Why Rails Makes Startups

The Lean Startup

#leanstartup

Eric Ries (@ericries)

http://StartupLessonsLearned.com
http://lean.st
watch what happens behind the scenes & see every experiment we’ve run at: http://lean.st
Lean Startup Principles

Entrepreneurs are everywhere

Entrepreneurship is management

Validated Learning

Build – Measure - Learn

Innovation Accounting
Lean Startup Principles

Entrepreneurs are everywhere
What is a startup?

• A startup is a *human institution* designed to *deliver a new product or service* under conditions of *extreme uncertainty*.

• Nothing to do with size of company, sector of the economy, or industry.
What is a startup?

STARTUP = EXPERIMENT
STOP WASTING PEOPLE’S TIME
Most Startups Fail
Most Startups Fail
Most Startups Fail
Who to Blame

- Father of *scientific management*
- Study work to find the best way
- Management by exception
- Standardize work into tasks
- Compensate workers based on performance

“In the past, the man was first. In the future, the system will be first.” (1911)

Frederick Winslow Taylor
(1856 – 1915)
Lean Startup Principles

Entrepreneurs are everywhere

Entrepreneurship is management
Entrepreneurship is management

• Our goal is to create an institution, not just a product
• Traditional management practices fail
  - “general management” as taught to MBAs
• Need practices and principles geared to the startup context of extreme uncertainty
• Not just for “two guys in a garage”
The Pivot
“I’m not leaving you. I’m pivoting to another man.”
The Pivot

• What do successful startups have in common?
  - They started out as digital cash for PDAs, but evolved into online payments for eBay.
  - They started building BASIC interpreters, but evolved into the world's largest operating systems monopoly.
  - They were shocked to discover their online games company was actually a photo-sharing site.

• Pivot: change directions but stay grounded in what we’ve learned.
Speed Wins

If we can reduce the time between pivots

We can increase our odds of success

Before we run out of money
Lean Startup Principles

Entrepreneurs are everywhere

Entrepreneurship is management

Validated Learning
Achieving Failure

• If we’re building something nobody wants, what does it matter if we accomplish it:
  
  On time?
  On budget?
  \textbf{With high quality}?
  With beautiful design?

• Achieving Failure = successfully executing a bad plan
Agile Product Development

Unit of Progress: A line of Working Code

“Product Owner” or in-house customer

Problem: known
Solution: unknown

Kent Beck
(still alive)
Lean Startup

Unit of Progress: Validated Learning

Customer Development

Problem: unknown
Solution: unknown

Hypotheses, Experiments, Insights
Data, Feedback, Insights

Customer Discovery → Customer Validation → Customer Creation → Scale Company

Agile Development

User Stories
Architectural Spike
Requirements → Release Planning → Iteration → Acceptance Tests → Small Releases

Test Scenarios
Bugs
Latest Version
Next Iteration

Extreme Programming

Steve Blank (still alive)
Lean Startup Principles

Entrepreneurs are everywhere

Entrepreneurship is management

Validated Learning

Build – Measure - Learn
Minimize **TOTAL** time through the loop
Lean Startup Principles

Entrepreneurs are everywhere

Entrepreneurship is management

Validated Learning

Build – Measure - Learn

Innovation Accounting
Innovation Accounting
The Three Learning Milestones

1. Establish the baseline
   - Build a minimum viable product
   - Measure how customers behave right now

2. Tune the engine
   - Experiment to see if we can improve metrics from the baseline towards the ideal

3. Pivot or persevere
   - When experiments reach diminishing returns, it’s time to pivot.
Questions

How do we know when to pivot?
Vision or Strategy or Product?
What should we measure?
How do products grow?
Are we creating value?
What’s in the MVP?
Can we go faster?
Thanks!

• Buy the book @ http://lean.st

• Startup Lessons Learned Blog
  - http://StartupLessonsLearned.com

• Getting in touch (#leanstartup)
  - http://twitter.com/ericries
  - eric@theleanstartup.com

• Additional resources
  - NEW: http://theleanstartup.com
  - SLLCONF 2011: http://sllconf.com May 23 in SF
Myth #1

Myth

*Lean means cheap.* Lean startups try to spend as little money as possible.

Truth

The Lean Startup method is not about cost, *it is about speed.*
Myth #2

Myth
The Lean Startup is only for Web 2.0/internet/consumer software companies.

