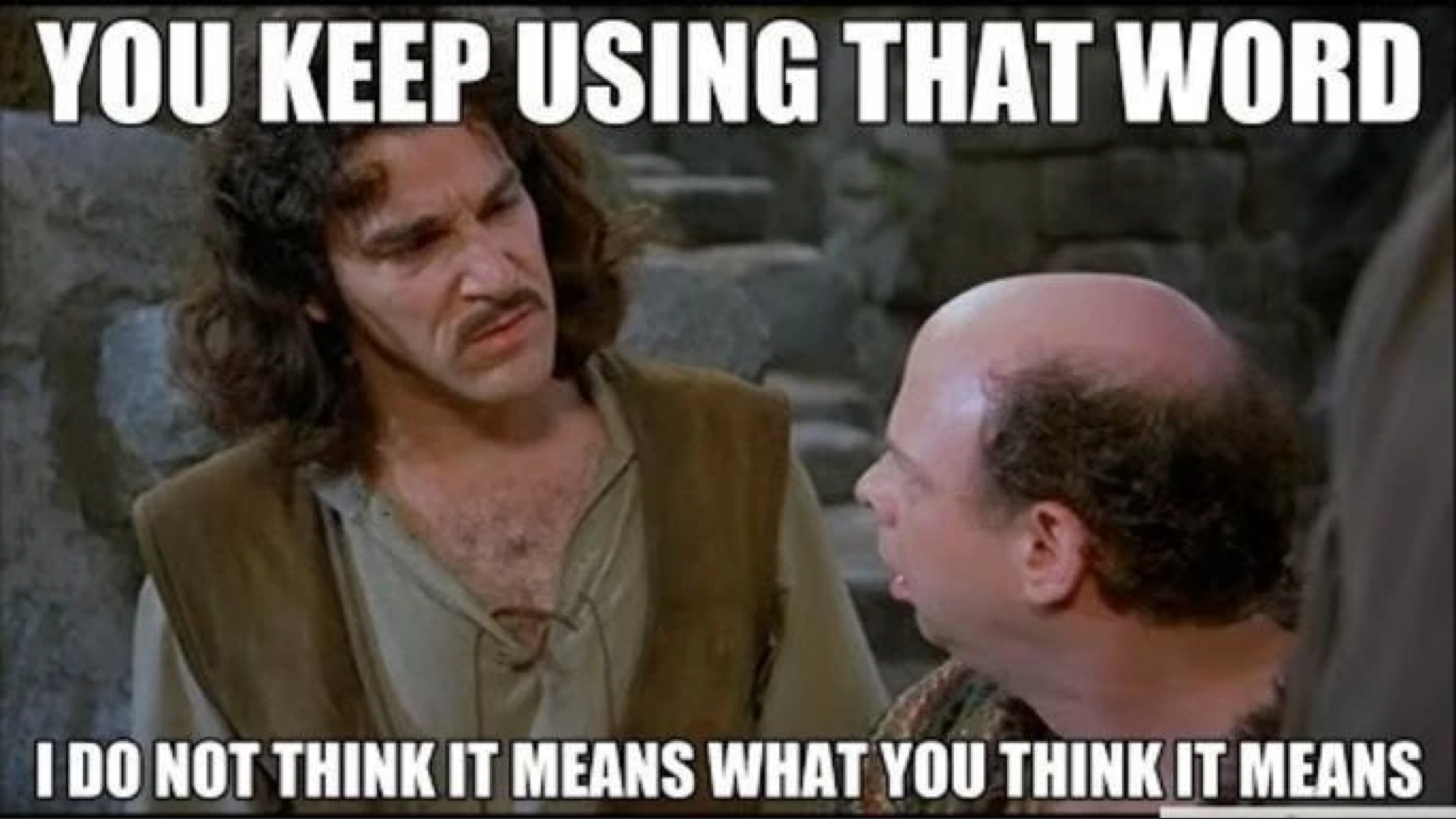
Information Architecture For The Soul

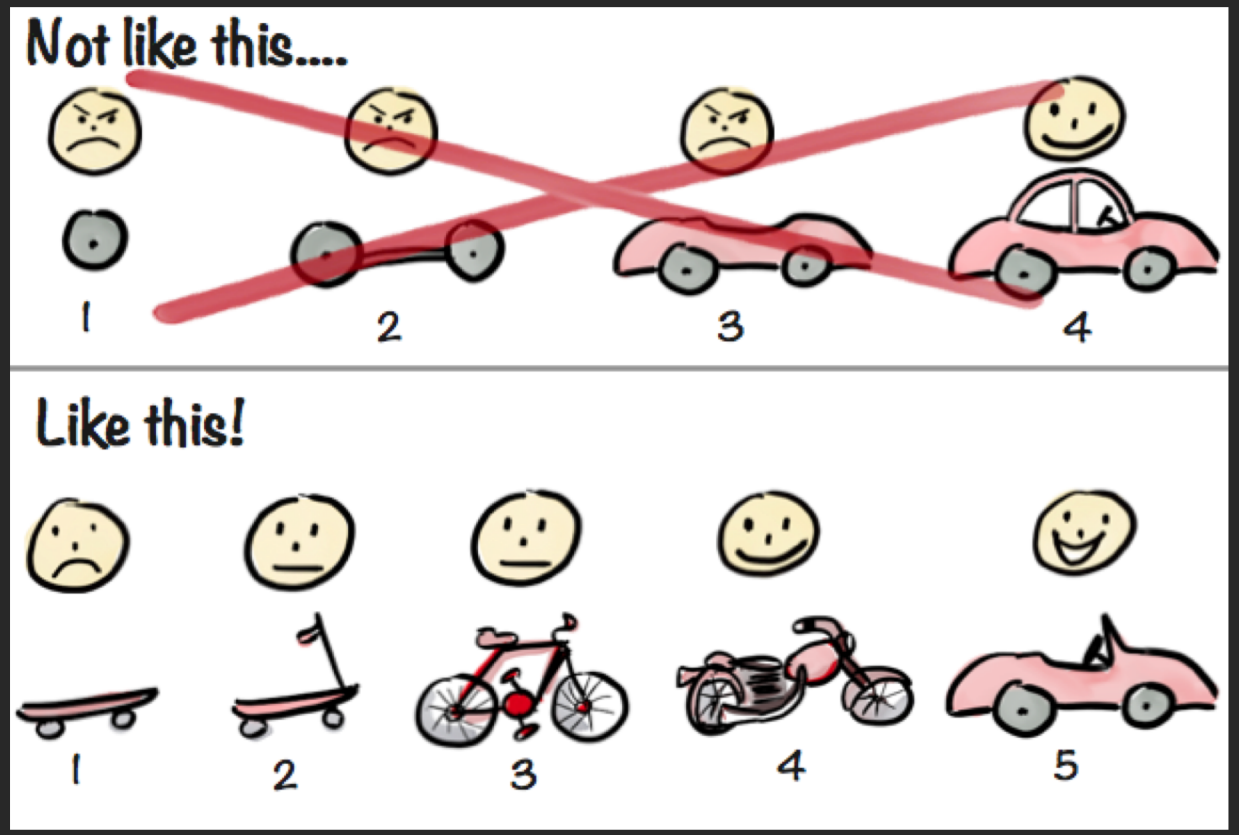
How to not build the wrong thing.

