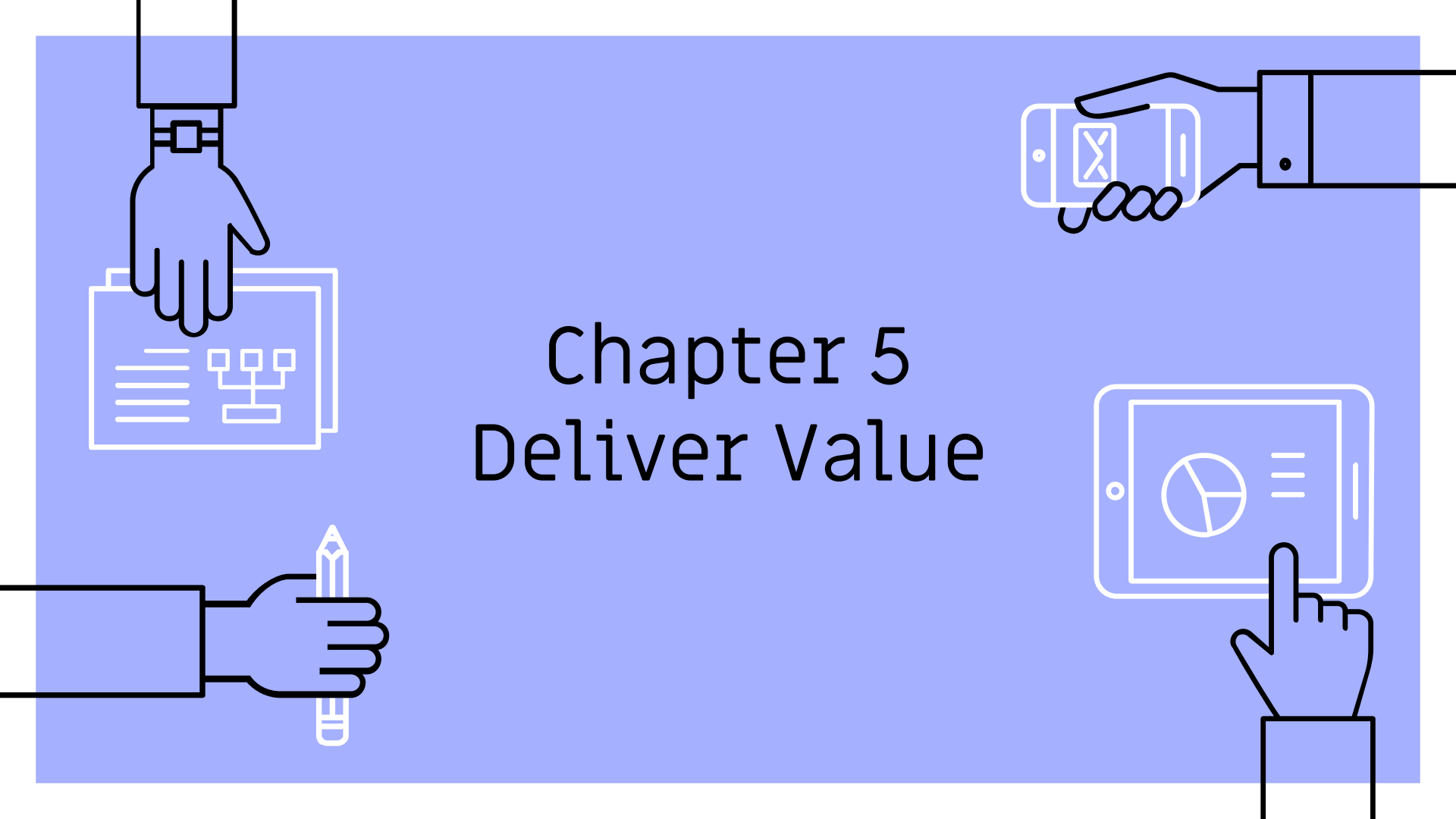
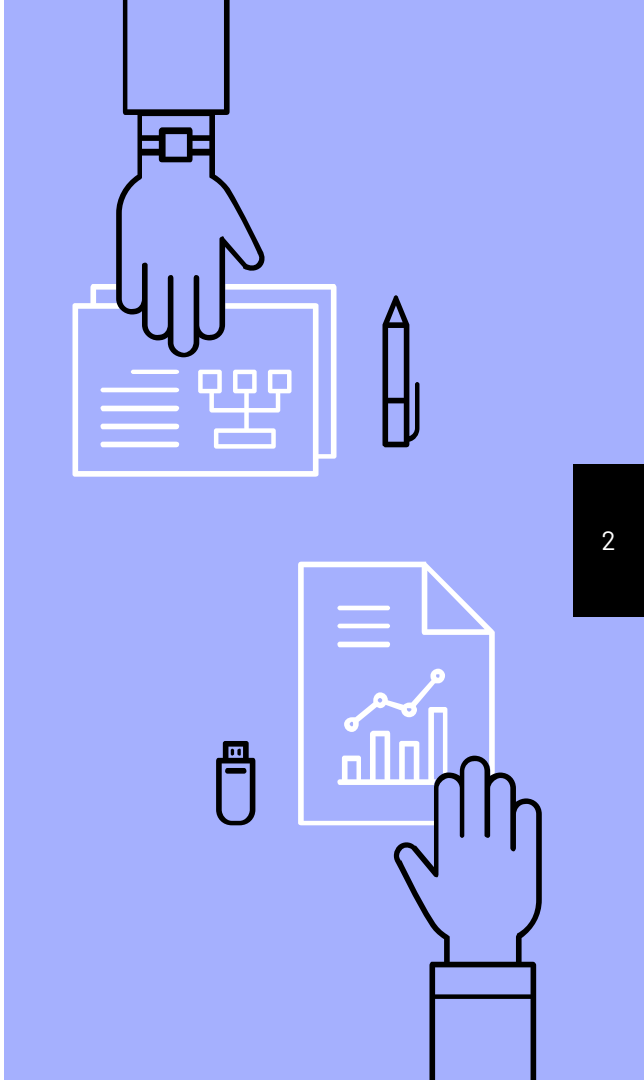


