APPLIED ENTREPRENEURSHIP

SPRING 2023/2024

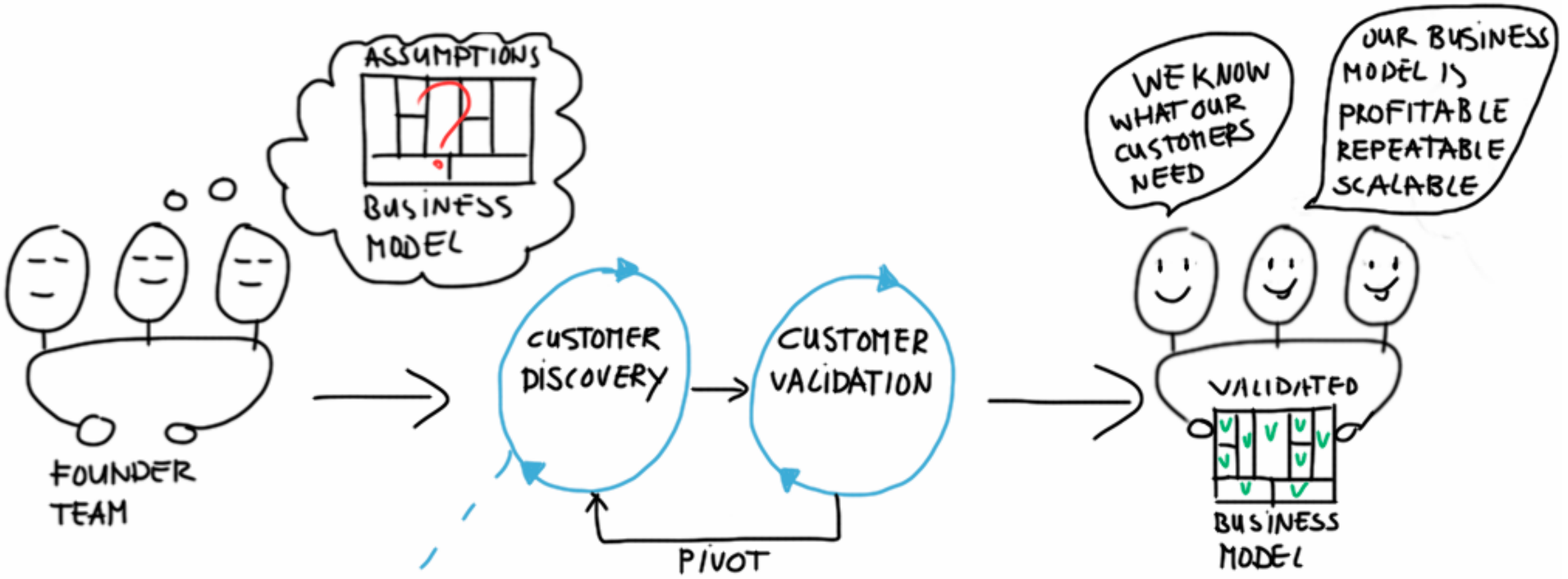
NOVA SCHOOL OF
BUSINESS & ECONOMICS

REMEMBERING.

Fucking social media— how does it work?

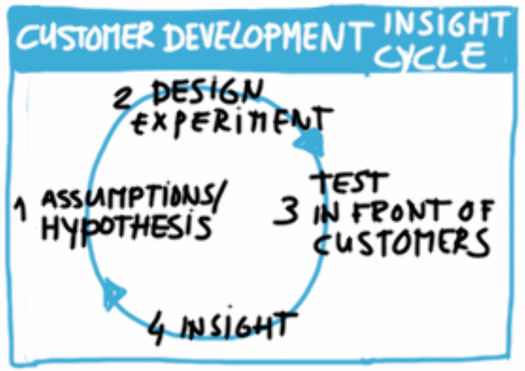
Uncertainty / patterns / insights

Clarity / Focus



CUSTOMER DEVELOPMENT

DESIGN: STEVE BLANK



The Business Model Canvas

Designed for:

Designed by:

Date:

Iteration:

Key Partners

Who are our Key Partners?
Who are our key suppliers?
Which Key Resources are we acquiring from partners?
Which Key Activities do partners perform?

Key Activities

What Key Activities do our Value Propositions require?
Our Distribution Channels?
Customer Relationships?
Revenue streams?

Key Resources

What Key Resources do our Value Propositions require?
Our Distribution Channels? Customer Relationships?
Revenue Streams?

Value Propositions

What value do we deliver to the customer?
Which one of our customer's problems are we helping to solve?
What bundles of products and services are we offering to each Customer Segment?
Which customer needs are we satisfying?

Customer Relationships

What type of relationship does each of our Customer Segments expect us to establish and maintain with them?
Which ones have we established?
How are they integrated with the rest of our business model?
How costly are they?

Channels

Through which Channels do our Customer Segments want to be reached?
How are we reaching them now?
How are our Channels integrated?
Which ones work best?
Which ones are most cost-efficient?
How are we integrating them with customer routines?

Customer Segments

For whom are we creating value?
Who are our most important customers?

Cost Structure

What are the most important costs inherent in our business model?
Which Key Resources are most expensive?
Which Key Activities are most expensive?

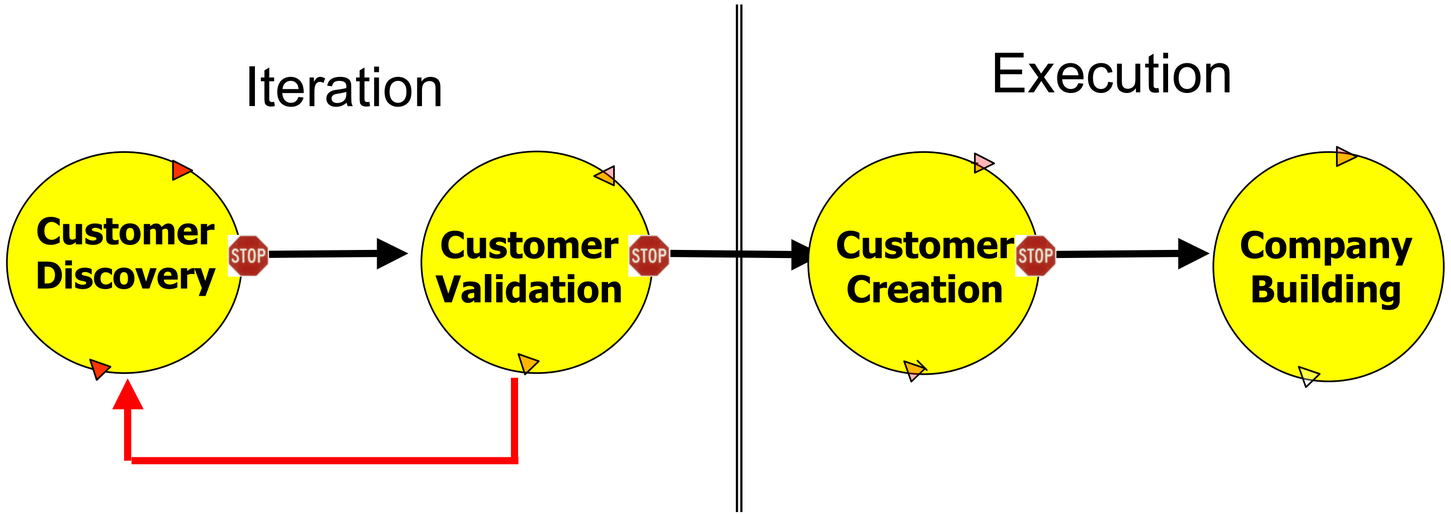
Revenue Streams

For what value are our customers really willing to pay?
For what do they currently pay?
How are they currently paying?
How would they prefer to pay?
How much does each Revenue Stream contribute to overall revenues?

www.businessmodelgeneration.com

This work is licensed under the Creative Commons Attribution-Share Alike 3.0 Unported License.
To view a copy of this license, visit http://creativecommons.org/licenses/by-sa/3.0/
or send a letter to Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.

3 Stages of a Startup



The Business Model Canvas

Designed for:

Designed by:

Date:

Iteration:

Key Partners

Who are our Key Partners?
Who are our key suppliers?
Which Key Resources are we acquiring from partners?
Which Key Activities do partners perform?

Key Activities

What Key Activities do our Value Propositions require?
Our Distribution Channels?
Customer Relationships?
Revenue streams?

Key Resources

What Key Resources do our Value Propositions require?
Our Distribution Channels? Customer Relationships?
Revenue Streams?

Value Propositions

What value do we deliver to the customer?
Which one of our customer's problems are we helping to solve?
What bundles of products and services are we offering to each Customer Segment?
Which customer needs are we satisfying?

this is what you're offering your customers

Customer Relationships

What type of relationship does each of our Customer Segments expect us to establish and maintain with them?
Which ones have we established?
How are they integrated with the rest of our business model?
How costly are they?

Channels

Through which Channels do our Customer Segments want to be reached?
How are we reaching them now?
How are our Channels integrated?
Which ones work best?
Which ones are most cost-efficient?
How are we integrating them with customer routines?

Customer Segments

For whom are we creating value?
Who are our most important customers?

Cost Structure

What are the most important costs inherent in our business model?
Which Key Resources are most expensive?
Which Key Activities are most expensive?

Revenue Streams

For what value are our customers really willing to pay?
For what do they currently pay?
How are they currently paying?
How would they prefer to pay?
How much does each Revenue Stream contribute to overall revenues?

www.businessmodelgeneration.com

The Value Proposition Canvas

Designed for:

Designed by:

On: Day | Month | Year

Iteration: No.

Products & Services

List all the products and services your value proposition is built around.

Which products and services do you offer that help your customer get either a functional, social, or emotional job done, or help him/her satisfy basic needs?

Which ancillary products and services help your customer perform the roles of:

Buyer
(e.g. products and services that help customers compare offers, decide, buy, take delivery of a product or service, ...)

Co-creator
(e.g. products and services that help customers co-design solutions, otherwise contribute value to the solution, ...)

Transferrer
(e.g. products and services that help customers dispose of a product, transfer it to others, or resell, ...)

Products and services may either by tangible (e.g. manufactured goods, face-to-face customer service), digital/virtual (e.g. downloads, online recommendations), intangible (e.g. copyrights, quality assurance), or financial (e.g. investment funds, financing services).

Rank all products and services according to their importance to your customer. Are they crucial or trivial to your customer?

Gain Creators

Describe how your products and services create customer gains.
How do they create benefits your customer expects, desires or would be surprised by, including functional utility, social gains, positive emotions, and cost savings?

Do they...

Create savings that make your customer happy?
(e.g. in terms of time, money and effort, ...)

Produce outcomes your customer expects or that go beyond their expectations?
(e.g. better quality level, more of something, less of something, ...)

Copy or outperform current solutions that delight your customer?
(e.g. regarding specific features, performance, quality, ...)

Make your customer's job or life easier?
(e.g. flatter learning curve, usability, accessibility, more services, lower cost of ownership, ...)

Create positive social consequences that your customer desires?
(e.g. makes them look good, produces an increase in power, status, ...)

Do something customers are looking for?
(e.g. good design, guarantees, specific or more features, ...)

Fulfill something customers are dreaming about?
(e.g. help big achievements, produce big reliefs, ...)

Produce positive outcomes matching your customers success and failure criteria?
(e.g. better performance, lower cost, ...)

Help make adoption easier?
(e.g. lower cost, less investments, lower risk, better quality, performance, design, ...)

Rank each gain your products and services create according to its relevance to your customer. Is it substantial or insignificant? For each gain indicate how often it occurs.

Pain Relievers

Describe how your products and services alleviate customer pains. How do they eliminate or reduce negative emotions, undesired costs and situations, and risks your customer experiences or could experience before, during, and after getting the job done?

Do they...

Produce savings?
(e.g. in terms of time, money, or efforts, ...)

Make your customers feel better?
(e.g. kills frustrations, annoyances, things that give them a headache, ...)

Fix underperforming solutions?
(e.g. new features, better performance, better quality, ...)

Put an end to difficulties and challenges your customers encounter?
(e.g. make things easier, helping them get done, eliminate resistance, ...)

Wipe out negative social consequences your customers encounter or fear?
(e.g. loss of face, power, trust, or status, ...)

Eliminate risks your customers fear?
(e.g. financial, social, technical risks, or what could go awfully wrong, ...)

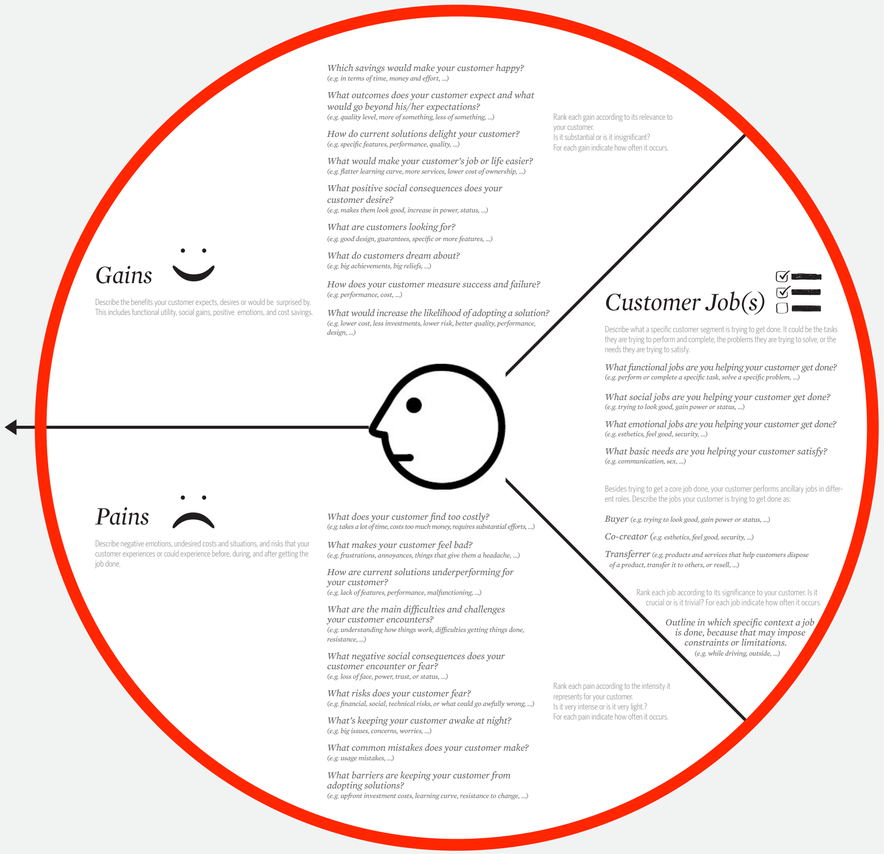
Help your customers better sleep at night?
(e.g. by helping with big issues, diminishing concerns, or eliminating worries, ...)