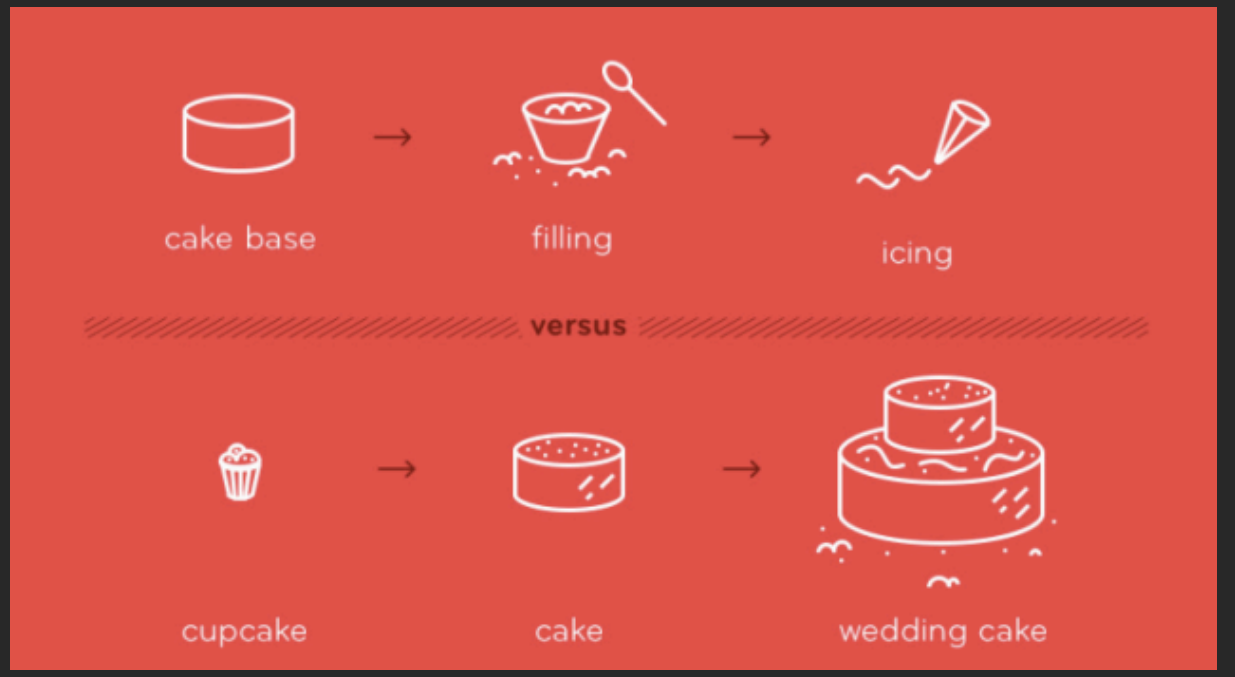
Ryan Albertson **me@ryan.ws**

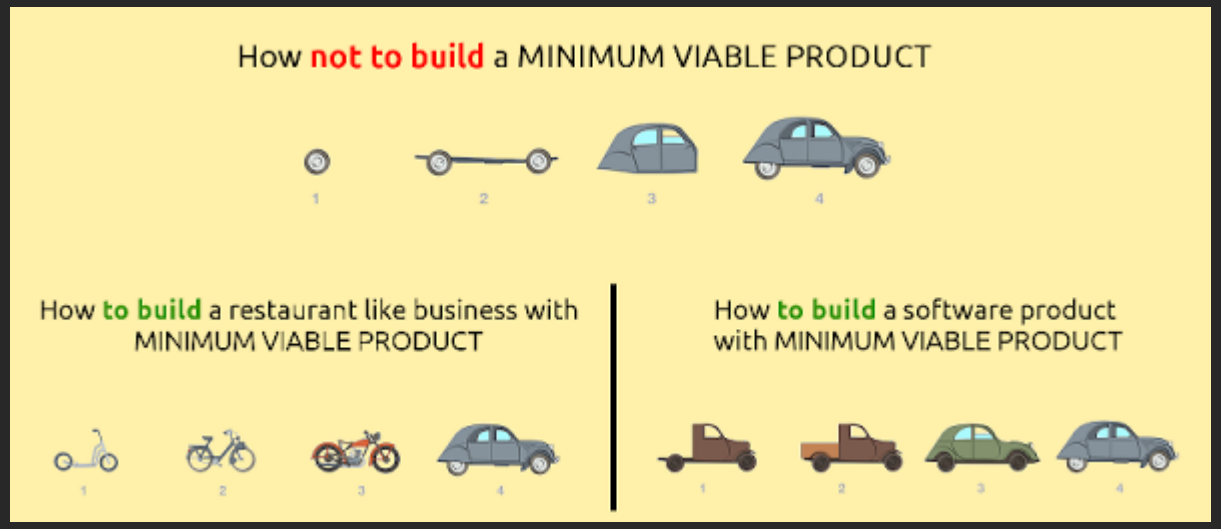

grb

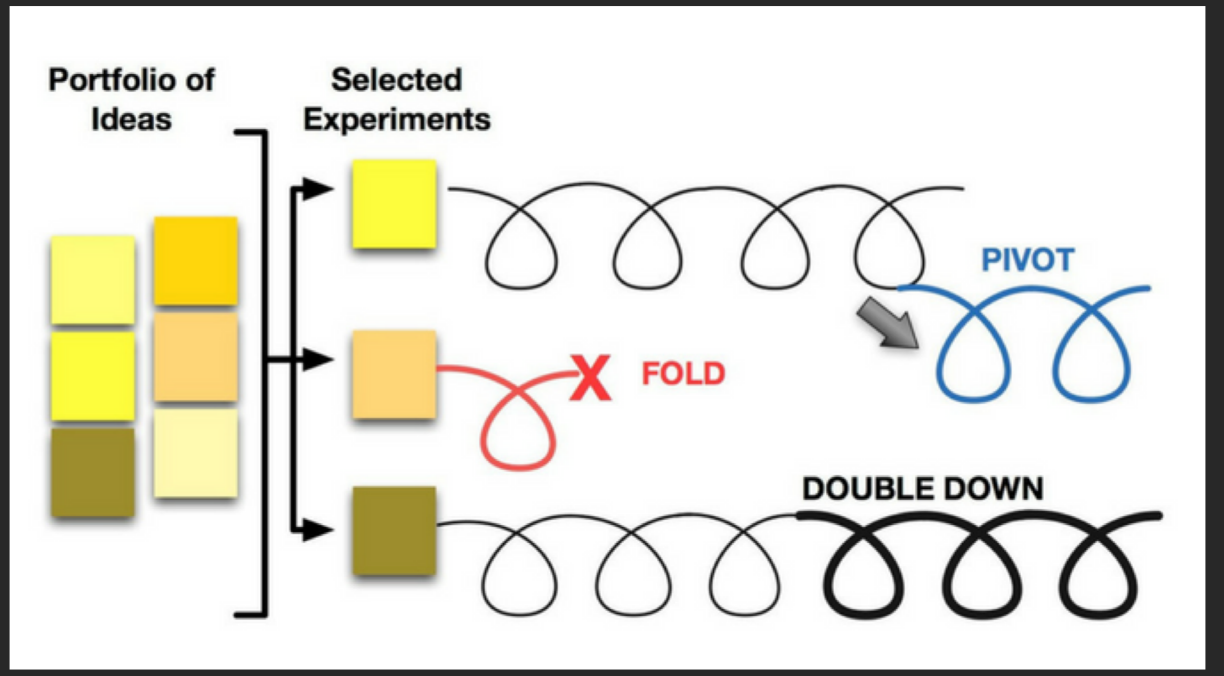
What is an MVP?


YOU KEEP USING THAT WORD
I DO NOT THINK IT MEANS WHAT YOU THINK IT MEANS

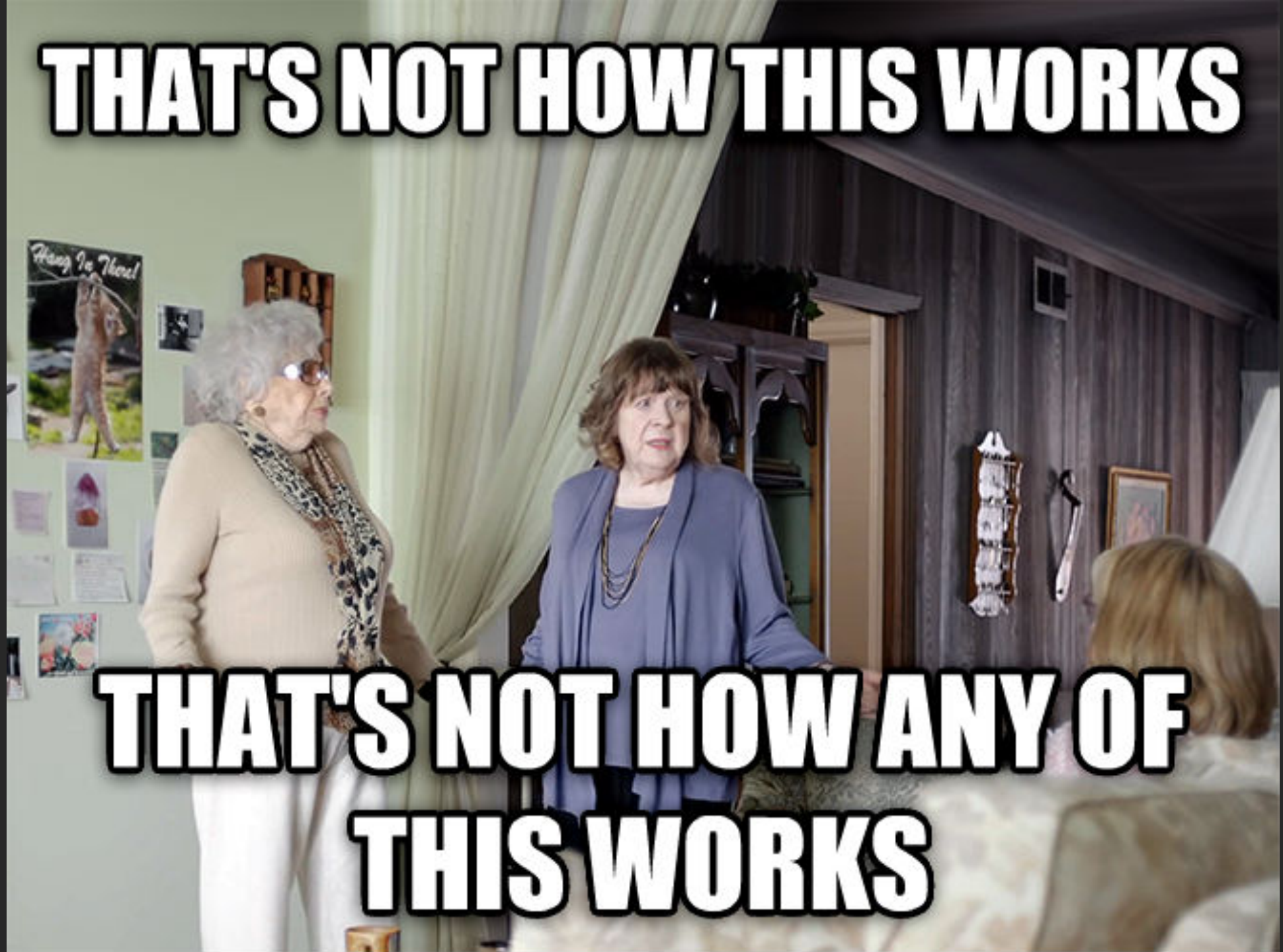

Not like this....
1
2
3
4
Like this!
1
2
3
4
5


cake base
filling
icing
versus
cupcake
cake
wedding cake


How not to build a MINIMUM VIABLE PRODUCT
1
2
3
4
How to build a restaurant like business with MINIMUM VIABLE PRODUCT
1
2
3
4
How to build a software product with MINIMUM VIABLE PRODUCT
1
2
3
4


Portfolio of Ideas
Selected Experiments
PIVOT
X
FOLD
DOUBLE DOWN

JAN | FEB | MAR | APR | MAY | JUN | JUL | AUG | SEP


THAT'S NOT HOW THIS WORKS
Hang In There!
THAT'S NOT HOW ANY OF THIS WORKS

Minimum viable product

From Wikipedia, the free encyclopedia

This article is about the product development strategy. For the pilot episode of Silicon Valley, see Minimum Viable Product.