Chapter 5
Deliver Value

Key concepts surrounding value delivery

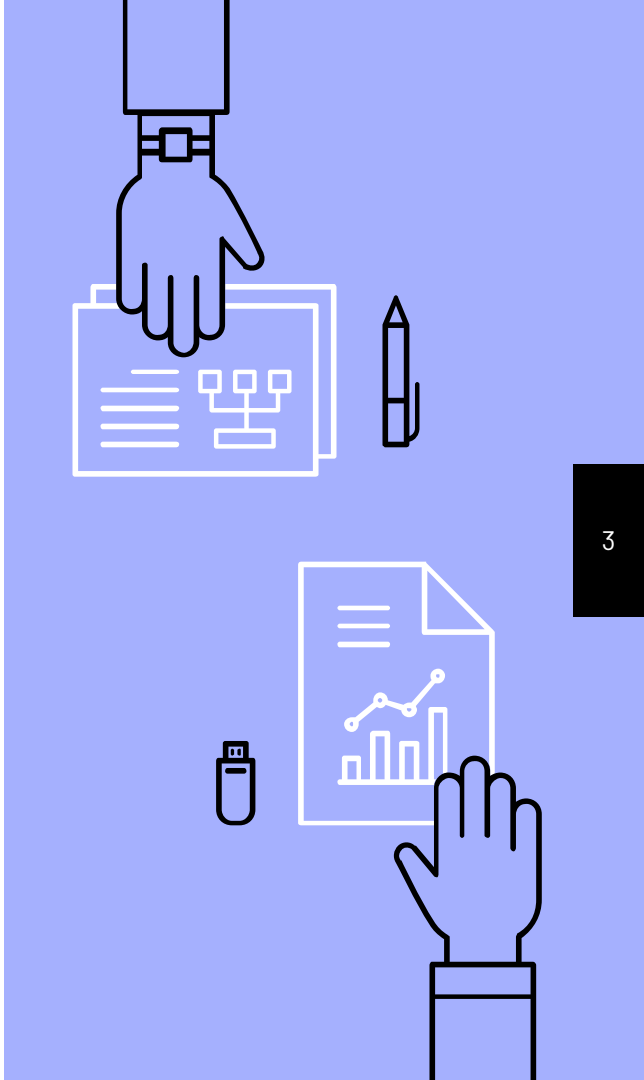
- Feature Injection
- Minimum Viable Product
- Minimum Marketable Feature



Feature Injections

- Inject features that represent the work the team does (outputs) to create that value (outcome)

- Key is knowing the value the initiative intends to deliver
  - Delivering the features that provide value
  - Communicating information about the features through examples
  - Lean Metaphor



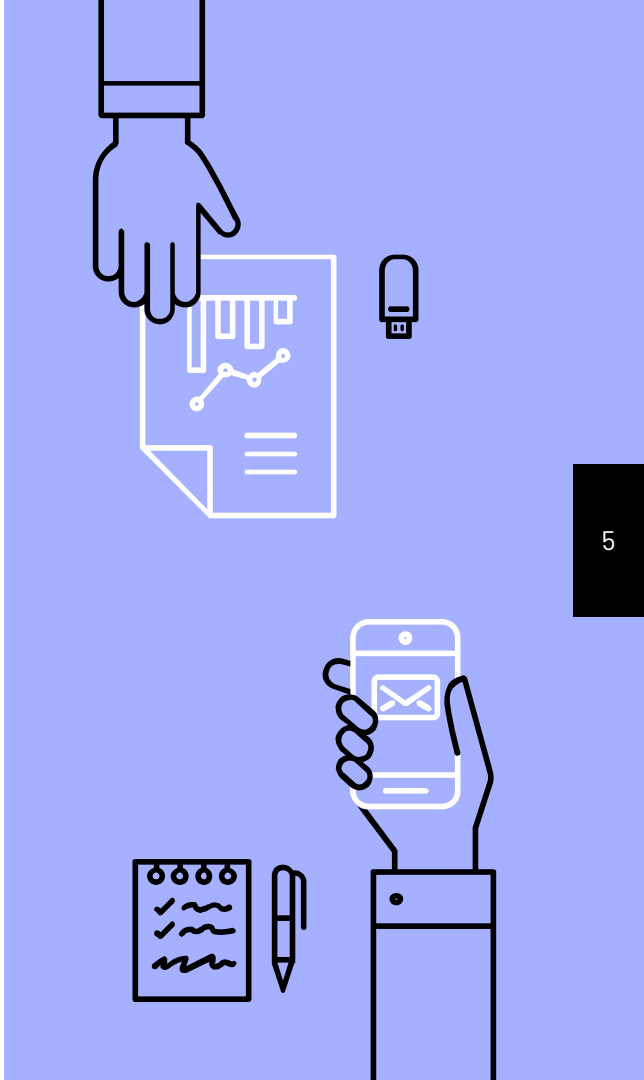
Ex: Toyota Factory

- Stand by the out door of a Toyota Factory that is idle, empty and with all workers waiting…
- Request a car!
- You would see the parts pulled to the out door, starting at the in door
- If you watch … you would see the Kanban cards flowing backwards against the flow of value

Identifying Value

Feature Injection

- Begins by creating an understanding about the value to be delivered
- The value is expressed in the form of a model
- The team can repeatedly test progress toward delivering value as the work proceeds
- Business Value?
  - A project delivers business value when it increases or protects revenue or reduces cost

