

ITI8740/ITX8522: Software Development Team Project

# 06. Minimal Viable Product

Martin Verrev

[martin.verrev@taltech.ee](mailto:martin.verrev@taltech.ee)

# Minimum Viable Product

2001. Frank Robinson. CEO of SyncDev:

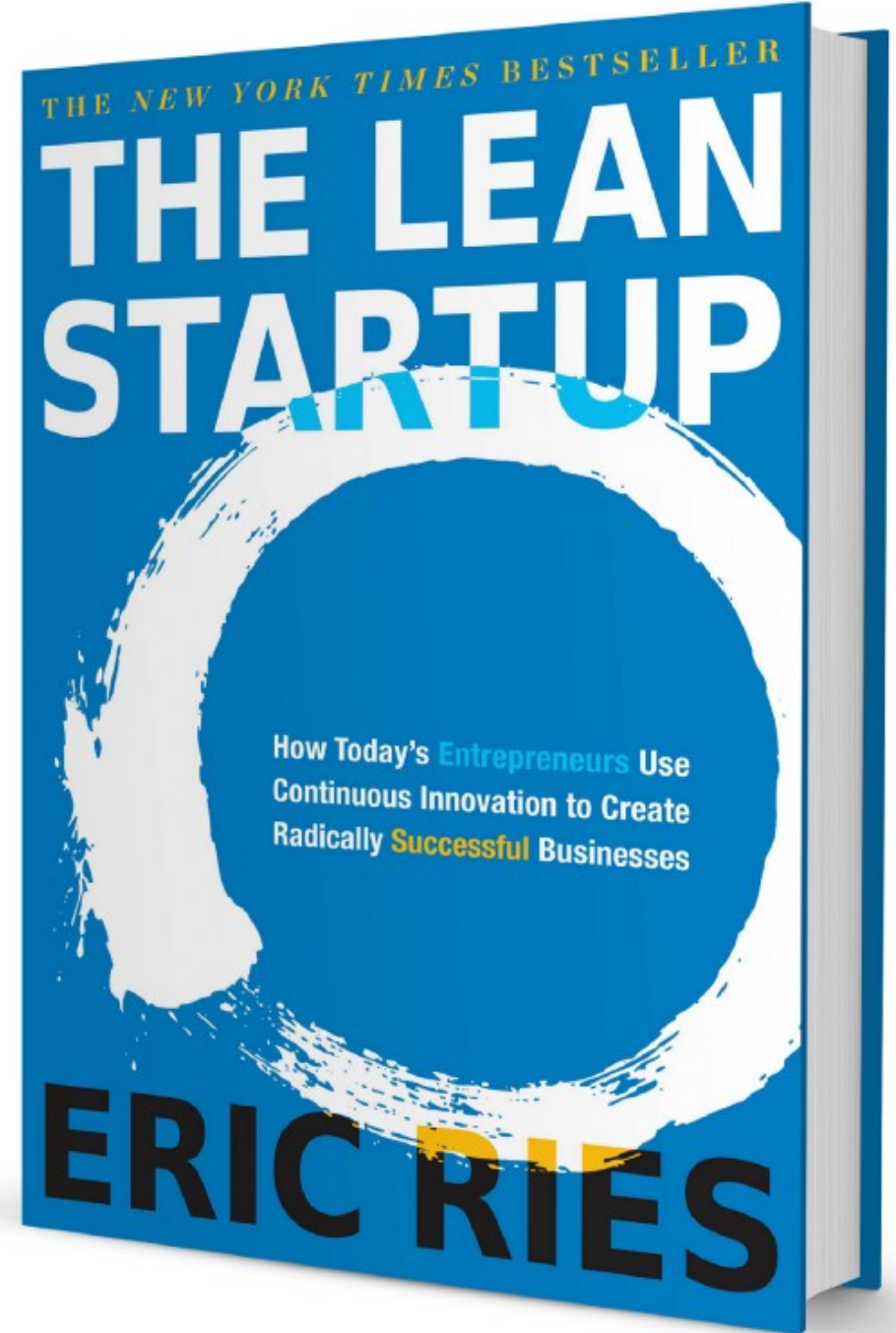
**Unique product that maximizes return on risk for the customer.**

or

**A simplest version of a deployable product**

Minimum Viable Product is that version of a new product that allows a team to collect the maximum amount of validated learning about customers with the least effort.

Eric Ries. Lean Startup



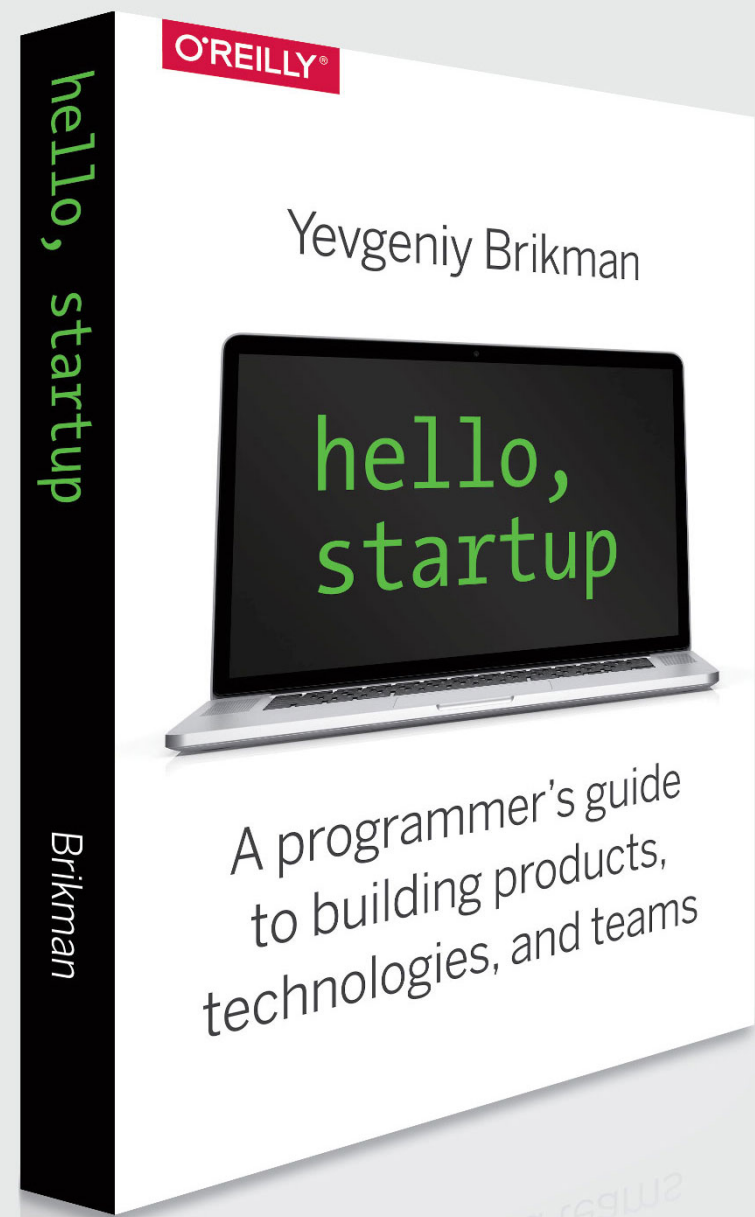
A Minimum Viable Product is the smallest thing you can build that delivers customer value (and as a bonus captures some of that value back).

Ash Maurya. Running Lean



**An MVP is a process that you repeat over and over again: Identify your riskiest assumption, find the smallest possible experiment to test that assumption, and use the results of the experiment to course correct.**

Jim Brikman. Hello Startup.



# MVP is ..

- A *tactic* for cutting back wasted engineering hours
- A *strategy* to get the product into earlyvangelists hands as soon as possible
- A *tool* for generating maximum customer learning in the shortest possible time.

*Steve Blank. The father of customer development*



**MVP** is not just a product with half of the features chopped out, or a way to get the product out the door a little earlier. In fact, the MVP doesn't have to be a product at all. And it's not something you build only once, and then consider the job done.