Limit or eradicate common mistakes customers make?
(e.g. usage mistakes, ...)

Get rid of barriers that are keeping your customer from adopting solutions?
(e.g. lower or no upfront investment costs, flatter learning curve, less resistance to change, ...)

Rank each pain your products and services kill according to their intensity for your customer. Is it very intense or very light?

For each pain indicate how often it occurs. Risks your customer experiences or could experience before, during, and after getting the job done?



Gains

Describe the benefits your customer expects, desires or would be surprised by. This includes functional utility, social gains, positive emotions, and cost savings.

Which savings would make your customer happy?
(e.g. in terms of time, money and effort, ...)

What outcomes does your customer expect and what would go beyond his/her expectations?
(e.g. quality level, more of something, less of something, ...)

How do current solutions delight your customer?
(e.g. specific features, performance, quality, ...)

What would make your customer's job or life easier?
(e.g. flatter learning curve, more services, lower cost of ownership, ...)

What positive social consequences does your customer desire?
(e.g. makes them look good, increase in power, status, ...)

What are customers looking for?
(e.g. good design, guarantees, specific or more features, ...)

What do customers dream about?
(e.g. big achievements, big reliefs, ...)

How does your customer measure success and failure?
(e.g. performance, cost, ...)

What would increase the likelihood of adopting a solution?
(e.g. lower cost, less investments, lower risk, better quality, performance, design, ...)

Rank each gain according to its relevance to your customer.
Is it substantial or is it insignificant?
For each gain indicate how often it occurs.

Pains

Describe negative emotions, undesired costs and situations, and risks that your customer experiences or could experience before, during, and after getting the job done.

What does your customer find too costly?
(e.g. takes a lot of time, costs too much money, requires substantial efforts, ...)

What makes your customer feel bad?
(e.g. frustrations, annoyances, things that give them a headache, ...)

How are current solutions underperforming for your customer?
(e.g. lack of features, performance, malfunctioning, ...)

What are the main difficulties and challenges your customer encounters?
(e.g. understanding how things work, difficulties getting things done, resistance, ...)

What negative social consequences does your customer encounter or fear?
(e.g. loss of face, power, trust, or status, ...)

What risks does your customer fear?
(e.g. financial, social, technical risks, or what could go awfully wrong, ...)

What's keeping your customer awake at night?
(e.g. big issues, concerns, worries, ...)

What common mistakes does your customer make?
(e.g. usage mistakes, ...)

What barriers are keeping your customer from adopting solutions?
(e.g. upfront investment costs, learning curve, resistance to change, ...)

Rank each pain according to the intensity it represents for your customer.
Is it very intense or is it very light.?
For each pain indicate how often it occurs.

Customer Job(s)

Describe what a specific customer segment is trying to get done. It could be the tasks they are trying to perform and complete, the problems they are trying to solve, or the needs they are trying to satisfy.

What functional jobs are you helping your customer get done?
(e.g. perform or complete a specific task, solve a specific problem, ...)

What social jobs are you helping your customer get done?
(e.g. trying to look good, gain power or status, ...)

What emotional jobs are you helping your customer get done?
(e.g. esthetics, feel good, security, ...)

What basic needs are you helping your customer satisfy?
(e.g. communication, sex, ...)

Besides trying to get a core job done, your customer performs ancillary jobs in different roles. Describe the jobs your customer is trying to get done as:

Buyer *(e.g. trying to look good, gain power or status, ...)*

Co-creator *(e.g. esthetics, feel good, security, ...)*

Transferrer *(e.g. products and services that help customers dispose of a product, transfer it to others, or resell, ...)*

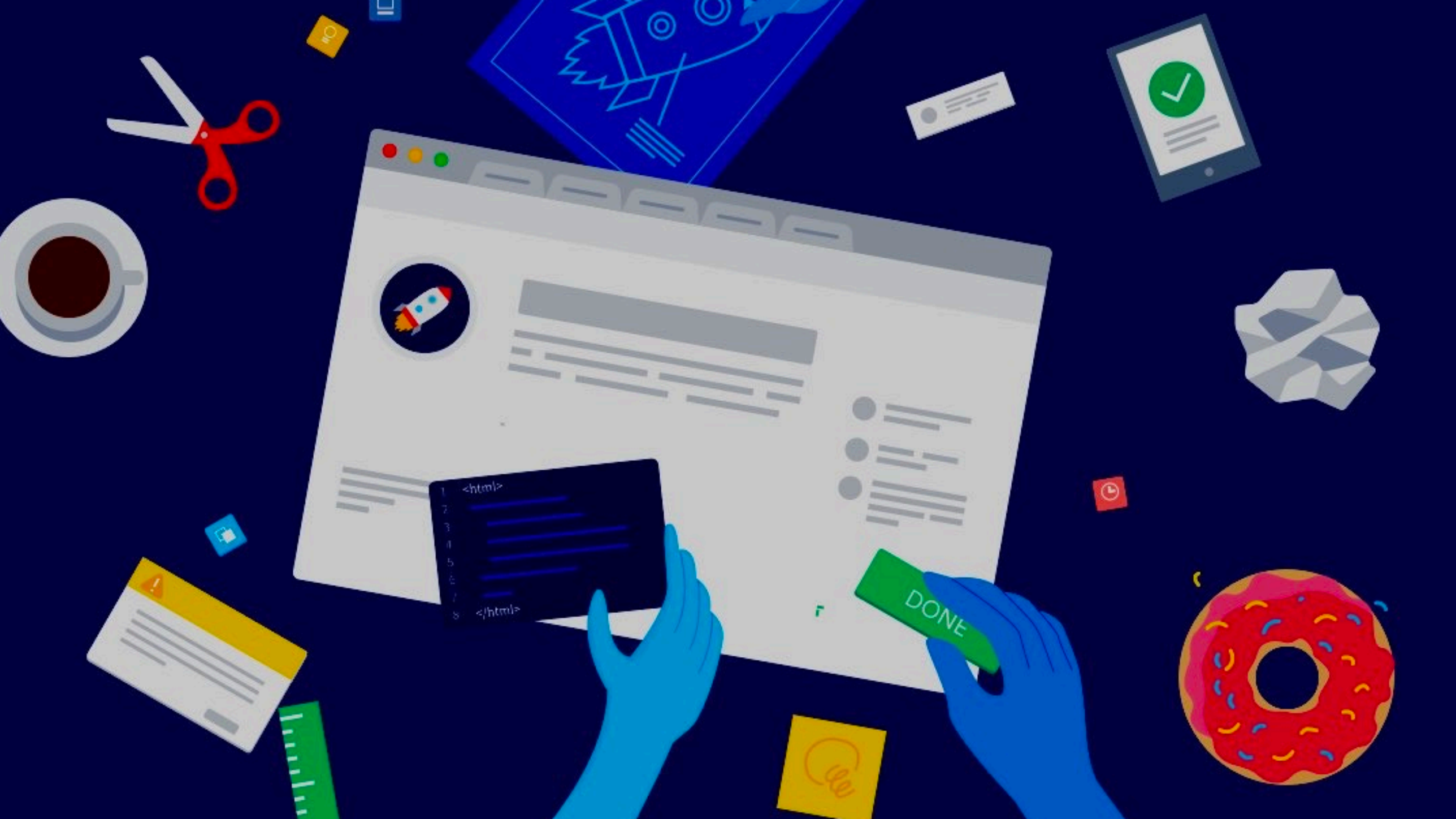
Rank each job according to its significance to your customer. Is it crucial or is it trivial? For each job indicate how often it occurs.

Outline in which specific context a job is done, because that may impose constraints or limitations.
(e.g. while driving, outside, ...)

design **observe**

Use in Conjunction with the Business Model Canvas

Copyright of Business Model Foundry GmbH

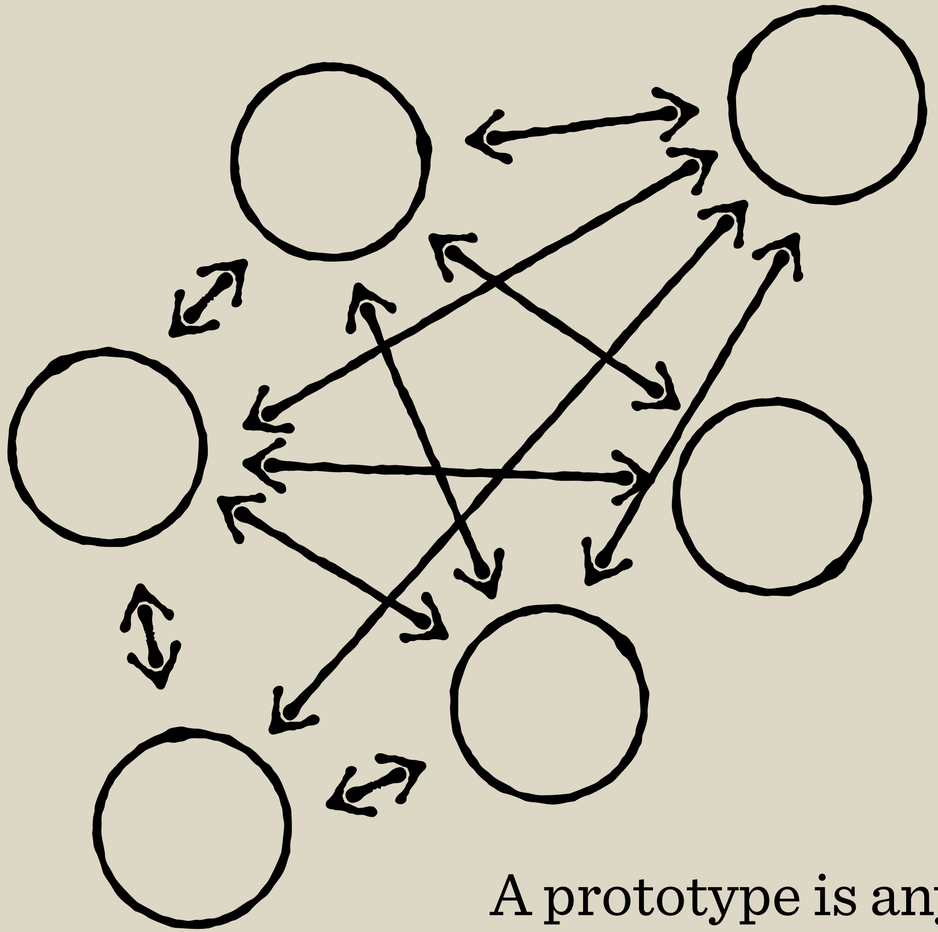

<html>
</html>
DONE

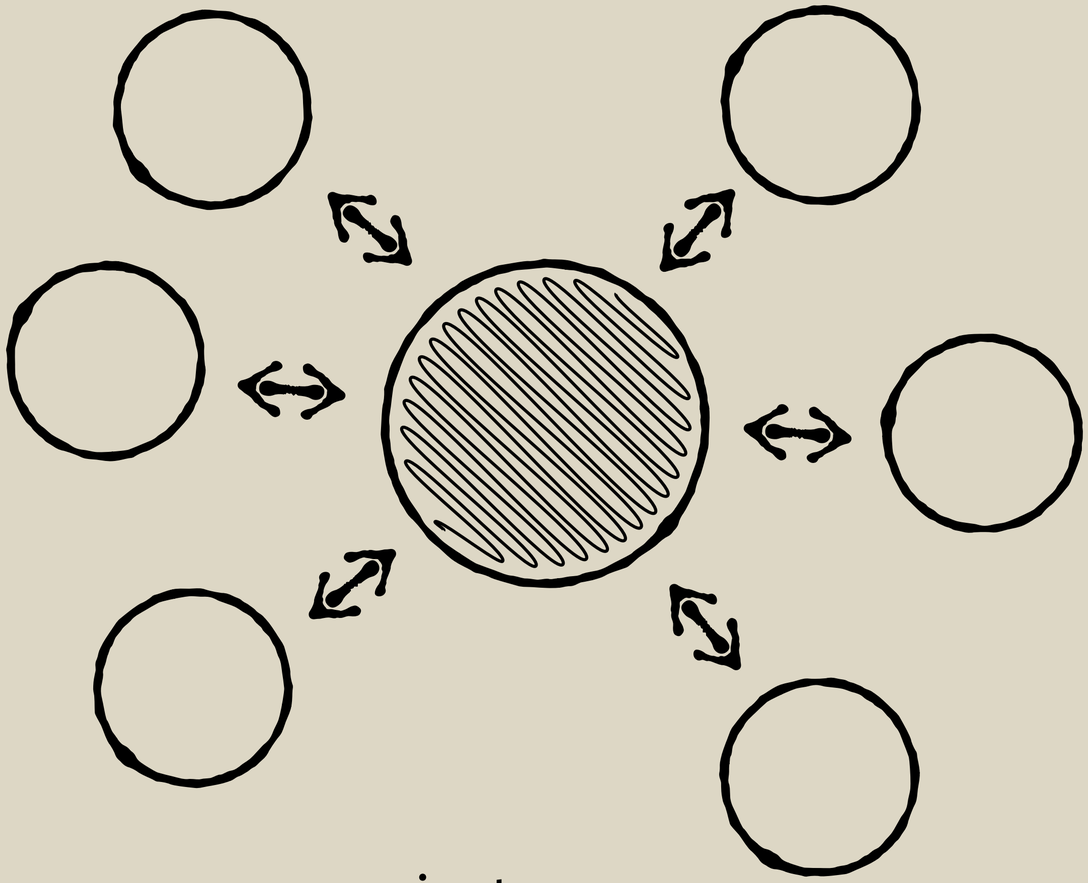
What’s a prototype?