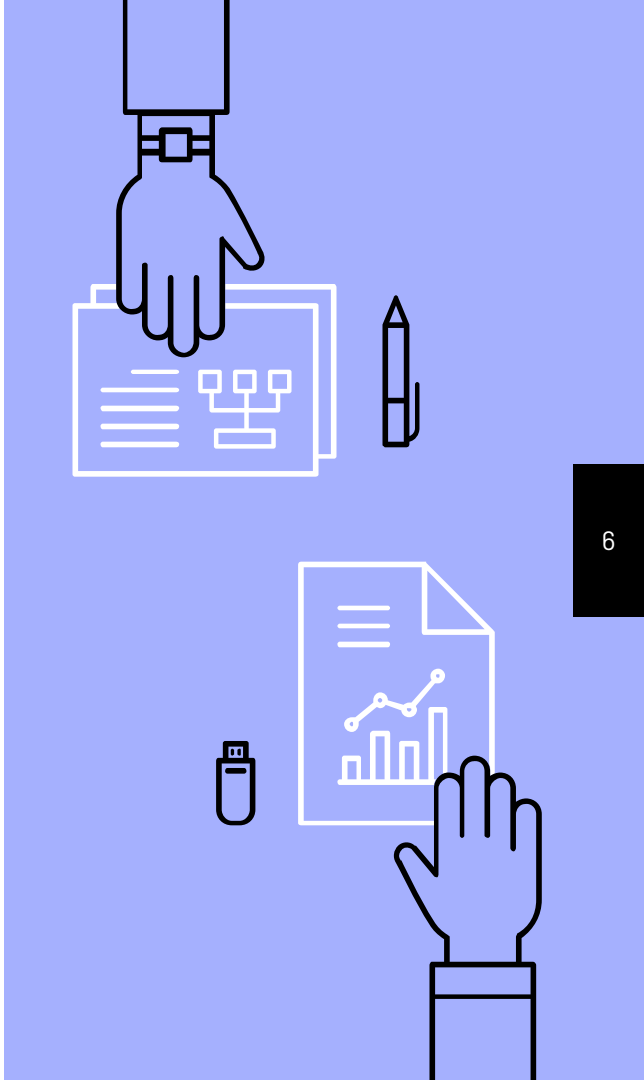


Business Value...?

Two Definitions

1. A project delivers value when it increases or protects revenue or reduces cost in alignment with organizational strategy
2. **IRACIS**
   - **I**ncrease **R**evenue
   - **A**void **C**osts
   - **I**mprove **S**ervice

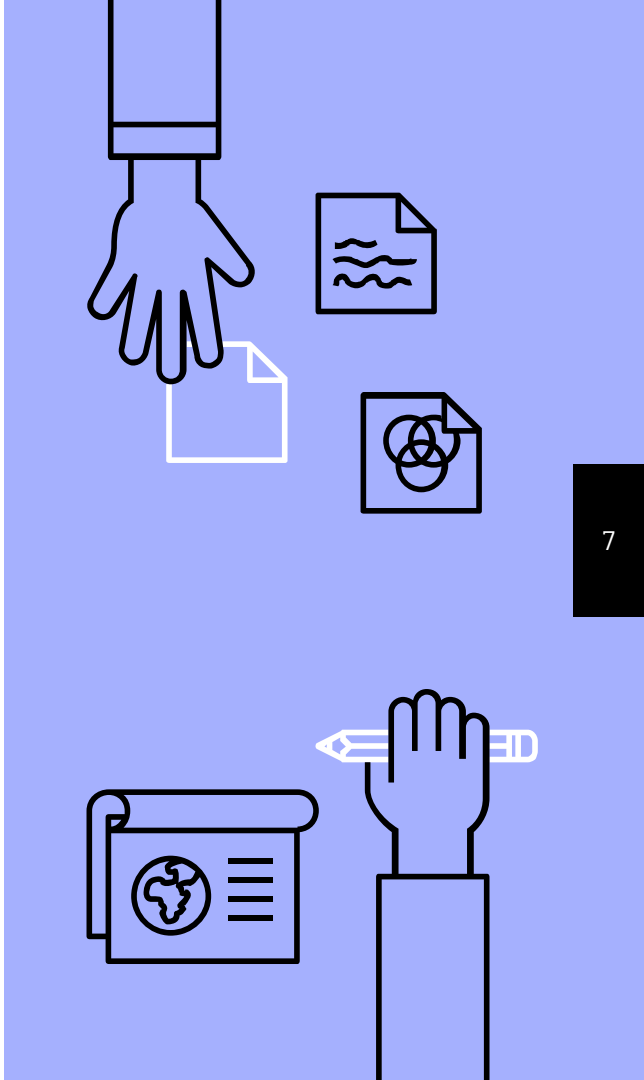
"... value should not be focused on profit or shareholders, but on delighting customers."



General way of describing **Business Value**

- The business value of an IT project can be gauged by whether it helps an organization meet one or more goals

- Measured based on the project's impact on one or more objectives



The Business Value Model

- Describes the impact of the project on your selected objectives so you can reevaluate expectations based on new information throughout the course of the project
- If it appears that the project will not accomplish what is needed, corrections can be made rather than continuing and finding out it's not going to "deliver"