A **minimum viable product** (**MVP**) is a product with just enough features to satisfy early customers, and to provide feedback for future product development.[1][2]

Gathering insights from an MVP is often less expensive than developing a product with more features, which increases costs and risk if the product fails, for example, due to incorrect assumptions. The term was coined and defined by Frank Robinson about 2001,[3] and popularized by Steve Blank and Eric Ries.[4][5][6][7] It may also involve carrying out market analysis beforehand.

Contents [hide]

Description [edit]

A minimum viable product has just enough core features to effectively deploy the product, and no more. Developers typically deploy the product to a subset of possible customers—such as early adopters thought to be more forgiving, more likely to give feedback, and able to grasp a product vision from an early prototype or marketing information. This strategy targets avoiding building products that customers do not want and seeks to maximize information about the customer per amount of money spent.

"The minimum viable product is that version of a new product a team uses to collect the maximum amount of validated learning about customers with the least effort."[2] The definition's use of the words maximum and minimum means it is not formulaic. It requires judgement to figure out, for any given context, what MVP makes sense. Due to this vagueness, the term MVP is commonly used, either deliberately or unwittingly, to refer to a much broader notion ranging from a rather prototype-like product to a fully-fledged and marketable product.[8]

An MVP can be part of a strategy and process directed toward making and selling a product to customers.[9] It is a core artifact in an iterative process of idea generation, prototyping, presentation, data collection, analysis and learning. One seeks to minimize the total time spent on an iteration. The process is iterated until a desirable product/market fit is obtained, or until the product is deemed non-viable.

Steve Blank typically refers to minimum viable product as minimum feature set.[10][11]

JAN | FEB | MAR | APR | MAY | JUN | JUL | AUG | SEP

JAN | FEB | MAR | APR | MAY | JUN | JUL | AUG | SEP

MVP

Who are you?

Are you the person on the team that…

Whiteboards the problems out.
Has The Actual Final Say. (Business Edition.)
Figures out your CSS.
Understands the legacy code.
Has the usage data.
Understands the business better than everyone else.
Writes most of your tests.
Makes your comps/wires/design documentation.
Writes requirements.
Has The Actual Actual Final Say. (Technology Edition.)
Cleans up the backlog.
Writes user stories.
Decides when the due date ***actually*** is.
Talks to the users.

Are you the person on the team that…

Whiteboards the problems out.
Has The Actual Final Say. (Business Edition.)
Figures out your CSS .
Understands the legacy code.
Has the usage data.
Understands the business better than everyone else.
Writes most of your tests.
Makes your comps/wires/design documentation.
Writes requirements.
Has The Actual Actual Final Say. (Technology Edition.)
Cleans up the backlog.
Writes user stories.
Decides when the due date ***actually*** is.
Talks to the users.

Are you the person on the team that…

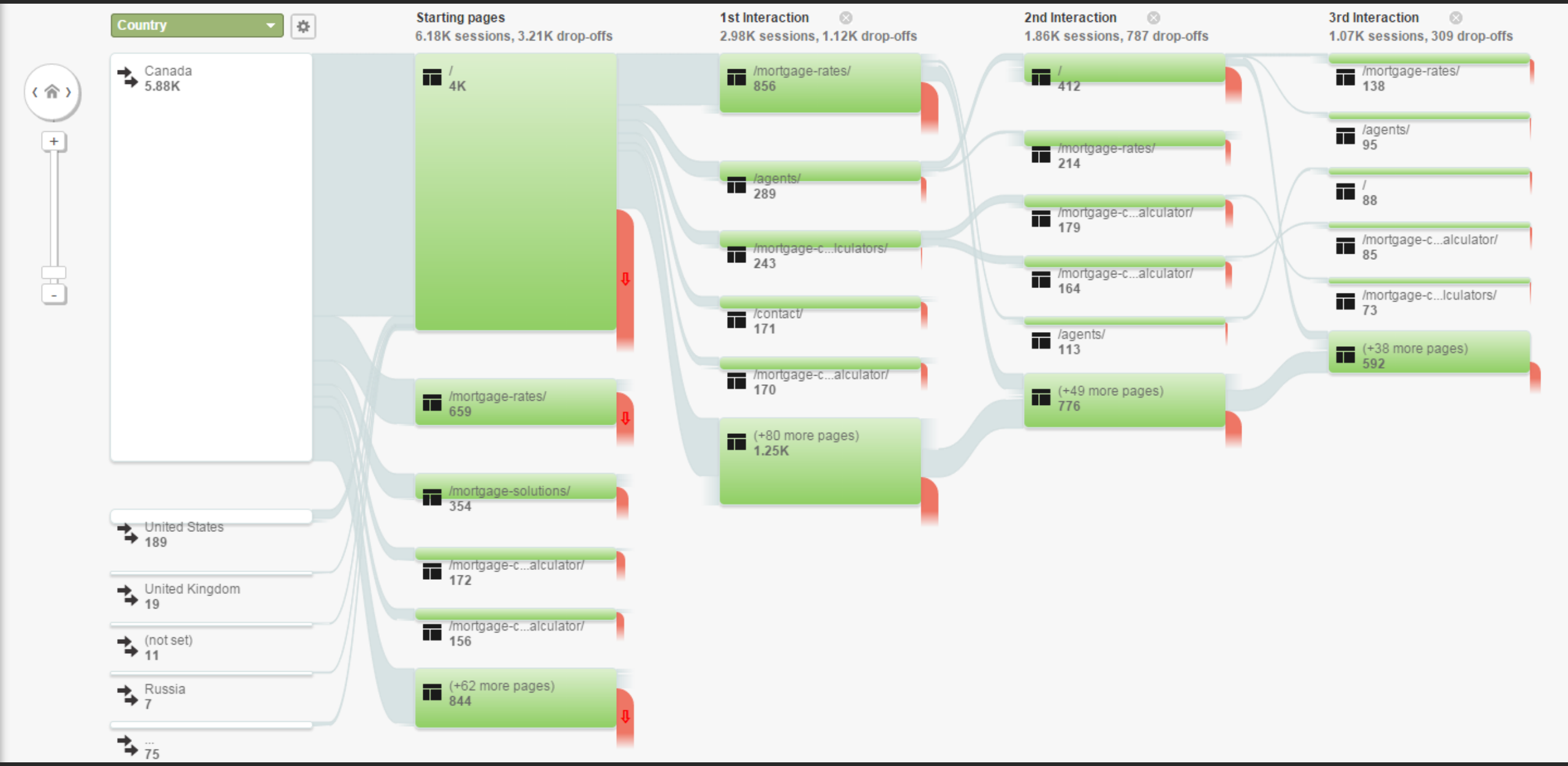
Whiteboards the problems out.
Has The Actual Final Say. (Business Edition.)
Figures out your CSS .
Understands the legacy code.
Has the usage data.
Understands the business better than everyone else.
Writes most of your tests.
Makes your comps/wires/design documentation.
Writes requirements.
Has The Actual Actual Final Say. (Technology Edition.)
Cleans up the backlog.
Writes user stories.
Decides when the due date ***actually*** is.
Talks to the users.

