

TA Hardcoded Title

TT Hardcoded issue #Title This is a \*Markdown\* formatted issue body. You can **\*\*bold\*\***, *\_italicize\_*, and create lists: - Item 1 - Item 2 - Item 3

TA Hardcoded Title

TT Hardcoded issue This is a \*Markdown\* formatted issue body. You can **\*\*bold\*\***, *\_italicize\_*, and create lists: - Item 1 - Item 2 - Item 3

TA Hardcoded Title

TT Hardcoded issue body

Certification for gong course

None

Forest Gong Puja 11 hours playing the gong.

None

22 website

None

Workshop Silence with Gongs

Lage en ny workshop

I made these 23 websites (and earned \$562,943)

[https://youtube.com/watch?v=hCMEGQfLIEM&si=t\\_\\_sBXcuyn5axEA9](https://youtube.com/watch?v=hCMEGQfLIEM&si=t__sBXcuyn5axEA9)

NodeJS can do WHAT now? #javascript #server #webdevelopment #coding

<https://youtube.com/shorts/J3W3s5O9wAs?si=euvScg7FrCavxXpw>

3-year-old boy leaves professional engineers SHOCKED ■

[https://youtube.com/shorts/IUIqVFpMquk?si=Z6\\_V9ldbKDSHtzc](https://youtube.com/shorts/IUIqVFpMquk?si=Z6_V9ldbKDSHtzc)

Nvidia CEO Jensen Huang and the \$2 trillion company powering today's AI ...

<https://youtube.com/watch?v=DpQQi2scsHo&si=RoKKhnSUfWWZTaI2>

I earn \$1,753/day with this SIMPLE tech stack

[https://youtube.com/watch?v=b\\_ANHgTHNSI&si=PADYWo0E1gmHCNiI](https://youtube.com/watch?v=b_ANHgTHNSI&si=PADYWo0E1gmHCNiI)

The Easiest Way to Build Websites

[https://youtube.com/watch?v=OjEg0IBR\\_ak&si=-T8FmB8Oo4G5MCFK](https://youtube.com/watch?v=OjEg0IBR_ak&si=-T8FmB8Oo4G5MCFK)

Websocket Stream API

[https://youtube.com/shorts/wkcEePjMMDg?si=1QrZePm\\_Y6VWd5IF](https://youtube.com/shorts/wkcEePjMMDg?si=1QrZePm_Y6VWd5IF)

Multiple income streams

[https://youtube.com/shorts/Ay\\_hu2hl0qE?si=hl2YGfFW26K8IA6W](https://youtube.com/shorts/Ay_hu2hl0qE?si=hl2YGfFW26K8IA6W)

Boitano

https://www.boitano.no/

Deal.ai

https://deal.ai/order-618701171715118724840?fbclid=IwAR3Ksyr7JwH6IEBUvUfyme7cbe0co8YSL-wCNXxaHFJbYQ-eJB6XEmCJYO4\_aem\_AQYg3vM6no-PsSCrFvr\_g

Pilot Studies

None

SlowYou Teacher Training

None

Hardcoded Title

Hardcoded issue body

Twillio

https://youtu.be/WMbsxYfqTWg?si=69EgcoALcoSHJjnz

Python Sendgrid?

Er dette et alternativ?

Python WEB SCRAPING in 30 Seconds! ■■■■ #shorts

https://youtube.com/shorts/AaWaE2CEUF0?si=9i4sYZ3sVu83aoi4

ScrapeGraphAI: A Web Scraping Python Library that Uses LLMs to Create Scraping Pipelines for Websites, Documents, and XML Files

https://www.marktechpost.com/2024/04/30/scrapegraphai-a-web-scraping-python-library-that-uses-llms-to-create-scraping-pipelines-for-websites-documents-and-xml-files/

Summit Food Revolution

https://summit.foodrevolution.org/broadcasts/?utm\_source=sfmc&utm\_medium=email-pro&utm\_campaign=frs-24&utm\_content=replay-weekend-p3-frs-broad-members&j=

