

COMP1531

 Other

10.3 - (Bonus) Building an MVP

Disclaimer

This lecture highlights some key principles in trying to rapidly develop and execute on software solutions you see in relation to a given problem. **Some aspects of these approaches may be incompatible with more fundamental software engineering principles discussed in the course** - don't get the two mixed up

This lecture would not ordinarily fall into this course - It's simply a bonus lecture for a lighter end of term.

What is an MVP?

An MVP is a **minimum viable product** that can demonstrate the fundamental value of a proposed solution without investing resources to build out an entire system.

The sole purpose of an MVP is to validate assumptions, not to build good technology. Good technology can come later.

It's like the smallest "demo" of something that still makes clear what the bigger idea is.

Building an MVP

1. **Ideation:** Understand the buss scope of your problem-solution
2. **Planning:** Scaffold some ideas down with design/modelling tools
3. **Building:** Use technology to bring it to life
4. **Tracking:** Have empirical understanding of how people use it

These are all part of a cycle, too. Sometimes learnings in planning or building send you right back to ideation etc.

Product-Market Fit

Every MVP is a demonstration that seeks a product-market fit

- Product-Market Fit (PMF) is the degree to which a product satisfies a strong market demand.
- Young companies thrive to reach PMF as soon as possible, as it's usually a key step in early growth
- One of the earliest steps of exploring PMF in a lean way, and verifying the business viability, is to build an **MVP**

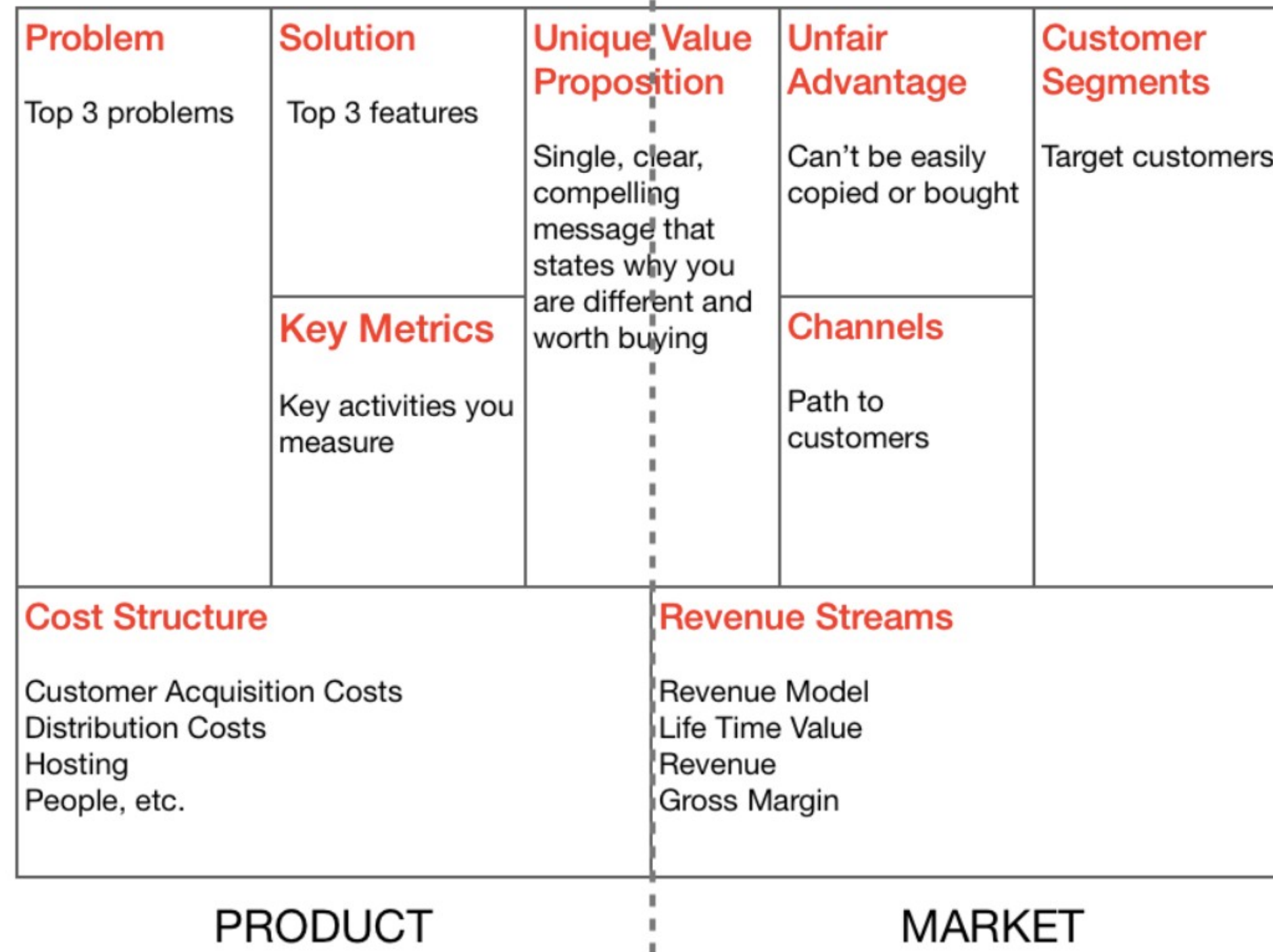
Ideation

- One of the easiest ways to try and capture ideas you have is to structure them into a lean canvas.
- A "Lean Canvas" is a common tool used to try and articulate the core principles of a business
- These canvases can then be used to inform what to design, build, or track.