A realistic façade is all you need to learn from customers.

“Prototyping allows us to *fail early* so we can succeed sooner”

David Kelley
IDEO Founder



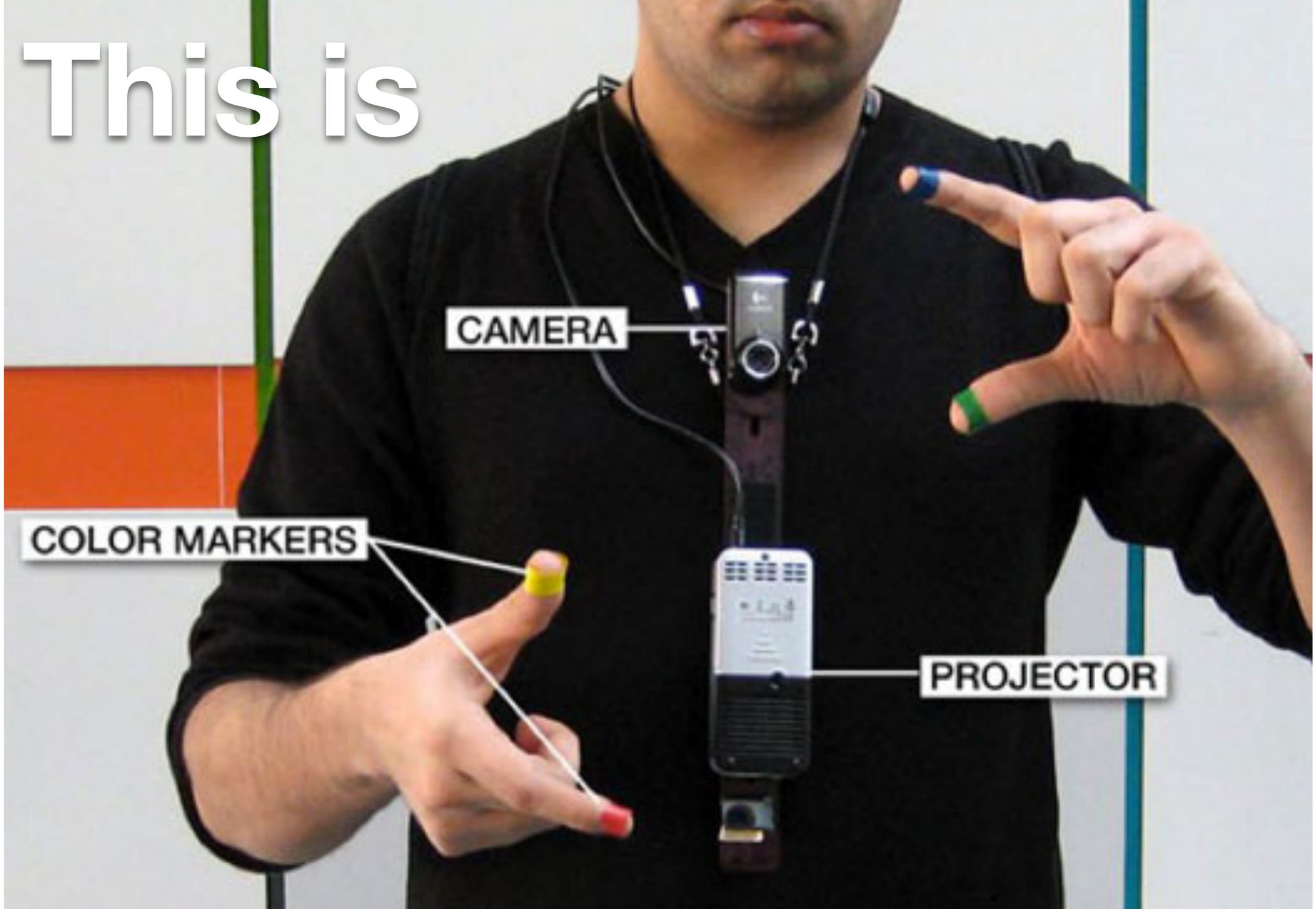


A prototype is anything that helps you communicate or test an experience with other people to get feedback

This is not a prototype

This is not a prototype

This is



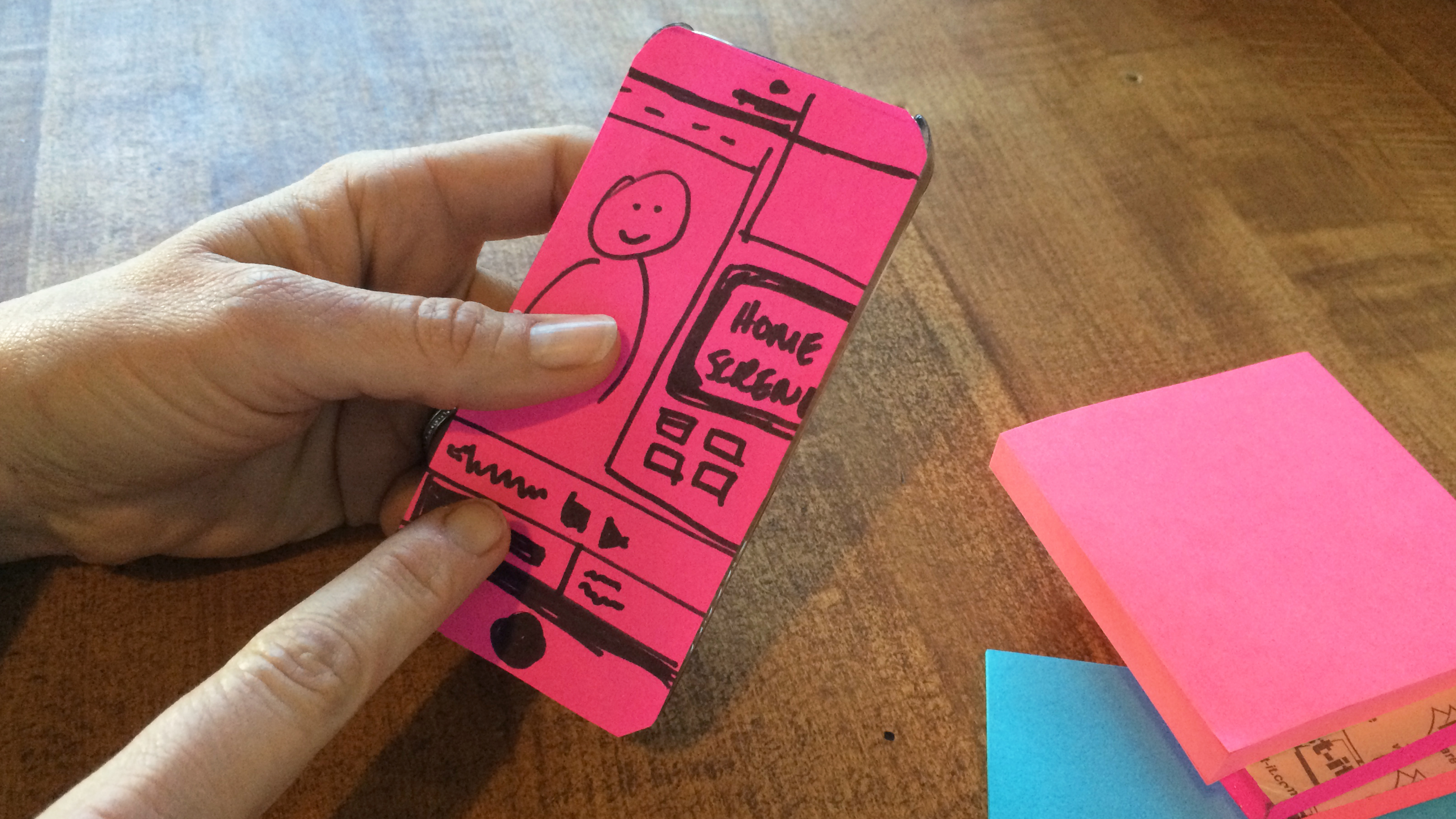
What makes a good prototype?

1. quick
2. cheap
3. minimal
4. testable
5. measurable

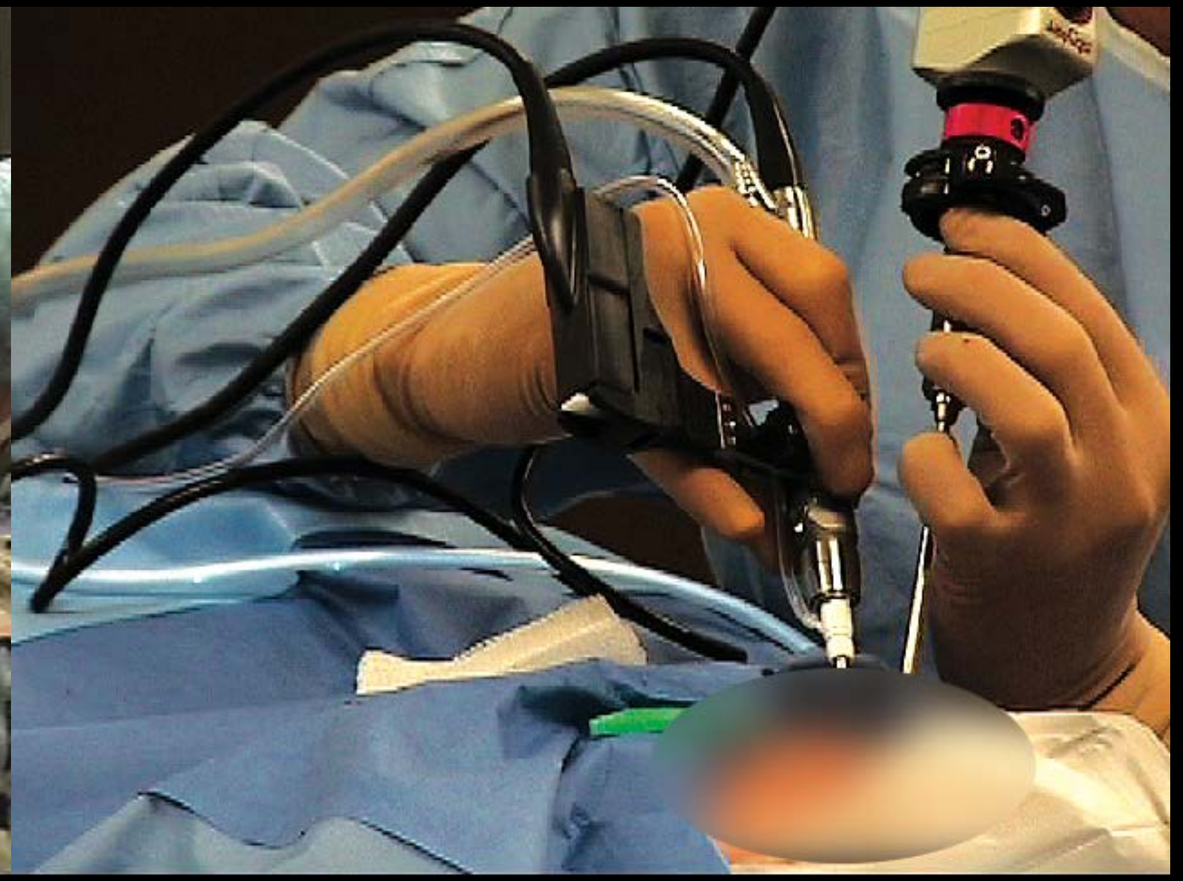
Let’s look at some examples

1.

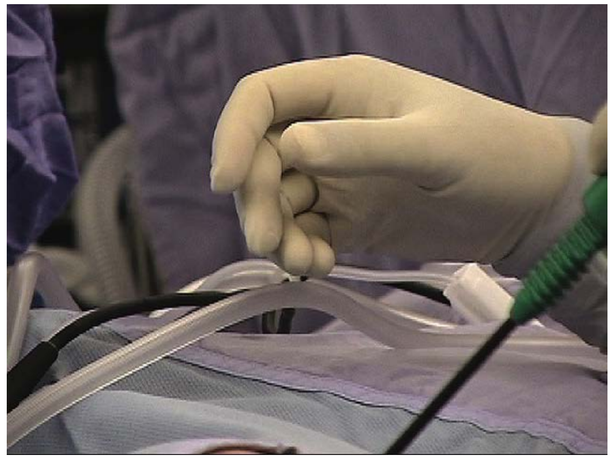
IF
Then
Recipient


HOME

2.