Branded intellectual property

For Universi AS, which owns the SlowYou Training concept, there are several areas where intellectual and sponsorship rights could be leveraged to create revenue streams: 1. **Licensing Content and Curriculum:** - Universi AS can license its structured training programs, including detailed exercise regimens and principles, to other wellness organizations or fitness trainers. This could include video tutorials, written content, and even certification programs for other trainers. - **Example:** Offering an official certification program for other trainers to become certified SlowYou practitioners could be a significant source of revenue. 2. **Branded Merchandise:** - Selling merchandise that supports the SlowYou lifestyle could be another avenue. This includes items like branded yoga mats, clothing, or wellness journals that align with the bioenergetic and holistic principles of SlowYou. - **Example:** Create and sell a line of eco-friendly, SlowYou-branded yoga wear. 3. **Digital Platforms (SlowYou.io and SlowYou.net):** - The development of digital platforms such as SlowYou.io for educational purposes and SlowYou.net for marketing and CMS solutions offers significant potential for monetizing intellectual property through subscription models, paid content, or even app-based coaching. - **Example:** A subscription-based mobile app providing daily SlowYou exercises and meditations. 4. **Sponsorship and Partnerships:** - Partnering with wellness-focused brands could offer mutual benefits. Sponsorships could help fund the expansion of SlowYou programs in exchange for brand visibility at events, in promotional materials, and across digital content. - **Example:** A partnership with a health food brand could involve co-branded content that promotes both the food products and SlowYou exercises as complementary for a healthy lifestyle. 5. **Publishing and Media:** - Producing and selling books, e-books, or audiobooks that guide users through the SlowYou principles and exercises. These could range from detailed guides to inspirational stories from long-time practitioners. - **Example:** A published series of guidebooks or a memoir by Tor Arne Håve outlining his journey with bioenergetic analysis and the creation of SlowYou. 6. **Special Events and Retreats:** - Hosting and organizing wellness retreats and workshops can be lucrative. These events can be sponsored or run as premium experiences where participants pay to learn SlowYou techniques in a retreat setting. - **Example:** Annual SlowYou retreats in scenic locations that offer deep dives into practice, sponsored by health and wellness brands. Each of these strategies incorporates elements of intellectual property inherent to the SlowYou training method and philosophy, creating a diversified portfolio of revenue streams while expanding the reach and impact of the SlowYou brand.

## Planning Node.js App

Absolutely, organizing the `views` folder in a module-based structure is a great strategy to maintain clarity and scalability as your application grows. This approach aligns the frontend structure with the backend modular architecture, making it intuitive for developers to locate and manage related frontend and backend code.

### Adjusted Application Structure with Module-Based Views

Here's how you can modify the `views` directory to reflect a modular structure:

```
``
/main-application |-- app.js // Main application entry point |-- package.json // Project dependencies and scripts |-- /config // Global configuration files | |-- index.js // Configuration loader |-- /core // Core functionality of the application | |-- /services // Shared services, e.g., database connections | |-- /middleware // Common middleware, e.g., authentication |-- /modules // Individual feature modules | |-- /email // Email module | | |-- emailController.js | | |-- emailRoutes.js | | |-- emailService.js | | |-- /views // Views specific to the Email module | | |-- emailDashboard.ejs | | |-- emailList.ejs | |-- /youtube // YouTube search module | | |-- youtubeController.js | | |-- youtubeRoutes.js | | |-- youtubeService.js | | |-- /views // Views specific to the YouTube module | | |-- searchResults.ejs | | |-- videoDetail.ejs |-- /public // Static assets | |-- /css | |-- /js | |-- /images |-- /views // Common or shared EJS templates | |-- /partials // Shared partials like headers, footers | |-- layout.ejs // General layout template |-- /test // Test scripts for the application |-- /scripts // Utility scripts, e.g., setup scripts ``
### Benefits of Modular Views
**1. Encapsulation:** Each module encapsulates its views along with its controllers and services, reinforcing the separation of concerns. This isolation ensures that changes in one module's views do not affect others, reducing the risk of cross-module dependencies and conflicts.
**2. Simplicity:** Developers can quickly understand the scope and functionality of a module by looking at its dedicated folder. This simplifies both development and debugging, especially in a team environment where multiple people might be working on different modules.
**3. Scalability:** As new modules are added, the existing structure doesn't need to be altered significantly. New modules can be developed and plugged in with their respective views and functionality, making the application scalable and extensible.
**4. Maintenance:** Updates, upgrades, or replacements of modules become straightforward. If a module needs to be updated or replaced, the impact on the overall system is minimized, as the changes are confined to the module folder.

### Additional Considerations



- Routing Consistency: Ensure that your Express routing logic is aligned with the views. Each module's router should correctly map to its own views to prevent any confusion or mismatches in UI rendering.
- Shared Components: Common components like headers, footers, and navigation bars should be placed in the `/views/partials` directory. This allows for easy inclusion across all module-specific views without redundancy.
- Security and Access Control: When modularizing views, consider implementing module-specific security policies or access controls to protect sensitive views, especially in modules that handle user or administrative functionalities. This modular approach to organizing views in a Node.js application using Express not only streamlines development workflows but also enhances the robustness and flexibility of the application architecture.

```

## THIS Is My Favourite F-String Trick In Python #code #programming #python

[https://youtube.com/shorts/jj\\_MzvOoHzg?si=uqzZkNQ2EBZv7Bhy](https://youtube.com/shorts/jj_MzvOoHzg?si=uqzZkNQ2EBZv7Bhy)

## YouTube API Blackbox

<https://www.blackbox.ai/share/1c922149-b49c-49a4-9861-bd5e2e7549c4>

## Python to write code summary in markdown

None

## Youtube API

None

## Sound API

Sound in issues?