Lean Canvas

- A "Lean Canvas" is a common tool used to try and articulate the core principles of a business
- These canvases can then be used to inform what to build.
- If you ever want to start a business - start here

Lean Canvas



Lean Canvas is adapted from The Business Model Canvas (<http://www.businessmodelgeneration.com>) and is licensed under the Creative Commons Attribution-Share Alike 3.0 Un-ported License.

Planning

Once you confidence that you've ideated correctly, it's time to bring your solution to life. Ways to do this include:

- Building conceptual models that help describe software in terms of structure or behaviour
- Creating wireframes in tools like **Figma** or **Zeplin**

This stage allows you to identify challenges with your approach, or alternatively to talk to potential consumers with something tangible to learn what works and what doesn't prior to building

Building

If you are confident you know what needs to build - it's time to build it. Get your checklist together:

- Somewhere to host your code (Github, Bitbucket, Gitlab)
- Somewhere to deploy your code (AlwaysData, Heroku, AWS, Azure)
- Programming languages for your systems, which typically include:
 - A front-end layer (e.g. command line interface, javascript frontend)
 - A back-end layer (e.g. python, nodejs, ruby)
 - A data layer (e.g. firebase, google sheets, airtable, sql database)

What technology should you pick? **Whatever you're most comfortable with.**

Tracking

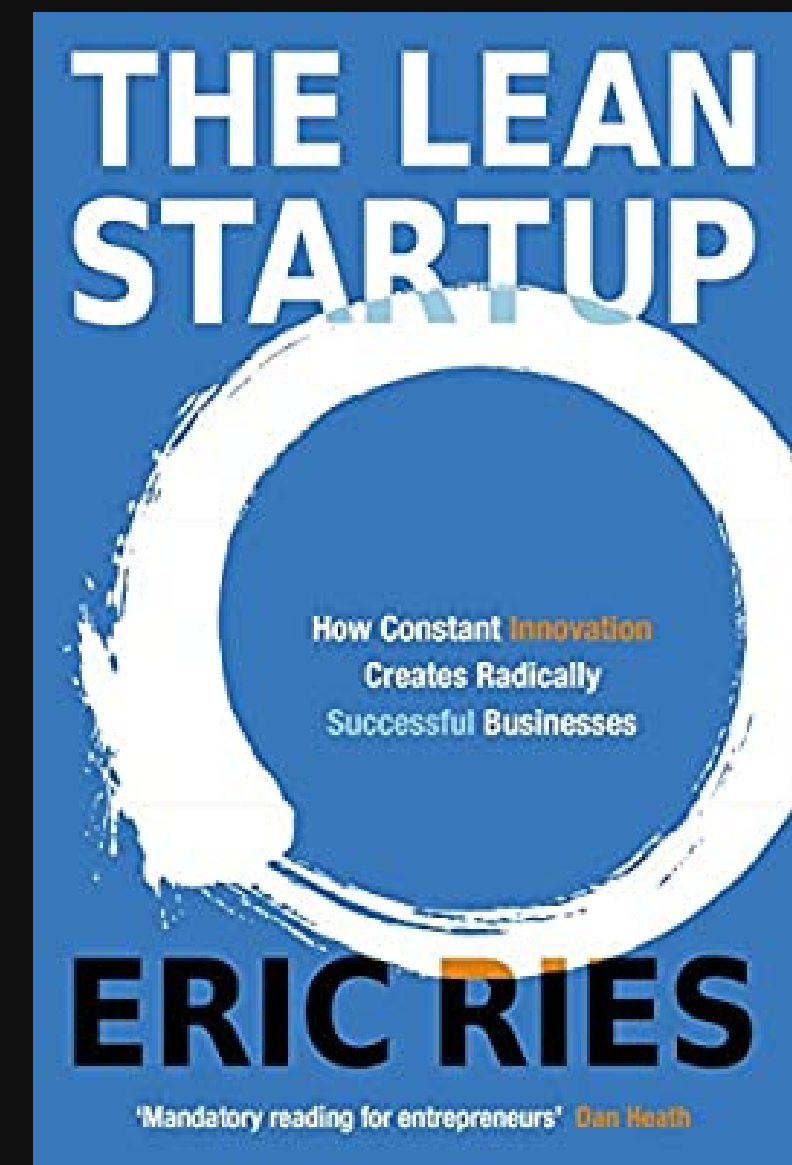
After you build something you will learn a lot just from hearing about people using it. However, this won't tell the full story.

Consider using data collection tools like Google Tag Manager or Segment that allow you to collect aggregated identity or event information about users of the solution, and then pass to a number of third party providers that typically:

- Help you track key events by users in your solution
- Help you track the nature of user types
- Help you understand hot spots

Being Lean

- Eric Ries produced a very popular book known as the "Lean Startup". The book focuses on methods to shorten the product development life cycle and determine if a business is viable or not more rapidly.
- "Build-measure-learn" principles



Seeking Support

- UNSW has a rich array of entrepreneurial programs and support for UNSW students and alumni looking to build out ideas into business.
- There is a massive under-supply of competent software engineering students in this industry.
- A good place to start if you feel unsure is to apply to be part of the Peter Farrell Cup.

Feedback

