

Winning Software Prototypes



How to Capture the Hearts and Minds of Your Key Stakeholders



It is possible to create a stunning prototype that will win hearts and minds –and, most importantly, budget and approval– in a short timeframe.

Introduction

As an executive or product manager, you are under pressure to explore new sources of revenue. If you're lucky, you have some options. You can create a new product, refresh an existing one, or expand your reach through new technologies and platforms.

First things first, you have to gain internal support, consensus, and budget. You have to excite key customers. And you need to create buzz to generate interest in the new offering.

One of the best ways to do this is to create a prototype: a functional and visually impressive representation of your vision that inspires and builds excitement. It is more than just a diagram or PowerPoint presentation. Your software prototype should look like a product that is ready to ship.

You will need to create a prototype quickly, but be prepared to face organizational and personnel constraints at every turn. The design team won't have enough time. The engineering team will remind you they are already overbooked. And if you don't plan carefully, instead of moving things forward, you run the risk of setting things back. It is a balancing act.

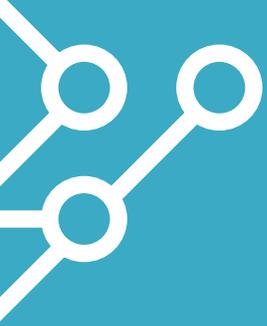
Despite these roadblocks, it is possible to create a stunning prototype that will win hearts and minds – and, most importantly, budget and approval – in a short timeframe. Here is how.

Get Clear On Strategy

At its core, strategy is about prioritization. You need to choose and spend time on the key priorities that are going to make the end product AND the prototype a success – nothing else.

Most software prototype projects don't start out this

way. Managers often assemble a long wish list outlining their need for lots of features, an impressive visual design, fast development, re-usable code, etc. But to ensure a successful prototype, you need to:

- 
1. Develop a very specific strategy
 2. Make sure everyone on the team—developers, designers, managers—understands the strategy
 3. Ensure that little or no time is spent on anything but the key priorities to bring the prototype to life.

It's critical to make sure the strategy for the product offering is clearly articulated and understood by all stakeholders. Usually, the prototype strategy, which also needs to be clearly defined, is targeting a different audience than the product strategy. For example, a product may be intended for consumers, but the prototype might be created for executives in your organization with an end-goal to secure budget, or to key clients to gather early-stage reactions and feedback.

So how do you actually determine the prototype strategy and key priorities? Start by tying things back to your business goals. Are you trying to make a business case to integrate WebRTC technology into your products to result in more seamless, cross-platform customer service? Are you trying to showcase the Modern UI of your new Windows 10 migration to get customer feedback?

Tie the features and use cases in the prototype back to business goals. You should be able to summarize it in one sentence or one slide. We don't mean this as a general philosophical idea. We mean it literally. Nail this one slide and you empower everyone on the team to make the right decisions independently, and quickly.

Tie the features and use cases in the prototype back to business goals. You should be able to summarize it in one sentence or one slide.

Weave The Story

Identify key use cases that can best demonstrate the business strategy and the differentiated product offering. Remember to consider the important factors to the audience, how this affects the story you tell, and how the prototype fits in.

The story could be fleshed out using a customer experience map, a series of personas, user stories, usage scenarios, or whatever medium your team is comfortable with that fully encompasses the details of the experience. This is your illustration of the vision and overall experience for the product.

From there, select which use cases are necessary to build and which can wait. Often just one use case, implemented exceptionally well, is more powerful than a multitude of features half-done.

Design and development teams need to work very closely together to create this product definition. Ideally, someone with a user experience design background is in a role to clearly document these details to help keep everyone on track and help the product manager gain consensus with other stakeholders.

Select The Right "Type" Of Prototype

You have a few options when developing a software prototype. Will it be a paper prototype or functional code? A throwaway concept or a reusable design and code base?

Depending on the strategy and the story, you will probably want to choose one of the following prototype "types".

TIP: To succeed here you need to really focus. Limit the work to one persona. That persona should be the most likely/intended audience of the product and should be specific.

A. Rapid Prototyping

PURPOSE: To demonstrate the feasibility of product idea. This is typically a throwaway prototype often used to get support for a new idea, to validate a new technology, or to illustrate a new concept at a trade show.

RESULT: Confirmation that the product idea is viable and that it can work.

In this case, you do not need to present a finalized product. Things can be faked and the code in your prototype might just be reference or sample code that could evolve later. Reusability is **not** the goal—the design should just be a rapid “best guess.” Basically, you are simply showing a design concept at this point.