## Cohesiveness

How to build a community

## Riley FOC Circle

<https://futureofcontent.circle.so/c/important-announcements/happy-may-1st-an-update> Hvordan svare han?

## Hardcoded Title

Hardcoded issue body

## Node express to pdf

<https://youtu.be/fKewAIUwRPk?si=wozY7-iA2emvkT6d>

## Issue Markup ,Bilde Maiken

- [ ] Husk ` Including > `indeterminate # Head 1 ##Head2 ###Head3 @torarnehave1  
![20240430\_202236.jpg](https://github.com/torarnehave1/SlowYouGPT/assets/159666877/ace166eb-338e-4034-9ce4-13a33e255ae9) **Bold** *Italic* - punkt  
liste - 1 - 2 ~~~~uujikkkkk~` ``sql Npm install express ````bash Sudo nginx -t ``

Kommentar



## Bootstrap

## Tilstedeværelse

Å mestre kunsten å være til stede i øyeblikket med bevissthet om kroppens posisjon, pust, stemme, og spenninger.

![image](https://github.com/torarnehave1/SlowYouGPT/assets/159666877/37f8612d-be63-4c29-ba0a-999d9f08ff66)

## Test

SlowYou følger en rekke grunnleggende prinsipper som er designet for å styrke individers selvforståelse og bevissthet om seg selv og deres omgivelser. Her er noen nøkkelpinsipper som veileder praksisen: 1. **Kroppens Visdom**: - **Tilstedeværelse**: Å mestre kunsten å være til stede i øyeblikket med bevissthet om kroppens posisjon, pust, stemme, og spenninger. - **Selvbevissthet**: Utvikling av en sterkere forbindelse med din egen kropp og lære å tolke dens unike språk. - **Grunning og Flyt**: Forflytte oppmerksomheten fra hodet til føttene og omfavne kraften av gravitasjon og naturlig bevegelse. 2. **Hjertets Sentralitet**: - **Kjærlighet og Medfølelse**: Dyrke en kjærlig og medfølelsende holdning til deg selv og andre. - **Hjertets Visdom**: Se på hjertet som en kilde til visdom og spirituell innsikt. - **Hjerterommet**: Skape et indre rom av fred og tilfredshet. 3. **Balanse og Harmoni**: - **Harmoni med Naturen**: Oppfordre til en livsstil som respekterer og etterligner naturens sykluser og prinsipper. - **Energi Flyt**: Lære å forstå og uttrykke "energi" som strømmer gjennom kroppen og hvordan påvirke denne flyten positivt. - **Wu Wei**: Oppnå mer med mindre anstrengelse gjennom en forståelse av kroppens naturlige prosesser. 4. **Universalitet og Inklusivitet**: - **Integrering av Praksiser**: Fremme en tilnærming som omfavner mangfoldet av menneskelige erfaringer åpen for alle bakgrunner. 5. **Helhetlig Tilnærming**: - **Helhetlig Syn**: Erkjennelsen av at kropp, sinn, og sjel er intrikat sammenvevd og samarbeider for å forme vår livserfaring. 6. **Naturlige Skjelvinger som Veien til Økt Vitalitet**: - **Bevissthet om Skjelvinger**: Utvikle en forståelse for kroppens naturlige skjelvinger som et verktøy for healing og fornyelse. Disse prinsippene legger grunnlaget for en reise av selvoppdagelse og transformasjon mot et liv med større bevissthet, kjærlighet, og harmoni.  [source](https://github.com/torarnehave1/SlowYouGPT/assets/159666877/4806473f-8a79-4dde-b282-3f0b50cfe34e)  [\[image\]\(https://github.com/torarnehave1/SlowYouGPT/assets/159666877/4806473f-8a79-4dde-b282-3f0b50cfe34e\)](https://github.com/torarnehave1/SlowYouGPT/assets/159666877/4806473f-8a79-4dde-b282-3f0b50cfe34e)

## sendgrid.com/guide/integrate Send Grid post@slowyou.net

<https://app.sendgrid.com/guide/integrate/langs/nodejs> Verify [ All languages](https://app.sendgrid.com/guide/integrate/langs/nodejs#guide/integrate/langs) How to send email using Node.js Make sure you have the prerequisites Our library requires Node.js version 0.10, 0.12, or 4. Create an API key This allows your application to authenticate to our API and send mail. You can enable or disable additional permissions on the [API keys page](https://app.sendgrid.com/guide/integrate/langs/nodejs#settings/api\_keys). My First API Key Name Create an environment variable Update your development environment with your SENDGRID\_API\_KEY. Run the following in your shell: Install the package The following recommended installation requires npm. If you are unfamiliar with [npm](https://npmjs.org/), see the npm docs. Npm comes installed with Node.js since node version 0.8.x, therefore you likely already have it: Send your first email The following is the minimum needed code to send an email: Implement and run the code above. If that runs without error, click "Next" and we'll check to see if your email came through SendGrid.