"What is the smallest or least complicated problem that the customer will pay us to solve"

# Examples: Spotify & Minecraft

<https://blog.crisp.se/2016/01/25/henrikkniberg/making-sense-of-mvp>



# Good Way to build MVP

Not like this...



Instead like this!



# ... and even better one

HOW **NOT TO BUILD** A MINIMUM VIABLE PRODUCT



1



2



3



4

ALSO HOW **NOT TO BUILD** A MINIMUM VIABLE PRODUCT



1



2



3



4

HOW **TO BUILD** A MINIMUM VIABLE PRODUCT



1



2

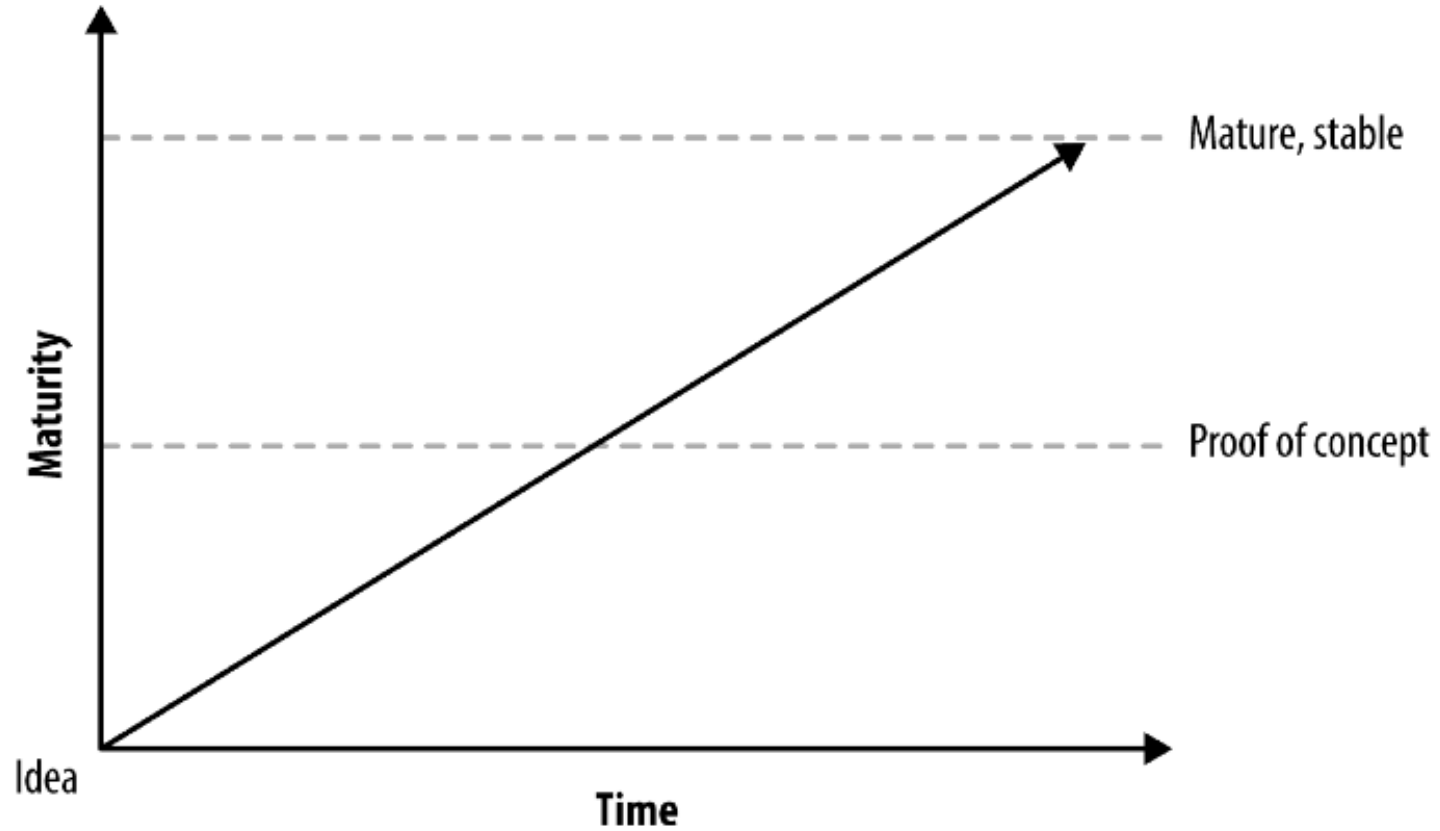


3

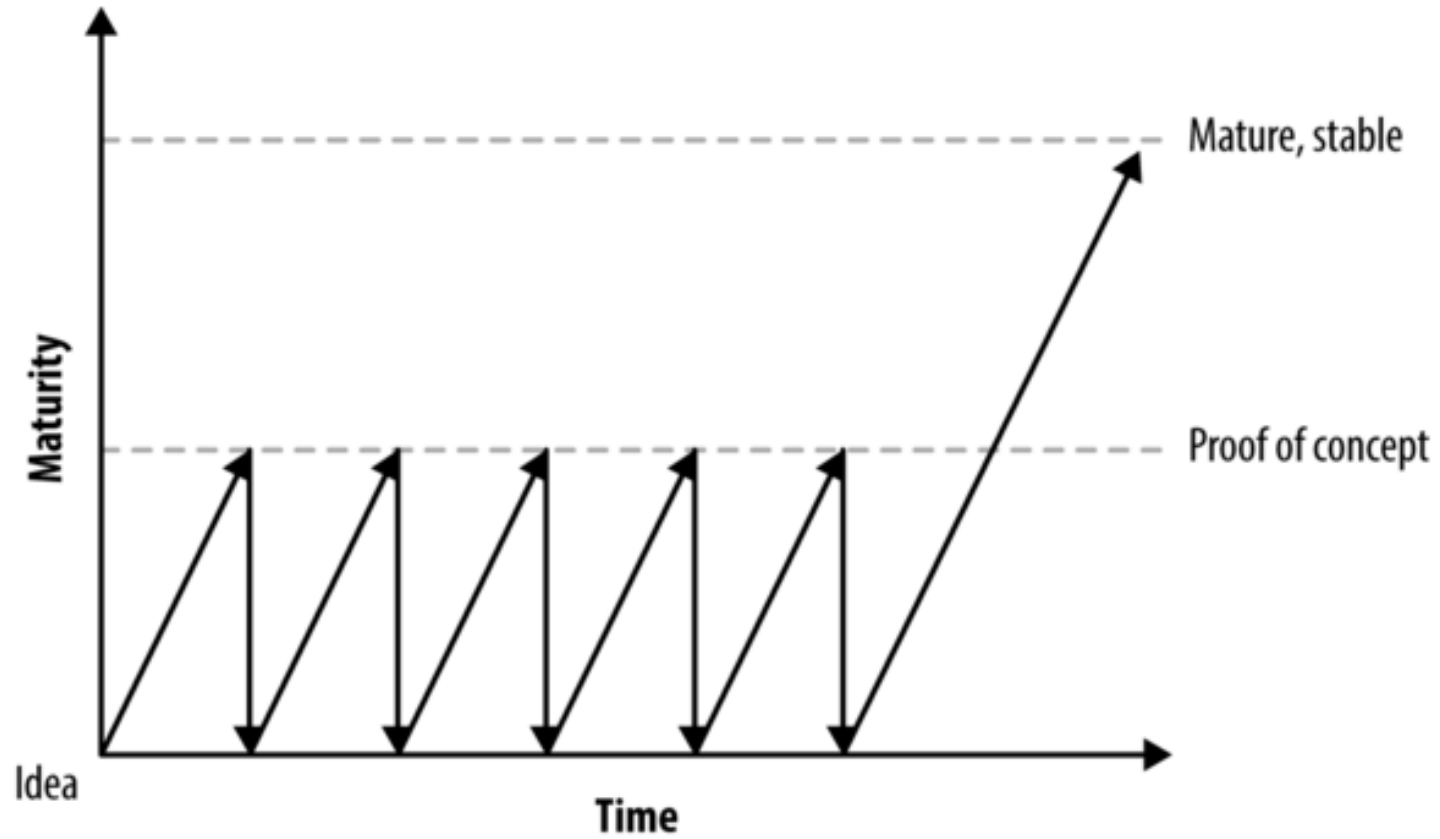


4

# Regular Product Development



# MVP Development



**MVP is a prototype that focuses on the most important part of a product or service.**

Typically offered to a group of potential customers, the most important aspect is viability in order to test the core of the value proposition; to learn from it; and to be able to replicate the idea

