

# Analysis

Software - Functional - **Business**

Onward to high-quality, successful ICT projects

**HO  
GENT**

# **H03 : Documenteren**

# Situering



# Functionele requirements beschrijven

- Use cases (Functional Analysis)
  - Analyse techniek
  - Specificatie
- User stories
  - Communicatie middel voor requirements
  - Moet aangevuld worden met extra technieken voor specificatie

# User stories



# Wat is een user story?

- Een user story is een **korte beschrijving** (story) van **wat een gebruiker** (user) **wil**.
- Een user story zijn enkele zinnen **gewone spreektaal** van de (computer)gebruiker waarin staat wat de gebruiker doet of moet doen, als onderdeel van z'n werk.
- User stories worden gebruikt **binnen agile software development** als een manier om de requirements te beschrijven. Hierin zegt men '**wie**', '**wat**', '**waarom**' wil.
- Het geheel is weinig gedetailleerd en moet passen op een **post-it**.



Als een bioscoopbezoeker

Wil ik mijn tickets laten sturen naar mijn gsm

Zodat ik niet hoeft te wachten aan de kassa

# Waarom user stories?

- Makkelijk te verstaan
  - Geen voorkennis vereist om het formaat te begrijpen
  - Geen technisch jargon
- Makkelijk te delen
  - Doel is communicatie tussen IT en Business
- Lage inspanning
  - Snel op te stellen, kan just-in-time
  - Geen uitgebreid dossier
- Moeten niet perfect zijn
  - Wanneer het werk moet gedaan worden, zal de discussie voor duiding zorgen
- Makkelijk te plannen
  - Stelt een werkitem voor
  - Prioriteren, wijzigen, opsplitsen, verduidelijken,...

# User stories schrijven

- Een goede user story is
  - Onafhankelijk (**I**ndependent)
  - Onderhandelbaar (**N**egotiable)
  - Waardevol voor gebruiker en opdrachtgever (**V**aluable)
  - Schatbaar (**E**stimatable)
  - Klein (**S**mall)
  - Testbaar (**T**estable)
- Focus op 6 elementen => **INVEST**



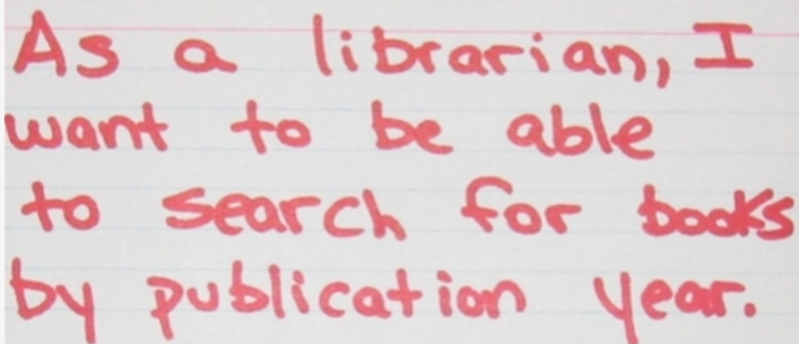
# Negotiable - Onderhandelbaar



- User Stories zijn geen contracten, ze beschrijven de requirements **zonder te specificeren**
- Start van een conversatie
  - Product Owners bepaalt richting (verwachtingen - wat)
  - Developers bepalen hoe er te geraken (specificaties - hoe)
- Mogelijks wijzigen User Stories doorheen de tijd
  - Meer kennis wordt vergaard doorheen het project
  - Stories worden gesplit
  - Stories worden onnodig
- Zo laat mogelijk details toevoegen

# Valuable - Waardevol

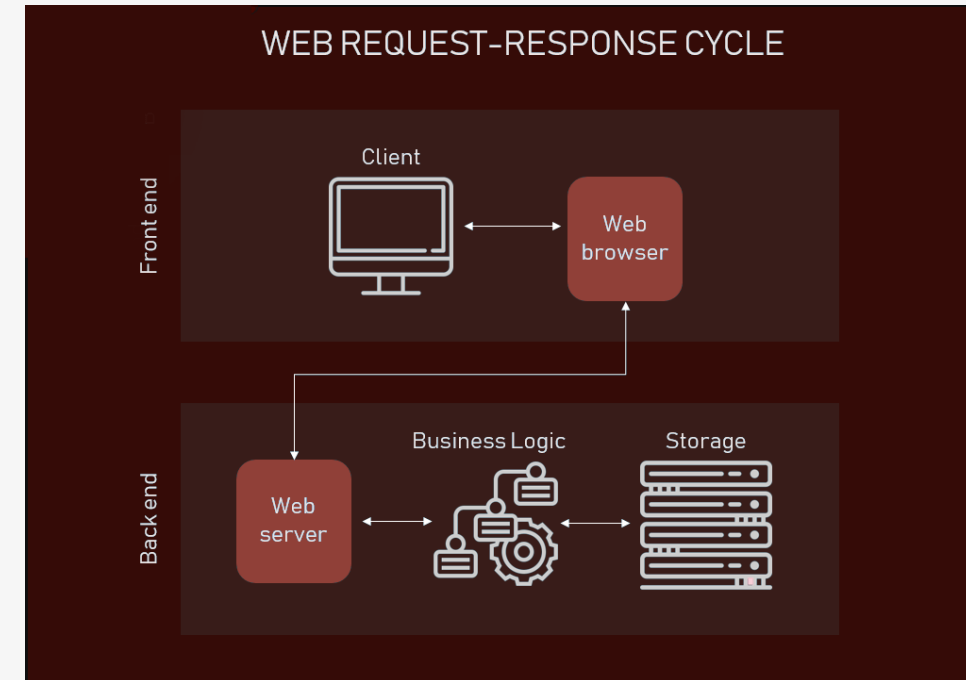
- User stories moeten een business waarde bevatten
  - Meer opbrengsten, lagere kosten, klanten aantrekken, personeel efficiënter maken, ...
  - Makkelijker als er een financieel voordeel aan kan gekoppeld worden -> Prioriteren wordt ook makkelijker
- Als er geen identificeerbaar voordeel is aan de story, waarom moet het dan gebouwd worden?
- “Gesloten” stories
  - Gesloten story eindigt met een user's goal (waarom)
  - Als user goal triviaal is, mag dit worden weggelaten



As a librarian, I want to be able to search for books by publication year.