## Meet CopilotKit: An Open-Source Copilot Platform for Seamless AI Integration in Any Application - MarkTechPost

<https://www.marktechpost.com/2024/04/24/meet-copilotkit-an-open-source-copilot-platform-for-seamless-ai-integration-in-any-application/>

## Create a Data Table in Bootstrap 5

[https://youtube.com/watch?v=EKSPK\\_Zk9X8&si=Ayq90AISQkoELk0X](https://youtube.com/watch?v=EKSPK_Zk9X8&si=Ayq90AISQkoELk0X)

## GitHub Copilot is like a second brain

<https://youtube.com/shorts/o3qURBlpGM?si=MZkfyiCXj4ot9AVr>

## setHTMLUnsafe and parseHTMLUnsafe

<https://youtube.com/shorts/cjYwuPMYREo?si=9UyhTUQxE-CeP-g4>

## Tiny URL | System Design in 60 Seconds #systemdesigninterview #tinyurl #...

<https://youtube.com/shorts/q964nYfgTL0?si=XBrCbV60LY5fz-H8>

## Dropbox / Google Drive | System Design in 60 Seconds #systemdesigninterv...

[https://youtube.com/shorts/f-bKxXVm\\_Kc?si=th50m6GN3tFzuYG-](https://youtube.com/shorts/f-bKxXVm_Kc?si=th50m6GN3tFzuYG-)

## The best feature every framework needs

<https://youtube.com/shorts/R-7eSCDwHPg?si=wLKrQjNdP21IBugV>

## WhatsApp / WeChat | System Design in 60 Seconds #systemdesigninterview #...

[https://youtube.com/shorts/5dUCikhnNLE?si=-msg-kUtbN\\_vibYh](https://youtube.com/shorts/5dUCikhnNLE?si=-msg-kUtbN_vibYh)

## Rag Ilm

[https://get.vectara.com/genai-platform-for-builders-r?rdt\\_cid=4418447320446752788&utm\\_ad=vid\\_genai\\_1x1gears\\_11.09.23\\_pm\\_NewCopy&utm\\_ad\\_group=Communities](https://get.vectara.com/genai-platform-for-builders-r?rdt_cid=4418447320446752788&utm_ad=vid_genai_1x1gears_11.09.23_pm_NewCopy&utm_ad_group=Communities)

## How to Build a SaaS in 30 Days ■■■■■ #shorts #viral #coding #saas

<https://youtube.com/shorts/VWV4L7cU584?si=8DVwuEnDYOBjxsLV>

## 4 Must-Read Computer Science Books ■ #coding #programming

[https://youtube.com/shorts/\\_zwrG0k8Eq4?si=AxEy9NIEsdB4hPTj](https://youtube.com/shorts/_zwrG0k8Eq4?si=AxEy9NIEsdB4hPTj)

## Notification System | System Design in 60 Seconds #systemdesigninterview...

[https://youtube.com/shorts/Qpe\\_LUsi\\_jM?si=fPYyeX9ntYg7B2-7](https://youtube.com/shorts/Qpe_LUsi_jM?si=fPYyeX9ntYg7B2-7)

## A great way to showcase your skills to employers

[https://youtube.com/shorts/IJLB0uU4pew?si=\\_SfnJYnATZbRZywl](https://youtube.com/shorts/IJLB0uU4pew?si=_SfnJYnATZbRZywl)

## Node.js 22 lar deg droppe npm run – skal være 6 ganger raskere - Kode24

<https://www.kode24.no/artikkel/nodejs-22-lar-deg-droppe-npm-run-skal-vaere-6-ganger-raskere/81318643>

## This makes adding payments so easy!

[https://youtube.com/watch?v=dHo7lvU6gs&si=\\_zdHoB-jlFbAkjtE](https://youtube.com/watch?v=dHo7lvU6gs&si=_zdHoB-jlFbAkjtE)

## 4 UI Libraries I Use In EVERY App ■■■■■ #shorts #viral #coding #ai

<https://youtube.com/shorts/FkNJo9d9Lzo?si=Z2FQ-FWPV650MbCF>

## Whisper

<https://huggingface.co/NbAiLab/nb-whisper-large>

## This is Why CSS is so Frustrating! ■

<https://youtube.com/shorts/F9KUw1WSHds?si=5ynxGiblfHpVwnLo>

## Stativer woodworking

<https://youtu.be/5AkXJSxOTzl?si=R-sdF6JFAMr6Vd94>

## Stativer Feste

<https://youtu.be/ex8b2gl0hvk?si=8k-gMIFNuvwNyeB5>

## Narrative Therapy

None

## Add a Honeypot to Your Forms ■

<https://youtube.com/shorts/4aVgn9px0il?si=whk-WOZP-xELVFMw>

## What tech I used to code my new startup in 2024

<https://youtube.com/watch?v=DYav0hb9piY&si=vRgYiYXN7-2j8Cpd>

## Do this when writing database queries with Appwrite

<https://youtube.com/watch?v=1ip2aljprvg&si=k9zlaX2TF7e7BkXQ>

## Be Lazy #shorts

<https://youtube.com/shorts/m6ltzaOpPl8?si=NIK71THYZplvgTUo>

## 7 Best VSCode Extensions

[https://youtube.com/shorts/W4Uj1\\_9RYVw?si=ThKkpamWA6ydr64O](https://youtube.com/shorts/W4Uj1_9RYVw?si=ThKkpamWA6ydr64O)

