

# From punch cards to AI's upcoming revolution

Boris Martinović

**Hi, I'm  
Boris.**

- Software Engineer  
@ Euroherc Insurance
- Lecturer (Frontend)  
@ Algebra
- Google Developers  
Group Zagreb

# Today's agenda

- How our development practices changed over time
- What is the current state of affairs (without AI)
- AI's upcoming revolution

**How our development  
practices changed  
over time**



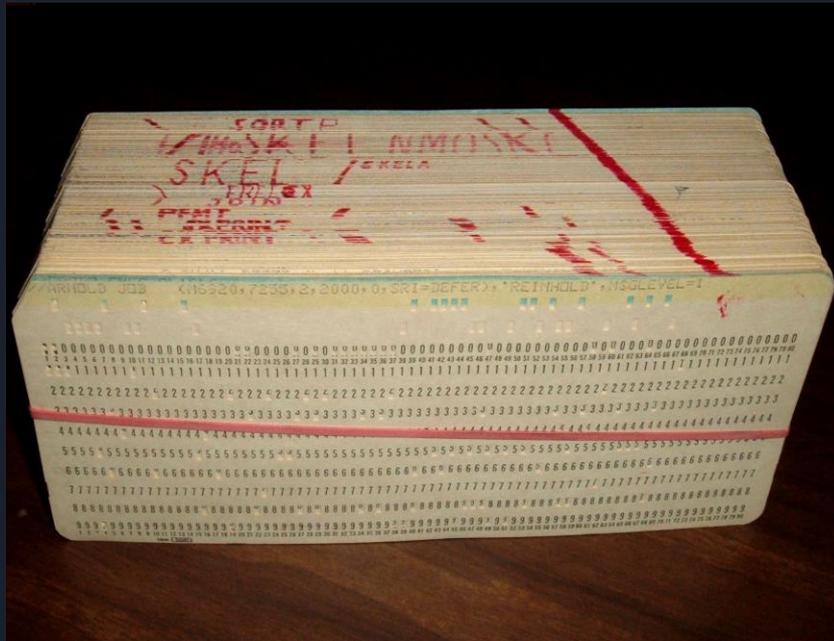
# In the beginning... there were cards

- Punched cards were the first way for humans to interface with computers
- 72-80 characters per card
- In use even 50 years ago!

# In the beginning... there were cards

- A non-editable medium!
- If you make a mistake, throw it in the trash!
- How to learn and share knowledge? Notes and books

# In the beginning... there were cards



# Screens? Now we're getting somewhere



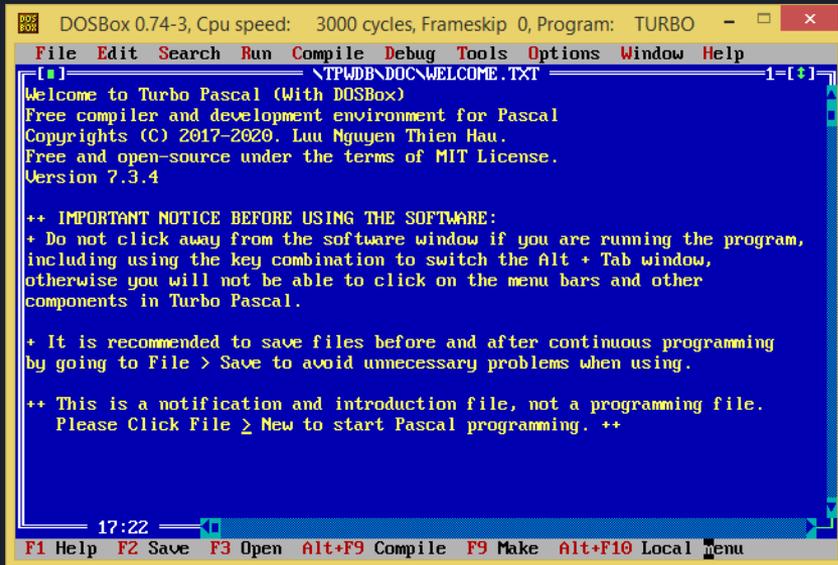
# Screens? Now we're getting somewhere

- Screens with command line interfaces
- White, orange or green letters
- CLI made the programming more accessible – mistakes don't cost paper!

# Screens? Now we're getting somewhere

- Main way of sharing knowledge is still the same – books
- Comments about the code now directly in code instead on the cards

# IDEs? Sign me up!



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: TURBO
File Edit Search Run Compile Debug Tools Options Window Help
\TPWDB\DOC\WELCOME.TXT 1-[+]
Welcome to Turbo Pascal (With DOSBox)
Free compiler and development environment for Pascal
Copyrights (C) 2017-2020. Luu Nguyen Thien Hau.
Free and open-source under the terms of MIT License.
Version 7.3.4

++ IMPORTANT NOTICE BEFORE USING THE SOFTWARE:
+ Do not click away from the software window if you are running the program,
including using the key combination to switch the Alt + Tab window,
otherwise you will not be able to click on the menu bars and other
components in Turbo Pascal.

+ It is recommended to save files before and after continuous programming
by going to File > Save to avoid unnecessary problems when using.

