

*TMCG*

---

# USER NEEDS → LOVED PRODUCTS

---

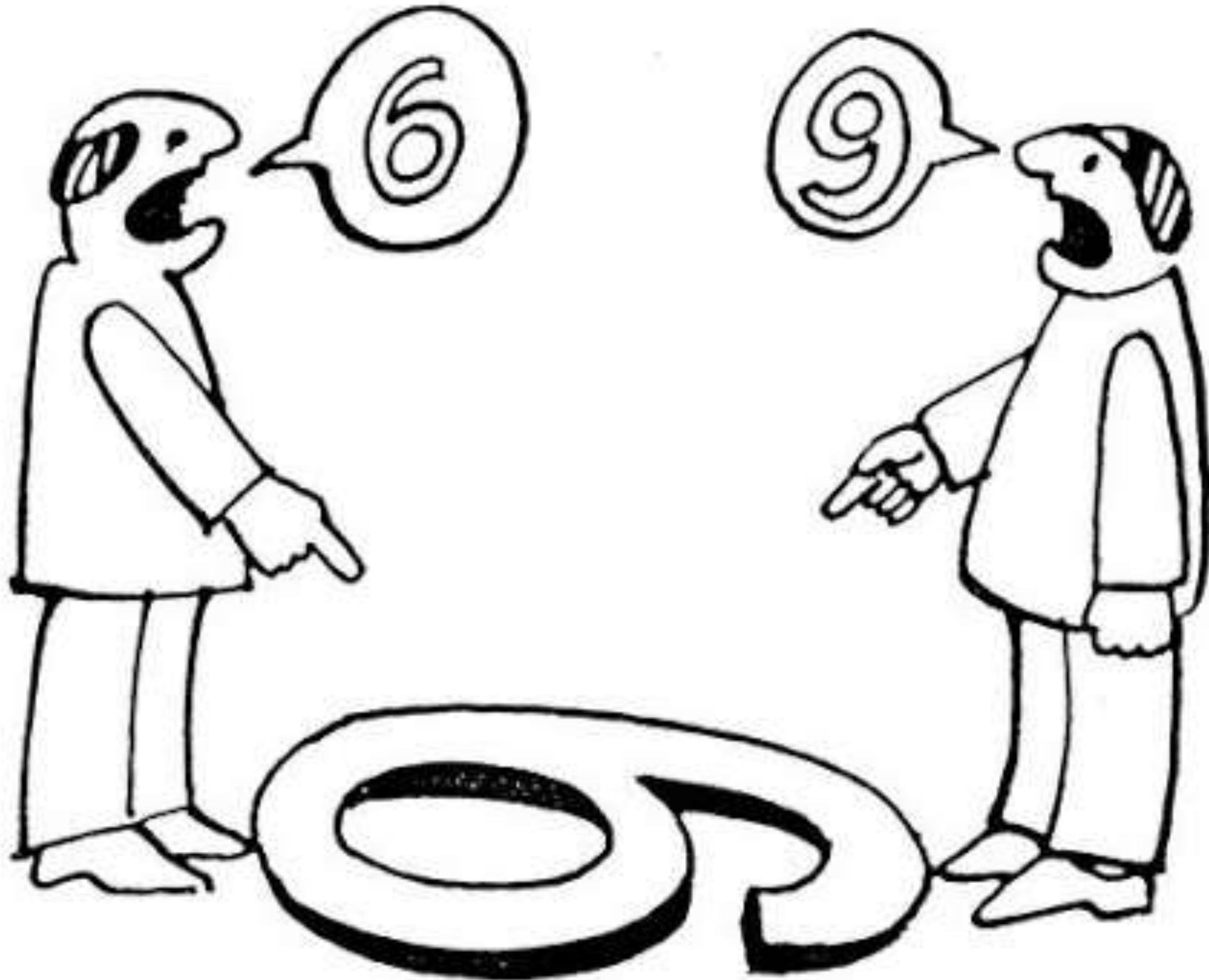
*Stephen Senkomago Musoke*

*@ssmusoke*

*February 2020*

WHAT DO WE WANT TO AVOID HY ARE WE HERE?WHY ARE WE HERE

---



# YOUR TYPICAL PROJECT

---

- ❑ After a long protracted negotiation
  - ❑ Fixed cost
  - ❑ Fixed timeline
  - ❑ Defined scope (it is obsolete by now)
- ❑ What happens along the way
  - ❑ Requirements keep changing along the way
  - ❑ Stakeholder influencing on scope
  - ❑ Timelines are too short
  - ❑ No single product owner
  - ❑ Tech stack how do deliver on the promises?