# Valuable - Waardevol

- “Slice the cake”
  - Bij het schrijven van user stories aandacht dat alle lagen van de applicatie voorkomen
- Database tabellen aanmaken, business logica schrijven, UI... zijn onderdelen van een user story, geen user story op zich



# Valuable - Waardevol



Als een klant  
Wil ik producten kunnen filteren  
op prijs  
Zodat ik snel de producten kan  
vinden binnen mijn prijs klasse

- Database model aanpassen
- DB Query voor filteren
- Business logica voor filter
- UI voor filter

# User stories

- Zijn niet gebonden aan IEEE-guidelines
- Zijn geen use cases
  - Scope
  - Volledigheid
  - Levensduur
  - Doel

# Stories, Epics en tasks

- Soms is er nood aan meerdere niveau's van detail
  - Business niveau vs technisch niveau

Epics

Taken uitvoeren

Stories

Als lid wil ik een overzicht van alle taken die ik moet uitvoeren

Als lid wil ik de details zien van een taak die ik moet uitvoeren

'''

Tasks

GET request op basis van lid

Taken Business Logic

Overzicht taken view

Test schrijven

'''

# Epics

- Groot stuk functionaliteit die business wil hebben en waarde oplevert
- Wordt opgeleverd aan de hand van kleinere user stories, gespreid over meerdere iteraties
- Format van user stories moeten niet gevolgd worden
  - Mag (moet) groot zijn
  - Belangrijk om toch te weten: **wie, wat en waarom?**

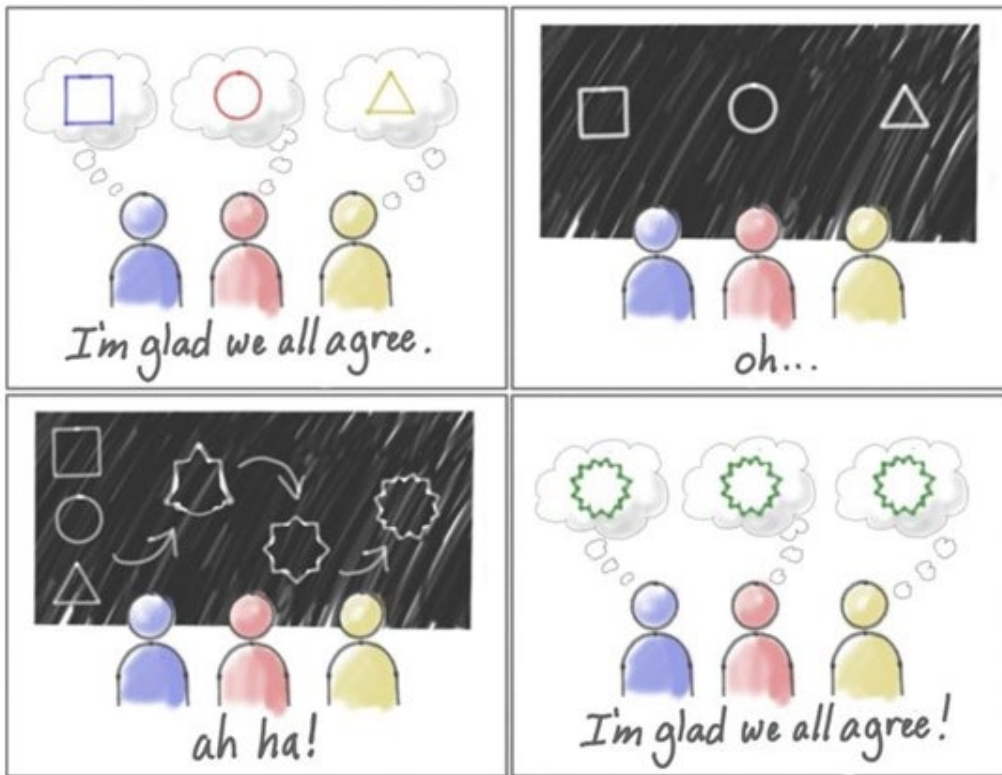
# Tasks

- Story kan onderverdeeld worden in uitvoerbare taken
- Geschreven door het Dev Team, voor het Dev Team
  - Mogen dus technische zaken bevatten
  - Hoeft geen directe business value te hebben
  - Moeten niet onafhankelijk te zijn
  - Hoeven niet verticaal (end-to-end) te zijn
- Input van DevOps is belangrijk hierbij
  - Nodige systemen/hardware
  - Beperkingen naar security toe
  - Aparte technical tasks voor DevOps