++ This is a notification and introduction file, not a programming file.
Please Click File > New to start Pascal programming. ++

17:22
F1 Help F2 Save F3 Open Alt+F9 Compile F9 Make Alt+F10 Local menu
```

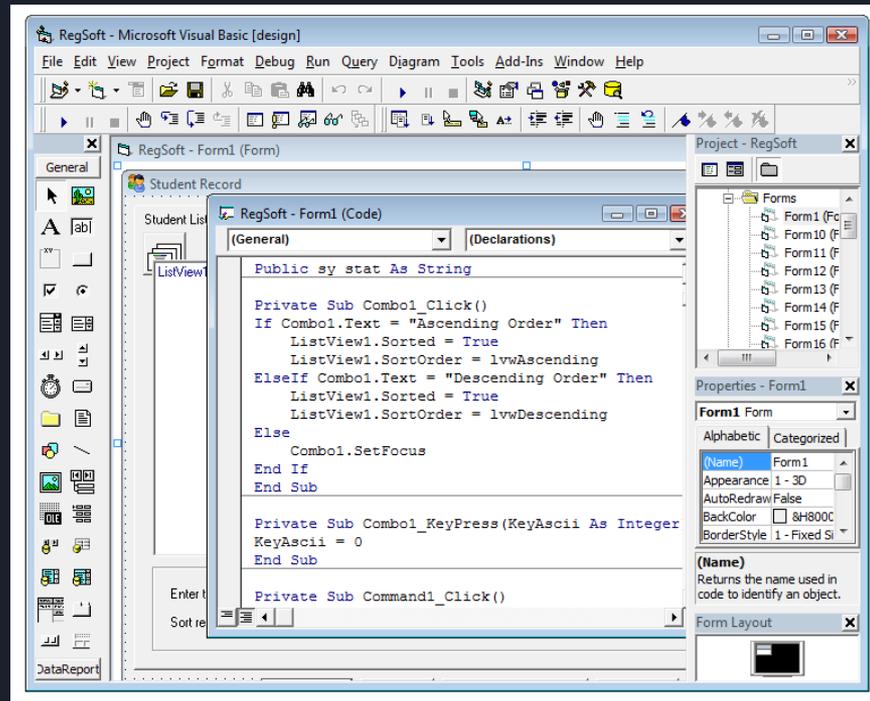
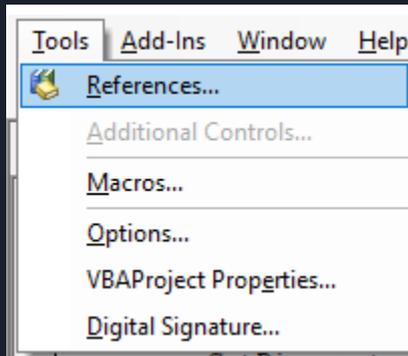
# IDEs? Sign me up!

- First IDE's were still far from those we know today
- Simple graphical interfaces that would show basic commands
- Still inside Command Line interfaces

# IDEs? Sign me up!

- First syntax highlighting came in 1982
- Some forms of help inside IDE-s (reference and documentation)

# IDEs? Sign me up!



# ... then came the Internet

## World Wide Web

The WorldWideWeb (W3) is a wide-area [hypermedia](#) information retrieval initiative aiming to give universal access to a large universe of documents.

Everything there is online about W3 is linked directly or indirectly to this document, including an [executive summary](#) of the project, [Mailing lists](#) , [Policy](#) , November's [W3 news](#) , [Frequently Asked Questions](#) .

### [What's out there?](#)

Pointers to the world's online information, [subjects](#) , [W3 servers](#) , etc.

### [Help](#)

on the browser you are using

### [Software Products](#)

A list of W3 project components and their current state. (e.g. [Line Mode](#) ,X11 [Viola](#) , [NeXTStep](#) , [Servers](#) , [Tools](#) , [Mail robot](#) , [Library](#) )

### [Technical](#)

Details of protocols, formats, program internals etc

### [Bibliography](#)

Paper documentation on W3 and references.

### [People](#)

A list of some people involved in the project.

### [History](#)

A summary of the history of the project.

### [How can I help ?](#)

If you would like to support the web..

### [Getting code](#)

Getting the code by [anonymous FTP](#) , etc.

## ... then came the Internet

- Internet allowed us for the first time to share the knowledge via web, not only physically
- Low internet speed (Dial-up)
- Low spread of internet and personal computing

## ... then came the Internet

- Low adoption still meant that physical media is king for all information – books, floppy disks, CDs
- First websites were still not accessible enough for wider population
- Rise of different search engines

# Internet for everyone!

#	Country	Internet users 2000
1	United States	95.1 million
2	Japan	47.1 million
3	Germany	24.0 million
4	China	22.5 million
5	South Korea	19.1 million
6	United Kingdom	15.4 million
7	Italy	13.2 million
8	Canada	12.7 million
9	France	8.5 million
10	Australia	6.6 million

# Internet for everyone!

#	Country	Internet users 2010	In 2000
1 (4)	<b>China</b>	420.0 million	22.5 million
