THE BLACK UNICORN

CONTENT CREATION CASE STUDY DOCUMENT

Client: Lexunit
Industry: IT (AI solutions, Data Science)

Title: What is a Proof of Concept, and How to Build One?

Published: September 2022

BENEDEK, Gergő
gergo@theblackunicorn.eu
+36206175791

theblackunicorn.eu/

What is a Proof of Concept, and How to Build One?

Published:

September 6, 2022

The PoC is a key stage of software development that is especially important in innovation partnerships and IT staff augmentation projects.

The PoC is a way to validate **scalability, technical potential**, and the **key functionalities** of your final product on a smaller scale.

Much like an MVP in the startup world, it is the first working iteration of the tool you plan to build.

But very much unlike the MVP, which can be a 'fake door' on your website to test the demand for a future solution (or just a video explaining what your product will do, as was the case with, for example, Dropbox), the PoC already contains the **core mechanics** of the service, and you build from there.

The PoC should be able to demonstrate the technical **feasibility** of your idea.

From here onwards, the task will be to implement the solution for real, make it efficient, eliminate room for error and deploy it into your systems, seamlessly.

Building a PoC, in general, is our primary goal right after we finish the **overview of our client's operations with an engineering mindset** and **identify a key area where we can generate business value** through higher efficiency.

This usually means some kind of automation, and it involves Machine Learning.

Okay, so how do we get **from establishing contact** with a business **to a working model of a solution** to one of their problems?

Looking for Business Value

Before you start working on any project, you need to define the business value first.

How will it increase profits?

How can it help you scale more quickly?

This is how we do this:

After the intro meeting, we enter the **Tech Consultancy Phase**, involving the tech experts from both sides. We assess the client's business needs, expectations, requirements, short/mid/long-term objectives, and technology.

In most cases, **we are not invited to solve a singular challeng**e - that would limit the scope of how we can help the organization. So, these meetings usually become brainstorming sessions about how we can bring value to you.

We aim to understand the **high-level vision**. In the case of startups, this vision is usually fluid. If the startup knew precisely how to reach the goal, it wouldn't be a startup but a small business.

Here are some examples of this **high-level vision**, picked from our projects:

'Lowering the cost of stock management for pharmacies through automation'

'Automating the job seeking process' [CASE STUDY]

'Finding available up-to-date expert analysis' about legislation changes' [CASE STUDY]

'Eliminating the chance of human error at a packaging station' [CASE STUDY]

The preliminary meetings are about understanding how the business leadership is thinking about business execution currently.

This could involve conditions like:

- what is the time frame,

- how are they timing the marketing efforts,

- will they employ test users,

- do they have a business model (not every startup has one at the beginning)

And there are also **external factors**, like the generic unit economics of the space they inhabit. What do paid services look like in this industry? How do these services scale?

In this phase, besides the talks, we are **researching** the available documentation and the startup's market segment.

During the continuous assessment of these inputs, we catalogue and organize our findings to make the learning process smoother.

Of course, if you strive for perfection, you could find yourself in a never-ending loop of talks, discussions, and iterations. There's always more to learn about a business. **So how do we know that our understanding is sufficient to start building?**

It's actually not that complicated.

We reach this milestone when:

- we can give useful feedback

- we can develop the available ideas further

- we can advise about what could be done to improve the execution

- and if we feel that we have a firm grasp on the ***priority order of the tasks*** *necessary for executing the high-level vision.*

Finally, and hopefully, soon enough, we arrive at the point where we can take the initiative and come forward with suggestions.

At this point, we are focusing on 'quick wins' and low-hanging fruits: we show you options with the best chances to create **the most value in the lowest amount of time**. We do this because we believe in getting tangible results within a reasonable time frame. (Read more on this and about how businesses can prepare well for AI projects here.)

Proof of Concept is about Risk Management

Once the Business Value is identified, you almost always instantly realize where you should go with your first PoC.

The very reason you need 'Proof' that your 'Concept is working is to eliminate (or rather contain and lower) risks. **If there is no risk, you don't need a PoC**.

What kind of risks are we talking about here?

The first one is the **Technological Risk**, which you usually think about in an IT project. We try to identify the most significant risk factors that could even derail the project.

Some typical examples of Technological Risks we usually encounter:

- accuracy variance of the Machine Learning model

- stability of the website crawler

- achieving optimal tuning of Big Data tasks on the speed/cost ratio

These are the typical risks we will need to navigate by carefully assigning where we accumulate Technical Debt.

