort or highest risk areas?
- ✓ Are person months believable by work package?
- ✓ Are high personnel costs explained?
- ✓ Is subcontracting limited to non-core work with best value for money logic?

- ✓ Are purchase costs above the threshold properly justified?
- ✓ Is equipment category usage conservative and defensible?
- ✓ Is the depreciation sheet correctly used and result transferred to BE1?
- ✓ Are large service categories clearly decomposed?
- ✓ Is the comments tab used well for anything non-obvious?
- ✓ Is Part A consistent with the workbook?
- ✓ Does the budget look neither inflated nor unrealistically lean?
- ✓ Would an evaluator trust this budget as a serious estimate built from execution logic?

## Ten questions on evaluator perception

- ✓ Does the proposal feel designed or assembled?
- ✓ Does it read like a company that understands execution?
- ✓ Does it read like a company that understands de-risking?
- ✓ Does it look specific enough to monitor?
- ✓ Does it look credible enough to fund?
- ✓ Are there visible contradictions that distract from the strengths?
- ✓ Does the company look in control of its own project?
- ✓ Does the budget strengthen the plan or weaken it?
- ✓ If the evaluator had to defend a GO, does the proposal give enough evidence to do so?

## Final kill points before submission

> *These issues should be treated as submission killers unless fixed: blank or incorrect depreciation logic next to visible equipment spend; core work package underdescribed; major budget blocks with no visible project purpose; contradictions between Part A, Part B, and the workbook; milestones with no meaningful means of verification; hiring-dependent execution with no hiring timing; broad regulatory or commercial claims with almost no visible work behind them.*