How did you get here?

And do you know where you are going?

Hard Data

Soft Data


Country
Starting pages
6.18K sessions, 3.21K drop-offs
1st Interaction
2.98K sessions, 1.12K drop-offs
2nd Interaction
1.86K sessions, 787 drop-offs
3rd Interaction
1.07K sessions, 309 drop-offs
Canada
5.88K
United States
189
United Kingdom
19
(not set)
11
Russia
7
...
75
/
4K
/mortgage-rates/
659
/mortgage-solutions/
354
/mortgage-c...alculator/
172
/mortgage-c...alculator/
156
(+62 more pages)
844
/mortgage-rates/
856
/agents/
289
/mortgage-c...lculators/
243
/contact/
171
/mortgage-c...alculator/
170
(+80 more pages)
1.25K
/
412
/mortgage-rates/
214
/mortgage-c...alculator/
179
/mortgage-c...alculator/
164
/agents/
113
(+49 more pages)
776
/mortgage-rates/
138
/agents/
95
/
88
/mortgage-c...alculator/
85
/mortgage-c...lculators/
73
(+38 more pages)
592

Are you the person on the team that…

Whiteboards the problems out.
Has The Actual Final Say. (Business Edition.)
Figures out your CSS .
Understands the legacy code.
Has the usage data.
Understands the business better than everyone else.
Writes most of your tests.
Makes your comps/wires/design documentation.
Writes requirements.
Has The Actual Actual Final Say. (Technology Edition.)
Cleans up the backlog.
Writes user stories.
Decides when the due date ***actually*** is.
Talks to the users.


J.MARIO
10
7

Personas:

- Keep them simple.
- You don’t need to build fake people with fake lives.
- Think about drives your users, especially in the context of your application.
- Keep the user’s goals in mind; and I don’t mean buying a Lamborghini. I mean their goals on your platform.
- When ever your Hard Data & Soft Data don’t correlate; you may be missing a Persona. (or your assumptions are wrong.)

I stole this list from smarter people than myself:

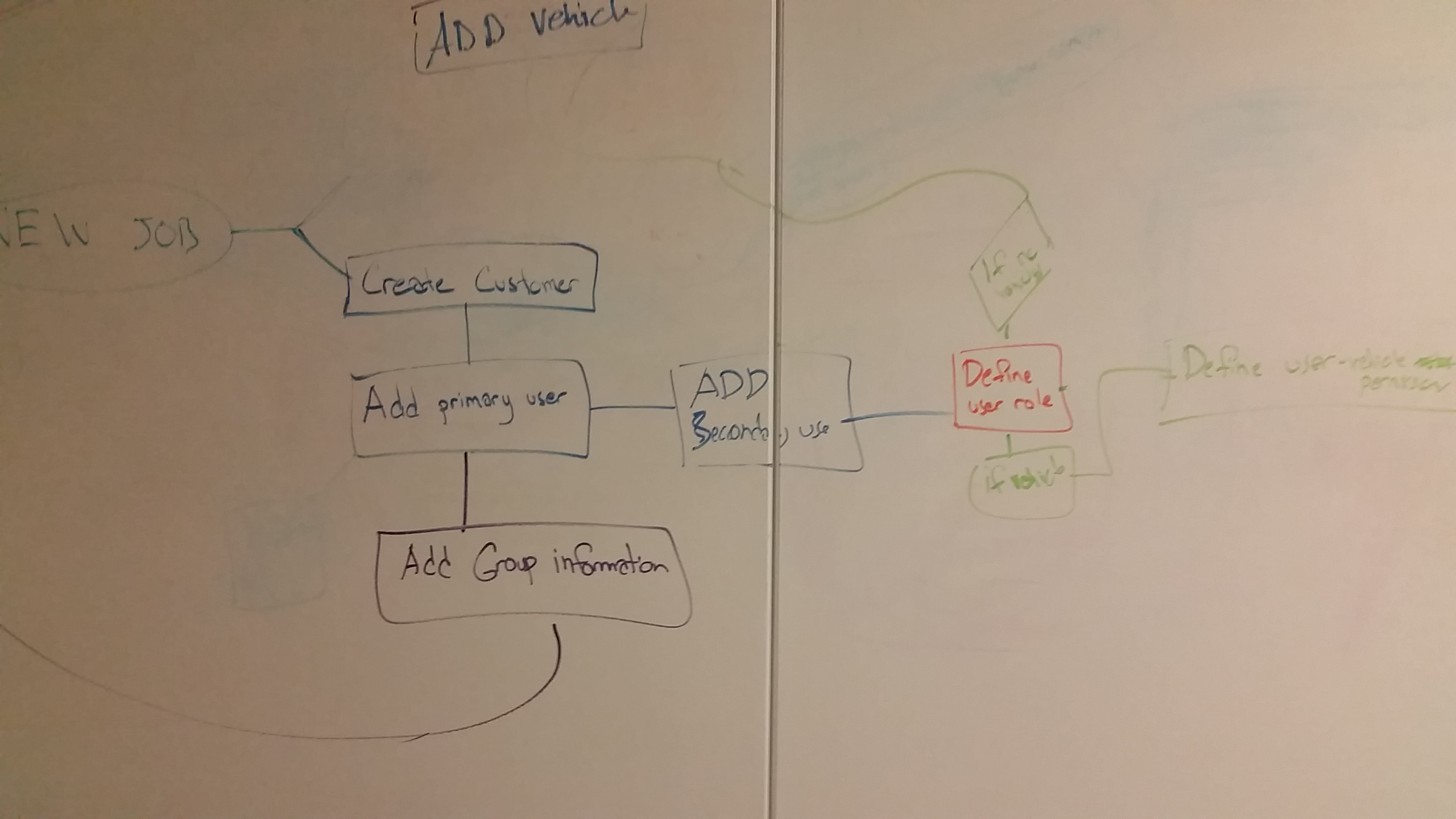
1. **Ask well crafted interview questions. (I.E Vague, open-ended.)**
2. **Dig deeper with the right follow-up questions.**
3. **Avoid jargon of any kind.**
4. **Embrace awkward silence.**
5. **Keep your reactions neutral.**
6. **Don’t mention solutions until the end.**