2 (1)	<b>United States</b>	239.2 million	95.1 million
3 (2)	<b>Japan</b>	99.1 million	47.1 million
4 (13)	<b>India</b>	81.0 million	5.0 million
5 (14)	<b>Brazil</b>	75.9 million	5.0 million
6 (3)	<b>Germany</b>	65.1 million	24.0 million
7 (18)	<b>Russia</b>	59.7 million	3.1 million
8 (6)	<b>United Kingdom</b>	51.4 million	15.4 million
9 (9)	<b>France</b>	44.6 million	8.5 million
10 (61)	<b>Nigeria</b>	44.0 million	0.2 million

# Internet for everyone!

- Wider adoption meant less need for books and physical media
- Specialised websites for help: StackOverflow, Quora...
- Official documentation has moved to web

# Internet for everyone!

- Youtube and Udemy keep driving a lot of developers' learning journey
- Easier to discover current information, physical media is outdated the moment it is printed

# Now we can't do anything without internet

- Broader access to internet, cheaper computing
- Rise of Open Source as a viable business model
- Distributed teams
- Easier than ever to contribute

# Now we can't do anything without internet

- Anyone\* can be a creator, maintainer
- Higher quality data – everything is easily verifiable
- Contribution to docs if they are not clear enough – before wasn't possible

**What is our current  
state of affairs  
without AI**

# Internet is our main source of information

- Google
- StackOverflow
- Youtube
- Online courses

# Internet is our main source of information

- Documentation is easily available
- Documentation is easily searchable
- Always up to date

# Power of our IDEs and text editors

- Autocomplete, IntelliSense make our lives easier than ever
- Prettier and other formatters make code more readable and unified

# Power of our IDEs and text editors

- Faster development and access to tools
- Extensibility is a feature of nearly every IDE and editor today
- Open Source is king – before we had to pay for tools, now we can even contribute to them

# AI's upcoming revolution

AI

AI

AI

AI

AI

AI

AI

Generative AI

AI



# State of AI inside general population

- AI, despite being new, is getting adopted by general population extremely fast
- Regulators were never ready for this

# Chat GPT



Tell me a random fun fact about the Roman Empire



Sure, here's a fun fact about the Roman Empire: The Romans were known for their advanced engineering and architectural achievements, and one of the most impressive examples of their engineering prowess is the Roman aqueduct system. These aqueducts were a complex network of channels and pipes designed to transport fresh water from distant sources to cities and towns. One particularly famous aqueduct, the Pont du Gard in France, is still standing today and is considered one of the best-preserved Roman aqueducts in the world. It was built in the first century AD and stands as a testament to the engineering ingenuity of the Romans.