Truth
The Lean Startup applies to all companies that face uncertainty about what customers will want.
Myth #3

**Myth**
Lean Startups are small *bootstrapped* startups.

**Truth**
Lean Startups are ambitious and are able to deploy *large amounts of capital.*
Myth #4

Myth
Lean Startups replace vision with data or customer feedback.

Truth
Lean Startups are driven by a compelling vision, and are rigorous about testing each element of this vision.
Lean Startup Principles

Entrepreneurs are everywhere

Entrepreneurship is management

Validated Learning

Build – Measure - Learn

Innovation Accounting
Minimum Viable Product

• Visionary customers can “fill in the gaps” on missing features, if the product solves a real problem

• Allows us to achieve a big vision in small increments without going in circles

• Requires a commitment to iteration

• MVP is only for BIG VISION products; unnecessary for minimal products.
Continuous Deployment

Learn Faster
- Customer Development
- Five Whys

Measure Faster
- Split Testing
- Actionable Metrics
- Net Promoter Score
- SEM

Build Faster
- Continuous Deployment
- Small Batches
- Minimum Viable Product
- Refactoring
Continuous Deployment Principles

Have every problem once
Stop the line when anything fails
Fast response over prevention
Continuous Deployment

• Deploy new software quickly
  - At IMVU time from check-in to production = 20 minutes

• Tell a good change from a bad change (quickly)

• Revert a bad change quickly
  - And “shut down the line”

• Work in small batches
  - At IMVU, a large batch = 3 days worth of work

• Break large projects down into small batches
Cluster Immune System
What it looks like to ship one piece of code to production:

- **Run tests locally** (SimpleTest, Selenium)
  - Everyone has a complete sandbox

- **Continuous Integration Server** (BuildBot)
  - All tests must pass or “shut down the line”
  - Automatic feedback if the team is going too fast

- **Incremental deploy**
  - Monitor cluster and business metrics in real-time
  - Reject changes that move metrics out-of-bounds

- **Alerting & Predictive monitoring** (Nagios)
  - Monitor all metrics that stakeholders care about
  - If any metric goes out-of-bounds, wake somebody up
  - Use historical trends to predict acceptable bounds

- **When customers see a failure**
  - Fix the problem for customers
  - Improve your defenses at each level
Minimum Viable Product

**Learn Faster**
- Customer Development
- Five Whys

**Build Faster**
- Continuous Deployment
- Small Batches
- Minimum Viable Product
- Refactoring

**IDEAS**

**DATA**

**CODE**

**MEASURE**
- Split Testing
- Actionable Metrics
- Net Promoter Score
- SEM

**LEARN**
Why do we build products?

• Delight customers
• Get lots of them signed up
• Make a lot of money
• Realize a big vision; change the world
• Learn to predict the future
Possible Approaches

• “Maximize chances of success”
  - build a great product with enough features that increase the odds that customers will want it
  - Problem: no feedback until the end, might be too late to adjust

• “Release early, release often”
  - Get as much feedback as possible, as soon as possible
  - Problem: run around in circles, chasing what customers think they want
Minimum Viable Product

• The minimum set of features needed to learn from early evangelists – visionary early adopters
  - Avoid building products that nobody wants
  - Maximize the learning per dollar spent

• Probably much more minimum than you think!
Minimum Viable Product

- Visionary customers can “fill in the gaps” on missing features, if the product solves a real problem
- Allows us to achieve a big vision in small increments without going in circles
- Requires a commitment to iteration
- MVP is only for BIG VISION products; unnecessary for minimal products.
Techniques

• Smoke testing with landing pages, AdWords
• SEM on five dollars a day
• In-product split testing
• Paper prototypes
• Customer discovery/validation
• Removing features (“cut and paste”)

Fears

• False negative: “customers would have liked the full product, but the MVP sucks, so we abandoned the vision”

• Visionary complex: “but customers don’t know what they want!”

• Too busy to learn: “it would be faster to just build it right, all this measuring distracts from delighting customers”
Five Whys

Learn Faster
- Five Whys Root
- Cause Analysis

IDEAS

MEASURE
- Measure Faster
- Rapid Split Tests

BUILD
- Code Faster
- Continuous Deployment

CODE

DATA

LEARN
Five Whys Root Cause Analysis

• A technique for continuous improvement of company process.