## Step inside the mind of a whirling dervish

[https://www.youtube.com/watch?v=x8YET0Bg\\_Ok&ab\\_channel=TRTWorld](https://www.youtube.com/watch?v=x8YET0Bg_Ok&ab_channel=TRTWorld) December 17 is the anniversary of the death of Mawlana Jalaluddin Rumi. The day is known as Seb-i Arus, or his “wedding night,” because it marked his reunion with his Beloved. The whirling ceremony is often a highlight of the annual celebration.

## 4 ways GitHub engineers use GitHub Copilot - The GitHub Blog

<https://github.blog/2024-04-09-4-ways-github-engineers-use-github-copilot/>

## Event Driven Arkititektur

[https://youtu.be/gOuAqRaDdHA?si=ux0\\_Hj5Ai-l\\_ISha](https://youtu.be/gOuAqRaDdHA?si=ux0_Hj5Ai-l_ISha)

## Web Developer's Roadmap 2024 - Step by Step

<https://youtube.com/watch?v=uWoFF9MkRbg&si=oCliiEarFJyQfn94>

## Vite in 100 Seconds

<https://youtube.com/watch?v=KCrXgy8qtjM&si=r8rf9wMO7pvVdZ7f>

## Controlling Arduino with Javascript Using the Johnny Five Library

[https://youtube.com/watch?v=6CmlidOxc2g&si=VYRNrVlp6i53PsT\\_](https://youtube.com/watch?v=6CmlidOxc2g&si=VYRNrVlp6i53PsT_)

## How I Code Any Design With This CSS Trick Flexbox

<https://youtube.com/watch?v=qMuBO87Y-8w&si=insYpcOtn6l1-54w>

## This Simple Bug Is Costing You a Lot of Money! | Easy to Fix Memory Leaks

<https://youtube.com/watch?v=NWlJBHwfR4M&si=GZbqkh-RU93QqeYw>

## Nvidia CEO's crazy work schedule ■

<https://youtube.com/shorts/vuDGmcW2ptA?si=xGXye8V-cCTjng3V>

## How to understand important concepts?

<https://youtube.com/shorts/tlU30R9eCDM?si=c3VDzGrKrBg1dYYt>

It is also about having patience or on the best case give your self fully to the highest truth in exploring the subject. Be clear that all you study take time and effort

## How to understand important concepts?

<https://youtube.com/shorts/tlU30R9eCDM?si=Wc13-NCa3JiOX-7>

## EXPOSING HAMAS: Son of Terrorist Leader UNLEASHES Truth from the Inside ...

<https://youtube.com/watch?v=tBiBiET3Y84&si=nSgl9AHqDTo7yMVZ>

43 min about heros

## Streamlining Development: Utilizing Hostinger's MEAN Stack VPS Template for Efficient Web Application Setup

<https://support.hostinger.com/en/articles/8805712-how-to-use-the-mean-stack-vps-template> You can view the guide on how to use the MEAN Stack VPS template on Hostinger's website [here](<https://support.hostinger.com/en/articles/8805712-how-to-use-the-mean-stack-vps-template>). This tutorial provides step-by-step instructions for setting up a MEAN stack application, starting from creating a project directory to setting up backend and frontend components with Express.js, Node.js, and Angular, and finally integrating them together. The guide also includes details on running the servers and viewing your application.

## USEFULL Github repos: free resources for developers! #shorts #coding #de...

<https://youtube.com/shorts/L405dwxoSOQ?si=VU84SH83wxd67XOY>

## He's Making HOW MUCH a Month?

<https://youtube.com/shorts/nsyMx5C0Bxc?si=WF947CWbBEa-BXlv>

## How to Open GitHub Repository Without Cloning With VsCode! #shorts #codi...

<https://youtube.com/shorts/y7fwyTh0F7Y?si=ghTVA7YewbNTxNcZ>

## Modular Open-Sources Mojo: The Programming Language that Turns Python into a Beast - MarkTechPost

<https://www.marktechpost.com/2024/03/31/modular-open-sources-mojo-the-programming-language-that-turns-python-into-a-beast/>

## Building draggable elements with Javascript & Sticky Notes app

[https://youtube.com/live/\\_NFdUC2W0W4?t=935&si=XqyO5FhTrueoAjBI](https://youtube.com/live/_NFdUC2W0W4?t=935&si=XqyO5FhTrueoAjBI)