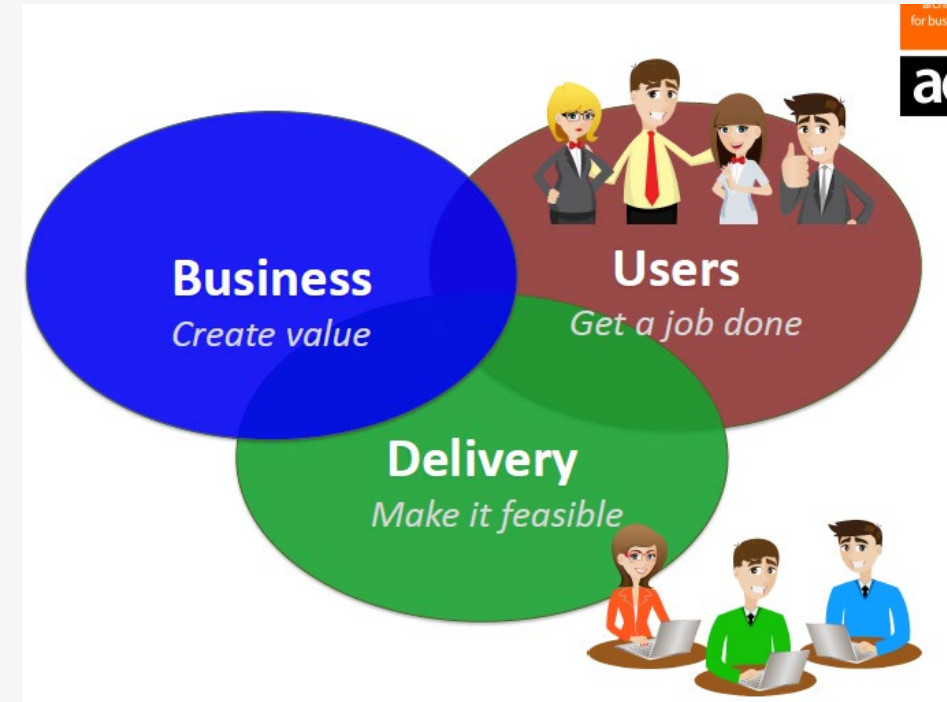


# Defining Story Writing

# Belang van story writing



Jeff Patton – User Story Mapping



Shared understanding

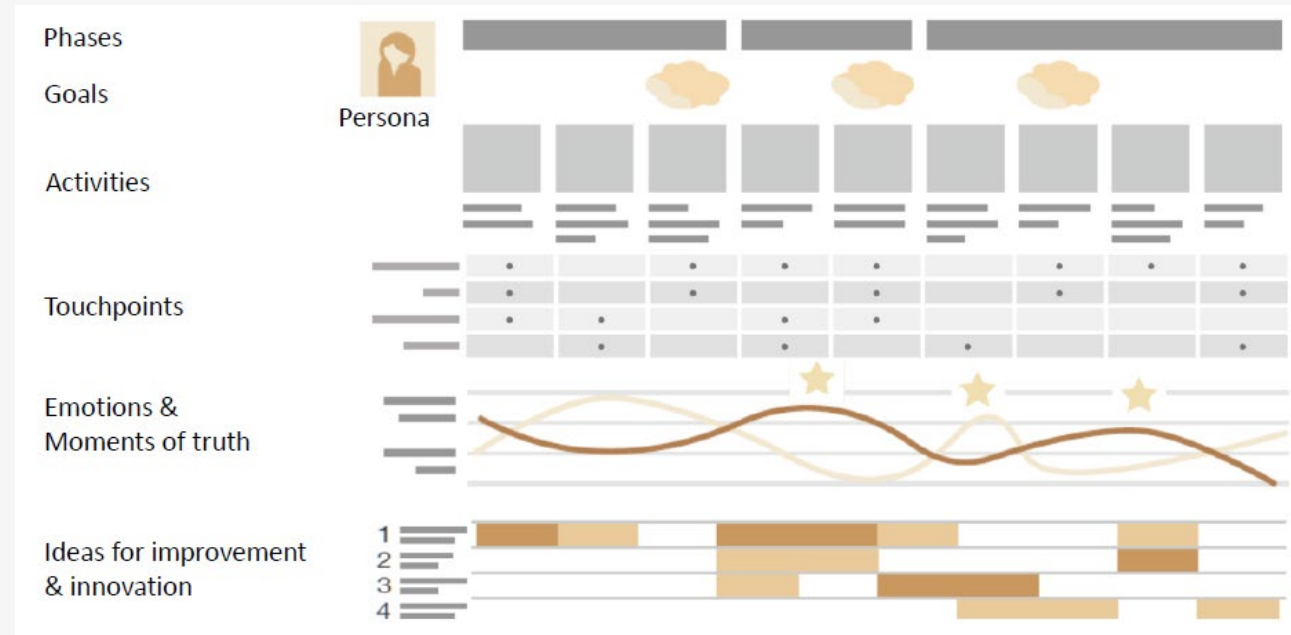
HO  
GENT

# **User Stories ontdekken**

Customer Journey – Story Mapping

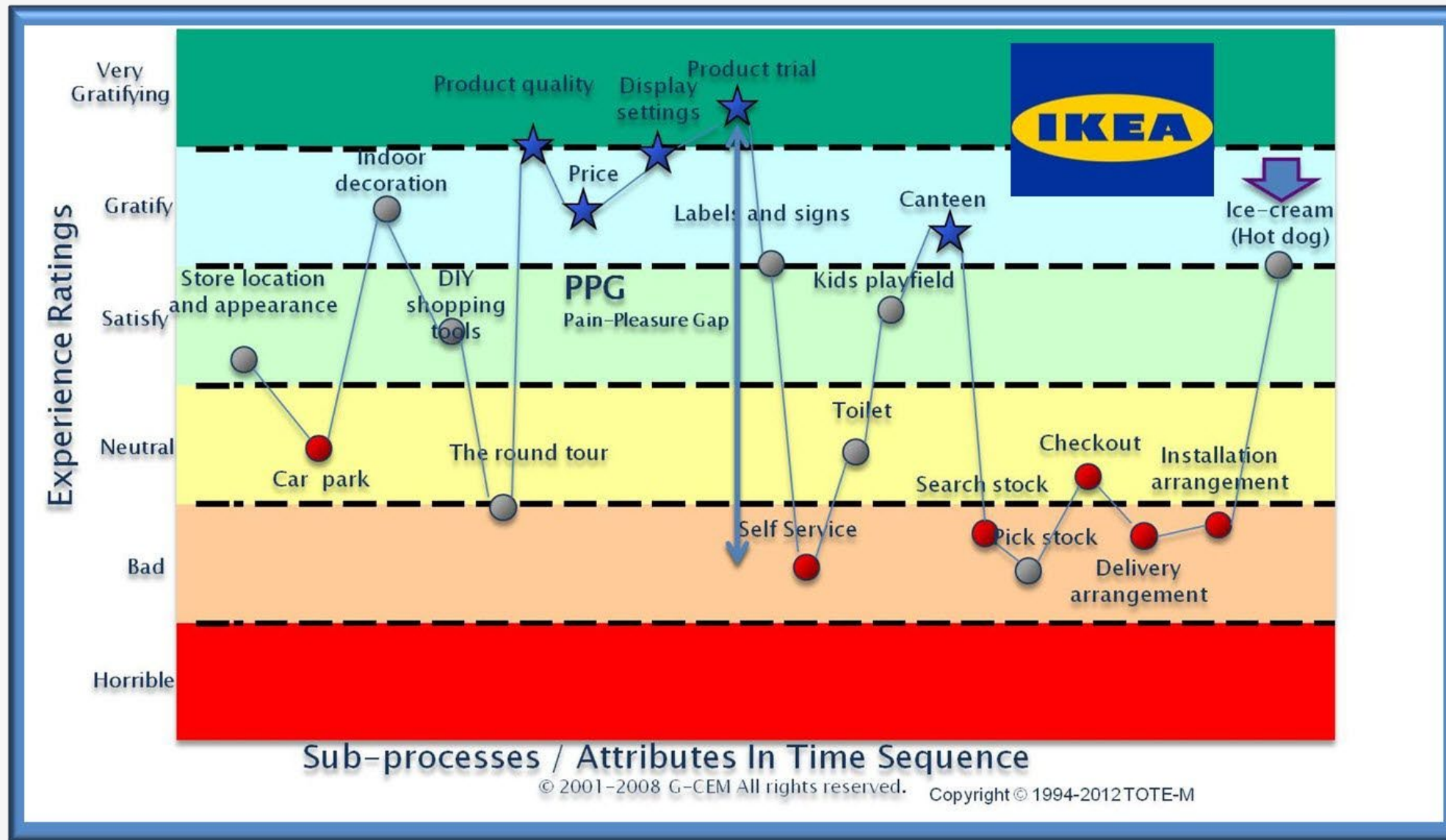
# Customer journey

- Visualiseert de ervaring van de klant van begin tot einde aangaande doelen, fases, activiteiten, contactpunten en emoties
  - Oorsprong in marketing
- Doel: Inzicht krijgen in het gedrag van de klant
  - Toont mogelijks **problemen**
  - Identificeert **opportuniteiten**
  - Aligneert de organisatie



# Customer journey

- Stappenplan
  - Creëren van persona's
  - Identificeren van de verschillen fases, elk met hun doel en beschrijving van de activiteit