To succeed here you need to really **focus**. Limit the work to one persona. That persona should be the most likely/intended audience of the product and should be specific, i.e. a physician, a data scientist, a CFO, etc.

Try to also limit your prototype to one usage scenario—the key scenario that highlights your end-to-end value proposition. In practice, you will probably include several scenarios but the point is to keep them **limited** and **clearly defined** for the entire team.

B. User Research Prototype

PURPOSE: To validate the design concept with users so it can be iterated on quickly.

RESULT: An optimized design concept that is approved by all stakeholders.

A user research prototype can vary in fidelity from a paper prototype to a Flash-based prototype. Often, you will do more than one round of prototyping and user testing. This will allow you to update the prototype before moving on to actual product development.

C. Production-Level “Prototype”

Our clients often ask us to “build a rapid prototype”, but not as a throwaway. They want to reuse the prototype design and code as the foundation of their new product. In fact, what they are really asking for is not a prototype at all, but the “first vertical slice” of the project.

These types of projects need to be treated with a lot more attention—especially when it comes to design, architecture, reuse, and the inherent trade-offs between building something quickly but that is also appropriate for reuse in production.

PURPOSE: Validate the software design and serve as a communication tool between the project leaders and team.

RESULT: Working, production quality code that contains all layers of the system and provides at least one usage scenario.

A re-usable prototype is the foundation of your app or product. Both the interaction design and technology choices are almost final. It is not, “We are exploring a new direction for our product on Android tablets and getting feedback from customers,” but rather “This is the design of the ERP client that we want to implement as a Native Android tablet application. It will communicate with the new version of our backend system for the Q4 release.”

Choosing the right kind of prototype is important, but it will not mean anything if you can not bring together the people to build it. This is the reason why it is so important to take some time to surround yourself with the **right** people.

Build a Specialized Team

No matter what kind of prototype you choose, you will need a very specialized team to actually bring it to life. Prototyping work, particularly in advance of a trade show or presentation to stakeholders, is fast paced, stressful, and requires specific skills. Not everyone has the stomach for it.

The individuals on the team will make all the difference, and they should be:

PASSIONATE

Build the team with experts passionate about innovation and who deeply believe in what you are trying to accomplish. If the project and learning feels like work, the team ramp-up will be slow and managing will be like pulling teeth.

SPECIALISTS

For technologies that are more established, find people who know them inside and out. Don't start learning Objective-C for the first time if your main technological unknown is a new set of social APIs.

FAST LEARNERS

When you innovate, team members will not know everything before they start, but that does not matter if they learn quickly.

COMMUNICATORS

Product managers, designers and software developers need to be in constant communication.

COMFORTABLE WITH AMBIGUITY AND CHANGE

People who can embrace ambiguity and frequent change of direction in the early going will help bring order to that ambiguity to drive things forward.

RESOURCEFUL AND SELF-MOTIVATED

Team members are either a +10 or a -5. In a prototype context there is rarely an in-between, and -5s will quickly drag your team down.

Stay True To The Design

Don't cut corners on the user experience of your prototype! Prototypes are made to convince buyers, stakeholders or investors, and nothing convinces better than stunning design and overall experience. Even a very technical prototype to demonstrate the use of a new low-level networking protocol will have better success if it is visually appealing and easy to interact with.

But in the rush to produce a working prototype, it can be very tempting to cut corners on elements of the design in the working prototype. A UI control that is not properly aligned will not cut it here. Never commit to poor-looking visuals with the expectation that you will go back and adjust them later. Get it perfect the first time!

When in a crunch timeframe, designers and developers should be communicating to ensure that standard controls provided by the OS or UI framework are used as much as possible, rather than custom UI controls (unless the goal of the project is to use a new control). Custom controls can take up to 10 times longer to implement.

If the development team is behind on implementing a supporting part of the design, consider using a PNG image for the presentation of the prototype. Depending on the strategy and goals you set out at the beginning, a static image that accurately represents the end-product may be a better choice than being late or presenting a demo that may be interactive but simply doesn't look right yet.