A Business Value Model

Used to assess the impact of decisions on revenue objectives

Example: The project is expected to generate an additional $15 million in profit

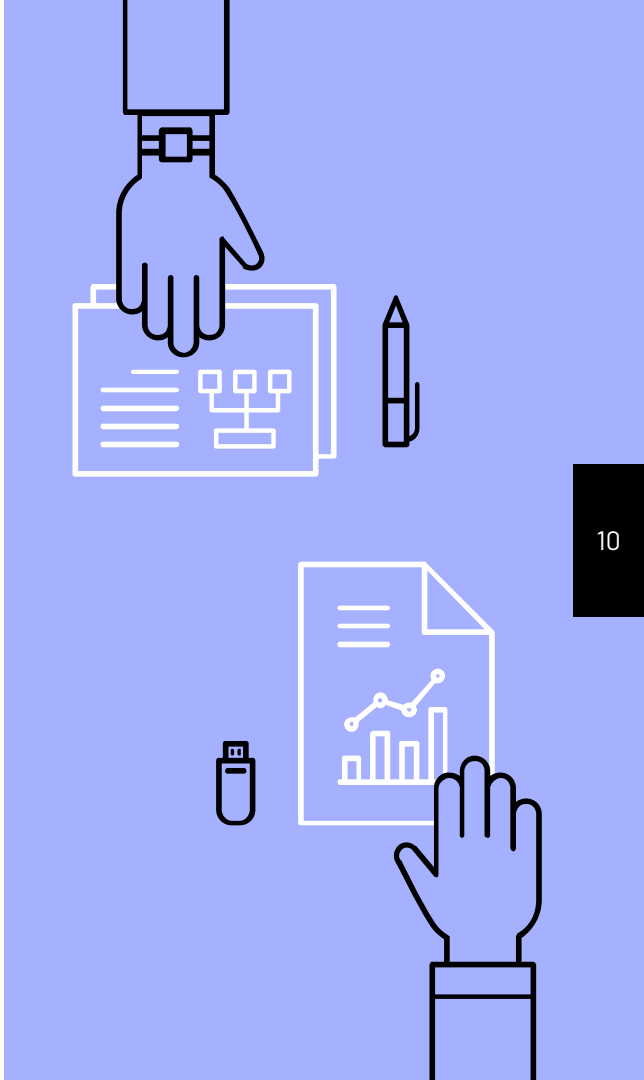
Based on the following assumptions:

1. We achieve 20% of the sales of existing project XYZ
2. The total cost of designing and producing the product is $5 million
3. Our product is the first to market
4. We are able to release the product two months before Christmas

Budget Spreadsheet

Business Value Model… Not just a number!

- Describe the **target** of the objective
- Explains the assumptions underlying the **target**
- Provides a test for what happens to the objectives if the underlying assumptions change

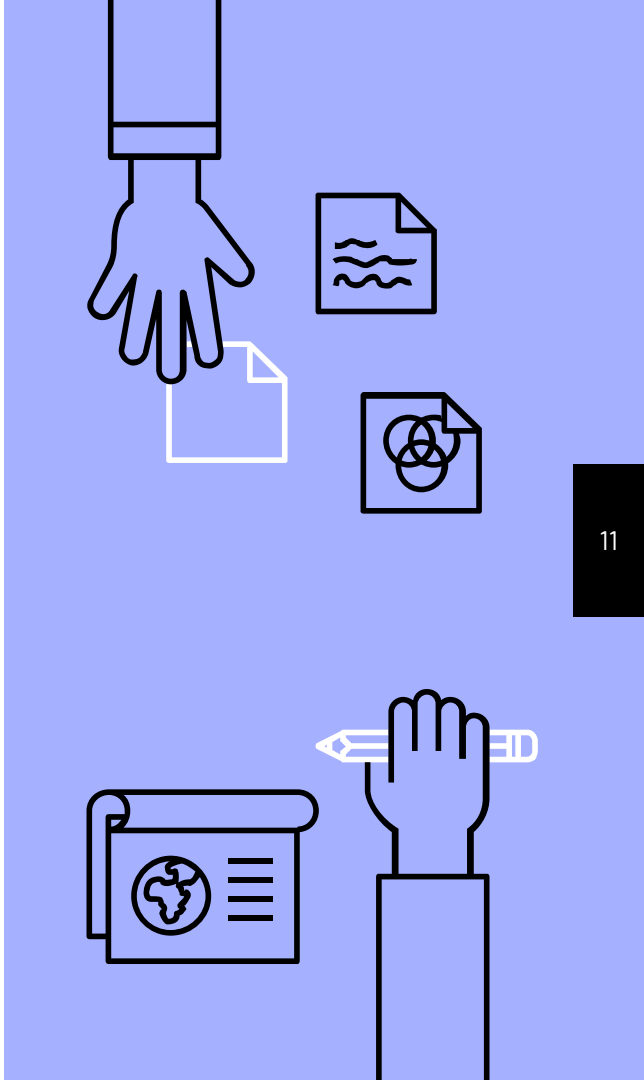


Another Perspective

Business value is determined by the “users”

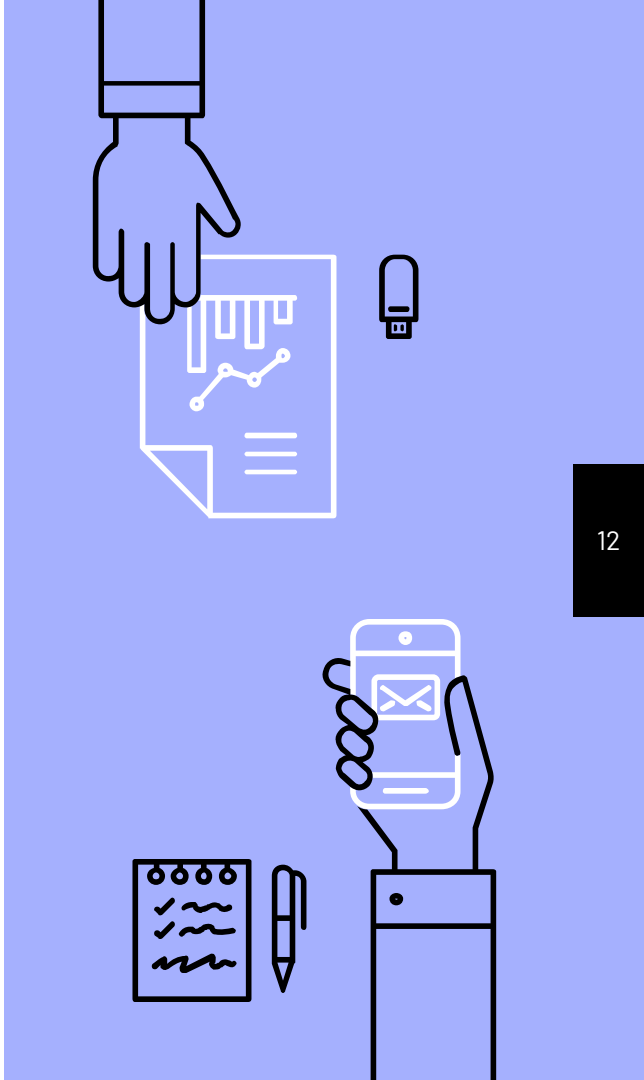
- Identify the relevant stakeholders–Identify their needs and goals
- Agree on how to measure/test the achievement of the user needs and goals
- Select the (few) most important measurements and tests (the “value drivers”)
- Define the relationship between the Value Drivers
- Use the Value drivers to focus and prioritize the work, from start to finish