# PROJECT KICKOFF MEETING

---

- Define roles and responsibilities
- When showcases will happen
- Whom to ask questions during development/delivery
- Who can make decisions
- Communication channels – project, admin
- A technical team representative, needs to attend to understand the business drivers

# REQUIREMENTS DOCUMENTATION

---

- Use cases/User Stories – Writing Effective Use Cases by Alastair Cockburn
- Components
  - Scope
  - Actors/Personas
  - Goals
  - Main Story
  - Alternate Paths
  - Acceptance Criteria
  - Technical Implementation approach

# WHY USE CASES/USER STORIES

---

- ❑ Driven by an end-user goal, driving business value of the solution
- ❑ Contain business language – communication across multiple persons
- ❑ Alternate paths drive further analysis
- ❑ Acceptance criteria simplify quality assurance
- ❑ Sizing of user stories – T-shirt sizes are common aid estimation of effort, hence placing into sprints
- ❑ Starting point for end-user documentation

# WHAT TOOLS CAN I USE?

---

- ❑ The best tool for the job is the one u know how to use
- ❑ However there are some good ones
  - ❑ Asana
  - ❑ Trello
  - ❑ GitHub/GitLab Issues
- ❑ These allow you to comment and include other assets such as documents, images, links etc. Use the best tool to illustrate that documentation
- ❑ Move tasks/cards/issues across multiple stages/states

# WHAT TOOLS CAN I USE?

---

- ❑ The best tool for the job is the one u know how to use
- ❑ However there are some good ones
  - ❑ Asana
  - ❑ Trello
  - ❑ GitHub/GitLab Issues
- ❑ These allow you to comment and include other assets such as documents, images, links etc. Use the best tool to illustrate that documentation
- ❑ Move tasks/cards/issues across multiple stages/states



# THE SHOWCASE

---

- ❑ Regular touch point with client
- ❑ Status check on delivery
- ❑ Progress update on the project
- ❑ Team catchup – end of sprint

# TIPS AND TRICKS

---

- Use Prototyping tools to give your users a sense of what the final solution will feel like and work
- Spend time collaborating with your clients in a manner and language that they understand – remember they have other full-time jobs not just your project
- Discipline – keep time, promises, manage expectations, pay attention
- Document, document, document – in a way that makes meaning to your clients, stakeholders & end-users

*On time is late!!!*

# TECH AIDS TO DELIVERY

---

- ❑ Automate & simplify deployment – to dev, staging and demo sites
- ❑ Fanatical, Meticulous version control – Trunk based development, GitFlow, short leaved feature branches
- ❑ Testing – unit, integration, stress, load, Travis CI
- ❑ Security – the price is too high at the end
- ❑ Document, document, document
- ❑ Use prebuilt frameworks, and libraries
- ❑ Contribute back to #OpenSource

*Every problem is a special case of a more general problem and you are not the first one to solve it*

# CHALLENGES

---

- ❑ Draft fatigue – too many back and forth cycles
- ❑ As requirements change, then goals and deliverables are forgotten
- ❑ Team dynamics – changing requirements are stressful
- ❑ Technology tools & choices – do not try to re-invent the wheel as much as you can, solve the problem at hand + know how to use your tools

*If I am given 4 hours to cut down a tree, I will  
spend 3 hours sharpening my axe  
~ Abraham Lincoln*

# READING LIST – BUSINESS & TECH

---

- ❑ Martin Fowler – Refactoring, Patterns of Enterprise Architecture & <http://martinfowler.com>
- ❑ Andy Hunt/Dave Thomas – The Pragmatic Programmer
- ❑ Basecamp blog - <https://m.signalvnoise.com/>
- ❑ Blue Ocean Strategy
- ❑ Clean Code
- ❑ Atomic Habits – James Clear
- ❑ Imposters Handbook by Rob Conery (I am reading it now)
- ❑ Blogs/Articles in whatever language or problem domain you are working in

# IN CLOSING

---

*DO NOT BE TOO BUSY*

*Enjoy your project*

*Keep learning*

*Read*

*Stand on the shoulders of Giants*

*Share your stories with others*

*Discipline, Discipline, Discipline*

*Document, Document, Document*

*AND MOST OF ALL*

*Deliver and satisfy your clients*

# THANK YOU

---

*For questions or suggestions*

*@ssmusoke*

*<http://ssmusoke.com>*

---