  - Identificeer de contactpunten tussen klant en bedrijf
  - Beschrijf het gevoel van de klant tijdens de actie
  - Identificeer de belangrijkste acties
  - Vind ideeën voor verbetering/innovatie



# Customer Journey IKEA

**HO  
GENT**



# Customer journey naar Story Map



1

Decide on what to investigate:  
existing/new product or service,  
touch points...



2

Examine the **Customer Journey Maps**  
and detect **opportunities** and/or **threats**

3

Describe your **hypothesis**:  
Because...  
We might...  
Which will lead to ...

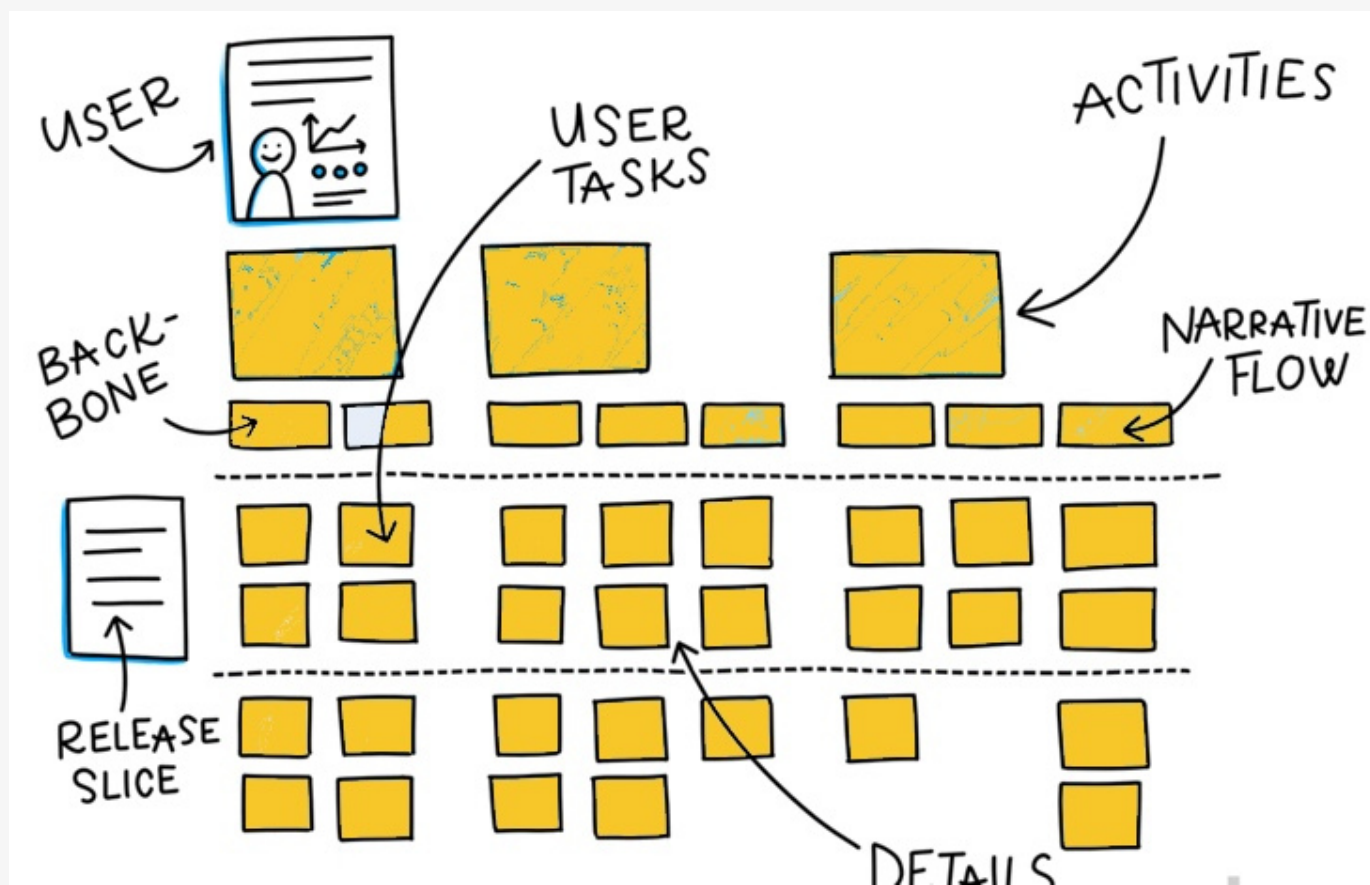
4

Identify and source the **capabilities** necessary to  
design/deliver the solution  
("who to invite to your story mapping workshops?")