Profit but also Customer Satisfaction



Inject Features

- Prioritize the Features … “identify value first”
- Then, iteratively identify features that are needed to deliver value
- Features and their User Stories
  - Measuring value should be at the Feature level
  - By working from value to features… you don’t assign value to individual stories or need to assign value to any change
- When a feature is “injected”… identify the output that represents the feature
- Work backward to understand what you need to do to deliver that output



Stakeholders

- **What outputs do they expect?**
- **Note:** outputs should be attached to stakeholder needs

- Understanding stakeholders (users) is necessary in order to know what outputs are actually needed

- This allows for the identification of the outputs that deliver the expected value!

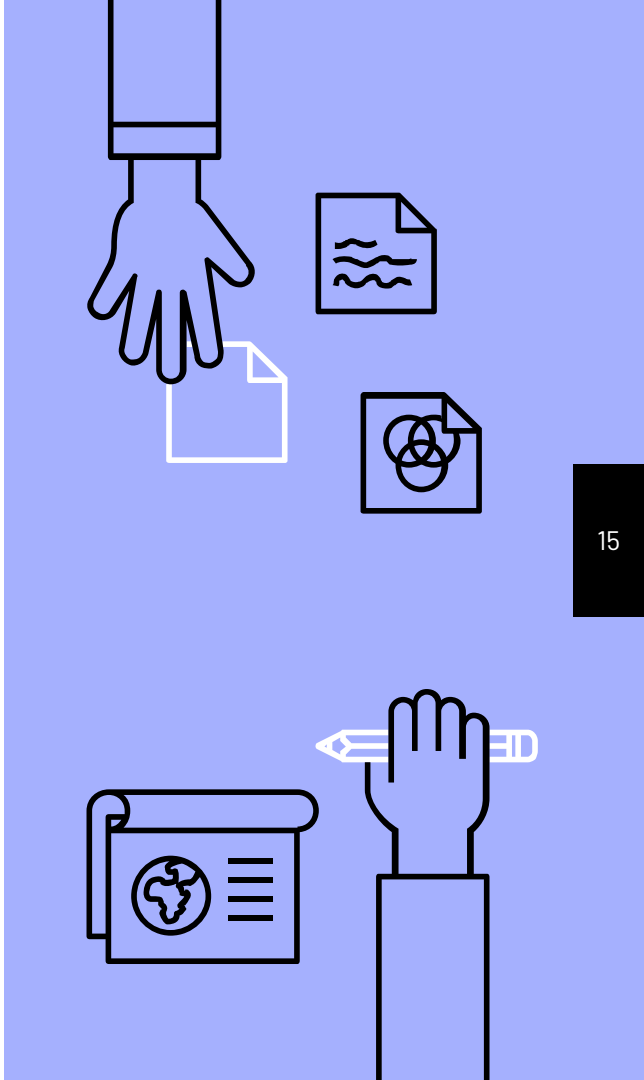
- Once the needed outputs are identified, you can work backwards to identify what processes are needed to provide the outputs and inputs the process needs to create the outputs

Table 5.1 Payroll Stakeholders & Their Expectations

Stakeholder	Expectations
Employees	If the payroll system did not exist, they might not get paid.This probably not the case, but it certainly may be a more complicated process to get payroll completed. From the employees' perspective, the payroll system adds value because it generates checks. When employees get paid, they continue to work, so the value is in protecting revenue by keeping employees satisfied that money is still coming in the door. (Yes, I know this is kind of a stretch, but it's the one that seems to fit best.)
Payroll department	Assuming that the organization would still pay its employees even without a payroll system, the lack of a system would make the process inefficient and more prone to error; so creating a payroll system reduces costs. And to some extent, it protects revenue by reducing the risks of incorrect paychecks.
Employee relations manager	Even if you could pay employees without a payroll system, the lack of organized data makes it much more complicated to perform analysis on payroll information, which indirectly leads to increased risk for the organization, similar to the items described previously.