# Define the Features

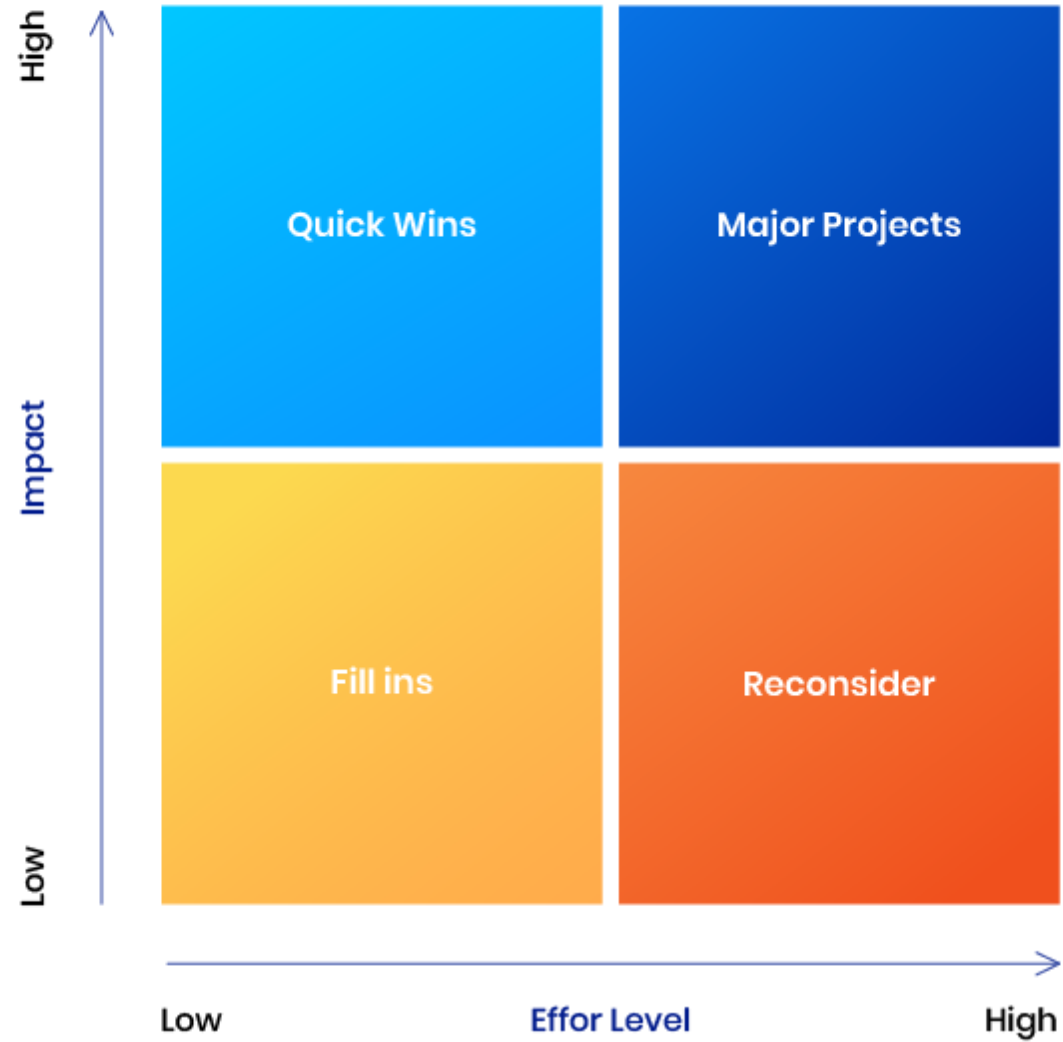
- Define your target audience.
- Find out their biggest issues.
- Decide how your product will solve these issues.
- Learn how customers solve these issues right now.
- Research who your direct competitors are.
- Research who your indirect competitors are.
- Define killer features.

# Scoping the Features

- *Business value*: Can we measure the cost of a feature?
- \_Relation between the feature cost and its [https://sloboda-studio.com/blog/how-to-decide-on-mvp-features/complexity:\\_](https://sloboda-studio.com/blog/how-to-decide-on-mvp-features/complexity:_) How complex is a new feature? Can we use a simpler one?
- *Timing*: How much time do we need to create a new feature?

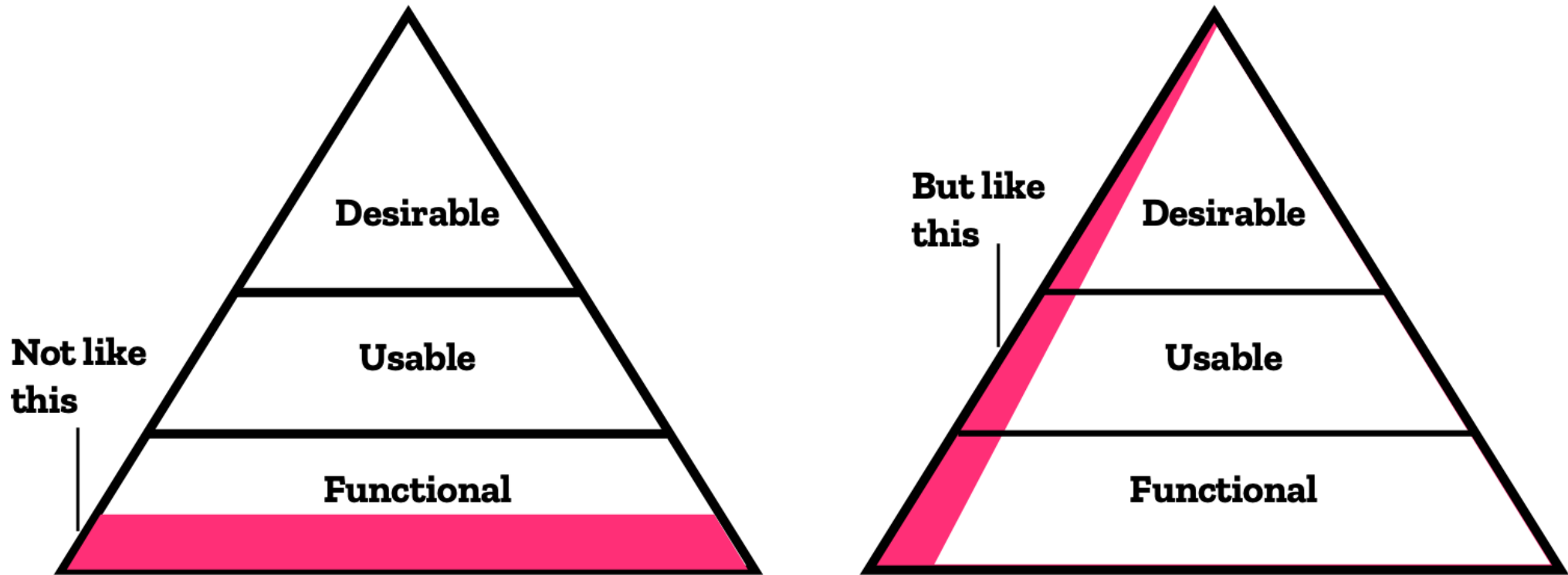
See: <https://sloboda-studio.com/blog/how-to-decide-on-mvp-features/>