Keep a Relentless Focus

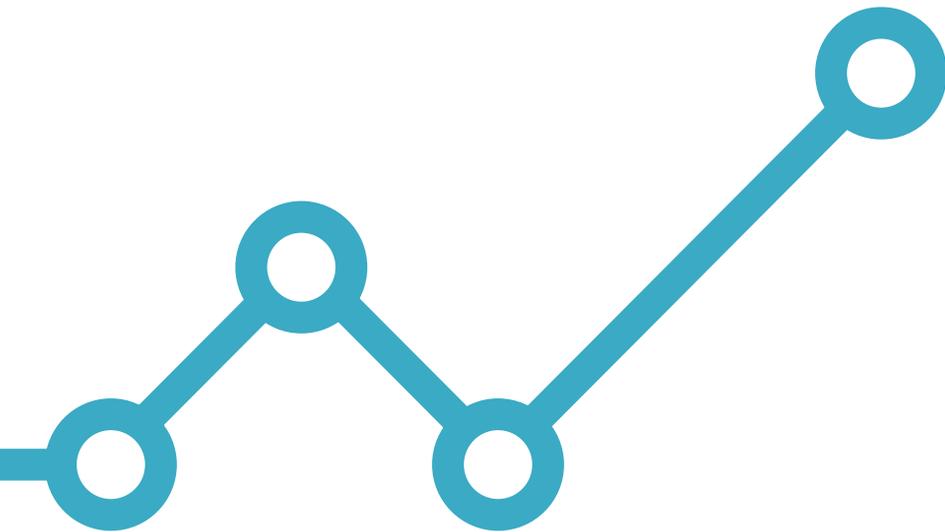
Even if your development team is made up of superstars, they'll need you to keep the big picture in mind and the project on track. The project manager or development leader needs to set the tone to ensure the vision is carried out in implementation. Every team member carrying out the prototype should be comfortable with an environment where people:

- 1. MAKE QUICK DECISIONS.** When faced with a dilemma, make a decision quickly. Help others determine whether a decision is crucial enough to spend time on and only ask first if it really matters.
- 2. DON'T GET LOST IN THE DETAILS.** Your prototype will not be using production code so don't worry about access control, password protection, data integrity, etc.
- 3. KEEP YOURSELF IN CHECK.** If you are blocked on something even for a couple hours, stakeholders need to know so that you can decide together whether it is worth the time or effort.
- 4. COMMIT NEW CODE TO THE COMMON CODE BASE OFTEN.** Ideally, you should be adding new code multiple times per day.
- 5. REACH OUT AND ASK FOR HELP.** It's worth five minutes of a colleague's time if it will save you a couple of hours.
- 6. BREAK THE RULES.** Normal product development rules don't apply to prototypes if you are working with throw-away code. You will not need production-level quality and reliability, so make sure your team members are not developing state machines when a series of "if" statements would do just fine.
- 7. TAKE THE QUICKEST SOLUTION** that gets you closer to the prototype goal.

Your Winning Software Prototype

By developing a strategy, choosing the right style of prototype, assembling a solid team, and breaking the rules when it counts, it really is possible to build an attractive and successful prototype.

At Macadamian, we have helped customers of all sizes create stunning prototypes and gain the internal or external buy-in needed for new projects. If you are thinking about creating a prototype, or are struggling to get one off the ground, contact us!



Winning Software Prototypes

How to Capture the Hearts and Minds of Your Key Stakeholders

by Didier Thizy, VP of Digital Health and Martin Larochelle, Chief Architect

About the Authors



Didier is Macadamian's VP of Digital Health and the leader of its growing healthcare practice. Responsible for a cross-functional unit of design and development consultants, his areas of focus include Electronic Medical Records, consumer health software, usability of complex systems, and modern mobile technologies. Didier is an active member of HIMSS, MGMA and Health 2.0. When Didier is not on the road, you can find him rocking out to 80s music, and on certain rare mornings, sleeping in because his kids decided to cut him some slack. Didier has been a software professional for 14 years, holding a variety of positions in Software R&D and Product Management.

didier@macadamian.com



Martin Larochelle has been with Macadamian since 2005. In his ten years with the company, he has tackled projects both big and small as Chief Architect. An expert in C++ and VOIP, his focus has been on mobile platforms. Martin was instrumental for all things BlackBerry providing technical leadership and project oversight. Martin now leads the Macadamian Innovation Lab, a team focused on developing concepts to solve the needs of small and medium businesses and key verticals such as healthcare. While we're all a little nuts at Macadamian, Martin counts himself as the biggest HeadBlade fan in Canada.

martin@macadamian.com

Thank you.

m a c a d a m i a n

Macadamian is a full-service software design and development firm that provides a complete range of highest quality usability, design, and software engineering services to industry leaders across North America. We care about the user and, as a result, create products that delight you and your customers. Our creativity and collaborative approach spark the ideas that ensure development of the best products. Our advanced, consistent experiences will be rapidly adopted by your desired audience right from the outset, because we think about the way your products are going to be received. The result? Your product will entice customers while standing out, maximizing your opportunity and delivering inspired results.

Visit macadamian.com

Call 1-877-779-6336