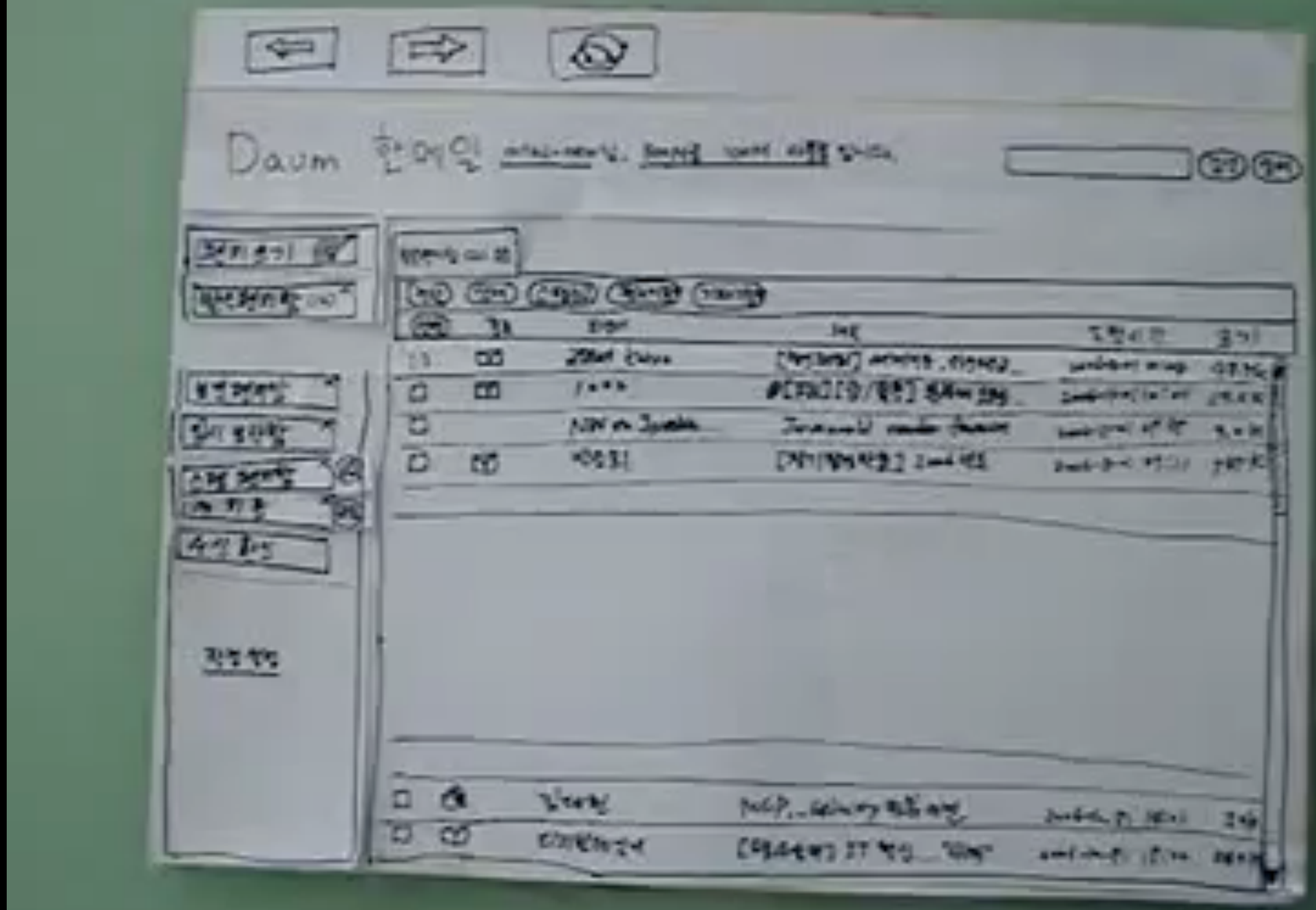
“One of the first things we noticed was the tangle of wires and awkward hand positions”- Andrew



“During an early brainstorm, we used office supplies to prototype an instrument for a neutral hand position.”

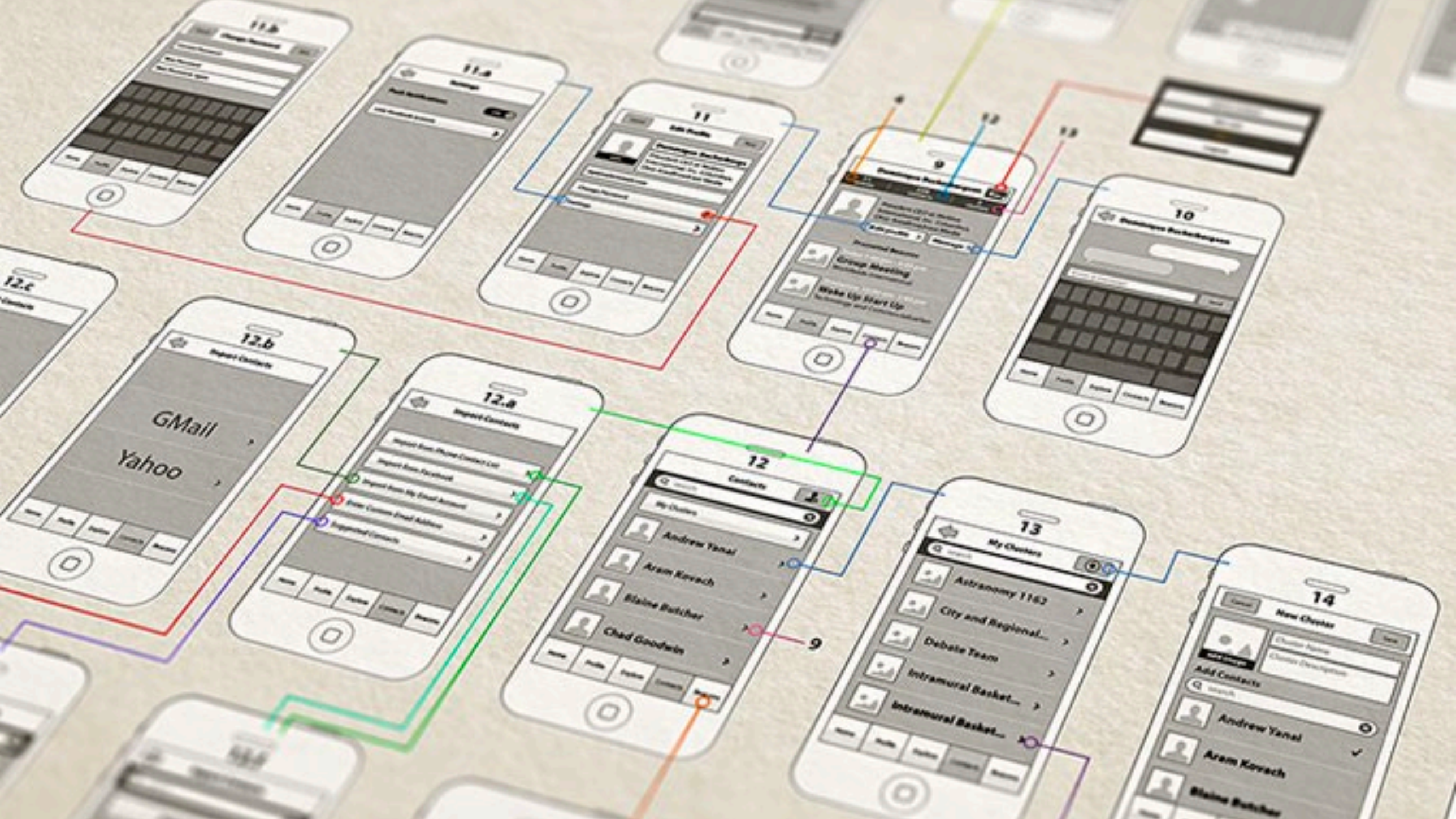
diego™

3.


Daum 한메일

4.

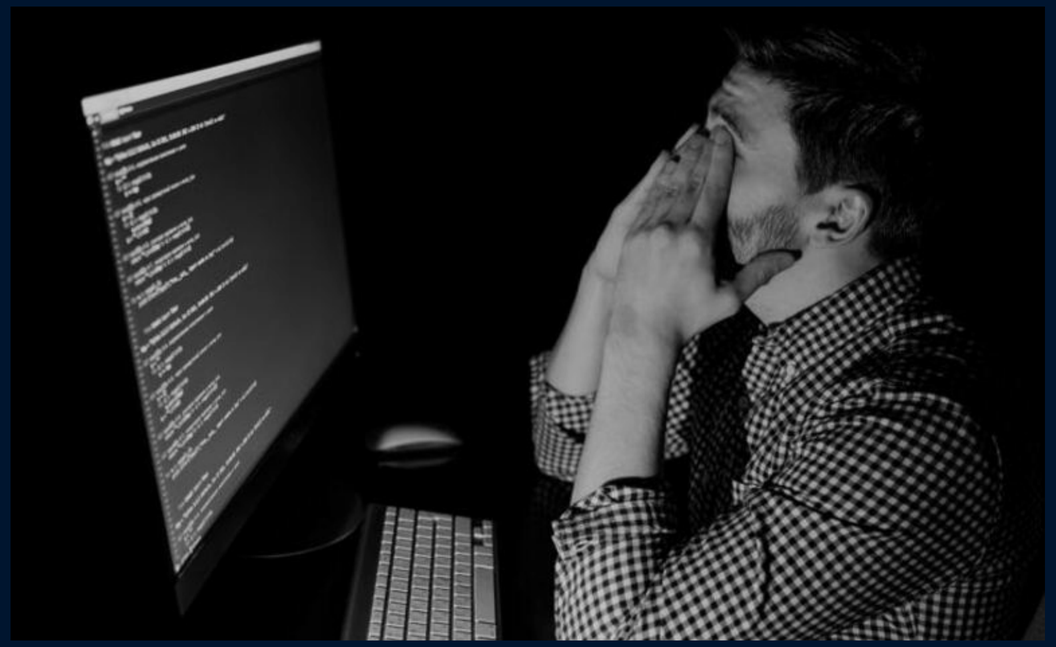
And there are other creative ways to do it…it's really up to you!


11.b
11.a
11
9
10
12.c
12.b
GMail
Yahoo
12.a
12
Andrew Yanai
Aram Kovach
Blaine Butcher
Chad Goodwin
9
13
My Clusters
Astronomy 1162
City and Regional...
Debate Team
Intramural Basket...
Intramural Basket...
14
New Cluster
Add Contacts
Andrew Yanai
Aram Kovach

Solutions?

Despite what it’s said…

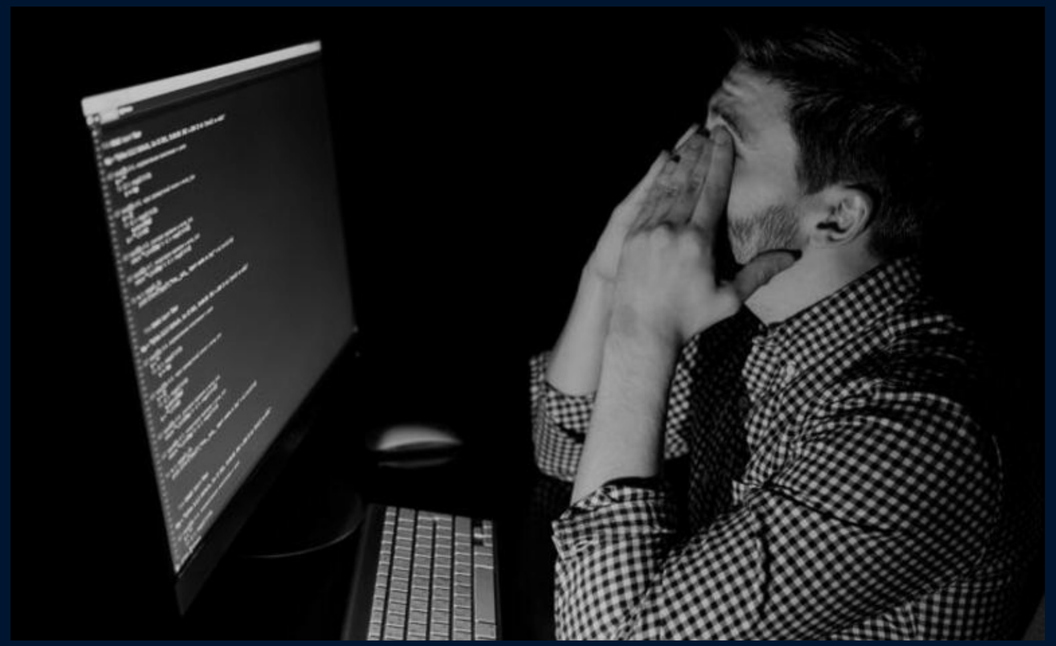
Learn to CODE?



Study DATA SCIENCE?

Despite what it’s said…

Learn to CODE?

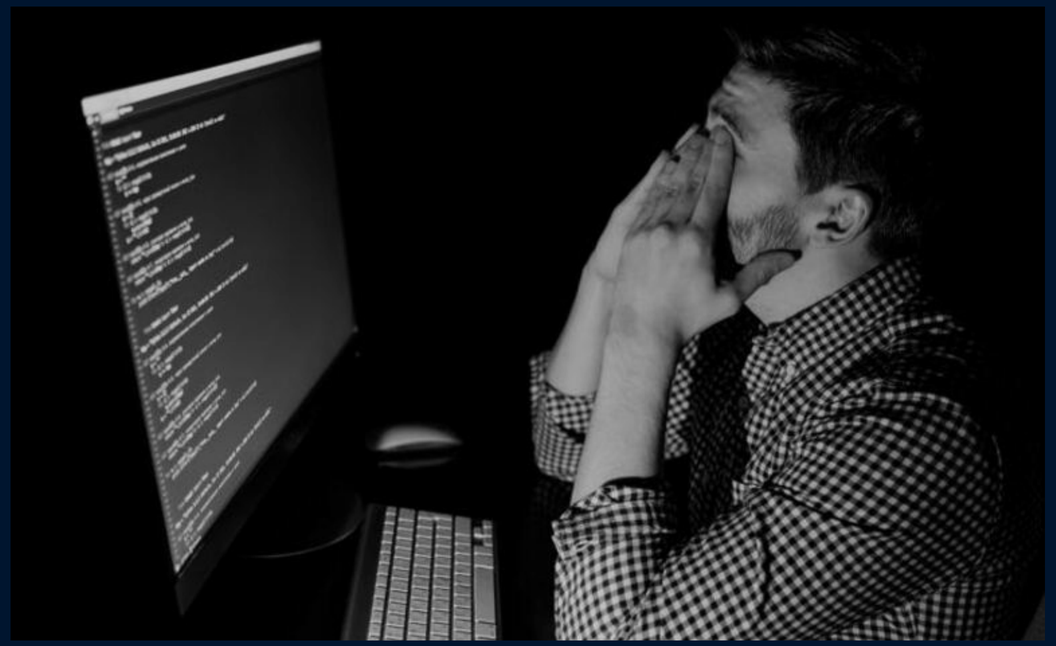


Study DATA SCIENCE?