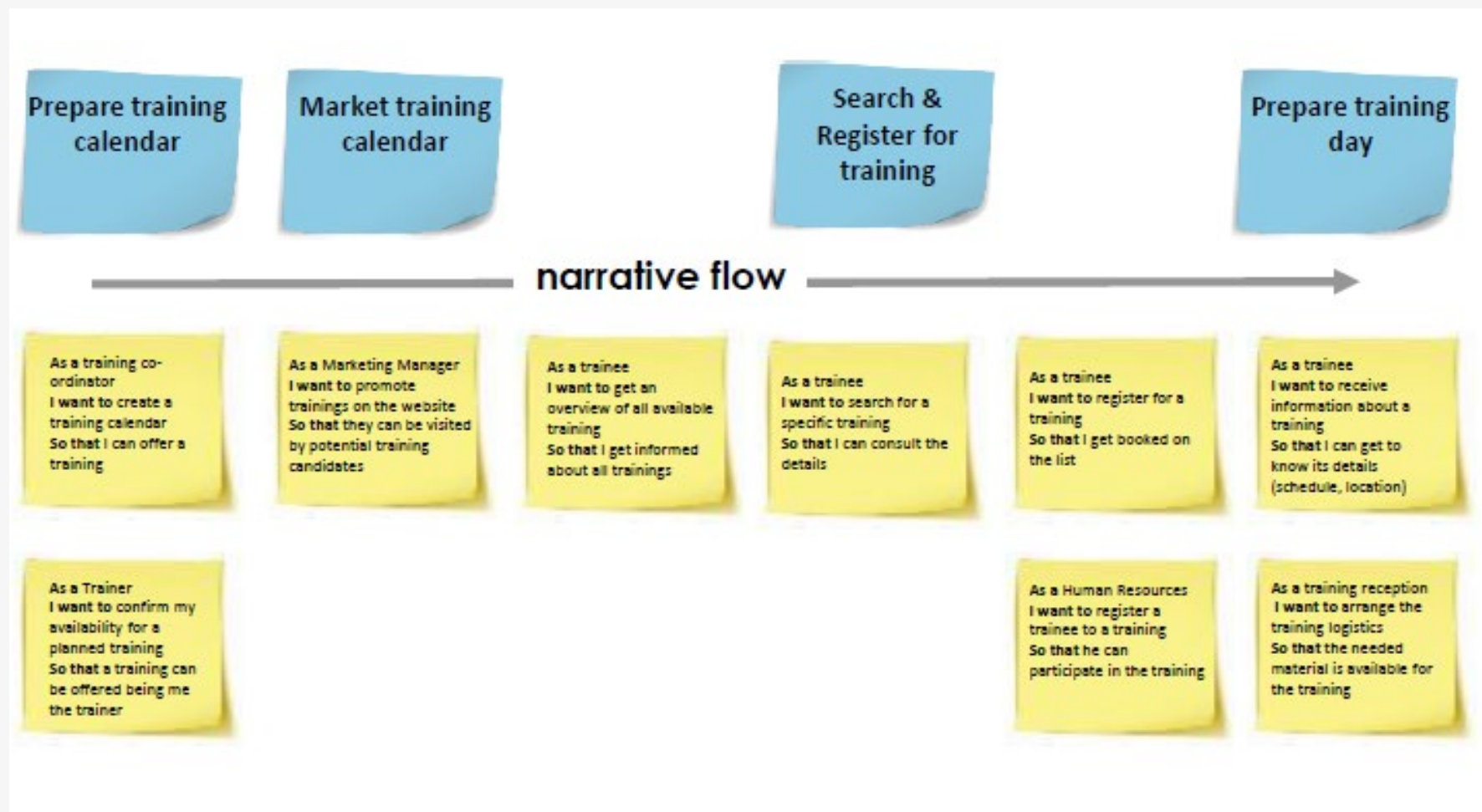
# Story Map



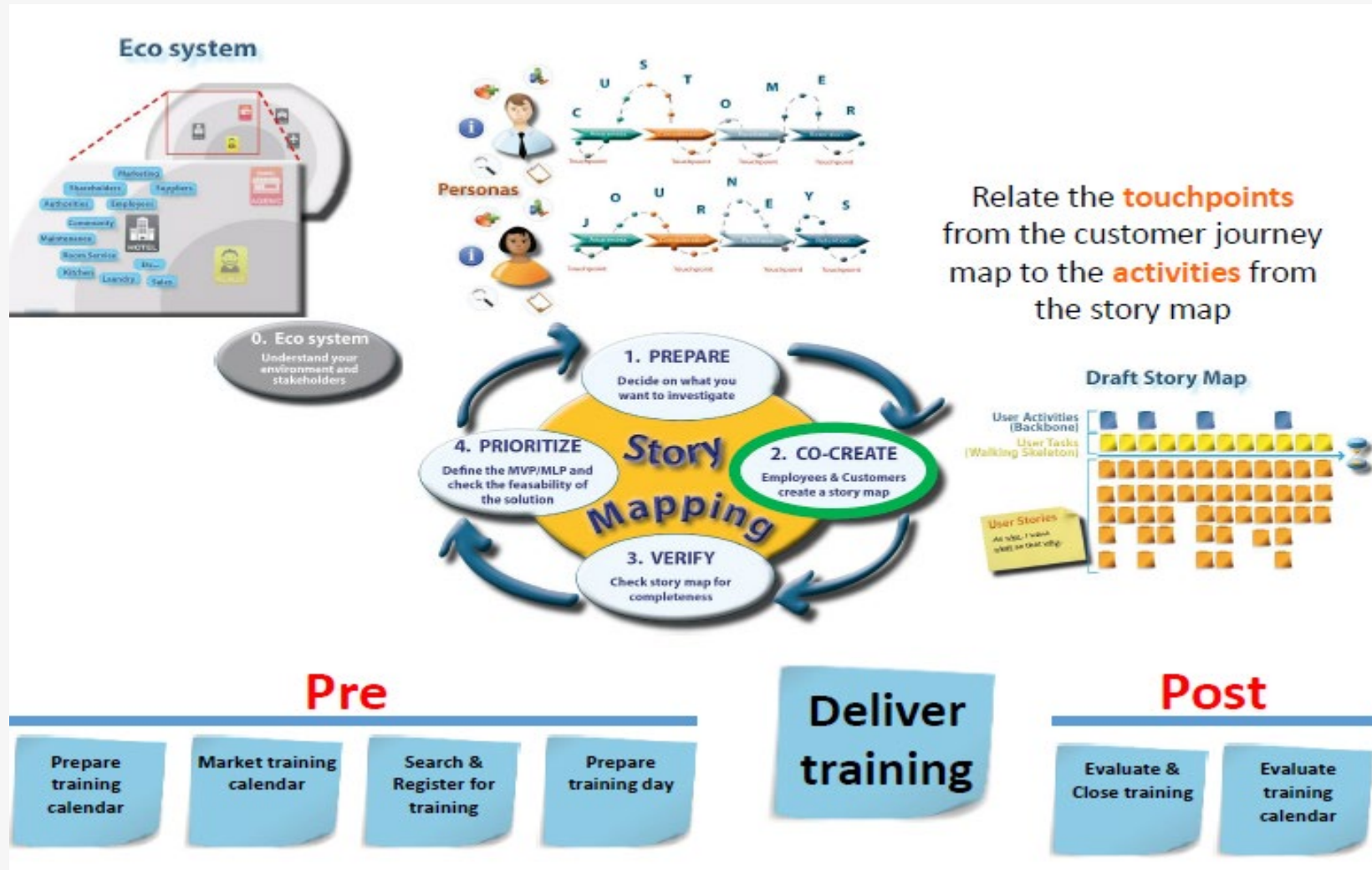
# Story map

- Een visuele oefening om shared understanding te creëren
  - Business en IT zijn hierbij aanwezig
- Horizontaal wordt een verhaal verteld, die activiteiten voorstellen in een product/site
- Iedere activiteit wordt verticaal verder uitgewerkt in details/verschillende taken
  - User story formaat kan hiervoor worden gebruikt
- Als alle activiteiten ontdekt zijn, kan er geprioriteerd worden
  - Vervolgens kunnen verschillende MVP's vastgesteld worden
- Is de start van een Product Backlog