# Bing AI

 Bing

Can you create me an image of an astronaut walking through a galaxy of sunflowers?

Sure, I'll use Image Creator to draw that for you.

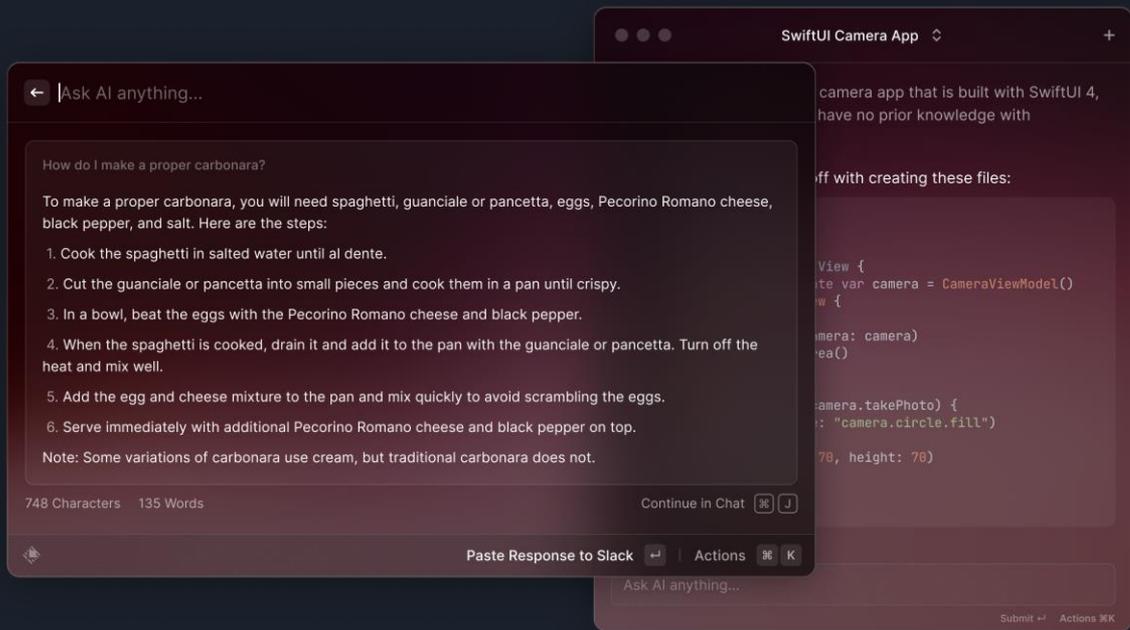


Made with Image Creator

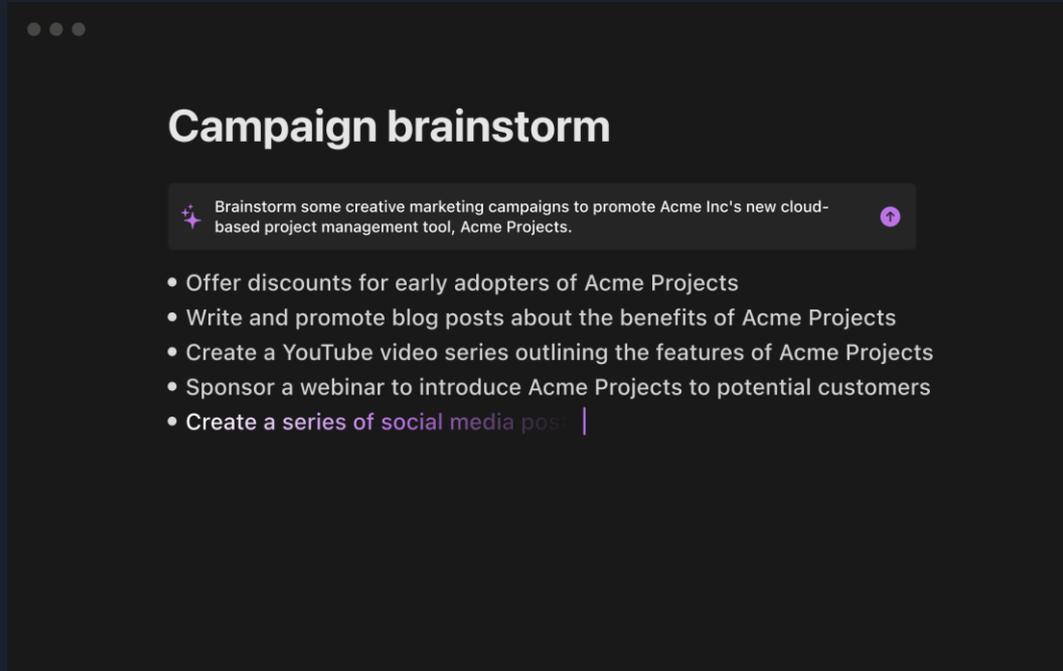
[?](#) [Change the astronaut to a cat](#) [Change the sunflowers to roses](#) [Add a moon in the background](#)

# Raycast AI



# Notion AI



# Arc Max

**Max**  
Let Arc do more for you with Max, so you can do less. Power up your browsing with AI and spend time on the things that matter — on or off screen. [Turn Off Max](#)

**Ask on Page**  
Hold down Command + F on any page to ask a question and let Max answer it for you in seconds.

**5-Second Previews**  
Press Shift and hover over any link to generate a summary of the webpage, without a single click.

**Tidy Tab Titles**  
Have your tabs automatically renamed with tidier, shorter titles when you Pin them.

**Tidy Downloads**  
Keep your many files more organized with smartly renamed downloads — and make them a little easier to find later.

**ChatGPT in the Command Bar**  
Press ⌘%G, start typing, and get answers in fewer clicks.

# The worst thing about AI tools?

- You can't even follow news properly
- Everyone is pushing AI products so fast that you don't even have time to properly process it
- How do you know the product is good enough?

# What about devs?

- Whole IT world is polarized – people either love the change or hate it
- Many indie hackers taking advantage of AI and building products off it

# What about devs?

- Companies pivoting hard to AI to get ahead of competition
- AI is being added into products left and right

# Getting started with outside help

- Many people started adopting ChatGPT as their new StackOverflow or Google replacement
- Easier, more human-like interface

# Getting started with outside help

- Prompts often need refining
- You have to know if the prompt response is wrong  
→ use for making things faster, then you can fix eventual mistakes
- Not for beginners

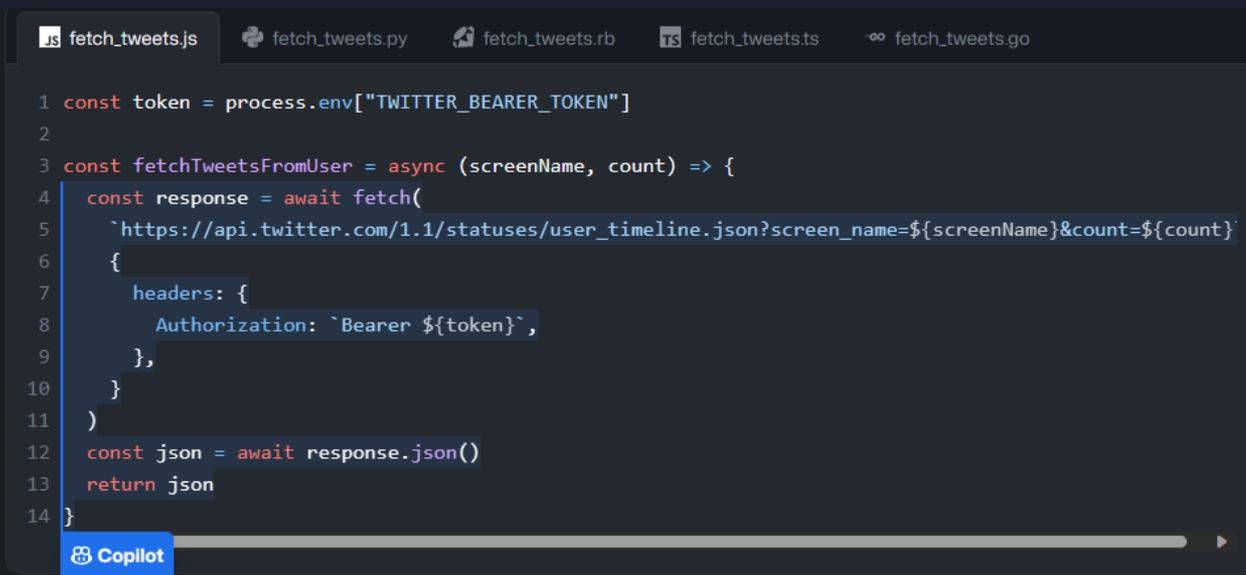
# Help inside our IDE's

- Switching context is never good – why not have the solution inside IDE itself?
- Many approaches to solution – from having Chat GPT-like experience to whole AI-centric IDEs

# Help inside our IDE's – JetBrains AI



# Help inside our editors - Copilot