NOT that easy. NOT that fast. NOT for everyone.

Despite what it’s said…

Learn to CODE?



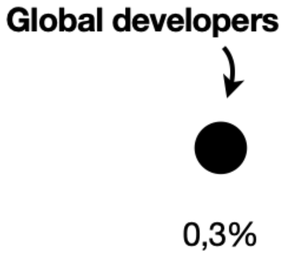
Study DATA SCIENCE?

NOT that easy. **NOT that fast.** **NOT for everyone.**

NOT ENOUGH.

There must be a better way…

CODE



CODE

Global developers

0,3%

LOW-CODE

Potential Low-code developers

5-10%

CODE

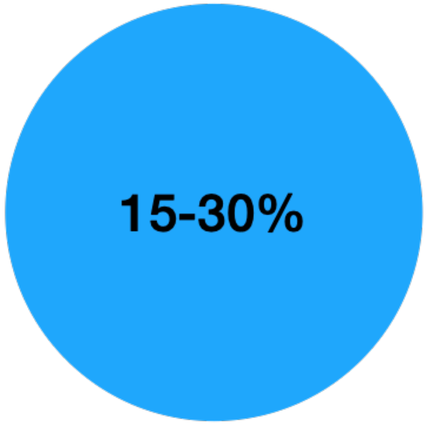
NO-CODE

LOW-CODE

Global developers

0,3%

Potential NO-code builders
(or Citizen developers)



Potential Low-code developers

5-10%

What’s NoCode?

Building software without code.

NoCode a.k.a. visual programming,
consists in creating digital products and software
without writing a single line of code.

What’s NoCode?

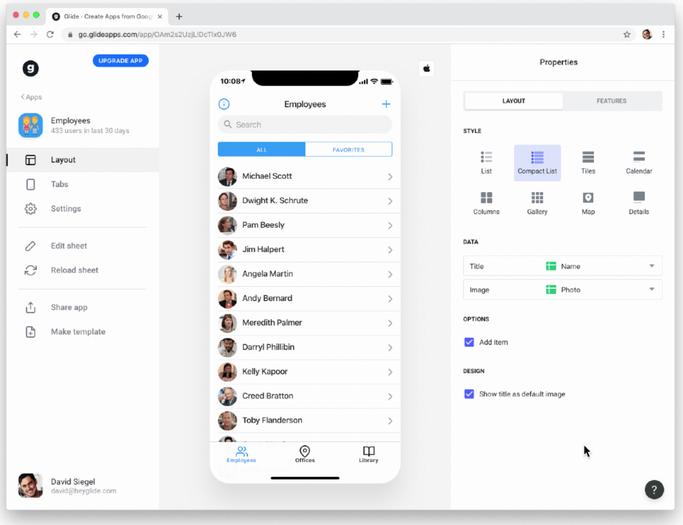
Building software without code.

NoCode a.k.a. visual programming,
consists in creating digital products and software
without writing a single line of code.

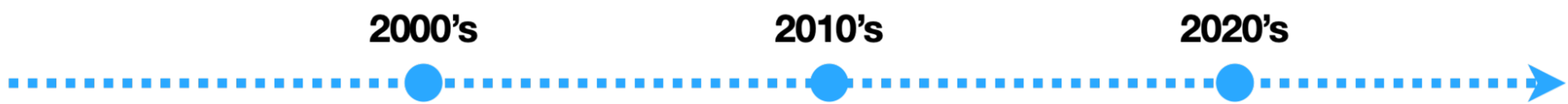
Code

```
 */
function my_theme_enqueue_scripts(){

    wp_enqueue_style( 'my_theme-style', get_stylesheet_uri() );
    wp_enqueue_script( 'my_theme-scripts', get_stylesheet_direct

}
add_action( 'wp_enqueue_scripts', 'my_theme_enqueue_scripts' );

/**
 * Filter themeworks titles
 */
function my_blog_title(){
    return 'News';
}
```

NoCode



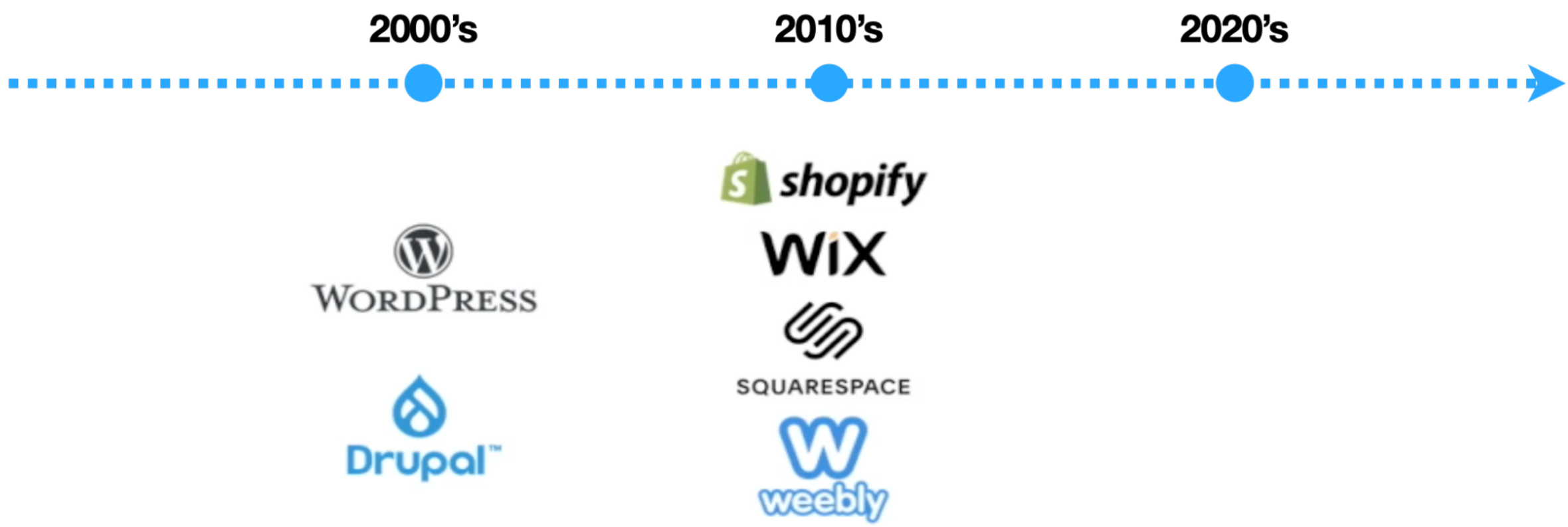
The evolution of NoCode technology



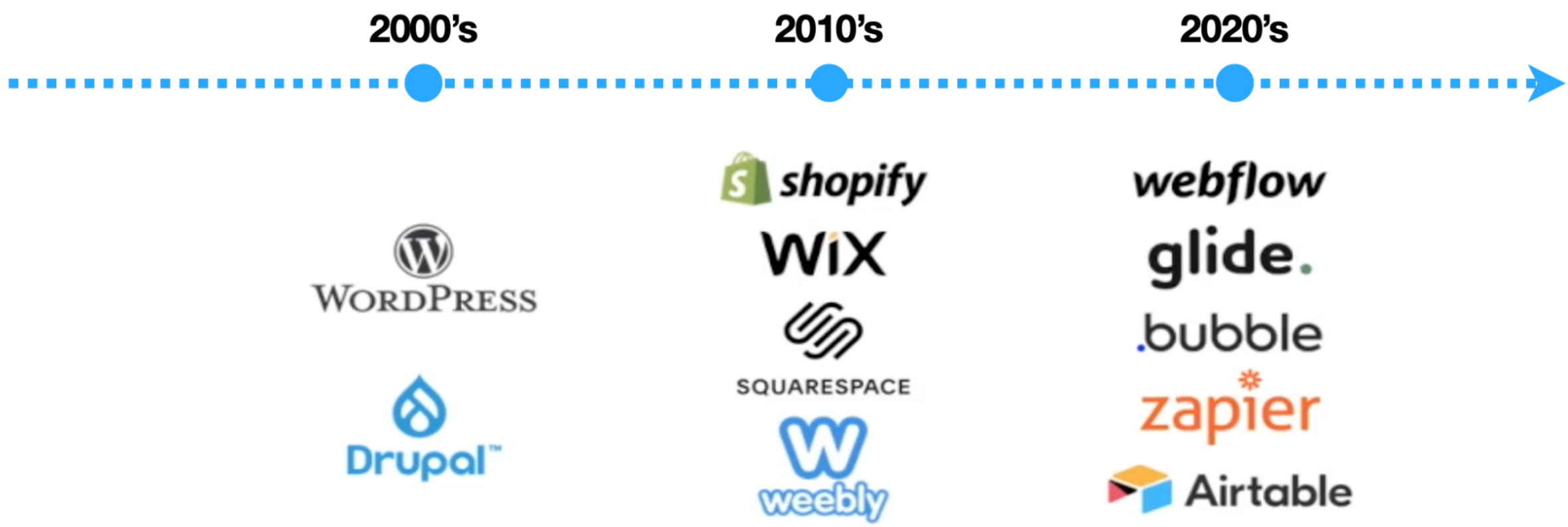
WordPress

Drupal™

The evolution of NoCode technology



The evolution of NoCode technology



Benefits of NoCode?

A movement that is already shaping the future

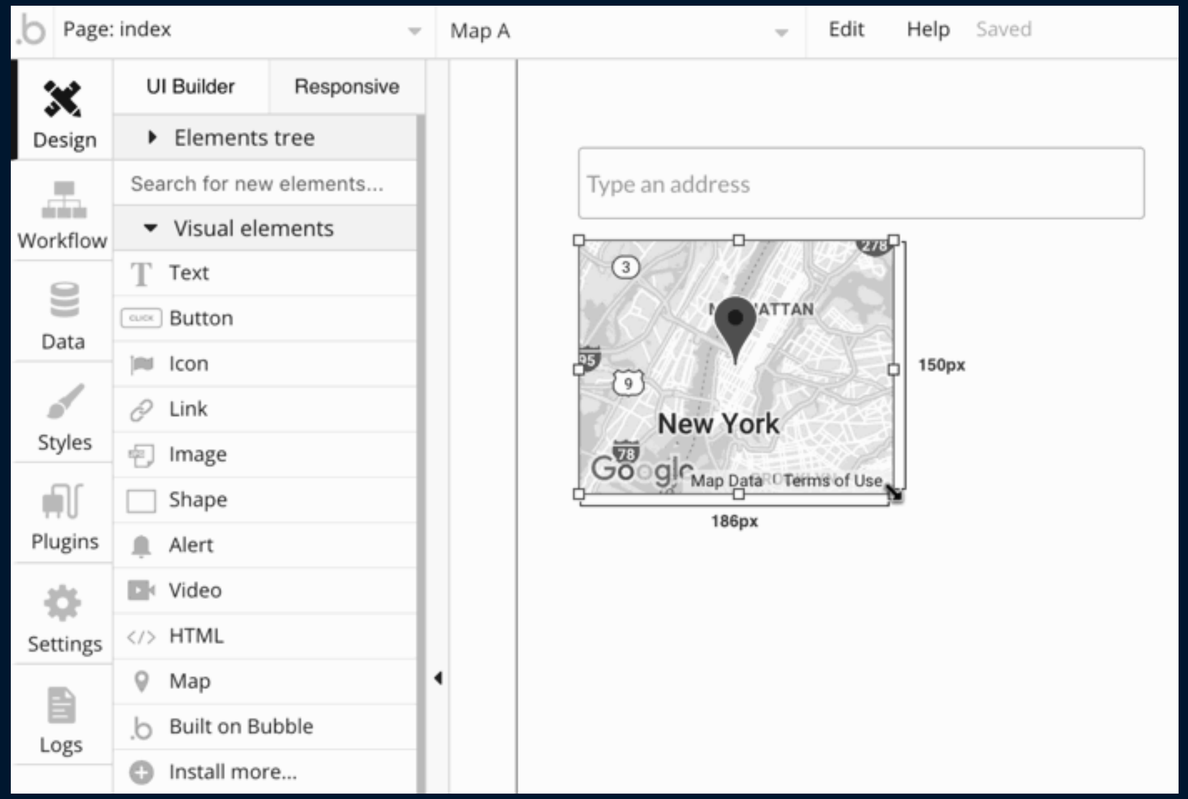
What is NoCode?

“The future of coding is no coding at all”
- CEO Github, 2017

NoCode is the power of code ***without writing it.***

NoCode refers to tools and platforms which allow users to build software without the need to write code.

NoCode tools usually feature a visual interface, where users can drag and drop elements to create their app.