https://uxplanet.org/6-tips-for-better-user-interviews-ad925afc52df


6 7 7 8 8 9 8 7

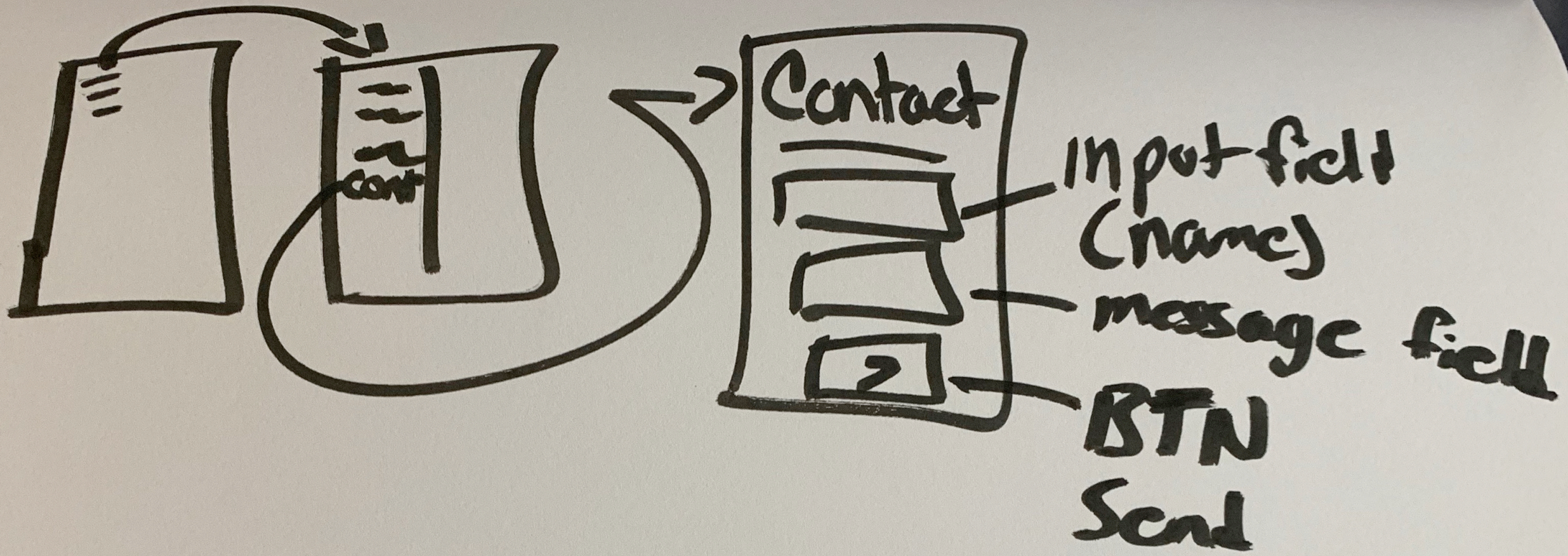
Are you the person on the team that…

Whiteboards the problems out.
Has The Actual Final Say. (Business Edition.)
Figures out your CSS .
Understands the legacy code.
Has the usage data.
Understands the business better than everyone else.
Writes most of your tests.
Makes your comps/wires/design documentation.
Writes requirements.
Has The Actual Actual Final Say. (Technology Edition.)
Cleans up the backlog.
Writes user stories.
Decides when the due date ***actually*** is.
Talks to the users.

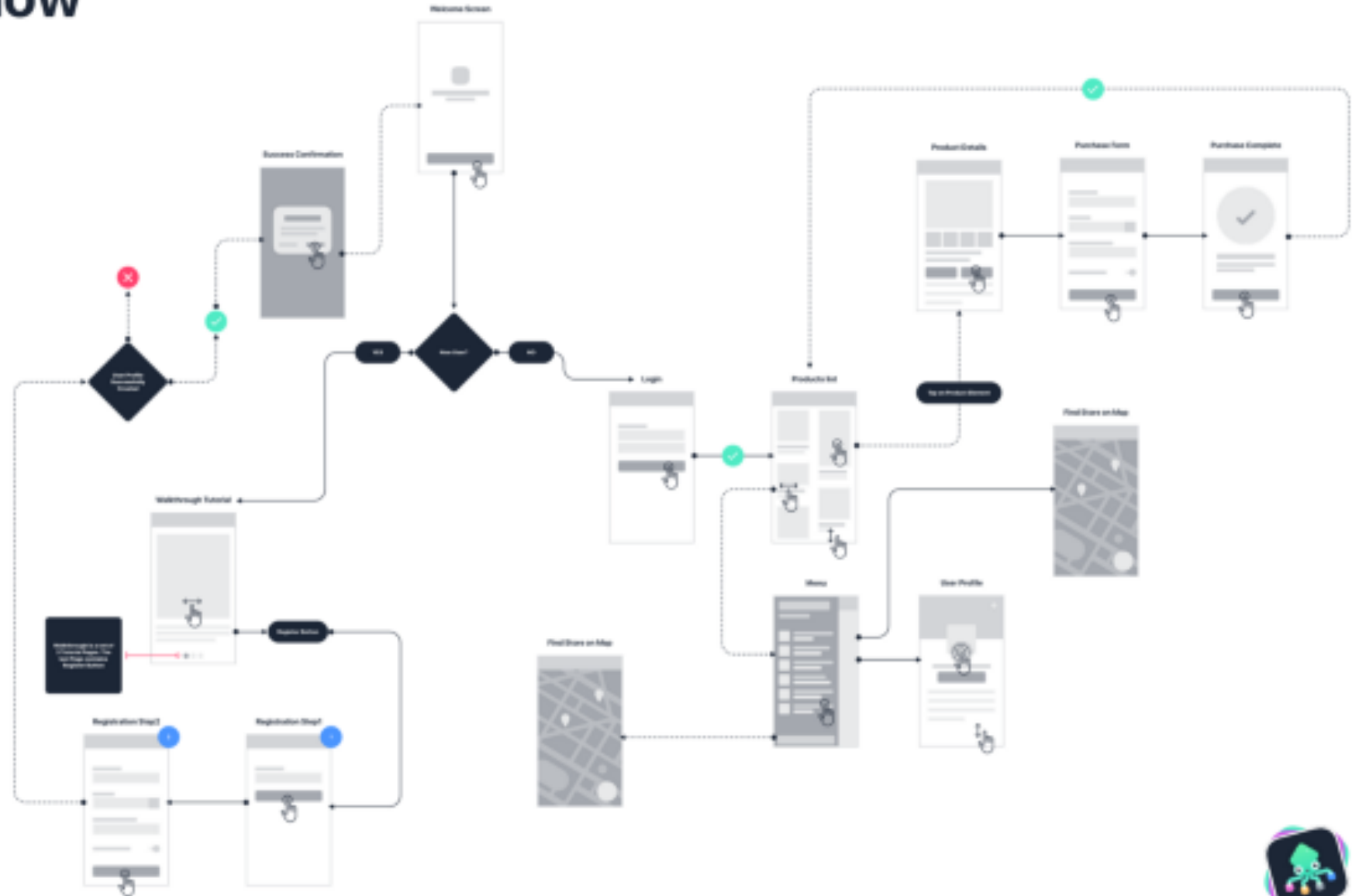

ADD vehicle
Create Customer
Add primary user
Add Group information
Define user role

The lowest amount of clicks doesn’t always mean it’s the best solution.