## Effort and impact prioritization



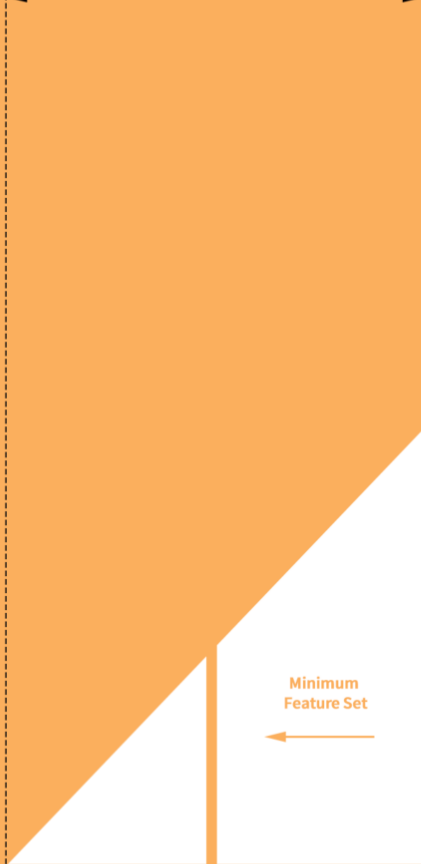


# Minimum viable product



## Early Market

People who want the newest things

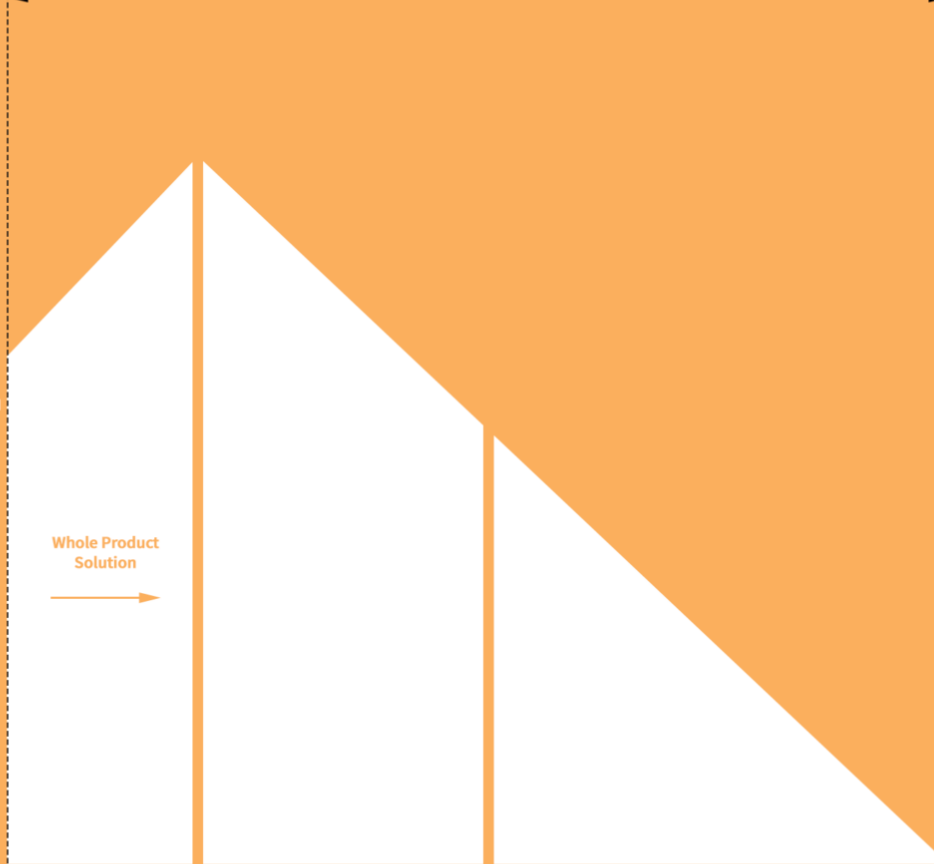


Innovators: 2.5%

Early Adopters: 13.5%

## Mainstream Market

People who want complete solutions and convenience



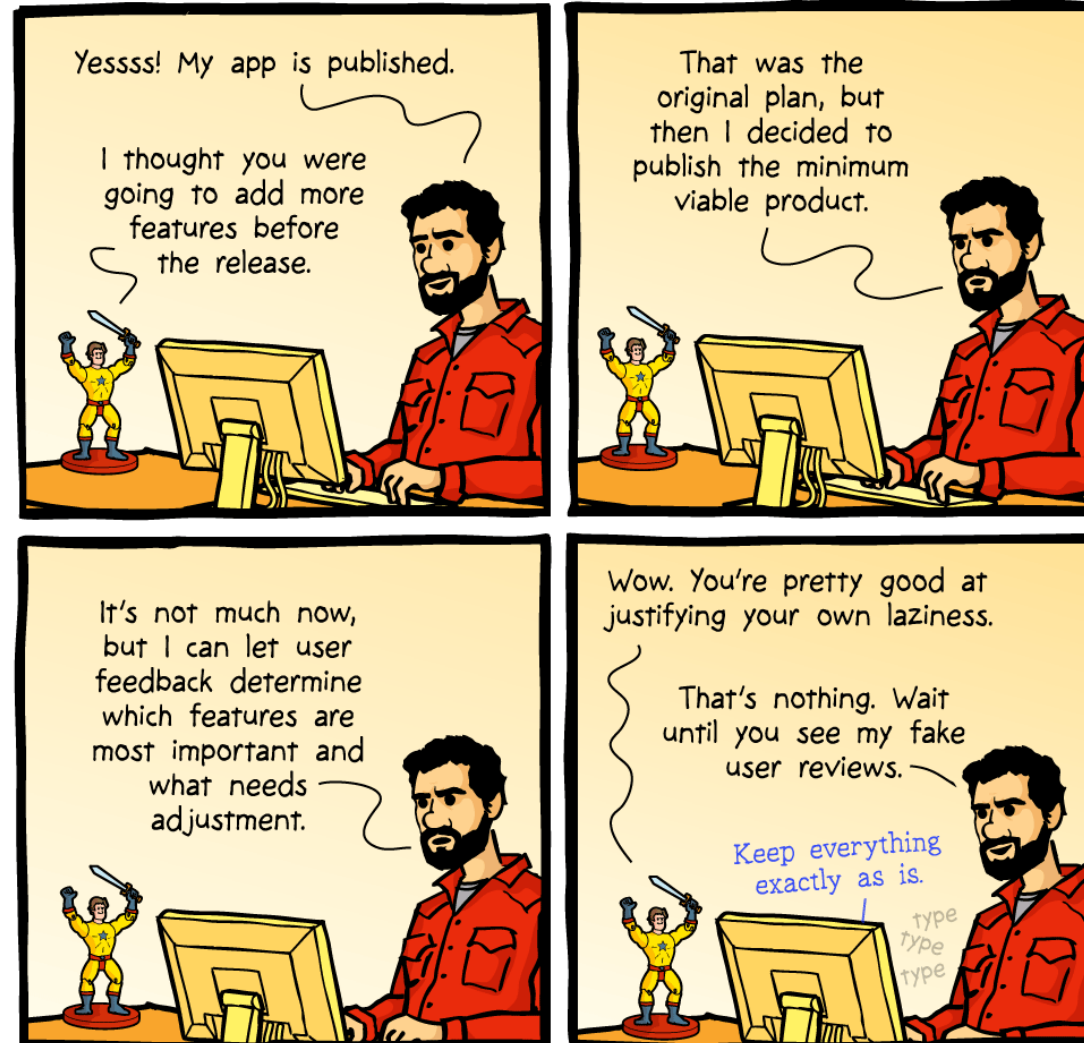
Early Majority: 34%

Late Majority: 34%

Laggards: 16%

The CHASM

# Sometimes little is too little



# Outcomes of MVP

- The hypothesis is **completely correct**: development continues
- Hypothesis is **partially true**: customer feedback helps to refine the idea
- Hypothesis is **refuted**: at least most of the funds are retained that would have been used or implementation.