## This New Node Feature Is Super Convenient

[https://youtube.com/shorts/x\\_GTNRxLtS8?si=mA2g\\_fDrSnMauRM7](https://youtube.com/shorts/x_GTNRxLtS8?si=mA2g_fDrSnMauRM7)

## 3 Best GihHub Repos that every developer must know! #shorts #developer #...

[https://youtube.com/shorts/CKY-Is\\_AyR0?si=cl4dFh890TUdvkQ6](https://youtube.com/shorts/CKY-Is_AyR0?si=cl4dFh890TUdvkQ6)

## I Recreated 3 Popular Apps for FREE ■■■ #tech #coder #developer #mobileapp

<https://youtube.com/shorts/6RgQ9CCLjkE?si=T3-1sVqrAvzyrAr2>

## Machine Learning made simple

Gradient enables individuals and teams to quickly develop, track, and collaborate on Machine Learning models of any size and complexity

<https://www.paperspace.com/pricing>

## NbAiLab/nb-whisper-large · Hugging Face

<https://huggingface.co/NbAiLab/nb-whisper-large>

## NbAiLab/nb-whisper-large · Hugging Face

<https://huggingface.co/NbAiLab/nb-whisper-large>

## Meet Open Interpreter: An Open-Source Project that Lets GPT-4 Execute Python Code Locally

<https://www.marktechpost.com/2024/03/28/meet-open-interpreter-an-open-source-project-that-lets-gpt-4-execute-python-code-locally/>

### I et bur

Jeg sitter i et bur jeg har skapt selv, et bur som heter "jeg er ikke god nok", "Jeg oppnår ikke nok". Så viser det seg en dag at det buret ikke var ekte, men jeg savner buret allikevel. Det var så trygt å ikke vise meg frem. " - Tor Arne Håve inspirert av [Maria Kristoffersen]

![image](https://github.com/torarnehave1/SlowYouGPT/assets/159666877/b83060e8-62e4-47bf-8073-5c5a0bf7b37c)

### Set Up .env

To import and use environment variables from a `.env` file in your `main.js` or any other JavaScript file within a Node.js application, follow these steps: ### Step 1: Install the dotenv package First, you need to install the `dotenv` package, which is a zero-dependency module that loads environment variables from a `.env` file into `process.env`. This package makes it easy to manage your application's configuration based on the environment it's running in. You can install `dotenv` using npm or yarn: `sh npm install dotenv` or `sh yarn add dotenv` ### Step 2: Create a .env file In the root directory of your project, create a `.env` file if you haven't done so already. Inside this file, define your environment variables. For example: `API_KEY=your_api_key_here DB_HOST=localhost` ### Step 3: Load the .env file in your main.js At the very beginning of your `main.js` file (or as early as possible), require and configure `dotenv` to load the variables: `javascript require('dotenv').config();` This line reads your `.env` file, parses the contents, assigns it to `process.env`, and makes it accessible throughout your application. ### Step 4: Access environment variables After configuring `dotenv`, you can access the environment variables anywhere in your application using `process.env.VARIABLE_NAME`. Here's an example of using an environment variable in `main.js`: `javascript console.log('The API Key is:', process.env.API_KEY);` ### Important Notes: - **Security**: Never commit your `.env` file to version control. It should be added to your `.gitignore` file to prevent sensitive information from being exposed. - **Default Values**: For variables that you expect to be defined but might not be (especially in development), you can provide a default value using `||`. For example: `const apiKey = process.env.API_KEY || 'defaultApiKey';` - **Scope**: Variables defined in `.env` are available only at runtime and within the Node.js process. They won't be accessible in client-side JavaScript unless explicitly passed through by your server-side code. By following these steps, you can easily manage and use environment variables in your Node.js applications, enhancing both security and configuration flexibility.