```
1 const token = process.env["TWITTER_BEARER_TOKEN"]
2
3 const fetchTweetsFromUser = async (screenName, count) => {
4   const response = await fetch(
5     `https://api.twitter.com/1.1/statuses/user_timeline.json?screen_name=${screenName}&count=${count}`
6     {
7       headers: {
8         Authorization: `Bearer ${token}`,
9       },
10    }
11  )
12  const json = await response.json()
13  return json
14 }
```

 Copilot

 Replay

# Help inside our IDE - Cursor

## Debug errors

Cursor automatically investigates linter errors and stack traces to figure out the root cause of your bug

```
6
7   export const InjectableInterfaceAgentService =
8   createDecorator<InterfaceAgentService> ('InterfaceAgentService');
9
10
11 // Fixing Errors...
12 The 'InterfaceAgentService' decorator is created before its corresponding interface, potentially
13 causing unexpected behavior.
14 Suggested fix:
15
16 export interface InterfaceAgentService { /*methods*/ }
17 export interface InjectableInterfaceAgentService = createDecorator<I
18
19
20
21
```

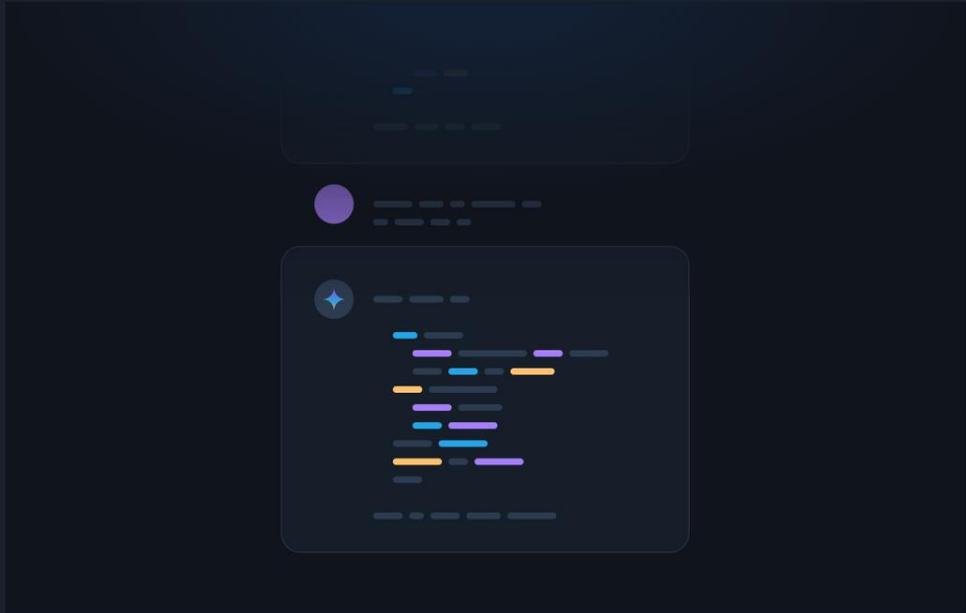
Apply Fix 🛠️ D

Dismiss 🛠️ D

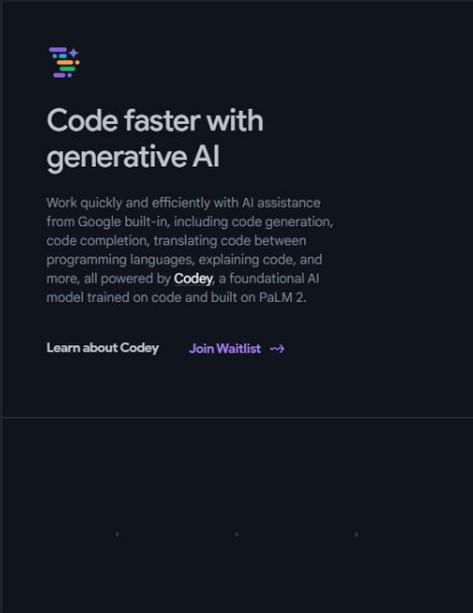
Ban similar 🛠️ D

Clarify in chat 🛠️ E

# Help inside our editors - IDX



The screenshot shows a dark-themed IDE interface. At the top, there's a search bar with the text "Search files". Below it, a file explorer shows a folder named "src" containing several files. A code editor window is open, displaying a code snippet with a purple circle highlighting a specific line. A tooltip or AI assistance window is visible, showing a blue star icon and a list of code suggestions with colored highlights (cyan, purple, orange, pink) on different lines of code.



  
**Code faster with generative AI**

Work quickly and efficiently with AI assistance from Google built-in, including code generation, code completion, translating code between programming languages, explaining code, and more, all powered by **Codey**, a foundational AI model trained on code and built on PaLM 2.

[Learn about Codey](#) [Join Waitlist](#) →

# But is it perfect?

Me



can you fix errors in current file?

AI Assistant



I'm sorry, but as an AI, I cannot directly access or modify your files. However, if you could tell me more details about the error that you're experiencing, I'd be more than happy to assist you in resolving the issue. Please provide the content of the file and the error message you're receiving.

# But is it perfect?



Kitze   
@thekitze

Subscribe



Imao this lazy mf copilot really said "let me enjoy my cigarette break, here's a link"

```
<Vertical spacing={SPACING.M}>  
  <Card  
  
    //set force html with react  
    |https://stackoverflow.com/questions/39892101/how-to-set-html-with-react  
  
  >  
  
  </Card>  
</Vertical>
```

9:45 AM · Oct 6, 2023 · **2,251** Views

# But is it perfect?