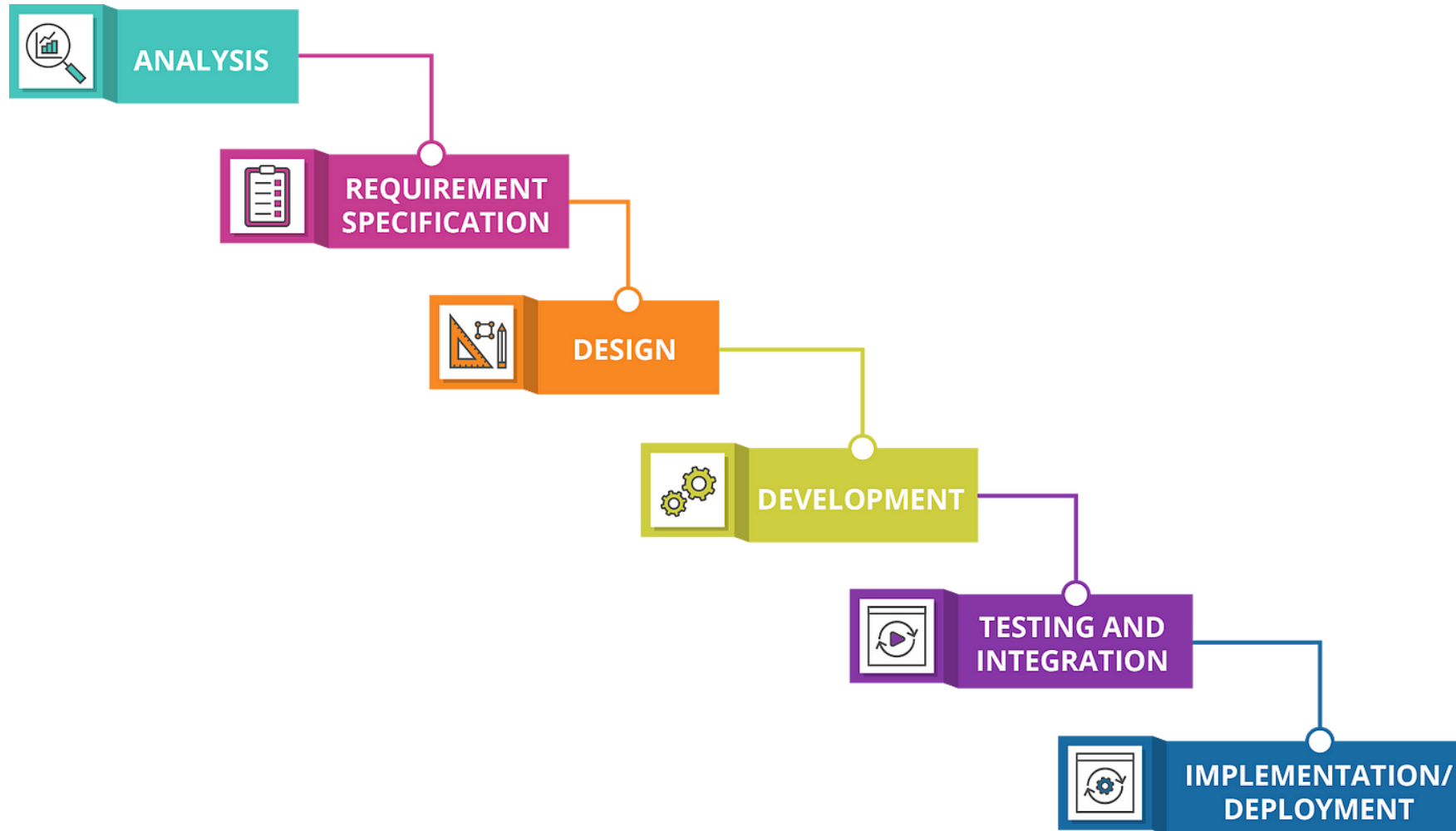
# Key takeaways

- MVP is a process not a one-off thing.
- Figure out the minimal for your specific case.
- MVP is not an excuse for a s\*\*\*y product.
- Frontend-oriented for especially business-to-customer cases.

# Overview of Methodologies

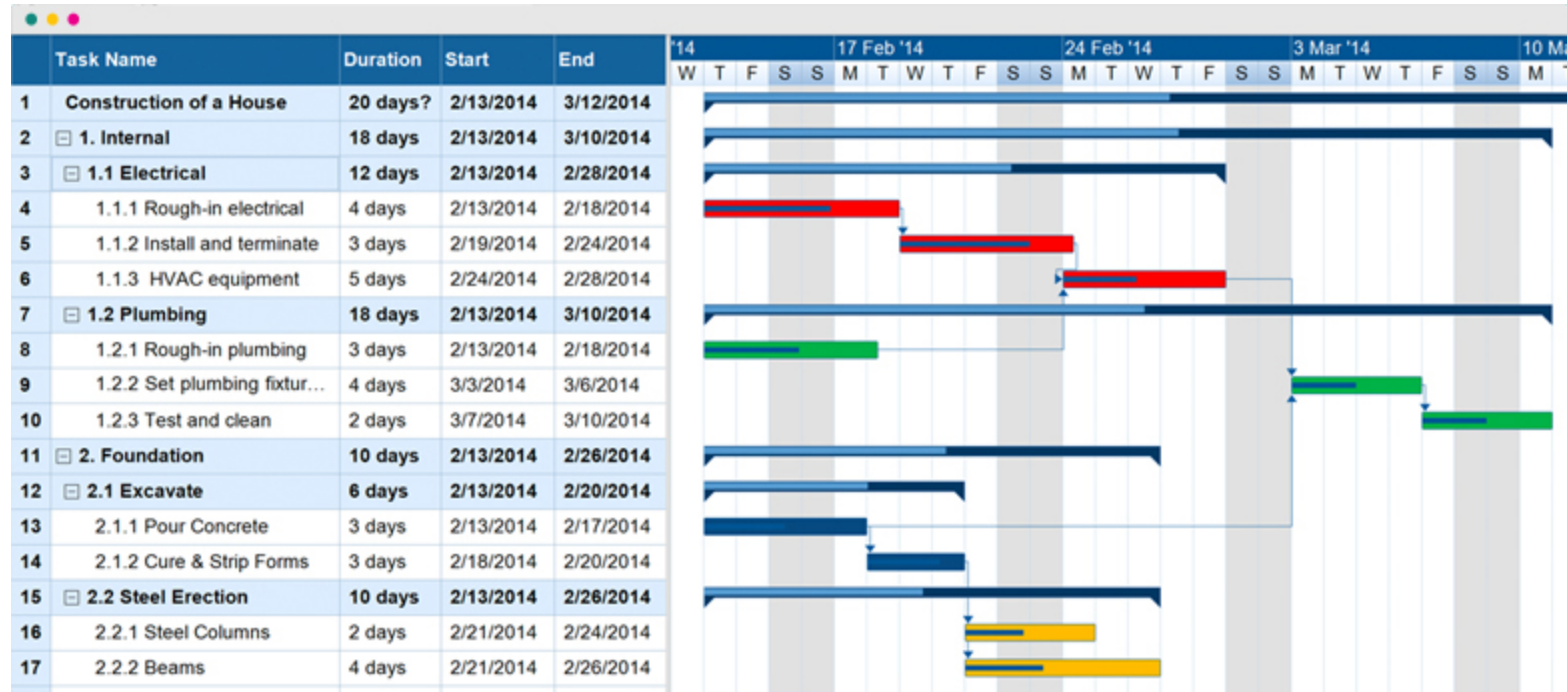
Two different approaches.

