The question of whether to **build** software or

buy software has been debated for a long time.

Analysis





Your Responsibility

	Build	Buy
Overall Vision	100%	0%
Project Planning		0%
Lending & Software Expertise		0%
Technology Decisions		0%
Software Maintenance		0%
New Feature Development		50%
Compliance		0%
Security		0%
Hosting		0%
Ownership		Data Only
Access		No Access to Source Code or Hardware

Requirements **Overall Vision** Individual in Control of the Overall Vision for

Build

the Software **Project Planning**

SCRA, TILA, etc.

Lending & Software Expertise Consulting Expertise: APR Calculation, Smooth Payments, Interest Calculation,

Product Owner, Product Manager

Technology

Planning to Ensure the Right Tech is Chosen

New Feature Development Coding the Software

Overall Vision

Hosting Hardware and Service Costs

Security Ensuring Data Safety

Compliance

Meeting Regulatory Requirements

Software Maintenance

Ongoing Updates, Fixes

Time to Launch When can you use the Software

How will the final product look and function?



Without someone who has a

strong overall vision, most software

projects are never completed.



tasks, and an understanding of

Project Planning

Typically software projects are created under the direction of a single individual who has the overall vision for the final product, but specification documents,

Feature planning

what goes into each feature are the responsibility of the Product Owner and Product Manager.



Lending & Software Expertise In-depth, subject-matter knowledge

about lending. They can learn about specifics by

The Product Owner and Product Manager know how to plan projects, but they probably don't know everything

consulting with experts on things like APR and interest calculations, overall software structure, compliance

While developers usually understand code, databases, servers, and similar tech, they will probably still need significant input when it comes to overall architecture. If your software isn't planned well from the beginning, it will

concerns, etc.

» SQL or NoSQL? » What are the best development languages? » What frameworks will you use?

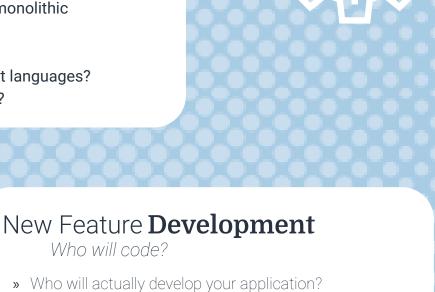
be hard to make the right technologies. After you start

building, any major change becomes very costly.

» Will you have an API? Will you be built on it?

» Will you use microservices, a monolithic

architecture, or a hybrid?





» Will you hire and manage developers in house?

» How will you ensure developers are prioritizing tasks

» What tools will you use to keep developers in sync? » Who will manage their access to code and hardware?

correctly and not creating things that don't fit the vision

Who will code?

» Will you outsource?

for the software?

services in a cost-effective way? Compliance

Are there rules you have to follow?

Once you've developed your application, you may need to add features, update software, scale your business, deal with server issues,



You may have compliance requirements from your funding source. You probably need to comply with state and federal lending laws. You will definitely need to worry about PCI compliance if you take

Where will your software live?

Will you host your software onsite or on the cloud?

Who will make sure you are utilizing hardware and

Time to Launch The best time to plant a tree is 20 years ago, the second best time is today In most cases, it takes around 3

months to launch a simple website. For a complex website with a back end, it will take more like 6 months to a year.

For a full software application, it usually takes between 3 and 10 years.

You are a lender, not a software developer. Development is much more than finding someone who can code. Focus on your primary business and let LoanPro provide software solutions. If you choose to BUY software, you get some additional benefits:

Default Loan Processes (e.g. SCRA, fraud, deferred payment) Built-In Consulting

Comprehensive Help Documentation

- Onboarding
- An Experienced Team
- Conclusion

payments.

Software Maintenance

Keeping your software up and going

fix bugs, or any number of things.