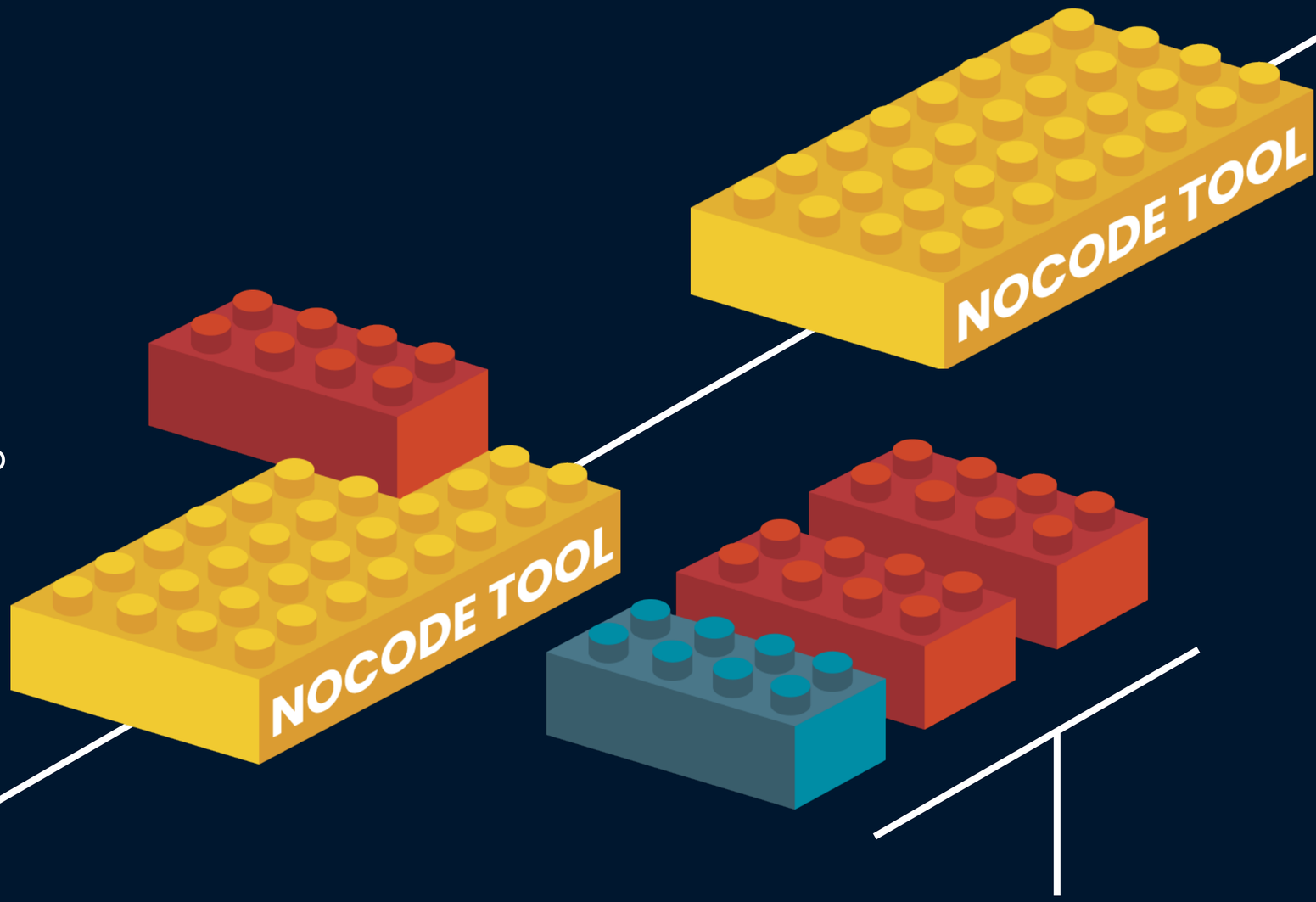


A ***new way*** of building products

NoCode tools *are like Lego*

Nocode tools come with pre-built components that you can drag and drop to your screen. These components are similar to lego blocks that are like your text, buttons, input fields…etc.

It is your job as a no-coder to stack these components together, just as you would with Lego blocks.



⚡ **Connect** different no-code **tools** together

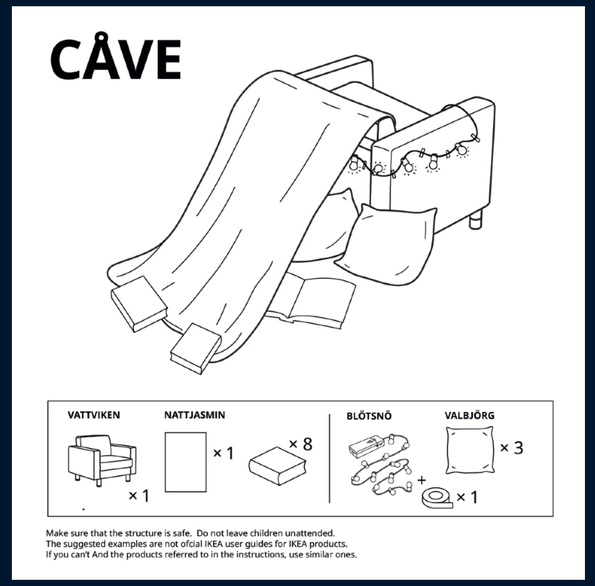
⚡ **Connect** pre-built **blocks** together

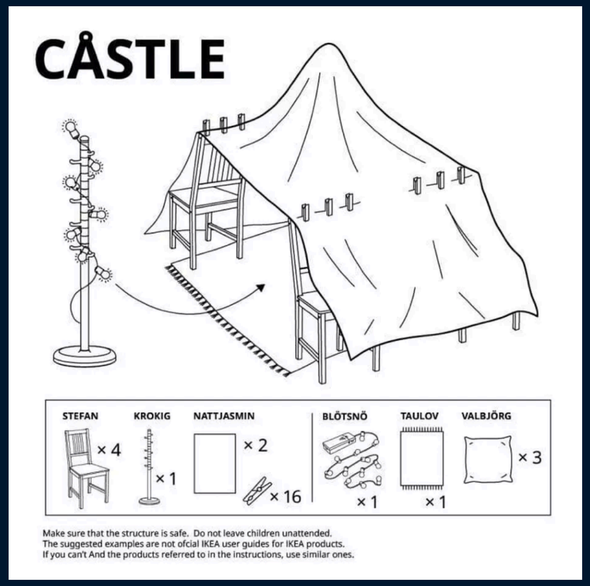
NoCode is *the new IKEA*

You receive a set of pieces to assemble the furniture. It is just like entering a nocode tool- you see a list of pre-built blocks you can choose from to build the solution you want.

Although you don’t have as much power or customizability as building from scratch with code, nocode makes it more accessible, cost affective and faster.

⚡ Piece pre-built blocks together





⚡ Simpler

⚡ More Affordable

⚡ Faster

Why is NoCode *a game changer*

Although you don’t have as much power or customizability as building from scratch with code, nocode makes it more accessible, cost affective or faster way.

⚡ Reduced development cost

Comparing with conventional development where you will need to recruit multiple technical developers, most of the nuances of front-end and back-end development are abstracted for NoCode application development. You don’t have to think about underlying systems, or security when you’re building on a platform.

Why is NoCode *a game changer*

Although you don’t have as much power or customizability as building from scratch with code, nocode makes it more accessible, cost affective or faster way.

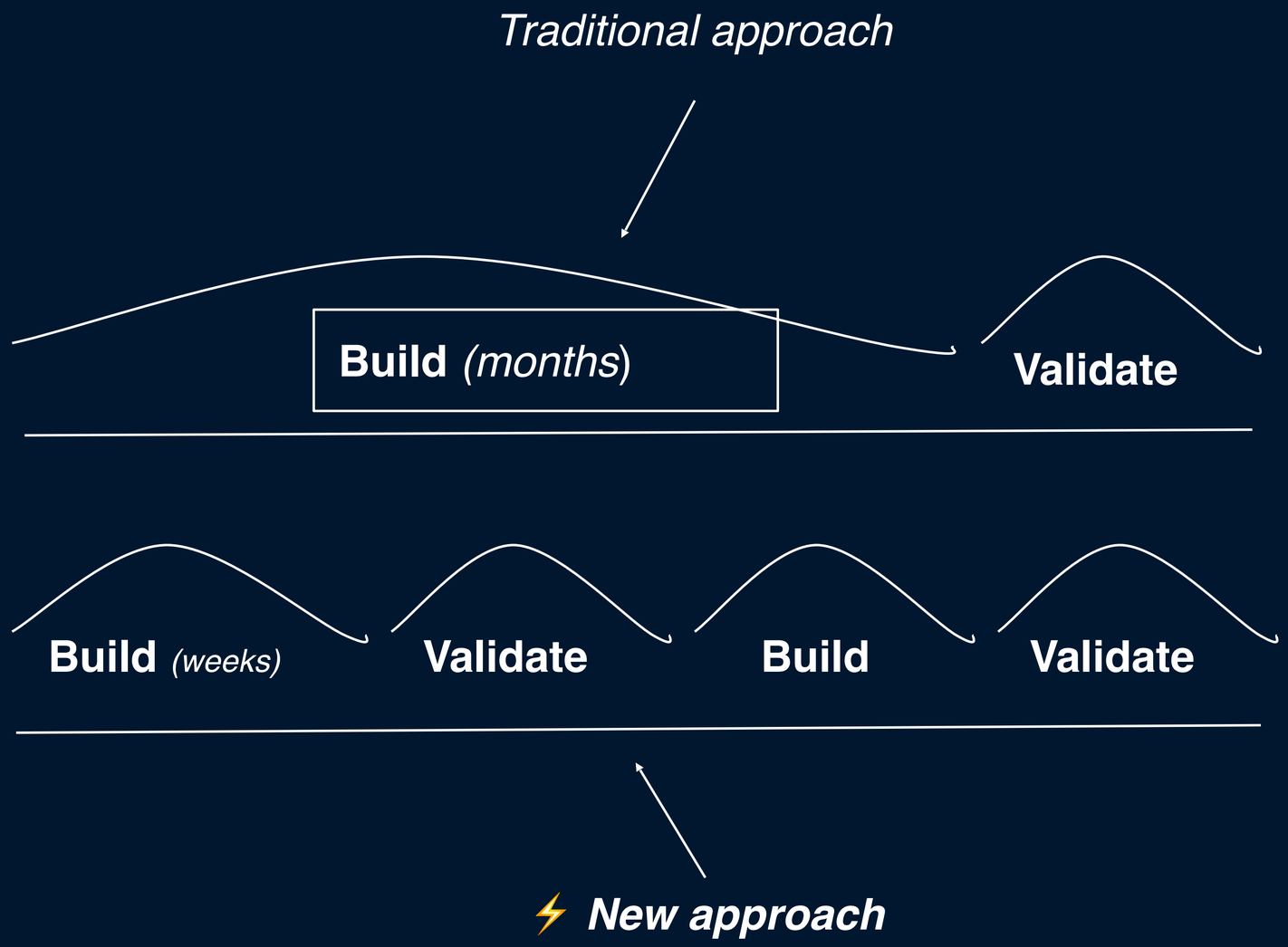
⚡ Focus more on your core business

NoCode simplifies a lot of the complexity when it comes to design the frontend and backend, giving you more time to focus on our customers and provide a better customer experience.

Build on your ***own terms.***

Everyone want to build without dependency. Making building faster, leading to a quicker turnaround, which allows us to spend more time focusing on the business and customers, saving time that would otherwise be spent on building with dependency on developers.

⚡ Short feedback loop



How it works?

Takes care of common functionality.

Things like Login, user accounts, payments, email, etc.. come almost ready out of the box.

NoCode

brings to the table

Takes care of common functionality.
Things like Login, user accounts, payments, email, etc.. come almost ready out of the box.

Makes user interface easier
With the option to drag and drop elements to build the interface, NoCode tools take care of most of the layout rules for you.

NoCode
brings to the table

Takes care of common functionality.
Things like Login, user accounts, payments, email, etc.. come almost ready out of the box.

Makes user interface easier
With the option to drag and drop elements to build the interface, NoCode tools take care of most of the layout rules for you.

NoCode
brings to the table

Handles most of the security and maintenance
Running a software will eventually require maintenance, but with NoCode tools most of it is done for you in the background.

Takes care of common functionality.
Things like Login, user accounts, payments, email, etc.. come almost ready out of the box.

Makes user interface easier
With the option to drag and drop elements to build the interface, NoCode tools take care of most of the layout rules for you.

NoCode
brings to the table

Handles scaling
Whether you have 1 user or 1 million users, NoCode tools will take care of the scaling for you in the background.
No worries about servers scalability.

Handles most of the security and maintenance
Running a software will eventually require maintenance, but with NoCode tools most of it is done for you in the background.

Primary builder *components*

Editor / Design
Allow you to drag and drop elements and build the interface of your app.

Database / information
Where you will store the information related to the app, like users, events, operation data, etc...

Workflows / Logic
They are where you will be able to create the rules to make the logic of your app work.

Elements & Blocks
The visual elements visuais of the app, website, … that we drag & drop to build the experience

List or Repeating groups
Element that allow you to show lists of information you have in your database.

What is actually possible to build with NoCode tools?

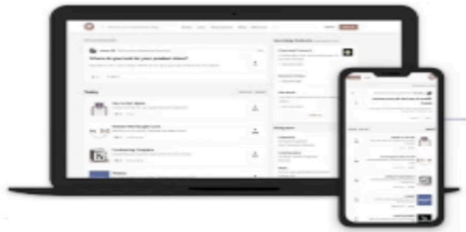
Limitless ***possibilities.***

The number of features available inside NoCode tools is vast. Even though it is impossible to cover everything here, we can at least break down the most common categories of apps most people want to build.

Landing pages
Websites
Apps
Software / Web-Apps
E-Commerce
Marketplaces
Bots
Games
Automations
Etc..