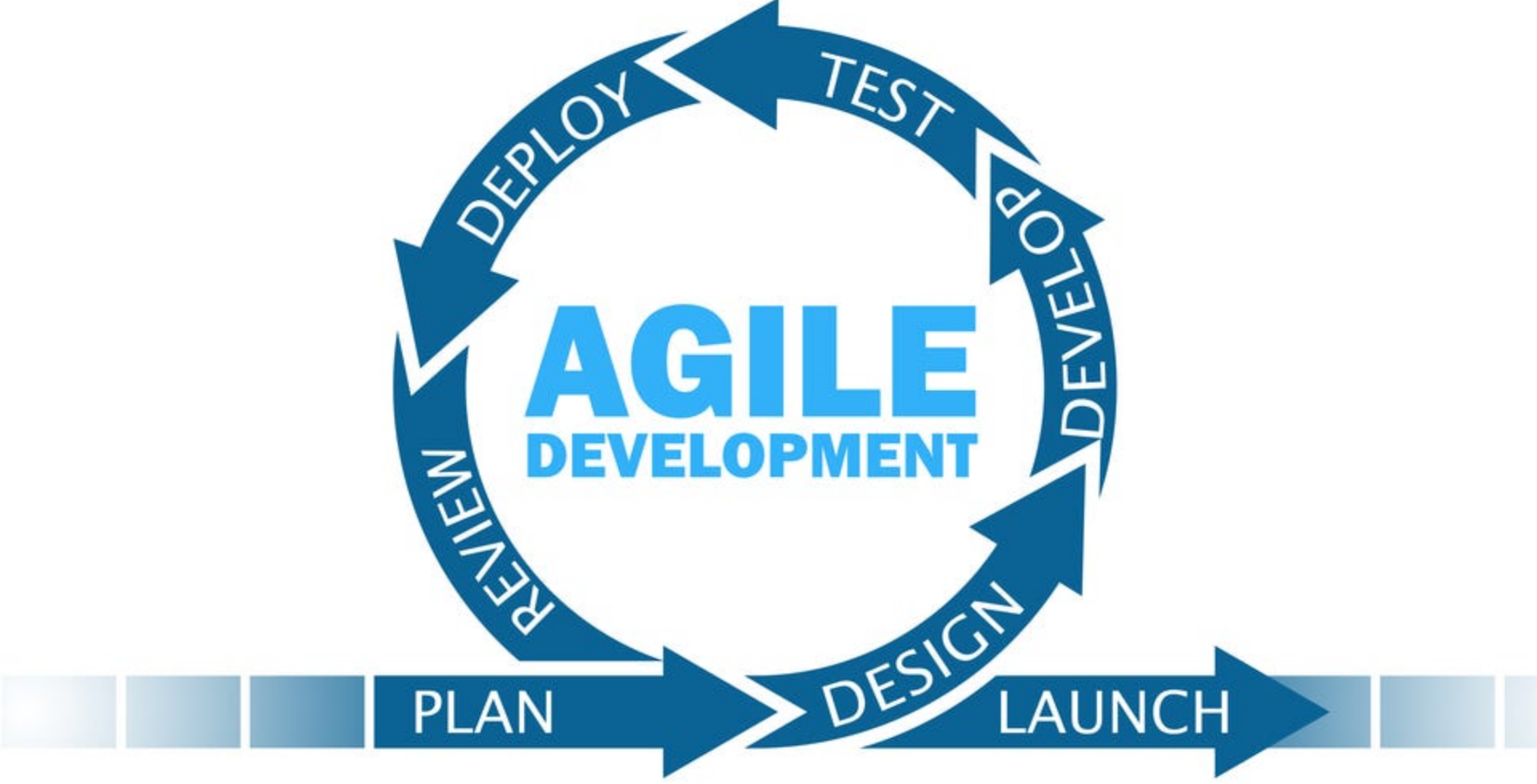
# Waterfall Development











# Benefits

- Reduction of uncertainty
- Minimal time to market
- Measurability
- Traceability
- Right action right time

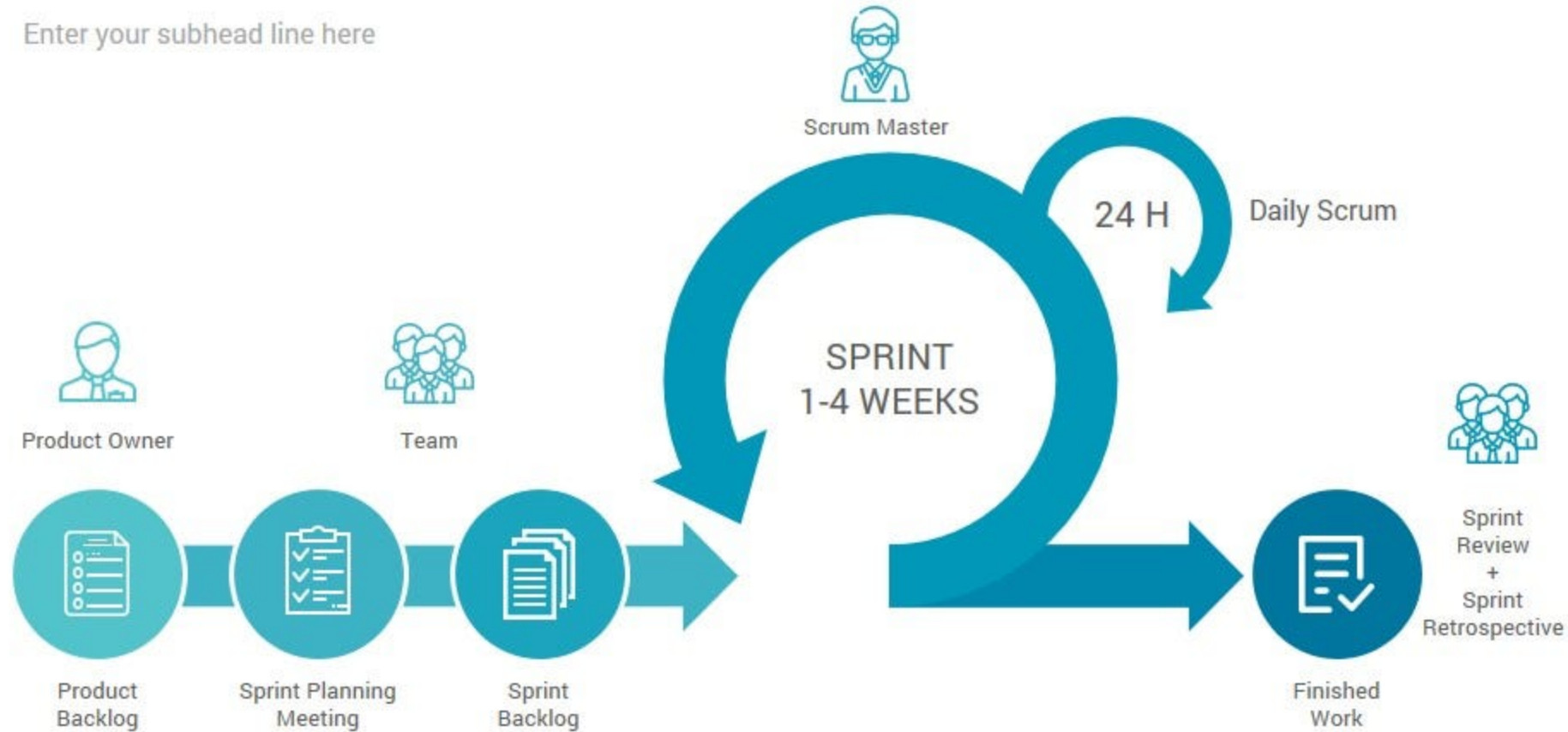
# AGILE

## Manifesto for Agile Software Dev.

- INDIVIDUALS AND INTERACTIONS OVER PROCESSES AND TOOLS
- WORKING SOFTWARE OVER COMPREHENSIVE DOCUMENTATION
- CUSTOMER COLLABORATION OVER CONTRACT NEGOTIATION
- RESPONDING TO CHANGE OVER FOLLOWING A PLAN

# Scrum Process

Enter your subhead line here



# Scrum: Roles

- **Product Owner** - Represents the stakeholders and is the voice of the customer.
- **Development Team** - Self-organizing, cross-functional team of people who are responsible for building the actual product.
- **Scrum Master** - accountable for establishing the Scrum process

# Scrum: Elements

- **User Story** - Informal, general explanation of a software feature written from the perspective of the end user. Its purpose is to articulate how a software feature will provide value to the customer.
- **Sprint** - short, timeboxed period when a team works to create a done, usable, and potentially releasable product increment.
- **Product Backlog** - prioritized list of all the functionalities that we need in the product. It is a single source of truth for all the product requirements.
- **Sprint Backlog** - Set of Product Backlog items that we select for the sprint.