# Story map

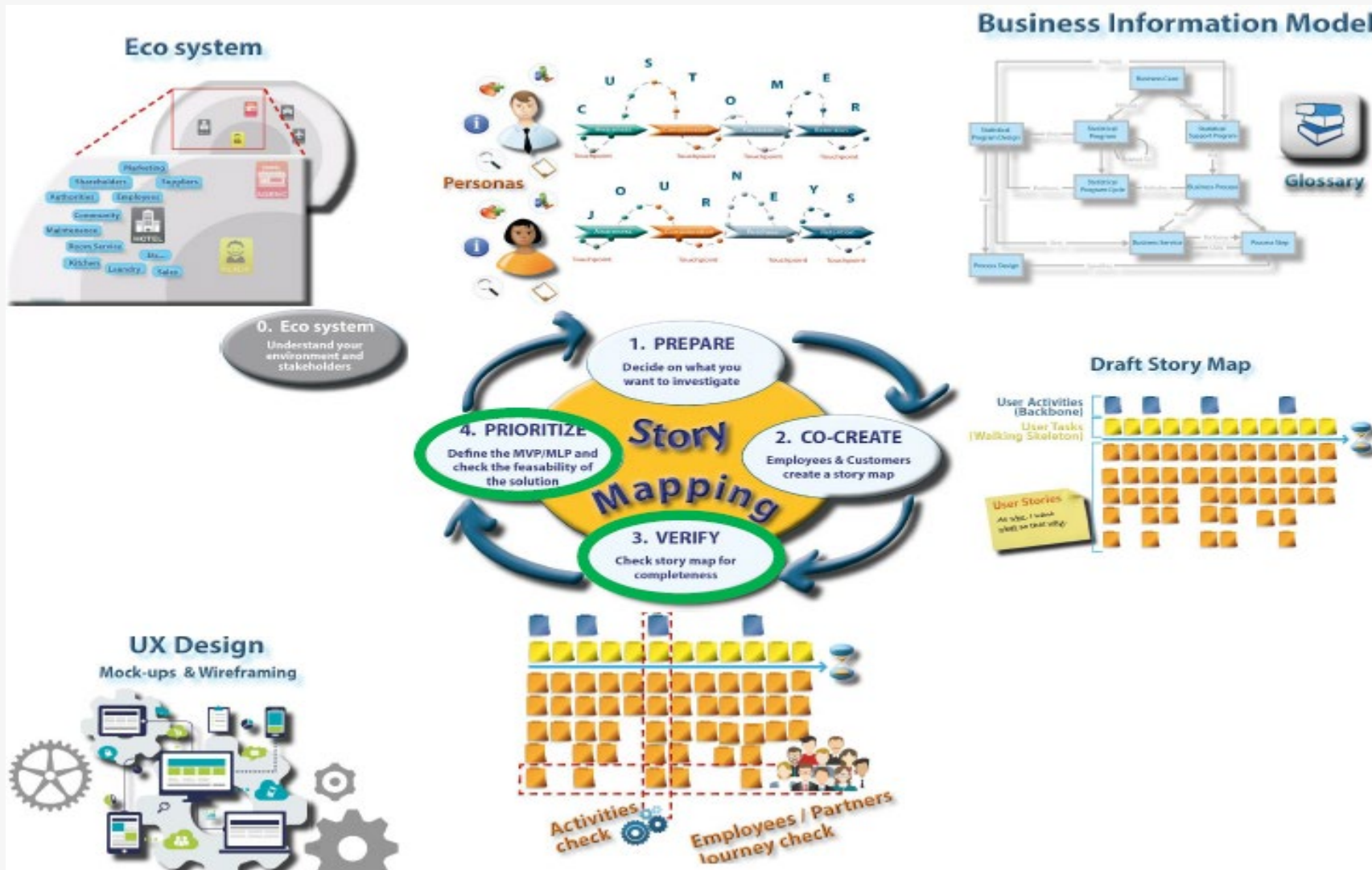


# Story map





# Story map - Verifiëren

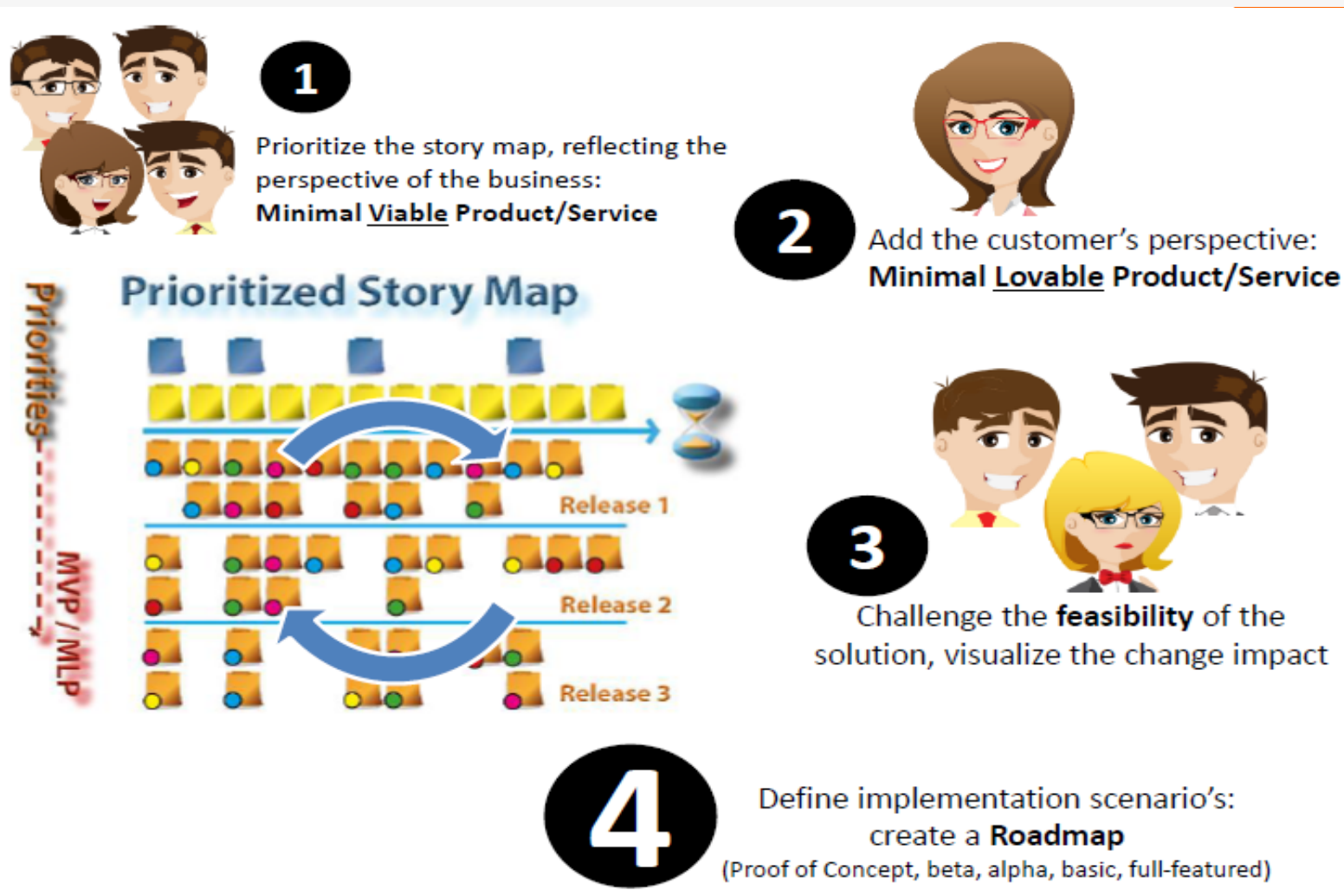


# Story map - Prioriteren

- Prioriteren is belangrijk
  - Niet alles kan even belangrijk zijn
- User feedback is belangrijk na iedere release