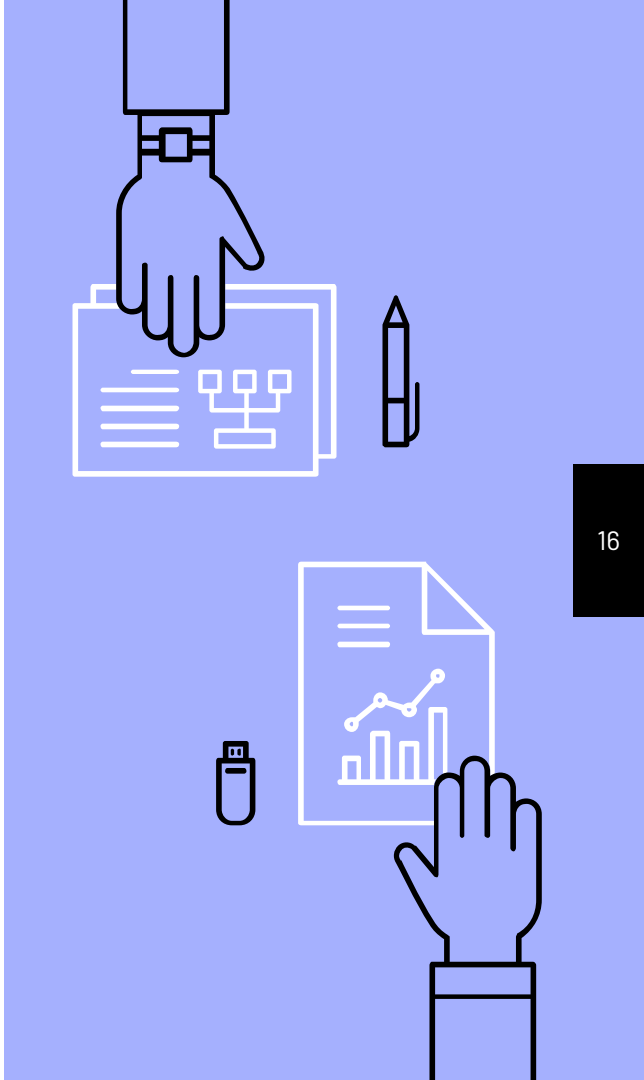
Spot the Examples

- Use models to describe the outputs & processes and the inputs used to create them
- You want / need a shared understanding by all those involved in **delivering** features



Spot the Examples (cont.)

- Create examples to get a better understanding
  - They provide concrete situational guidance for developers
  - They give the team a way to test the models… to make sure all situations are accounted for
  - The author… for the submission system and all the features, examples were used to describe the features
  - Features were described in terms of examples
  - The examples were used as the basis for preparing automated acceptance test



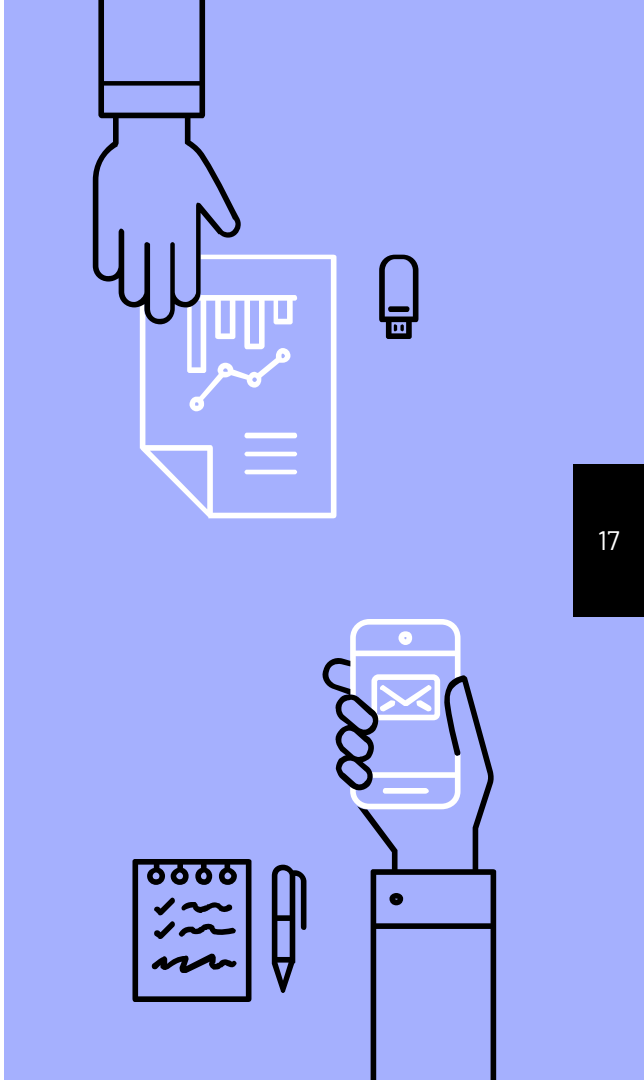
Example (Page 62)

Feature: Edit conference dates

As a conference chair, I want to change session submission deadlines.

Background

- Given I am logged in as “Connie”
- The session submissions should be open
- And session edits should be open
- And accepted session edits should be open



Conference Date Example (cont.)

Scenario: Update Open Date

When I change the session submission start date to 1 day from now. Then session submissions should be closed.

Scenario: Update Close Date

When I change the session submission end date to 1 day before now. Then session submissions should be closed.

Scenario: Update Edit Date

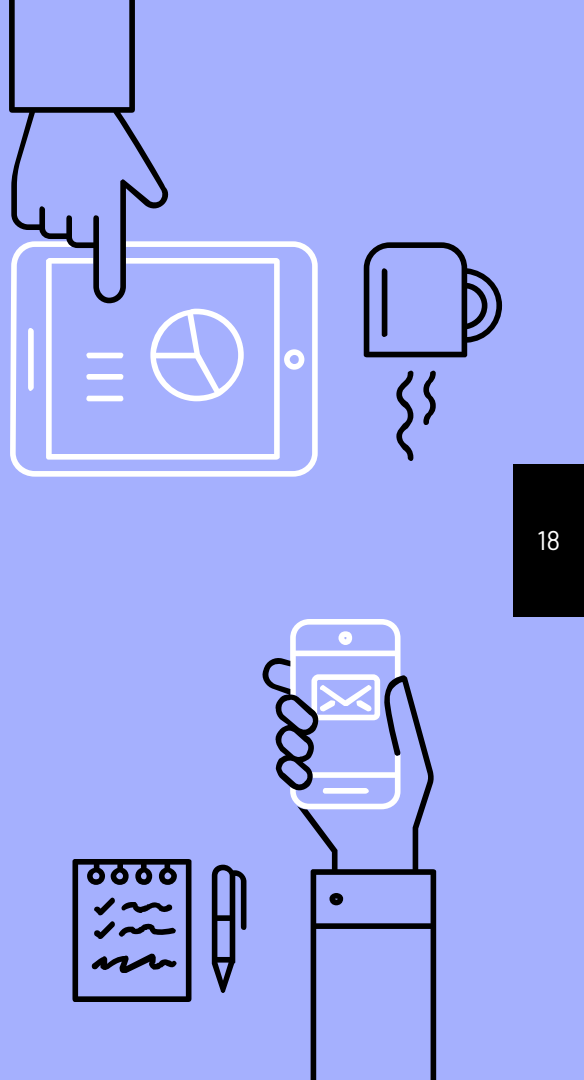
When I change the session submission edit date to 1 day before now. Then session edits should be closed.