### Do you make this mistake with border-radius? #css

<https://youtube.com/shorts/D0lIR1qVJOk?si=sehx2D6qG4fjyNwV>

### Use this instead of :focus in your css Hamburger Menu-bar

<https://youtube.com/shorts/FHgnhnUwwQM?si=1pVIU8wOfvh76k7t>

### How to make a table searchable and sortable using jQuery DataTables

<https://youtube.com/shorts/ETSWJxpQRNY?si=niNFFgi8wb63hLGt>

### What I Learned after 5000 Hours of Studying" by Dr. Justin Sung

The video "What I Learned after 5000 Hours of Studying" by Dr. Justin Sung discusses the speaker's experiences and insights gained from studying various subjects over a decade. The speaker emphasizes the importance of effective learning techniques and shares three mind-blowing realizations that significantly improved their learning ability[2][5]. These realizations are: 1. Overusing learning techniques can hinder progress and limit learning: The speaker notes that they initially spent twice the amount of time studying than necessary, which led to burnout and inefficiency. They suggest focusing on a few effective techniques and mastering them rather than trying to learn many methods[5]. 2. Interleaving improves memory and learning: The speaker discusses the importance of interleaving, which involves alternating between different topics or tasks, to improve memory, test performance, and the ability to learn new concepts[8]. 3. Learning from mistakes is crucial: The speaker acknowledges making mistakes during their learning journey but emphasizes the importance of learning from them. They suggest that mistakes are an inevitable part of the learning process and can lead to significant improvements if used as learning opportunities[2]. Overall, the video provides insights into effective learning strategies and the importance of focusing on a few key techniques, interleaving, and learning from mistakes. Citations: [1] <https://www.youtube.com/watch?v=XeKMjwiF3Pc> [2] <https://www.youtube.com/watch?v=u2Ogza-n3i0&t=0> [3] <https://www.youtube.com/watch?v=sPCiv1WN9k0> [4] [https://www.reddit.com/r/mikew\\_reddit\\_selfhelp/comments/1blvls9/what\\_i\\_learned\\_after\\_5000\\_hours\\_of\\_studying/](https://www.reddit.com/r/mikew_reddit_selfhelp/comments/1blvls9/what_i_learned_after_5000_hours_of_studying/) [5] <https://glasp.co/youtube/p/what-i-learned-after-5000-hours-of-studying> [6] <https://www.youtube.com/watch?v=NqxUExCZJ5Y> [7] <https://chattube.io/summary/education/u2Ogza-n3i0> [8] <https://glasp.co/youtube/p/what-i-learned-after-5000-hours-of-revision>

### Awsm - Ownership is Awsm, Awsm is Ownership



<https://www.awsm.com/>

## Markedsføring

<https://awesm.com/?fbclid=IwAR1eQbmKydeOrgs3Djqu7b2b-MeQer78T2HMINxxzruhQ7BFbT0VsrhV7w>

## 7 ways to center with CSS

[https://youtube.com/shorts/7l6pb2MHRQg?si=FNFNQ\\_Y4-2yYkuBO](https://youtube.com/shorts/7l6pb2MHRQg?si=FNFNQ_Y4-2yYkuBO)

## Select all siblings of an element using :has() CSS

[https://youtube.com/shorts/Ue3VUsZwAJ0?si=f\\_RBf2J23DQIm-33](https://youtube.com/shorts/Ue3VUsZwAJ0?si=f_RBf2J23DQIm-33)

## Claude AI

<https://x.com/sourabhrrathourr/status/1769483847039938580?s=20>

## Appwrite QuickStart project with vanilla JavaScript

<https://youtube.com/watch?v=E7xjuKJI7Ac&si=lxNewiEJrWb9CSuA>

## Maximizing Community Engagement with Kajabi's Custom Page Feature

Given the content and the focus of the video, here's a suggestion for a page title and introduction, a concise video summary, key points with timestamp links, and additional resources that could form the basis of a well-structured article. ### Page Title: "Maximizing Community Engagement with Kajabi's Custom Page Feature" ### Introduction: Welcome to a deep dive into the transformative potential of Kajabi's Community 2.0, particularly through the lens of its custom page functionality. This feature stands as a beacon of customization and user engagement within the digital community space. In this guide, we unravel the essentials of setting up and optimizing a custom page, illustrating its critical role in fostering a cohesive, engaging, and resource-rich environment for members. Navigate through the intricacies of enhancing member experience, ensuring your community not only grows but thrives. ### Video Summary: The video serves as an instructional guide on leveraging the custom page feature within Kajabi's Community 2.0. It emphasizes the significance of this feature in creating a more personalized and resourceful environment for community members. The presenter walks viewers through the process of integrating a custom landing page into their community, aiming to streamline the member journey and enrich the learning experience. By highlighting the importance of a well-structured welcome page, including instructional resources, FAQs, and access to additional courses or materials, the video underscores the custom page's role in achieving educational objectives and enhancing community value. ### Key Points with Timestamp Links: 1. \*\*Introduction to Kajabi Community 2.0's Custom Page Feature\*\*: - Overview and benefits of custom page integration. [Start of Video](https://www.youtube.com/watch?v=Ppj5VbZ2wMQ) 2. \*\*Creating a Welcoming Environment\*\*: - Importance of a "Start Here" page and what to include for new members. [Timestamp: 2:30](https://www.youtube.com/watch?v=Ppj5VbZ2wMQ&t=150s) 3. \*\*Content Ideas for the Custom Page\*\*: - Suggestions for valuable content to include, such as downloadable resources and access to bonus courses. [Timestamp: 4:15](https://www.youtube.com/watch?v=Ppj5VbZ2wMQ&t=255s) 4. \*\*Step-by-Step Setup Guide\*\*: - Detailed walkthrough on setting up and customizing the custom page feature. [Timestamp: 6:00](https://www.youtube.com/watch?v=Ppj5VbZ2wMQ&t=360s) 5. \*\*Future Updates and Functionalities\*\*: - Speculation on potential enhancements to the custom page feature. [Timestamp: 8:45](https://www.youtube.com/watch?v=Ppj5VbZ2wMQ&t=525s) 6. \*\*Closing Thoughts and Call to Action\*\*: - Encouragement to utilize Kajabi for online education businesses, including a special trial offer. [Timestamp: 10:30](https://www.youtube.com/watch?v=Ppj5VbZ2wMQ&t=630s) ### Additional Resources: - [Kajabi Help Center](https://help.kajabi.com/): Comprehensive guides and articles on utilizing Kajabi's features. - [Building Online Communities](https://www.feverbee.com/building-online-communities/): A book by Richard Millington, offering insights into creating vibrant digital communities. - [Kajabi Community Best Practices](https://blog.kajabi.com/): Tips and strategies for engaging and growing your online community. [Link to the video](https://www.youtube.com/watch?v=Ppj5VbZ2wMQ) This structured approach not only facilitates a deeper understanding of the custom page feature's utility but also empowers creators to cultivate an enriched community space, driving member satisfaction and success.