With that in mind, use the fewest interactions possible.


cont
Contact
input field
(name)
message field
2
BTN
Send

User Flow



made with

SQUID

UI FLOW TEMPLATE

There is no way to do this wrong.

There is no way to do this wrong.

No Really.

Action -> Reaction.

Cause -> Effect

User Clicks

->

Thing Happens

Repeat Until Journey Mapped.

Okay. Back to the pretty font.

ARTICLE BY 99PI

Norman Doors: Don't Know Whether to Push or Pull? Blame Design.

CATEGORY Objects | DATE 02.26.16 | PRODUCER 99pi

Some doors require printed instructions to operate, while others are so poorly designed that they lead people to do the exact opposite of what they need to in order to open them. Their shapes or details may suggest that pushing should work, when in fact pulling is required (or the other way around). Roman Mars teamed up with Joe Posner of Vox to interview Don Norman and bring you this story of terrible doors:

EXPLORE

Architecture
Infrastructure
Cities
Objects
Sounds
Visuals
Technology
History

MORE IN OBJECTS


KURT KOHLSTEDT
Paint & Press: "Pirate...


EPISODE 193
Tube Benders


99PI
Norman Doors: Don't Know Whether to...

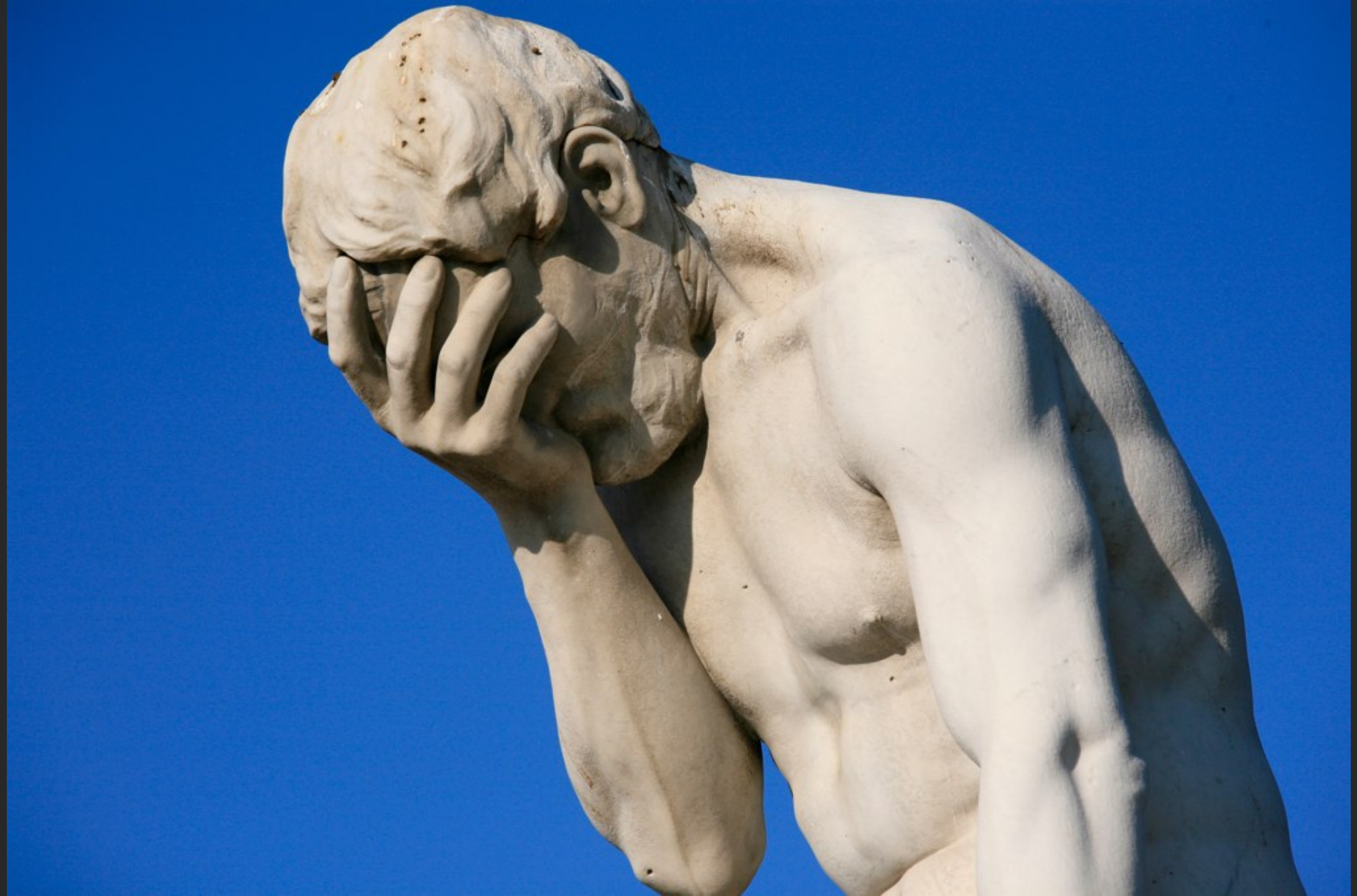

KURT KOHLSTEDT
Tsunami Stones: Ancient Japanese Markers...

Are you the person on the team that…

Whiteboards the problems out.
Has The Actual Final Say. (Business Edition.)
Figures out your CSS .
Understands the legacy code.
Has the usage data.
Understands the business better than everyone else.
Writes most of your tests.
Makes your comps/wires/design documentation.
Writes requirements.
Has The Actual Actual Final Say. (Technology Edition.)
Cleans up the backlog.
Writes user stories.
Decides when the due date ***actually*** is.
Talks to the users.

How to competitive analysis good:

1. Compare your Journey Maps.
2. See where you can improve.
3. If something seems off, look deeper.
4. Identify what they did right, and steal it.
5. Identify what they did wrong, and learn from their mistakes.



Are you the person on the team that…

Whiteboards the problems out.
Has The Actual Final Say. (Business Edition.)
Figures out your CSS .
Understands the legacy code.
Has the usage data.
Understands the business better than everyone else.
Writes most of your tests.
Makes your comps/wires/design documentation.
Writes requirements.
Has The Actual Actual Final Say. (Technology Edition.)
Cleans up the backlog.
Writes user stories.
Decides when the due date ***actually*** is.
Talks to the users.