Examples of NoCode Developments

Login / Signup
User Management
Database
Complex Logic / Flows
Payments
Notifications
Etc…



CONNECTED
WITH NO-CODE

AI Model

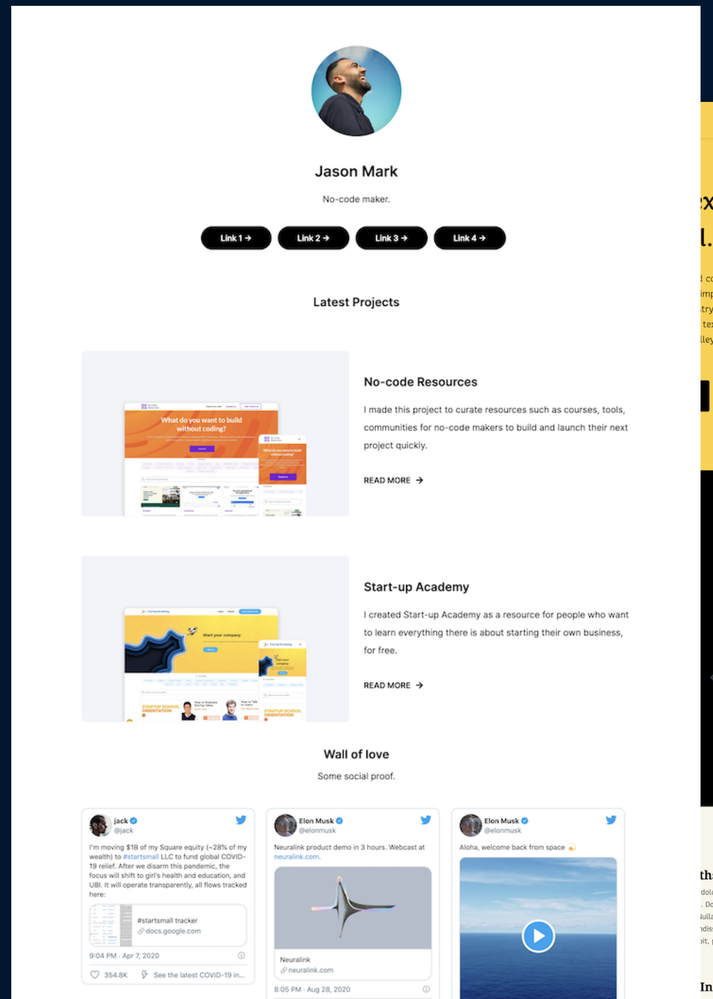
Data
Sources

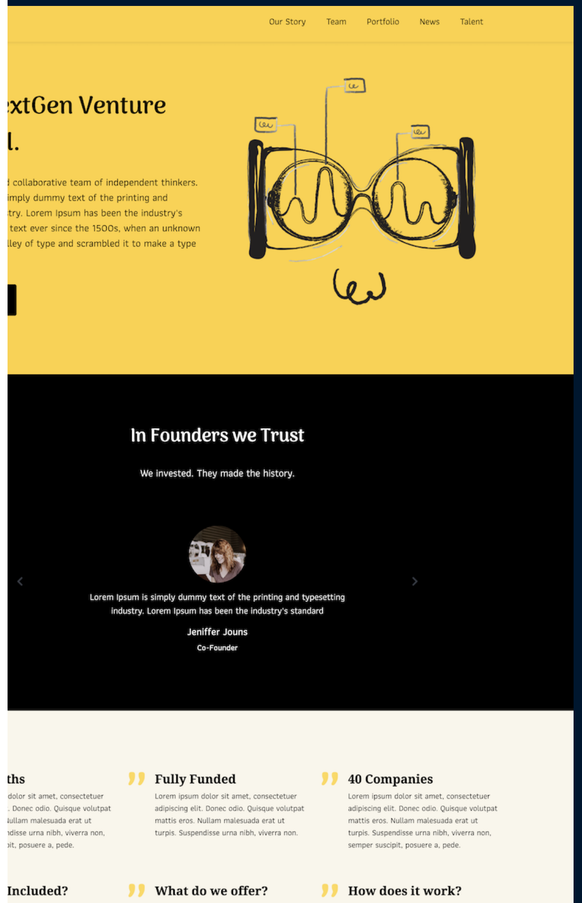
Amazon/
Google/…

Websites

Think of sites like your favorite brand. At their core, websites let vendors show their products, and then customers search for and buy those products.

These are common features and functionalities for building with nocode tools.

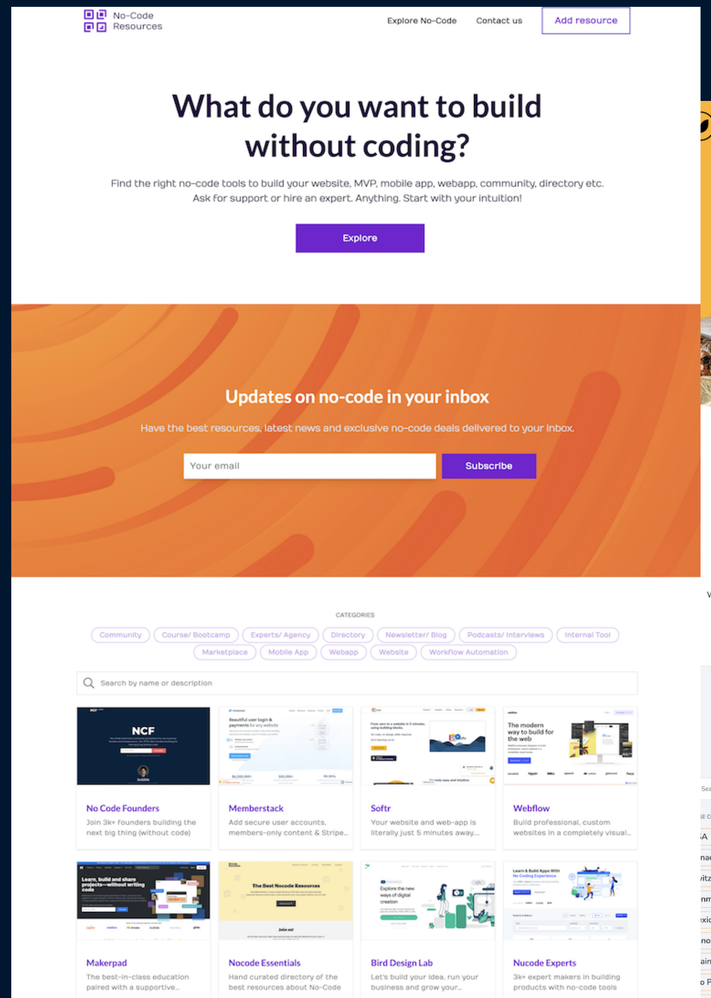


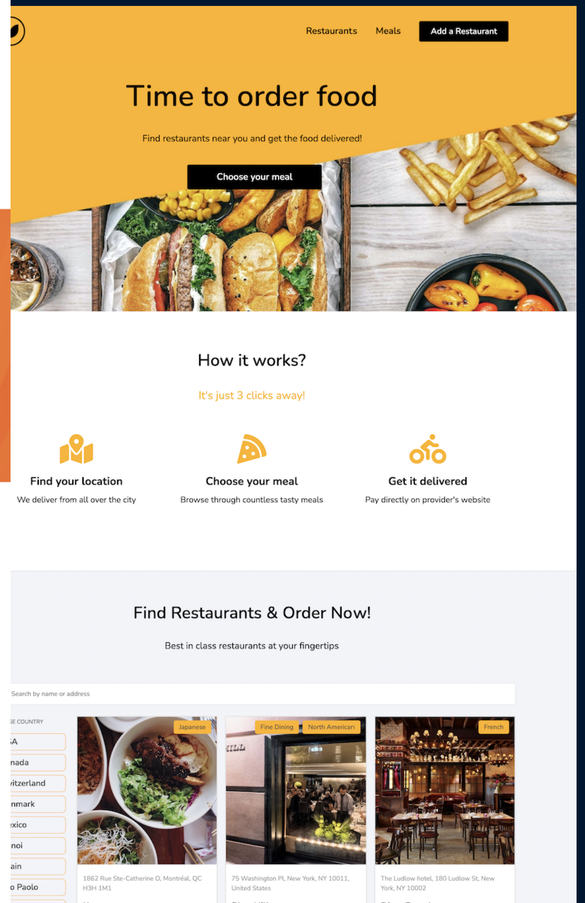


Directories

Directories are the most capable solutions for fulfilling requirements of organised and specific type of data.

This includes job board apps like Indeed or real estate listing apps like Zillow. Listing directories for small niches are a very common type of nocode app.

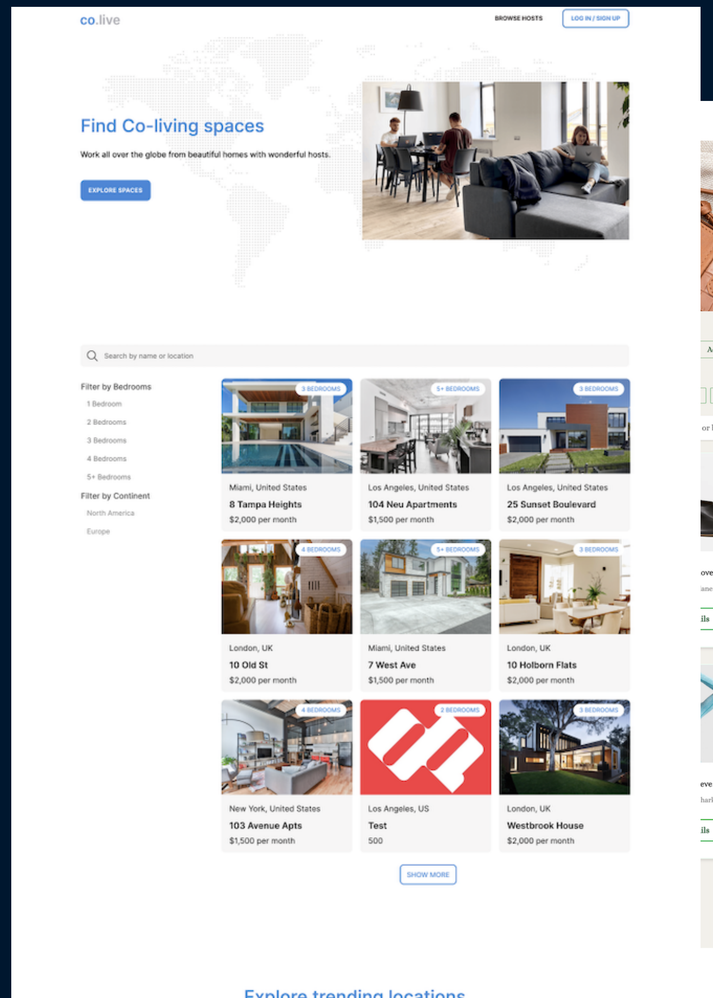


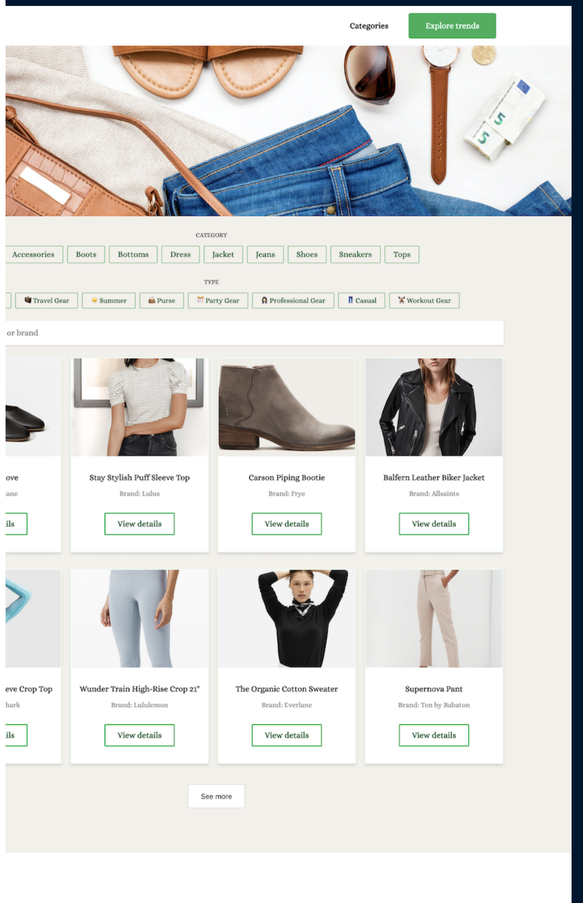


Marketplaces

Marketplace apps connect users. Big marketplace apps include Uber, DoorDash, and Airbnb.

These apps are huge, but at their core, they are pretty straightforward. Broad no-code platforms like Bubble.io can accommodate marketplace apps for your niche (many of our own clients are building marketplace apps).