• Ask “why” five times when something unexpected happens.

• Make proportional investments in prevention at all five levels of the hierarchy.

• Behind every supposed technical problem is usually a human problem. Fix the cause, not just the symptom.
Rapid Split Tests

Learn Faster
Five Whys Root
Cause Analysis

LEARN

IDEAS

MEASURE

Rapid Split Tests

Build Faster
Continuous Deployment

Code Faster

DATA

BUILD

CODE
Split-testing all the time

• A/B testing is key to validating your hypotheses

• Has to be simple enough for everyone to use and understand it

• Make creating a split-test no more than one line of code:

```plaintext
if( setup_experiment(...) == "control" ) {
    // do it the old way
} else {
    // do it the new way
}
```
The AAA’s of Metrics

- Actionable
- Accessible
- Auditable
Measure the Macro

- Always look at cohort-based metrics over time
- Split-test the small, measure the large

<table>
<thead>
<tr>
<th></th>
<th>Control Group (A)</th>
<th>Experiment (B)</th>
</tr>
</thead>
<tbody>
<tr>
<td># Registered</td>
<td>1025</td>
<td>1099</td>
</tr>
<tr>
<td>Downloads</td>
<td>755 (73%)</td>
<td>733 (67%)</td>
</tr>
<tr>
<td>Active days 0-1</td>
<td>600 (58%)</td>
<td>650 (59%)</td>
</tr>
<tr>
<td>Active days 1-3</td>
<td>500 (48%)</td>
<td>545 (49%)</td>
</tr>
<tr>
<td>Active days 3-10</td>
<td>300 (29%)</td>
<td>330 (30%)</td>
</tr>
<tr>
<td>Active days 10-30</td>
<td>250 (24%)</td>
<td>290 (26%)</td>
</tr>
<tr>
<td>Total Revenue</td>
<td>$3210.50</td>
<td>$3450.10</td>
</tr>
<tr>
<td>RPU</td>
<td>$3.13</td>
<td>$3.14</td>
</tr>
</tbody>
</table>
Lean Startup Principles

Entrepreneurs are everywhere

Entrepreneurship is management

Validated Learning

Innovation Accounting
Minimum Viable Product

• The minimum set of features needed to learn from early evangelists – visionary early adopters
  - Avoid building products that nobody wants
  - Maximize the learning per dollar spent

• Probably much more minimum than you think!
Minimum Viable Product

• Visionary customers can “fill in the gaps” on missing features, if the product solves a real problem

• Allows us to achieve a big vision in small increments without going in circles

• Requires a commitment to iteration

• MVP is only for BIG VISION products; unnecessary for minimal products.
Split-testing all the time

- A/B testing is key to validating your hypotheses
- Has to be simple enough for everyone to use and understand it
- Make creating a split-test no more than one line of code:

```python
if( setup_experiment(...) == "control" ) {
    // do it the old way
} else {
    // do it the new way
}
```
The AAA’s of Metrics

- Actionable
- Accessible
- Auditable
Measure the Macro

- Always look at cohort-based metrics over time
- Split-test the small, measure the large

<table>
<thead>
<tr>
<th></th>
<th>Control Group (A)</th>
<th>Experiment (B)</th>
</tr>
</thead>
<tbody>
<tr>
<td># Registered</td>
<td>1025</td>
<td>1099</td>
</tr>
<tr>
<td>Downloads</td>
<td>755 (73%)</td>
<td>733 (67%)</td>
</tr>
<tr>
<td>Active days 0-1</td>
<td>600 (58%)</td>
<td>650 (59%)</td>
</tr>
<tr>
<td>Active days 1-3</td>
<td>500 (48%)</td>
<td>545 (49%)</td>
</tr>
<tr>
<td>Active days 3-10</td>
<td>300 (29%)</td>
<td>330 (30%)</td>
</tr>
<tr>
<td>Active days 10-30</td>
<td>250 (24%)</td>
<td>290 (26%)</td>
</tr>
<tr>
<td>Total Revenue</td>
<td>$3210.50</td>
<td>$3450.10</td>
</tr>
<tr>
<td>RPU</td>
<td>$3.13</td>
<td>$3.14</td>
</tr>
</tbody>
</table>
“I’m not leaving you. I’m pivoting to another man.”