You need to start building somewhere, and you will encounter problems. You will consciously decide to solve these problems later to get to the first major milestones faster. A thorough understanding of the Business Value is crucial to performing well at these decision points.

And there is another category called **Market Risks.**

It doesn't matter that the product runs well when nobody is interested!

If the Product-Market Fit is insufficient due to a UX or business model problem, if your users can't get value out of it, your project will still fail without pivoting or some major tuning. The fact is, **you'll still need a PoC to figure this out.**

The key is that your Technological Partner can help build a PoC tailored for testing these Risks. The business leadership works together with the Partner in deciding **how the PoC should prioritize** these two kinds of risks.

In short, what kind of questions do you want answers for?

Here are some examples of the Risks from the projects we mentioned earlier:

<u>1. High Level Vision: 'Automating the job seeking process</u>

Technological Risks:

- automating the application process through a third-party web app carries significant stability risks

*- the **volume** of finding certain kinds of jobs need to have a similar average (if we find 3 bookkeeper positions and 5000 carpenters a week, the discrepancy threatens stability)*

Market Risks:

- where will new users come from?

2. High Level Vision: 'Eliminating the chance of human error at a packaging station'

Technological Risks:

- we need to make sure that the image recognition is 100% accurate

- feedback needs to be real-time, so we need to prioritize Speed when running the algorithm

This project also offers examples of what doesn't need to be included in a PoC: **all we need to prove the concept** is the core algorithm and ensure high data quality. We ran the PoC on-premise, on a laptop, with a single video camera - everything else (databases, UX, networking) we can figure out later because those components are already proven.

So how do you know which components we should include in the PoC? Let's find out.

Identifying the Proof of Concept Key Components

Now that we have a target and a project idea, we work together to create the process flow with a very low level of granularity.

What is absolutely required to achieve this goal? How will this be used? What are the main components?

Usually, we have

- an *Event*

- which triggers a *Business Logic*

- which is accessible via *UX*

- which requires some sort of *Authentication*

- and is in connection with a *Database*

- which has a *Monitoring* feature

After you have progressed through about half a dozen of these processes, there usually comes an a-ha moment:

*'I see what you are doing here. Now, have you thought about trying **this**?'*

Business leaders love this moment. This is when the project comes alive in the shape of feedback coming from an outside perspective. Everyone enjoys when others build upon their ideas.

We joined the ballgame now, and things will never be the same.

This is especially true for Components that run non-deterministic calculations or work with large amounts of data. Stochastic phenomena are often impossible to predict or create a plan for in advance. Sometimes, what you are looking at is not even stochastic. You just haven't figured out the deterministic mechanism behind it yet.

You must perform checks that measure how your components can work together reliably. Usually, you know by experience which components are riskier and move towards testing those.

When you reach a decision point, you always move towards the one that is easier to undo. Because in the end, the PoC will be checked against the business requirement, and if it isn't working out well, you roll that step back and try something else.

When is your Proof of Concept complete?

There are Risks you know and prepare for, and there are ones that emerge.

If you checked all the ones you prepared for **and** handled the emerging ones well, your Concept is Proven.

The quality of your PoC hinges on how you assess the risks; sadly, there is no way to ensure you get it 'right.' Only experience helps. The PoC-s we build today handle way more Risks by default than the ones we made three years ago.

That also means that we can't offer you a workaround or shortcut. You need a skilled technological partner with decent mileage. We like to think that we qualify as one and that our case studies prove this. We've been working with US startups remotely and Hungarian manufacturing companies on-site. We are a tightly knit, in-house team of curious developers focusing on the latest technologies.

We think that building a PoC is, in essence, an engineering challenge that suits our skills and mindset well. So far, it seems we can develop solutions that drive real value and generate results, maybe somewhat faster than most of our global competitors.

The best and worst thing about startups is that they usually come from a singular idea. But we enjoy and thrive in testing out the feasibility of these ideas, not only in a technical sense but from a business perspective as well. We are builders who have already helped many digital entrepreneurs realize their dreams.

AI is the frontier in technology right now, and we are happy to play an active part in this amazing journey. We can't wait for what's next in store for us.

How about you?

***

BENEDEK, Gergő
gergo@theblackunicorn.eu
+36206175791
theblackunicorn.eu/