

# Build vs Buy Analysis

The question of whether to **build** software or **buy** software has been debated for a long time.

Choose Wisley

“Nothing ever gets built on schedule or within budget.”

- Robert A. Heinlein

## 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

## Build Requirements

### Overall Vision

Individual in Control of the Overall Vision for the Software

### Project Planning

Product Owner, Product Manager

### Lending & Software Expertise

Consulting Expertise: APR Calculation, Smooth Payments, Interest Calculation, SCRA, TILA, etc.

### Technology

Planning to Ensure the Right Tech is Chosen

### New Feature Development

Coding the Software

### 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

## Overall Vision

How will the final product look and function?

Without someone who has a **strong overall vision**, most software projects are never completed.

### Project Planning

Feature 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, tasks, and an understanding of what goes into each feature are the responsibility of the Product Owner and Product Manager.

### Lending & Software Expertise

In-depth, subject-matter knowledge

The Product Owner and Product Manager know how to plan projects, but they probably don't know everything about lending. They can learn about specifics by consulting with experts on things like APR and interest calculations, overall software structure, compliance concerns, etc.

### Technology Decision

How will you ensure your platform is fast, efficient, and scalable?

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

### New Feature Development

Who will code?

- » Who will actually develop your application?
- » Will you hire and manage developers in house?
- » Will you outsource?
- » How will you ensure developers are prioritizing tasks correctly and not creating things that don't fit the vision for the software?
- » What tools will you use to keep developers in sync?
- » Who will manage their access to code and hardware?

### Security

How will you ensure your software is secure?

There are security companies you can use for things like penetration testing. It is usually best to have intrusion detection, file integrity management, and ongoing vulnerability scans. If someone gains access to your customer data, what is your plan?

### Where will your software live?

- » Will you host your software onsite or on the cloud?
- » Who will make sure you are utilizing hardware and services in a cost-effective way?

## Compliance

Are there rules you have to follow?

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 payments.

### Software Maintenance

Keeping your software up and going

Once you've developed your application, you may need to add features, update software, scale your business, deal with server issues, fix bugs, or any number of things.

### 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.

## Conclusion

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:

- ✓ Comprehensive Help Documentation
- ✓ Default Loan Processes (e.g. SCRA, fraud, deferred payment)
- ✓ Built-In Consulting
- ✓ An Experienced Team
- ✓ Onboarding

Power Your Next Big Idea

Claim Your Free Trial

LoanPro