Scenario: Update Accepted Start Date

When I change the accepted submission edit start date to 1 day from now. Then accepted session edits should be closed

Scenario: Update Accepted End Date

When I change the accepted submission edit end date to 1 day before now. Then accepted session edits should be closed



📦 Starting with the value you want to deliver

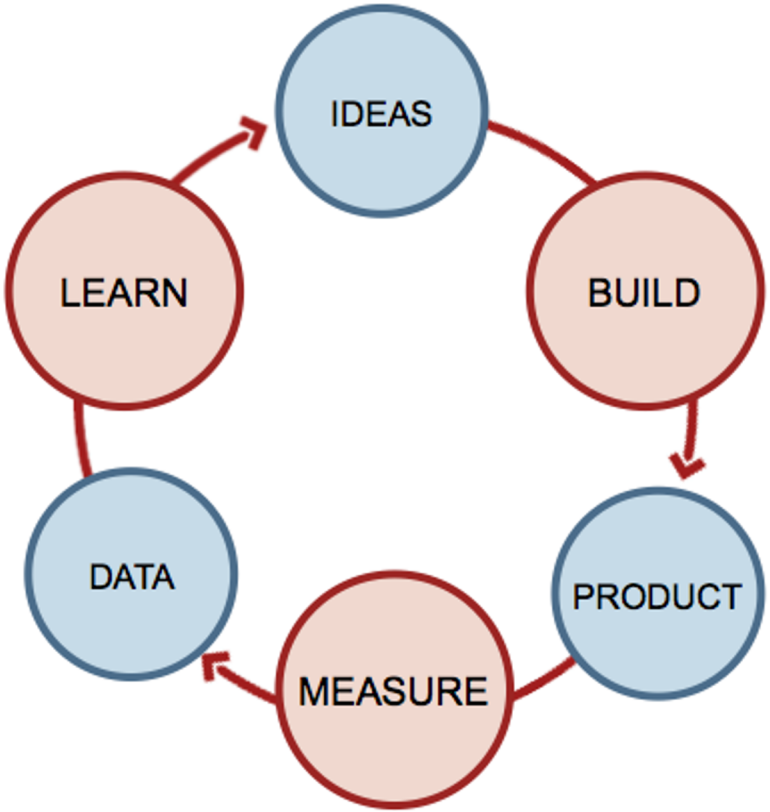
- Use that value to decide what features to build next

- Describe that feature through the use of real-life examples
  - This proves to be a simple, effective way to build the right thing
  - And not build things that are not right (or needed)!

Minimum Viable Product (MVP)

- Contrary to traditional development, which usually involves a long, thoughtful incubation period and strives for product perfection, the goal of the **MVP** is to begin the process of learning, not end it.
- A **MVP** helps entrepreneurs start the process of learning as quickly as possible.
- Not necessarily the smallest product imaginable but is simply the fastest way to get through the Build-Measure-Learn feedback loop with the minimum amount of effort.

The Loop was founded By Eric Ries
(https://www.youtube.com/watch?v=y-ozPfRHFt8)



