

python 后端

服务器: 42.193.241.119, root 登录

ID/名称	状态	实例类型	实例配置
ins-rbe2oki2 未命名	运行中	标准型SA2	4核 8GB 3Mbps 系统盘: 通用型SSD云硬盘 网络: vpc-pygktq89

用户名

新密码 × 眼

注意: 您的密码已经符合设置密码规则, 但密码需要具备一定的强度, 建议您设置12位及以上 ([a-z],[A-Z],[0-9]和[(`~!@#\$%^&*-+=_|\{};`<,.?/] 的特殊符号]) , 每种字符大于等于2位且不请输入实例密码

确认密码 × 眼

请再次输入实例密码

路径: /www/wwwroot/XY/campus-info-platform-main

先上传 scripts 中的 python 代码到服务器。找到目录, 打开终端

项目路径选择上传的代码路径

启动命令输入 nohup python api_server.py

```
[root@VM-0-4-centos scripts]# nohup python api_server.py
nohup: ignoring input and appending output to 'nohup.out'
```

小程序

```
> .swc
> admin-console
< config
  TS index.ts
> docs
> scripts
< src
  > assets
  > components
  > contexts
  > custom-tab-bar
  > pages
  > services
  > styles
  > utils
  TS app.config.ts
    11      },
    12      375: 2,
    13      828: 1.81 / 2,
    14    },
    15    sourceRoot: 'src',
    16    outputRoot: 'dist',
    17    plugins: ['@tarojs/plugin-sass'],
    18    defineConstants: {
    19      // API 服务地址(开发环境使用局域网 IP)
    20      API_URL: JSON.stringify('http://192.168.1.12:5001')
    21    },
    22    copy: {
    23      patterns: [],
    24      options: {},
    25    },
    26    framework: 'react',
    27    compiler: 'webpack5',
    28    cache: {
    29      enable: false
    30    }
  
```

修改 config 里的 apiurl 为服务器的域名加端口, 然后在小程序开发软件里直接上传

管理后台：

```
> .next          8   },
> app           9   })
> components   10  }
> docs          11  const supabase = createServerClient(
> lib           12  "https://civlywqsdzrvsutlrx.supabase.co",
> node_modules 13  "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJzdXBhYmFzZSIsInJlZiI6ImNpdm
> {
> public         14  cookies: {
> scripts        15  getAll() {
> types          16  return request.cookies.getAll()
> .gitignore     17  },
> admin-console.rar 18  setAll(cookiesToSet) {
> eslint.config.mjs 19  cookiesToSet.forEach(({ name, value, options }) =>
> middleware.ts 20  request.cookies.set(name, value)
> next-env.d.ts 21  )
> next.config.ts 22  response = NextResponse.next({
>                   23  request,
>                   24  })
```

将所有的 env 部分替换成真实数据，然后 admin-console 文件夹里的代码单独上传一个 git 库

<https://vercel.com/>注册登录

打开 <https://vercel.com/new?teamSlug=mrjjs-projects>

The screenshot shows the Vercel dashboard. At the top, it says "Let's build something new". Below that is a search bar with the placeholder "Enter a Git repository URL to deploy...". To the right of the search bar is a "Clone Template" section with two options: "Next.js Boilerplate" and "Vite + React Start". The main area is titled "Import Git Repository" and contains a dropdown menu showing "Mr-J-J" and a search bar with "Search...". Below this is a list of repositories:

Repository	Last Updated	Action
ceshi	7m ago	Import
monopoly	12/19/24	Import
XGD_uniapp	6/11/24	Import

在这里搜索你的库，点击 import

Import Git Repository

The screenshot shows a user interface for deploying a Git repository. At the top left, there's a dropdown menu showing 'Mr-J-J' and a search bar with placeholder text 'Search...'. Below this, a list of repositories is displayed:

- ceshi · Just now (with an 'Import' button)
- monopoly · 12/19/24 (with an 'Import' button)
- XGD_uniapp 🛡 · 6/11/24 (with an 'Import' button)

Below the repository list is a section titled 'Build and Output Settings' with the following options:

- Build Command: 'npm run build' or 'next build'
- Output Directory: Next.js default
- Install Command: 'yarn install', 'pnpm install', 'npm install', or 'bun install'

At the bottom of this section is a link labeled 'Environment Variables'.

A large red arrow points from the 'Import' button of the 'ceshi' repository to the 'Deploy' button at the bottom of the page.

On the right side of the interface, there are sections for 'Clone' and 'Next' (with 'Get s' and 'React' links), and a 'Vite' section (with 'Vite / I' link).

点击 deploy

部署成功后

点

击

The screenshot shows the Vercel dashboard. At the top, there's a navigation bar with links like Overview, Integrations, Deployments, Activity, Domains, Usage, Observability, Storage, Flags, AI Gateway, Agent, and Support. Below the navigation is a search bar labeled "Search Projects...". The main area is divided into two sections: "Usage" and "Projects". The "Usage" section displays metrics for the last 30 days, including Fast Data Transfer (0 / 100 GB), Fast Origin Transfer (0 / 10 GB), Edge Requests (0 / 1M), and Edge Request CPU Duration (0 / 1h). An "Upgrade" button is located in the top right of this section. The "Projects" section lists two projects: "ceshi" (ceshi-blue.vercel.app) and "chat-gpt-next-web1-ljek" (chat-gpt-next-web1-swart-seven-67.vercel.app). Each project card includes the repository name, URL, a small profile picture, and a timestamp for the first commit.

就会自动跳转测试域名

The screenshot shows a browser window with the URL https://ceshi-92wo832z-mrjjs-projects.vercel.app/login. The page features a central login form with a header that reads "UniFlow 智汇流" and "管理后台登录". The form contains fields for "邮箱地址" (Email Address) and "密码" (Password), and a purple "登录" (Login) button.

如果需要更改域名则点击标题然后回跳转

The screenshot shows the Vercel dashboard again, but this time the "Domains" tab is active, indicated by a dark background. The title "ceshi" is visible at the top left. Below it, under "Production Deployment", there are three tabs: "Repository", "Usage", and "Domains", with "Domains" being the active one. At the bottom of the screen, there are three buttons: "Build Logs", "Runtime Logs", and "Instant F".

点击 Domains 则自动跳转配置域名