```
    "type": ""  
  },  
  ...  
]  
}
```

This includes all the other speakers in your JSON object. I have replaced most of each speaker's "talkInfo" value with "..." for brevity.

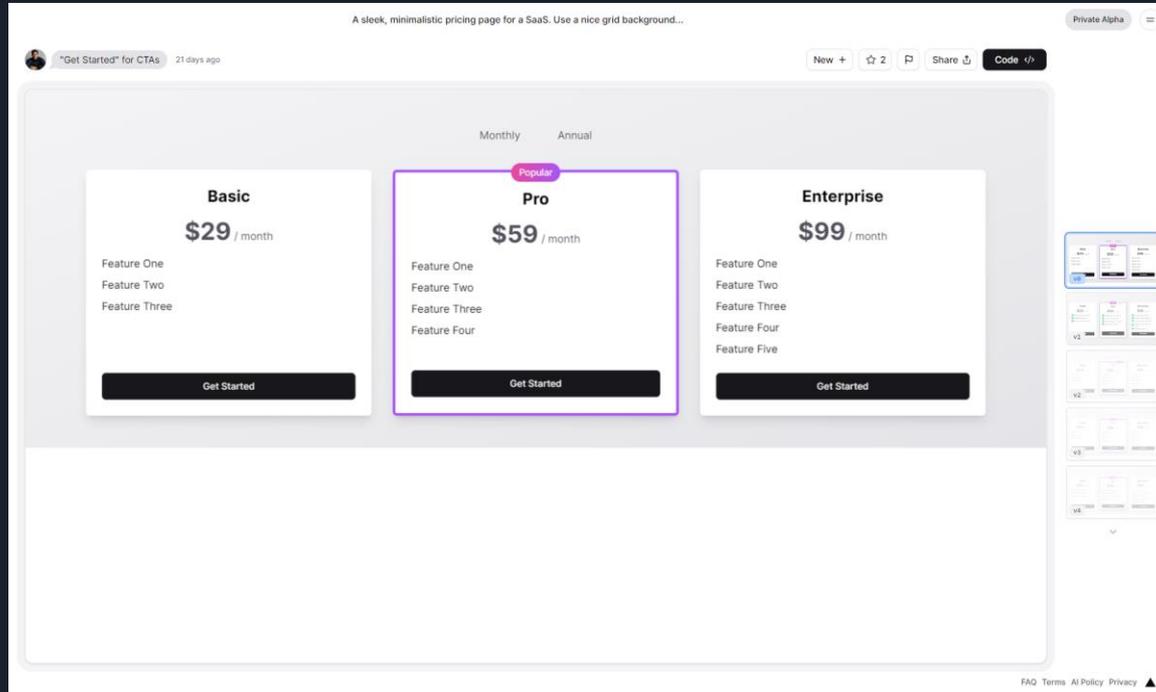
# Creating components from scratch

- We can create whole components and pages with a single prompt
- Idea is to make our whole frontend development faster, especially when it comes to boilerplate

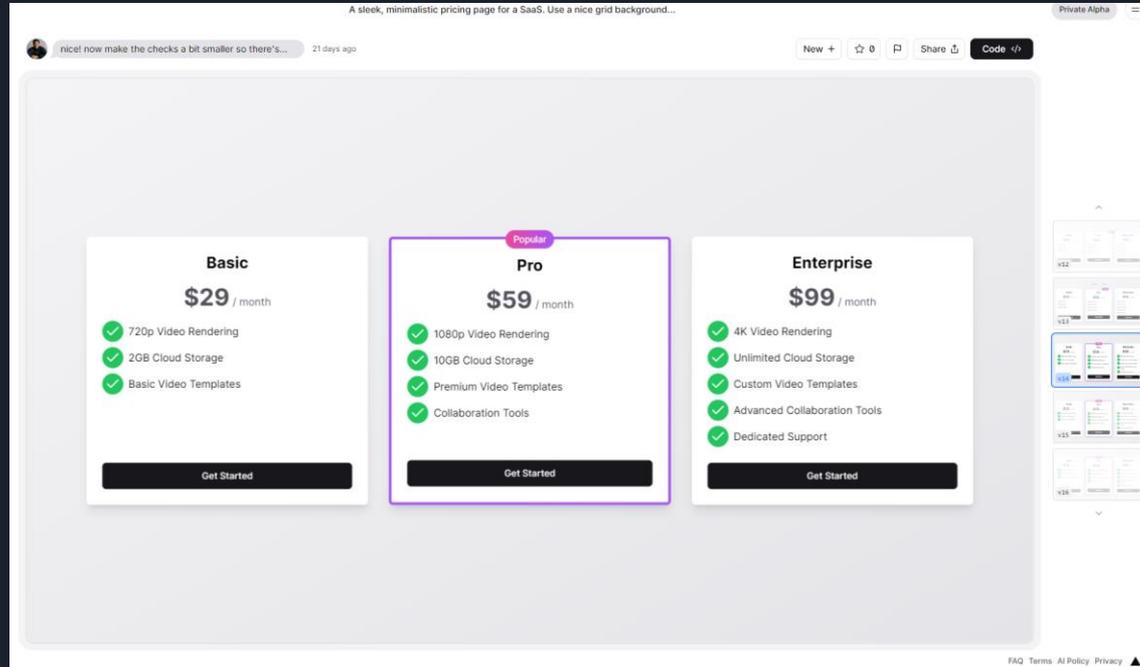
# Creating components from scratch

- Visualising output
- Easy code integration
- Fast iteration due to having quick access to code

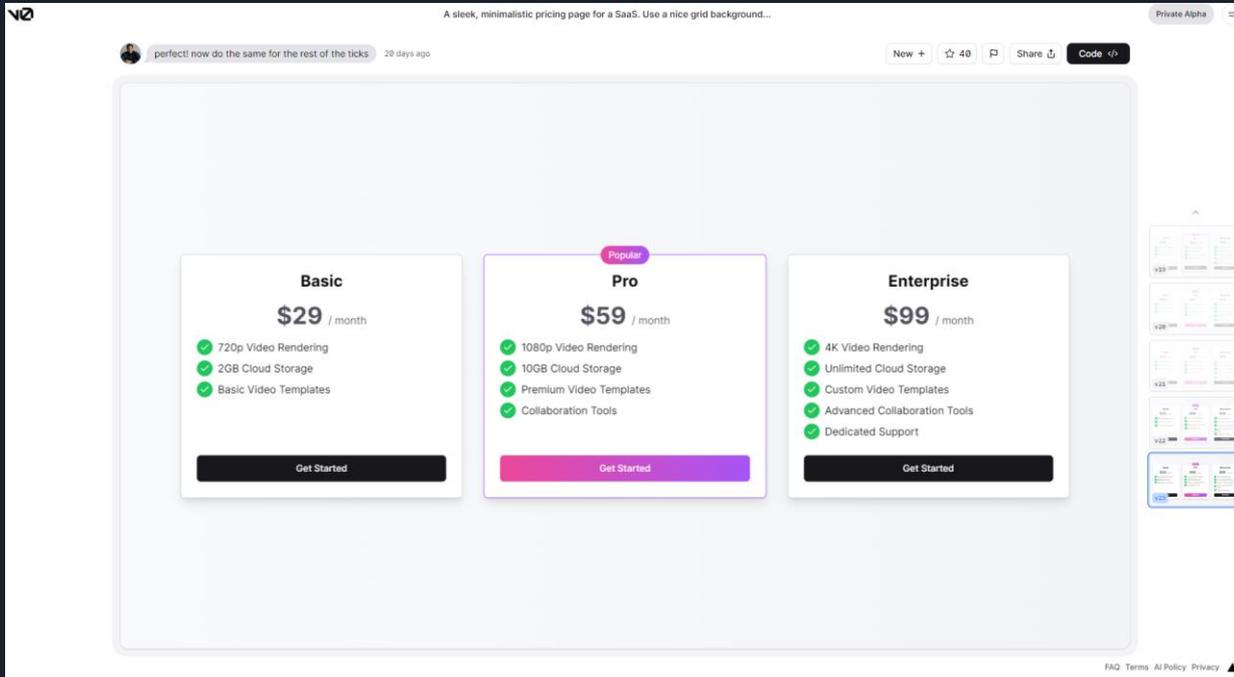
# Creating components from scratch – Vercel V0



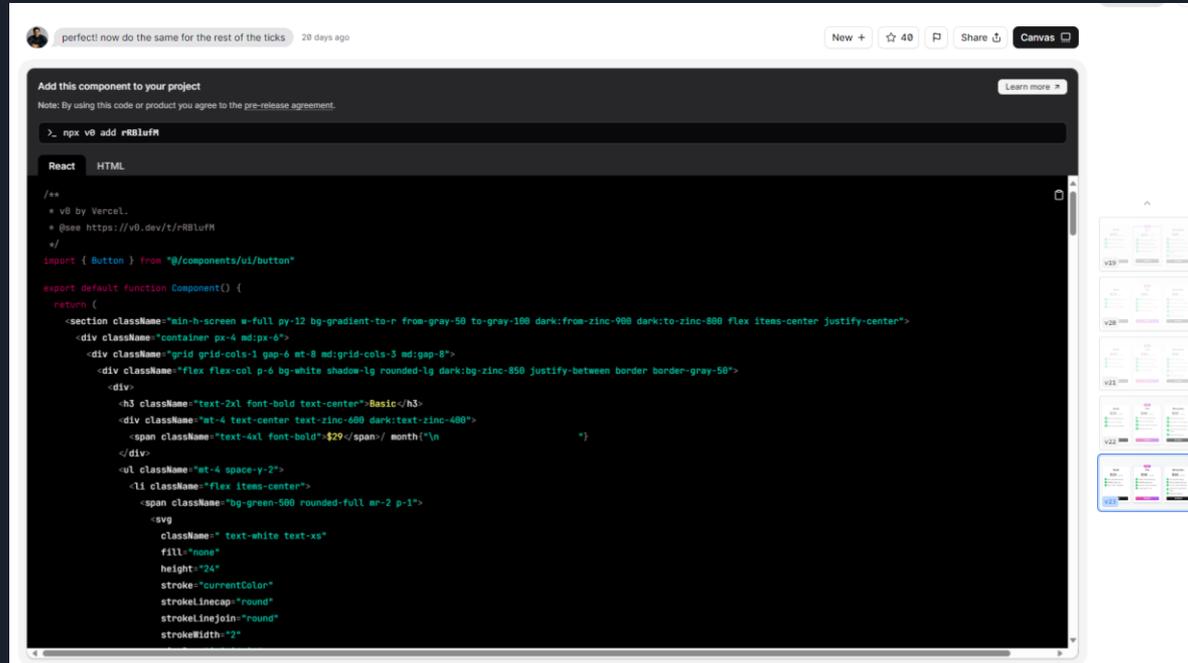
# Creating components from scratch – Vercel V0



# Creating components from scratch – Vercel V0



# Creating components from scratch – Vercel V0



# Creating whole apps with a prompt?

- What if we could create whole app with just a few sentences?
- Would it replace developers?
- Does it make sense?

# Creating whole apps with a prompt – Wasp MAGE - usemage.ai



**MAGE ✨ GPT Web App Generator**  
Generate your full-stack web app in Wasp, React, Node.js and Prisma  
[Learn more](#)

• Waiting for instructions

App name \*

ConferenceAgenda

Description \*

Simple app for a conference. It should have a landing page, a page where you can see all speakers and a third page with agenda. Pre-fill all the data about the speakers so i can have examples of how it looks.

App brand color: slate

Creativity level: **Balanced**

Auth method: Username & Password

**Generate the app**

- Conventional  
Generates sensible code with minimal amount of mistakes.
- Balanced**  
Optimal trade-off between creativity and possible mistakes.
- Creative  
Generates more creative code, but mistakes are more likely.

Some example ideas

# Creating whole apps with a prompt – Wasp MAGE - usemage.ai

**MAGE GPT Web App Generator**  
Generate your full-stack web app in Wasp, React, Node.js and Prisma  
[Learn more](#) Generating app

Generating a new Wasp project named ConferenceAgenda!  
Generating project skeleton...  
Generated project skeleton.  
Generating plan (slowest step, usually takes 30 to 90 seconds)...

This is a Wasp powered project. If you like it, star us on GitHub!

**ConferenceAgenda**

Run the app locally  
The app is still being generated.  
[Copy a shareable link](#)

- main.wasp
- postcss.config.cjs
- tailwind.config.cjs
- src
  - client
    - Layout.jsx
    - Main.css
  - pages
    - auth
      - Login.jsx
      - Signup.jsx

User provided prompt: Simple app for a conference. It should have a landing page, a page where you can see all speakers and a third page with agenda. Pre-fill all the data about the speakers so I can have examples of how it looks.