# Story map - Prioriteren



# Prioriteren met MoSCoW

Mo

## MUST HAVE

Essentiële elementen die onmogelijk kunnen ontbreken

S

## SHOULD HAVE

Belangrijke elementen maar minder van essentieel belang

Co

## COULD HAVE

Leuke extraatjes als er tijd en geld is, nice-to-haves

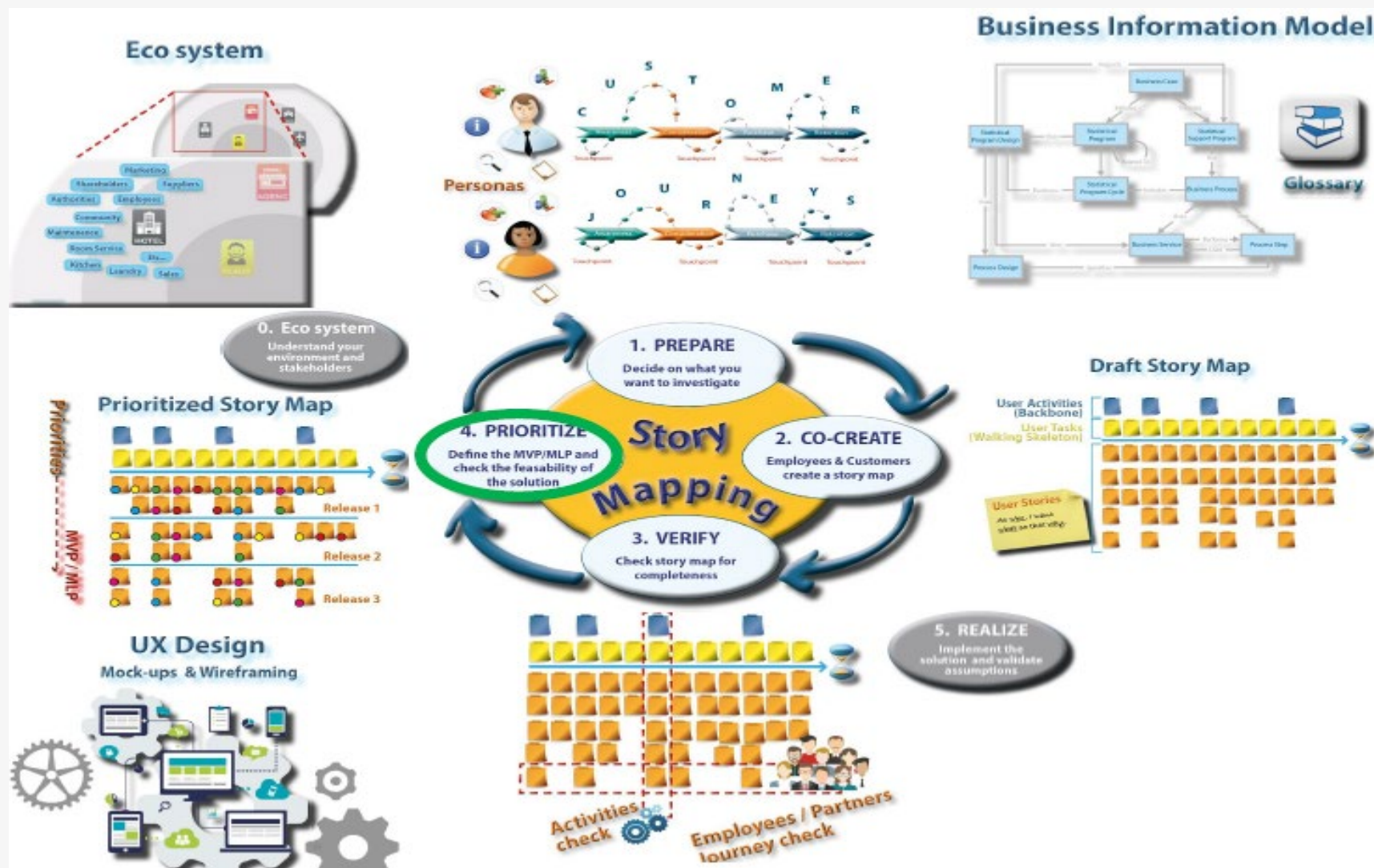
W

## WON'T HAVE

Elementen die geen waarde toevoegen

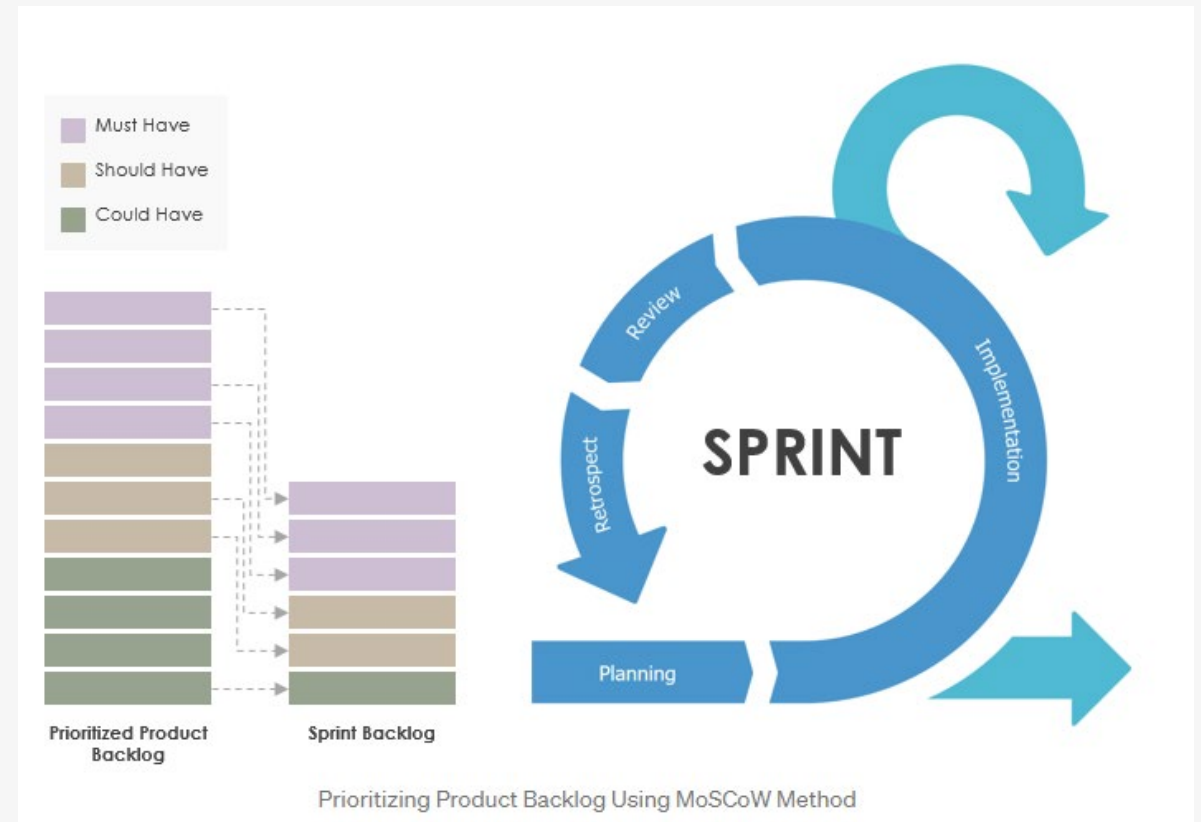


# Story map



# Product Backlog

- Het resultaat van een story mapping is een geprioriteerde product backlog, klaar om te realiseren
- Maar dit kan nog wijzigen doorheen het project
  - Prioriteiten wijzigen
  - Must have's kunnen should/Could have's worden en omgekeerd
  - Na iedere iteratie/feedback weten we beter wat er gewenst is.



# Tips

- <https://www.jpattonassociates.com/backlog-grooming-bugs-me/>