## How to Create a Mock-up in Figma

[https://youtu.be/oZAKb\_gs2Uo?si=-IBX82XTQFwrzCwi] (https://youtu.be/oZAKb\_gs2Uo?si=-IBX82XTQFwrzCwi) The video titled \*\*\*How to Create a Mock-up in Figma\*\*\* by Alienacai provides a comprehensive tutorial on creating high-fidelity design mock-ups using Figma. The content is aimed at those with some familiarity with Figma, suggesting viewers first watch a prior tutorial on creating wireframes. \*\*Key points from the video include:\*\* - \*\*Mock-up Creation:\*\* The process involves converting a wireframe into a detailed and accurate representation of a final product. Alienacai demonstrates this by recreating a screen from a favorite app, emphasizing the importance of matching the size, color, and position of every design element to the original. - \*\*Detailed Steps:\*\* The tutorial covers various steps in detail, including resizing and positioning elements, filling shapes with images, adjusting layering for visibility, and utilizing the color picker for exact color matching. Attention is given to both large elements (like images) and smaller components (such as icons and text), with tips on alignment and positioning. - \*\*Utilizing Resources:\*\* Alienacai highlights the utility of online resources for elements like icons, recommending the Figma community page for templates and specific design elements like the iOS kit for system-related icons. - \*\*Design Considerations:\*\* The video touches on broader design considerations, including the importance of a wireframe as a precursor to detailed mock-up work, and the need for designers to make informed decisions about spacing, colors, and fonts in their projects. For solo designers or those in startups, these decisions can be both challenging and rewarding. - \*\*Final Thoughts:\*\* Concluding the tutorial, Alienacai encourages viewers to experiment with creating interactive prototypes and to follow her for more design-related content. \*\*Conclusion:\*\* This tutorial serves as a valuable resource for anyone looking to deepen their understanding of Figma and the mock-up creation process. It offers practical guidance, from basic adjustments to more nuanced design decisions, suitable for designers at various levels of experience. Would you like to convert

this summary into a well-structured article in English?

## **Appwrite - Visual Studio Marketplace**

<https://marketplace.visualstudio.com/items?itemName=Streamlux.vscode-appwrite>

## **Show YouTube videos on your own website! | YouTube API Tutorial**

<https://youtube.com/watch?v=idYUy3hf3D0&si=DEES-ahZ8qbAI-6F>

## **Css Convention**

[https://www.linkedin.com/advice/0/what-best-practices-naming-css-classes-ids-front-end-ecv7e?utm\\_source=share&utm\\_medium=member\\_mobile\\_web&utm\\_campaign=](https://www.linkedin.com/advice/0/what-best-practices-naming-css-classes-ids-front-end-ecv7e?utm_source=share&utm_medium=member_mobile_web&utm_campaign=)

Another important practice for naming CSS classes and IDs is to follow a consistent naming convention throughout your project. A naming convention is a set of rules or guidelines that define how to format, structure, and separate your names. There are many different naming conventions that you can choose from, such as BEM, OOCSS, SMACSS, or SUITCSS, or you can create your own custom convention. The main benefits of following a naming convention are that it helps you organize your code, avoid conflicts and collisions, and apply styles more efficiently and modularly. For example, using a convention like BEM, which stands for Block, Element, Modifier, you could name your classes like `.blog-post__title`, `.blog-post__image`, or `.blog-post__image--large`.

## **Microsoft Introduces AutoDev: A Fully Automated Artificial Intelligence-Driven Software Development Framework - MarkTechPost**

<https://www.marktechpost.com/2024/03/19/microsoft-introduces-autodev-a-fully-automated-artificial-intelligence-driven-software-development-framework/>

## **Turn a Harry Potter Book into a Knowledge Graph - Graph Database & Analytics**

<https://neo4j.com/developer-blog/turn-a-harry-potter-book-into-a-knowledge-graph/>