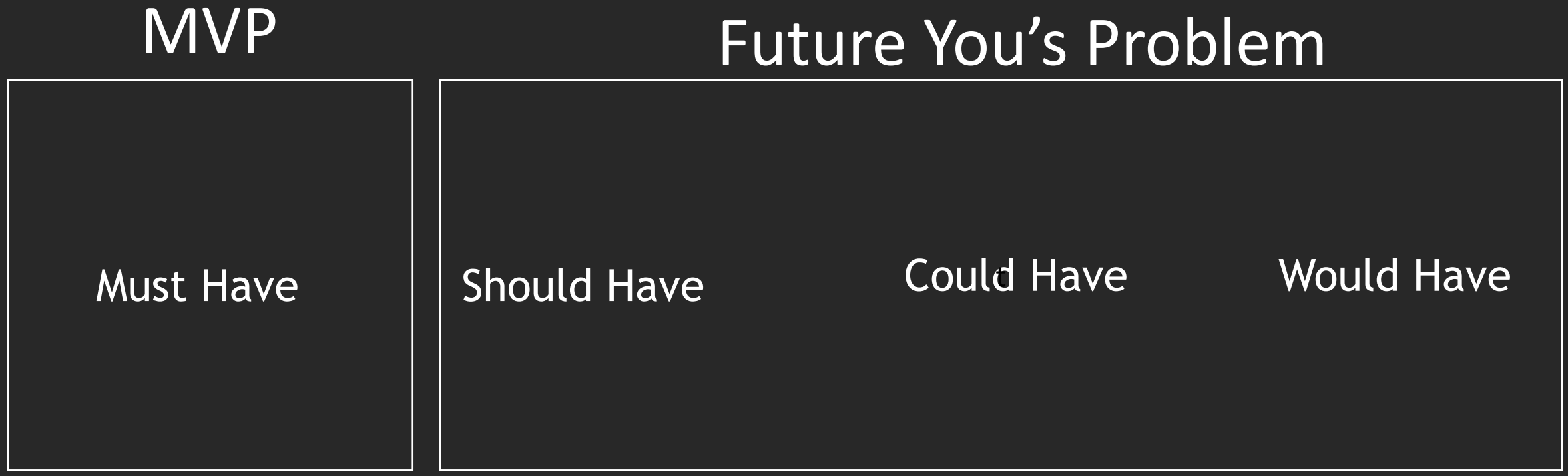
MoSCoW

Must Have

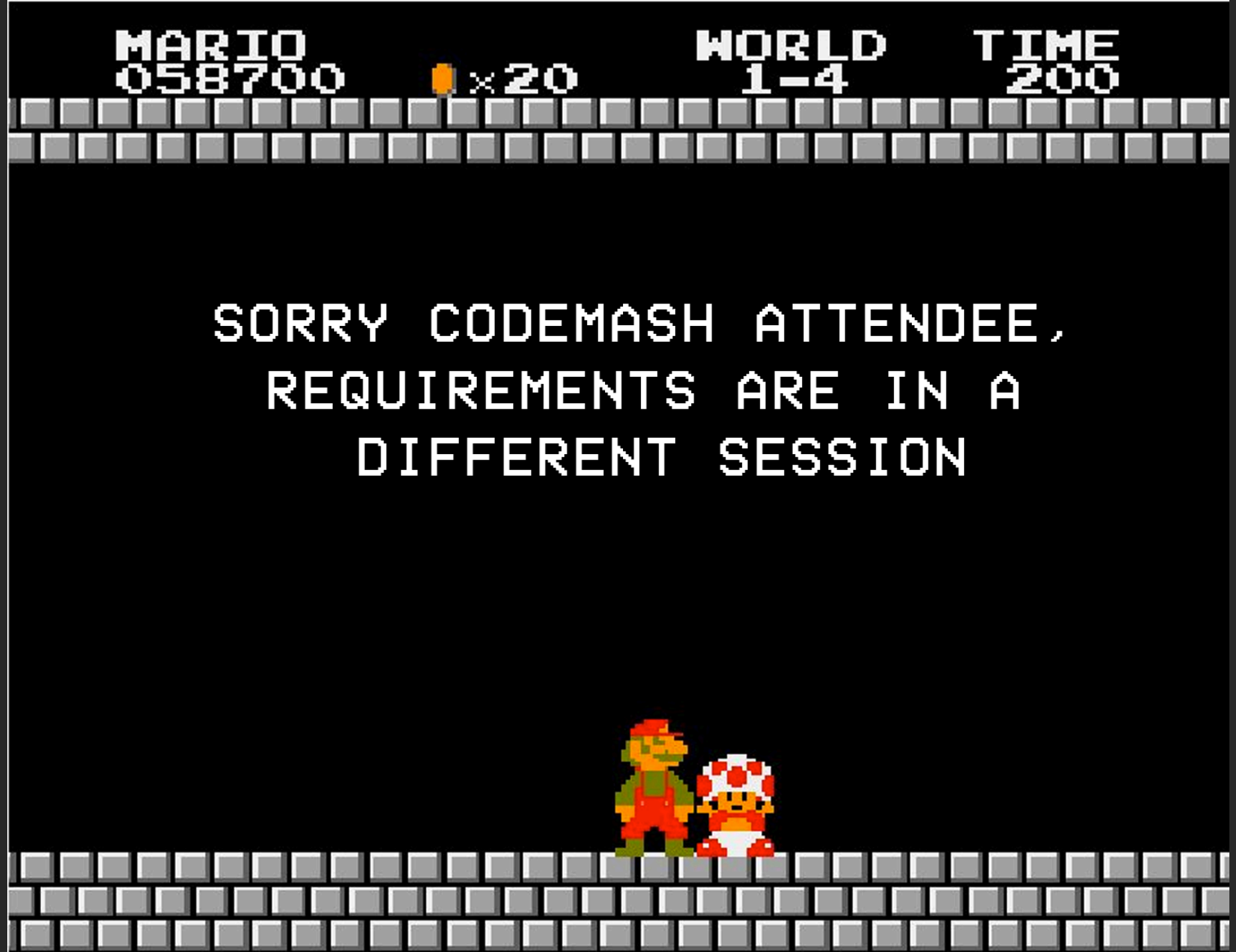
Should Have

Could Have

Would Have


MVP
Future You's Problem
Must Have
Should Have
Could Have
Would Have

Requirements


MARIO
058700
×20
WORLD
1-4
TIME
200
SORRY CODEMASH ATTENDEE,
REQUIREMENTS ARE IN A
DIFFERENT SESSION

Conculsion

Questions?

Fin

Ryan Albertson me@ryan.ws