# Scrum: Practices

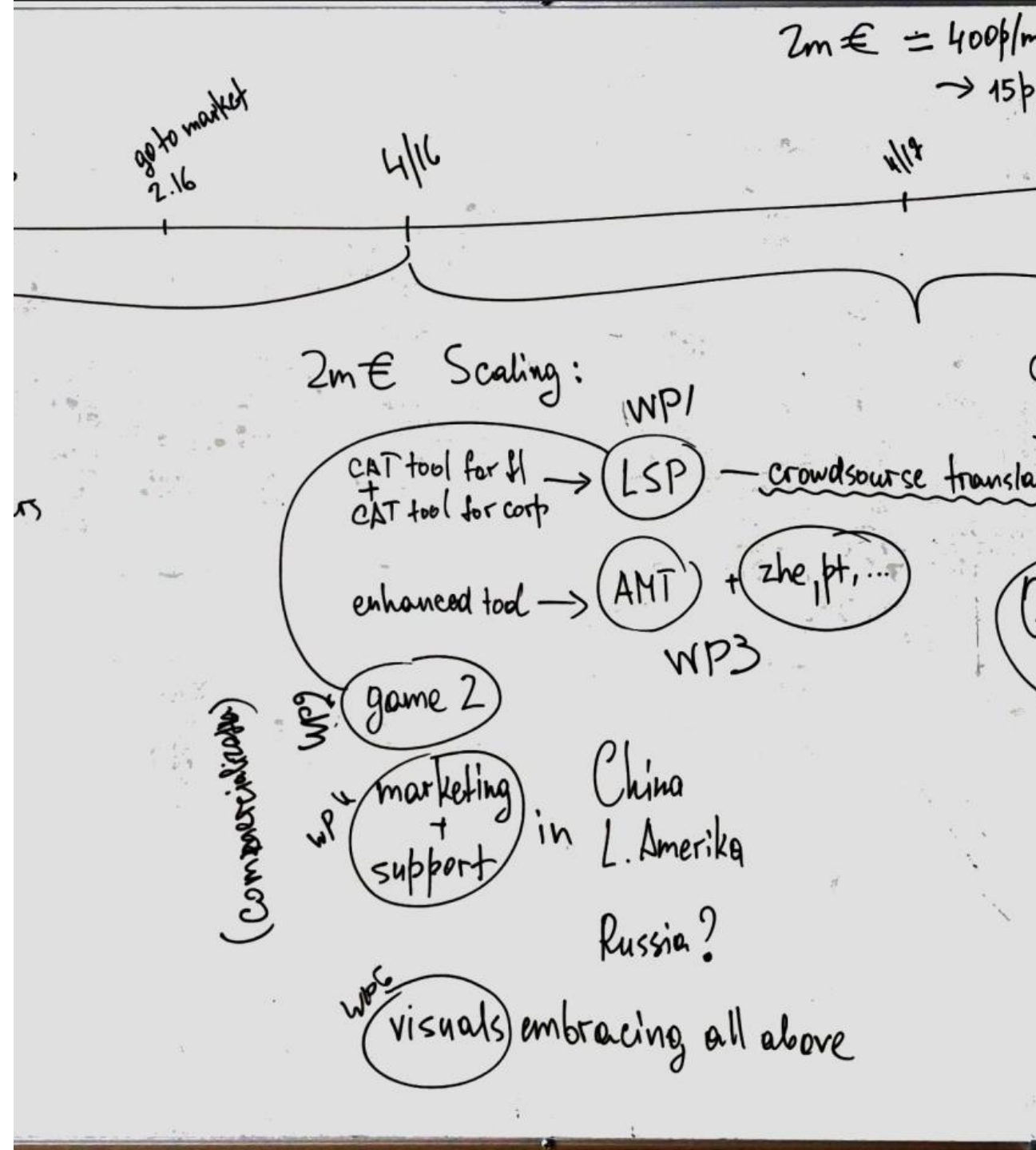
- **Sprint Planning** - event in Scrum that kicks off the sprint by defining the scope of delivery and how to accomplish that work.
- **Stand Up (Daily Scrum)** - 15 minute PPP event to keep track of progress.
- **Sprint Review** - event that takes place at the end of the sprint to inspect the increment and adapt the product backlog.
- **Retrospective Meeting** - event that solely focuses on improvements on every aspect of Sprint



# Scrum: How does it work in practice?

## Prerequisites

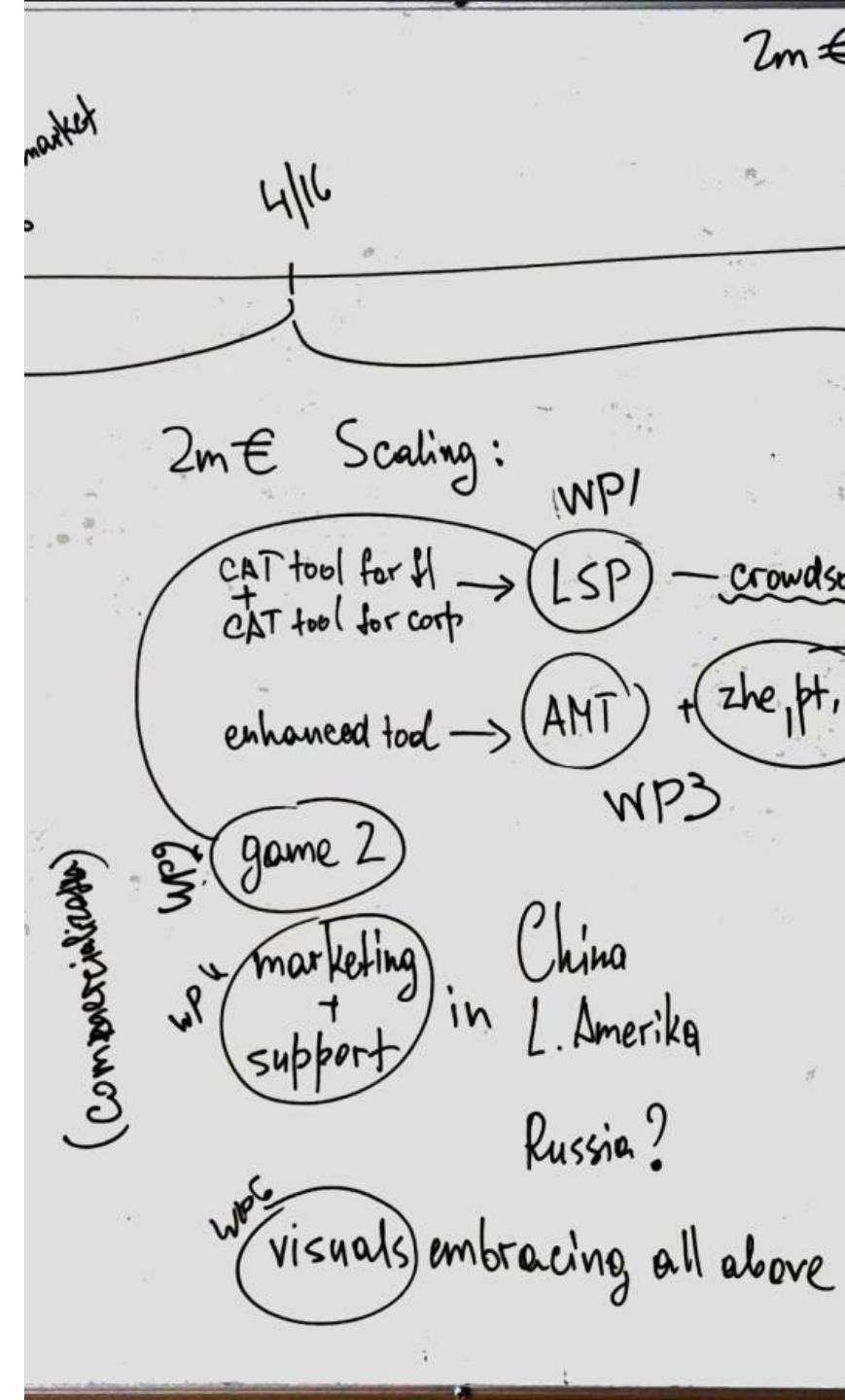
- Describe the desired end result in general terms.
- Define the sprint cycle length.