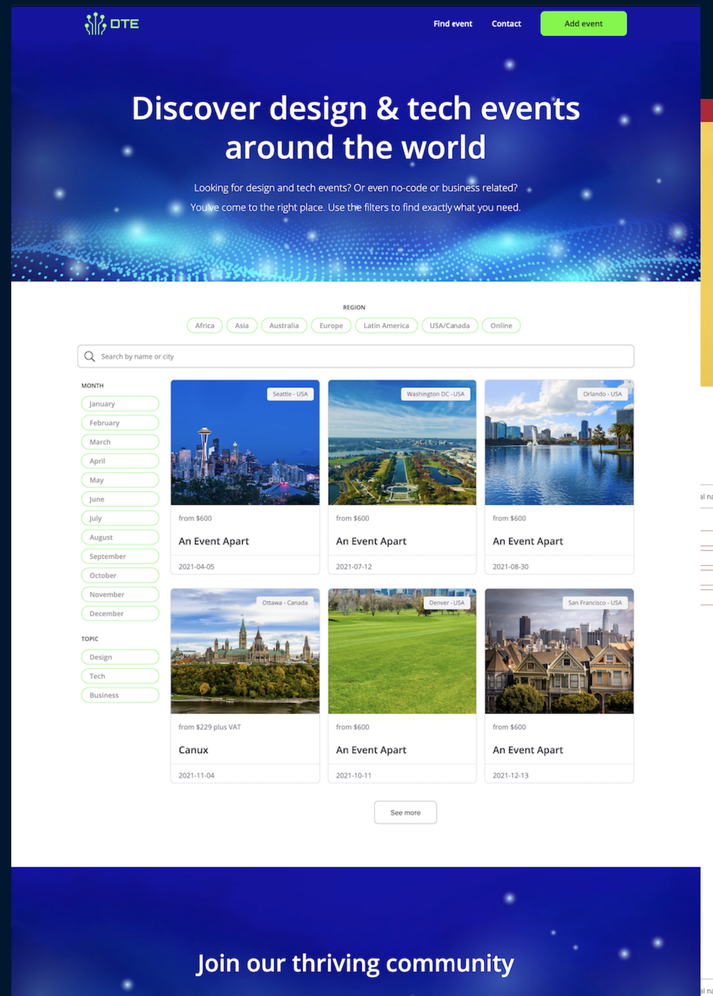


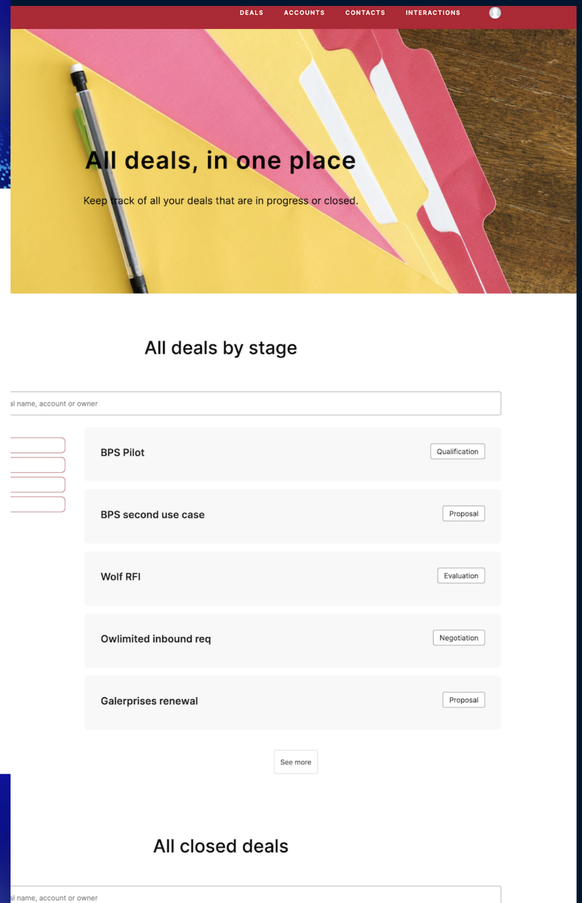


Internal Tools

Internal tools & dashboard apps are great for optimizing your processes and tracking statistics and analytics.

Think about Salesforce. Nocode apps can help you build dashboards that are customized to exactly what you want to track your industry or business.

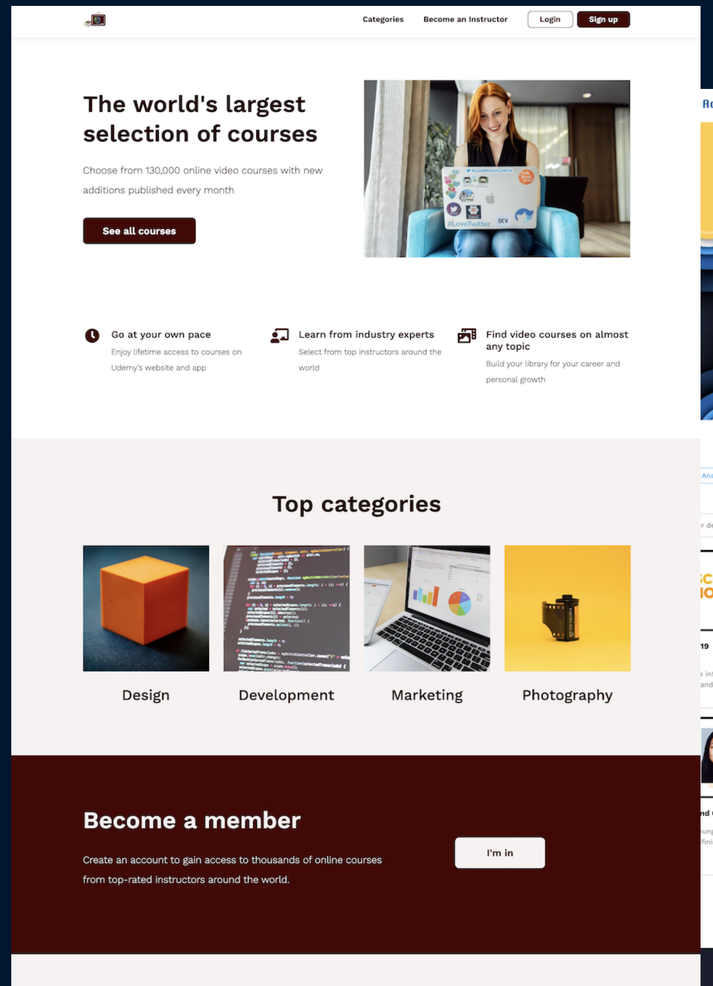


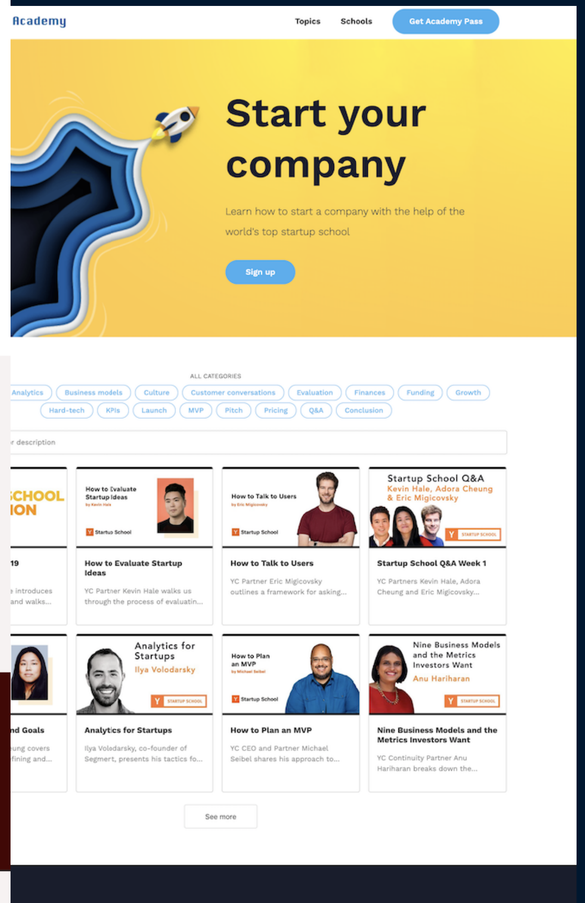


Communities

Online communities are the best solutions for people looking to create a private space for like-minded people.

This includes onlines communities, courses for students and other niche communities around specific topics.





Software

Progressive web apps (PWA) or Software As A Service (SaaS) can be built with NoCode.

Some examples:

Fintech

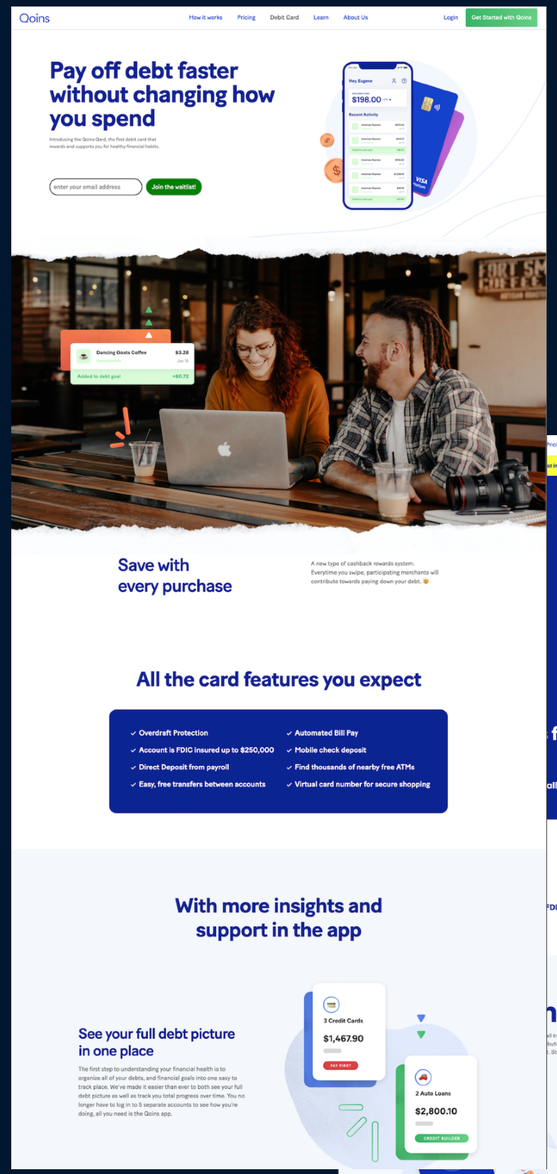
- *Qoins*
- *Dividend Finance*
- *Paytable*

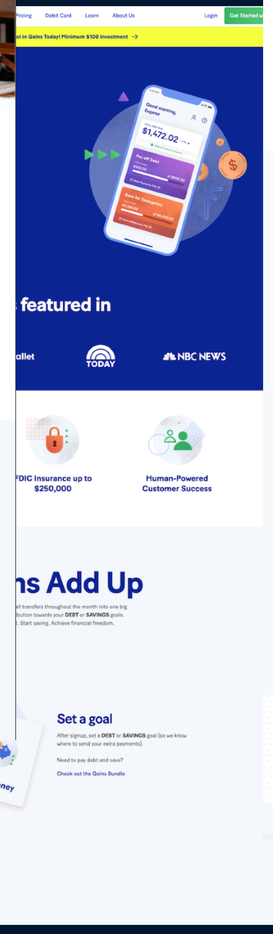
SaaS

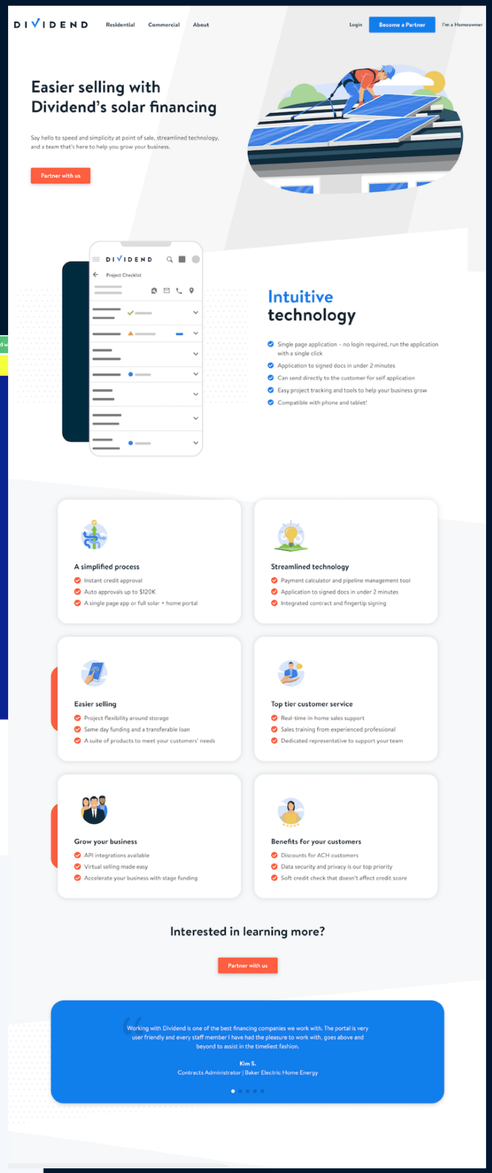
- *Userloop*
- *Benchmarq*
- *MooneHQ*

Health

- *Equinas*
- *Messly*







There’s a ***nocode tool*** for that.

We have reached a point where there’s a nocode tool popping up every week.

However, Nocode is not about building simple solutions. Some people have made everything from Facebook, Airbnb to Uber clones just with nocode to showcase what is possible.

Generic all-in-one builders
Which allow you to build software for one or more platforms (mobile, web, and others).

Niche builders
Which let you build a specific use-case, for example marketplaces, dashboards and landing pages.

A NO-CODE MAPPING

Product builders

WEB

Websites

webflow
WIX
SQUARESPACE
Carrd
Landen

Ecommerce

shopify
weebly
eCommerce by Square
Elliot
gumroad
shōgun

Web apps

bubble
unqork
budibase
tabbli
Boundless

Payment & Membership

Buy me a coffee
PICO

MOBILE APP

Adalo
glide.
AppSheet
thunkable
openasapp
Bravo
mendix
draftbit

AI & DATA SCIENCE

GYANA
lobe
ACCERN
obviously.ai
colabel
intersect labs

GAMING

BUILDBOX
CELESTERY
DREAMCRAFT ENTERTAINMENT
RIVE

AR & VR & 3D

MINSAR
scapic
Plug

AUTOMATION

tonkean
BRYTER
LEAPWORK
airSlate
axiom.ai

CHATBOT

ada
chatfuel
Landbot.io
ManyChat
mindsay

VOICE

voiceflow
vuix

INTERNAL TOOL

process.st
jet admin
Ninox
.internal
Retool

API

Sheetsu
Sheety
NoCodeAPI

DATA SCRAPING

PHANTOM BUSTER
Spider Pro
APIFY
Simplescraper

Base blocks

DATABASE & SPREADSHEET

Airtable
EXPLO
tadabase
dashdash
Google Sheets
actiondesk

CONNECTORS

zapier
integromat
PARABOLA
IFTTT
tray.io
Clay
Paragon
n8n.io

ENTREPRENEURSHIP

THANK YOU.

NOVA SCHOOL OF
BUSINESS & ECONOMICS