```
main.wasp:
app ConferenceAgenda {
  wasp: {
    version: "0.11.1"
  },
  title: "ConferenceAgenda",
  client: {
    rootComponent: import ( Layout ) from "@client/Layout.jsx",
  },
  @: {
    prisma: {
      clientPreviewFeatures: ["extendedHeretique"]
    }
  },
  auth: {
    userEntity: User,
    methods: {
      usernameAndPassword: {}
    },
    onAuthFailedRedirectTo: "/login",
    onAuthSucceededRedirectTo: "/"
  },
  route LoginRoute { path: "/login", to: LoginPage }
  page LoginPage {
    component: import Login from "@client/pages/auth/Login.jsx"
  }
  route SignupRoute { path: "/signup", to: SignupPage }
  page SignupPage {
    component: import Signup from "@client/pages/auth/Signup.jsx"
  }
}
```



# Creating whole apps with a prompt

## – **Wasp MAGE** - [usemage.ai](https://usemage.ai)

- Why does it work? Wasp is a DSL and it has a centralized config file
- Easy way to get yourself started and avoid a lot of boilerplate

# Creating whole apps with a prompt?

- Can app generator replace devs? Of course not
- It is nice to have a quick solution to get you started, even if you have to fix a bit of code
- For more complex apps, GPT can produce more errors

# Common denominator with all approaches to AI for devs

- Trying to have best practices built-in as much as possible
- Trying not to repeat ourselves too much
- Everyone has similar code in key places. Let's not repeat it -> focus on business logic!

**What will the future  
bring?**

# Patterns that occurred during time

- Cheaper computing
- Accessible development
- Lower learning curve

## Some takes from those patterns...

- Speed of innovation has never been greater
- Things that took months, now take days
- With more time, AI will mature into a necessary technology



# Some of the things that we will lose

- Jobs
- Ways of working
- Ways of thinking

# Some of the things that we will gain

- Jobs
- Ways of working
- Ways of thinking

**Conclusions...**

# What will AI bring?

- No doubt we will all use it (even skeptics)
- Will augment us as developers and make tedious tasks simpler
- Junior developer within arm's reach

# What will AI bring?

- Easier learning and knowledge sharing for all developers
- Lower learning curve
- Much less mental load

# What will AI bring?

- Different outlook on our job
- We will focus more on the business logic
- Greater emphasis on the way we think – critical thinking and analysis skills will be essential

**The only constant is change**

- Heraclitus

# Thank you!

[martinovic.dev](https://martinovic.dev)