Build Measure Learn Loop

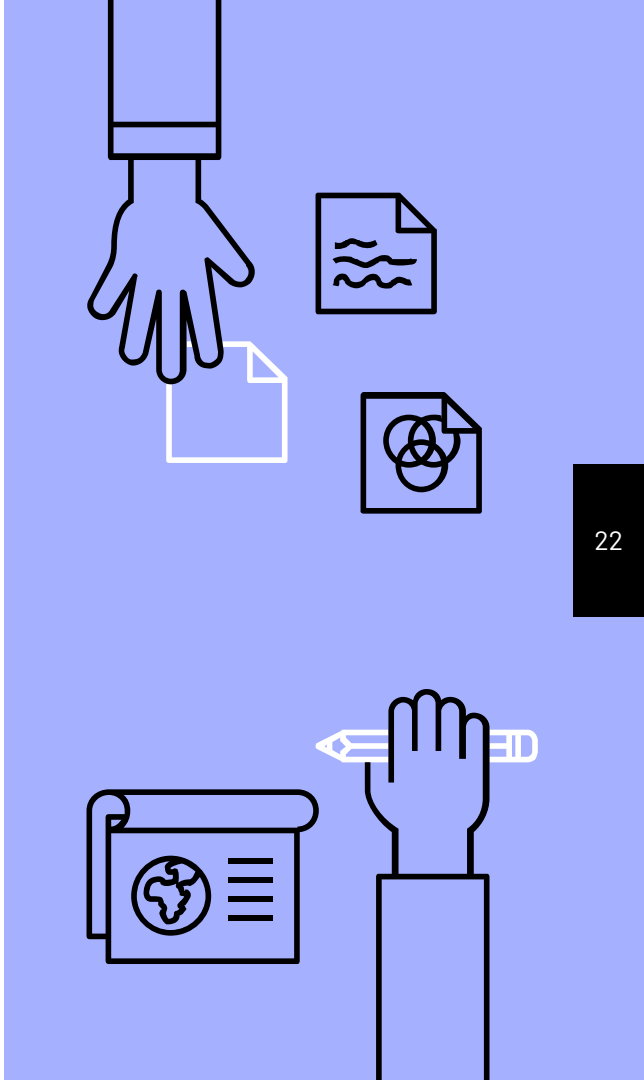
Minimum Marketable Features

Features

The main planning unit for releases

User Stories

The main planning unit for iterations

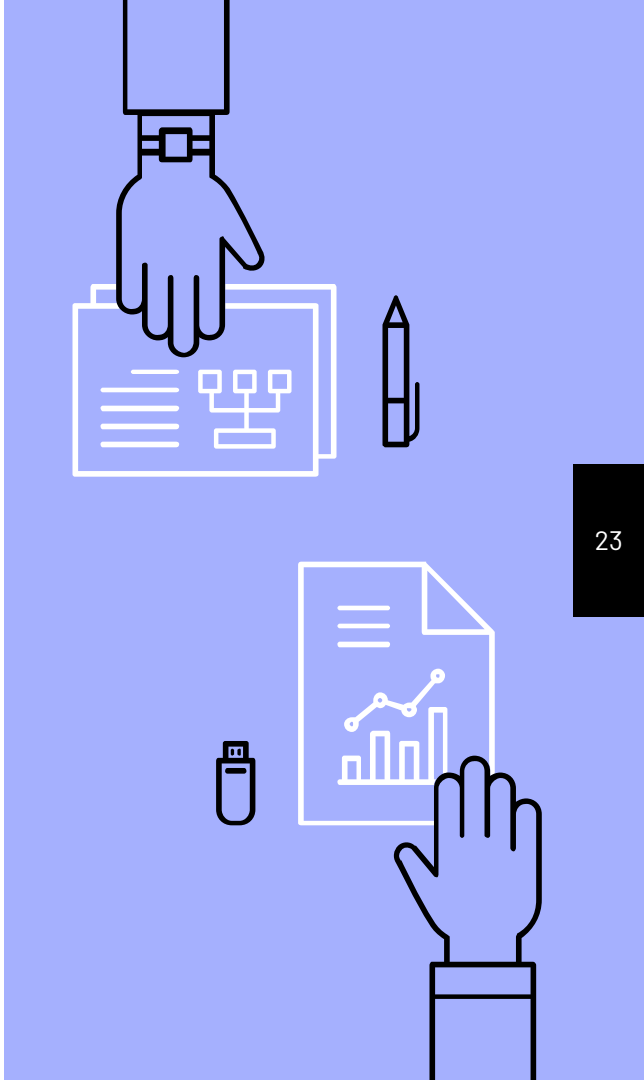


Minimum Marketable Features (MMF)

Unlike a prototype or concept test, an **MVP** is designed not just to test a fundamental business hypothesis

- Minimum:the smallest possible group of features that deliver significant value to the user
- Marketable: provides significant value to the customer
- Feature: something that is observable to the user

The **MMF** is a small, self-contained feature that can be developed quickly that delivers significant value to the user (stakeholder)



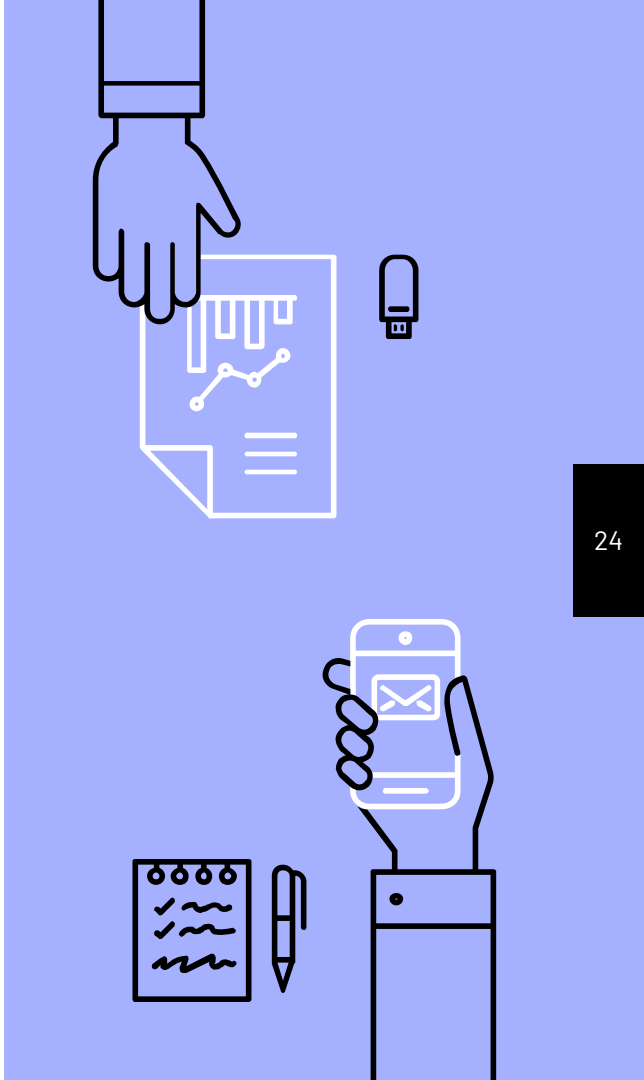
MVP and MMF

Minimum Valuable Product

Is a version of the product that lets the team use the Build-Measure-Learn loop quickly with least amount of effort. It delivers value to stakeholders and was created mainly for use in new businesses.

Minimum Marketable Feature

Is a small, self-contained feature that can be developed quickly and provides significant value to the user.



If You Remember Nothing Else

- Effective analysis starts with the outcome and works its way backward through output, process and input.
- Minimum Viable Products (**MVP**s) are intended to get information.
- Minimum Viable Features (**MVF**s) are intended to capture value.