# Scrum: How does it work in practice?

## Running the sprint

- Set general goals for next sprint
- Brainstorm
- Build consensus what to build next
- Define clear tasks and clear measure of success
- ... work ... and share the testing load
- Evaluate the results



# TO-DO

CHI - Assembly Done  
 VM  
 deployed

DEBUG GAME  
 TRACKERS  
 MACEIN/  
 YOUNG

GAME 1.7  
 TIME CODING  
 IF TIME < 1 MIN

SEARCH /  
 REPTAL  
 THINK OF LOCAL  
 MODULE FEATURES  
 VSB  
 Subpage to support  
 and content / design  
 MACEIN/  
 YOUNG  
 TAVL

VSB  
 CAT TOOL - SHOW  
 TARGET VIEW  
 MOBILE VIEW - STEW  
 OF CAT GAME

VSB  
 LANGUAGE PAIRS C  
 INSTEAD OF COLLECT

GAME  
 BASIC MESSAGES  
 SCREEN UPDATE

GAME | PROFILE  
 HAVE MESSAGES  
 (CLEANUP) (MACEIN)

GAME  
 Touch board (fully)  
 change (control)

GAME  
 touch to follow  
 long to view to  
 hand

GAME  
 SET MESSAGE LINE  
 FOR API CALLING  
 CODE

GAME | LOCAL  
 PEDS ON WINDOWS  
 CONFIGURATION  
 (LEADERBOARD  
 TRACKING)

GAME  
 MESSAGE MESSAGE  
 DIALOGUE  
 TO BRANCH  
 (DIALOGUE)

AUTH | API  
 authentication method

GAME #1  
 kate's conversation

GAME  
 TEXT BUBBLES  
 FOR CHARACTERS

CAT  
 VIEW

CAT-TOOL  
 VIEW #2  
 ☹️

STICKERS  
 ON  
 GAPS

VISITKA  
 ☹️

PILTIE  
 BROTHER  
 LIVED ☹️

GAME  
 User profile screen  
 MACEIN

GAME  
 kate's board screen  
 MACEIN

MESSAGE  
 MESSAGE  
 (VSB)

VISITKA

GAME  
 AUTH  
 FIX

Responsive  
 Web  
 10.11  
 MACEIN

GAME  
 AUTH  
 FIX

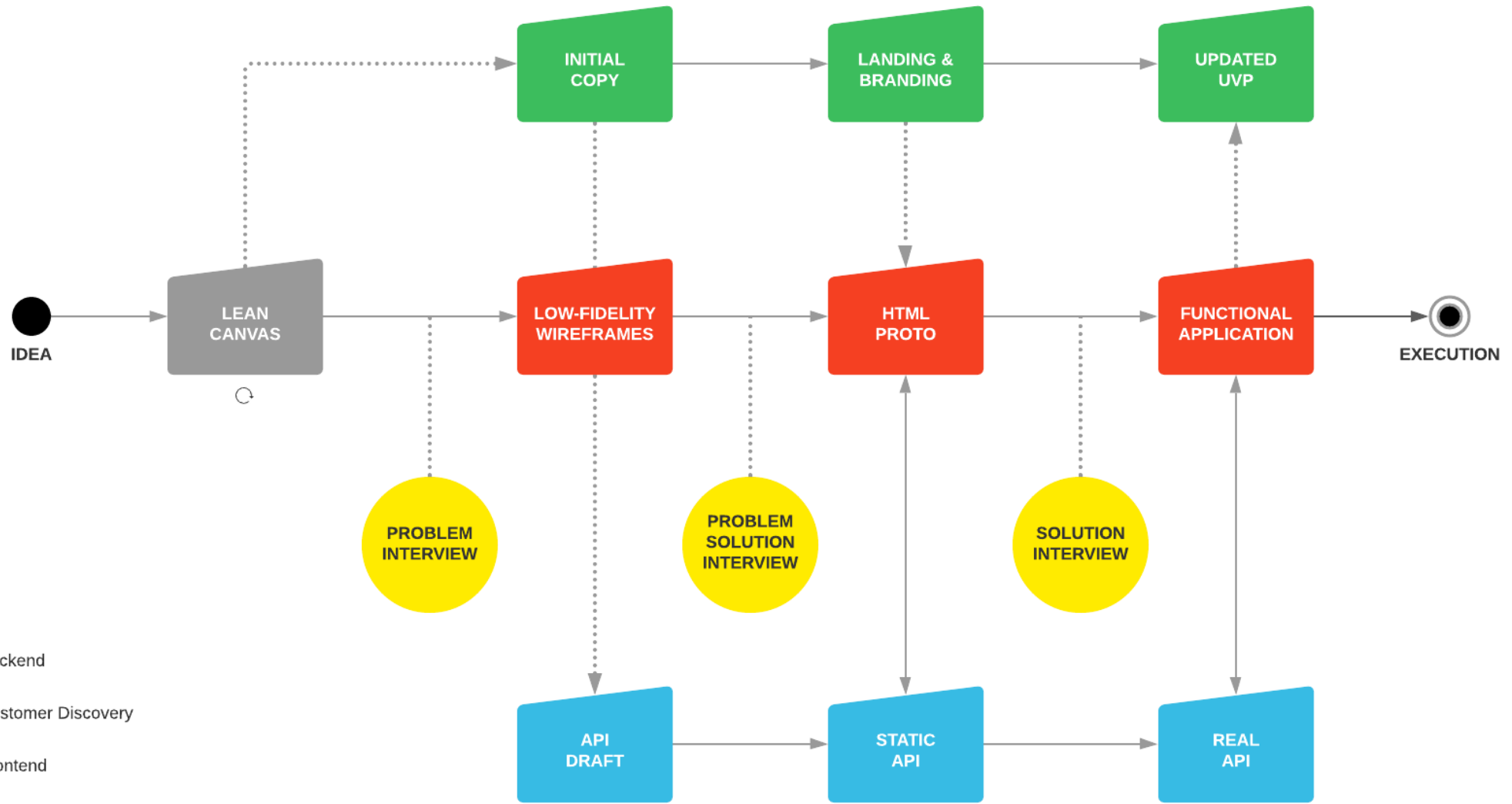
HOW IT  
 WORKS  
 (MACEIN?)  
 FIX ☹️

TEXT TRANSLATED  
 TEXT

Web  
 How it  
 displays

VSB  
 MAKE MORE  
 IN TO  
 VISUALIZE LANGUAGE  
 PAIRS  
 ☹️  
 MACEIN

KEEP



- Backend
- Customer Discovery
- Frontend
- Marketing
- Single Source of Truth

# Do not invent the wheel

- Lean Validation Playbook: <https://guides.co/g/the-lean-validation-playbook/60783>
- Startup Playbook <https://playbook.samaltman.com/>
- Thoughtbot Playbook <https://thoughtbot.com/playbook>

# References

- What is a Minimum Viable Product. <https://blog.leanstack.com/minimum-viable-product-mvp/>
- Don't Start With an MVP. <https://blog.leanstack.com/dont-start-with-an-mvp/>
- How DropBox Started As A Minimal Viable Product. <https://techcrunch.com/2011/10/19/dropbox-minimal-viable-product/>
- Our Dangerous Obsession With The MVP. <https://techcrunch.com/2014/03/01/our-dangerous-obsession-with-the-mvp/>.
- A Minimum Viable Product Is Not a Product, It's a Process. <https://www.ycombinator.com/library/4Q-a-minimum-viable-product-is-not-a-product-it-s-a-process>

# References

- How to set up a Minimum Viable Product (MVP).  
<https://medium.com/innovation-machine/from-a-problem-to-a-well-defined-mvp-in-8-steps-88efa12c1345>
- A Step-by-Step Guide to Build a Minimum Viable Product (MVP).  
<https://www.netsolutions.com/insights/how-to-build-an-mvp-minimum-viable-product-a-step-by-step-guide/>
- Making sense of MVP (Minimum Viable Product) – and why I prefer Earliest Testable/Usable/Lovable.  
<https://blog.crisp.se/2016/01/25/henrikkniberg/making-sense-of-mvp>
- Why a minimum viable product can be bad for your business.  
<https://qvik.com/news/mvp-minimum-viable-product-can-be-bad-for-your->

**Thank you!**