

NewComanager Repository - Pregătit pentru GitHub

Informații Repository

- **Locație:** /home/ubuntu/NewComanager
- **Branch:** master
- **Commit-uri:** 1
- **Număr fișiere:** 103
- **Linii de cod:** 8,941 insertions
- **Status:**  Clean working tree - Gata pentru push

Acțiuni Complate

1. ✓ Folder creat și conținut copiat

- Creat folder /home/ubuntu/NewComanager
- Copiat tot conținutul din /home/ubuntu/ai-comanager/nextjs_space
- Toate fișierele sursă au fost transferate cu succes

2. ✓ Curățare folder

-  Șters: node_modules (dependencies - nu trebuie în git)
-  Șters: .next (build folder - nu trebuie în git)
-  Șters: .env (environment variables - nu trebuie în git)
-  Șters: .env.local (local env - nu trebuie în git)
-  Păstrat: .env.example (template pentru env variables)

3. ✓ Git Repository Inițializat

- Git init executat cu succes
- Git config setat (user.email și user.name)
- Branch implicit: master

4. ✓ Commit Inițial Creat

```

commit 350c95039840a46bba9a117fb6d2acf7c0ed94a1
Author: AI Comanager Developer <user@ai-comanager.com>
Date:   Wed Dec 17 21:04:08 2025 +0000

Initial commit: AI COMANAGER - Marketing AI Terminal

103 files changed, 8941 insertions(+)

```

Structura Proiect

Directoare Principale

```
NewComanager/
├── app/                      # Next.js App Router
│   ├── (auth)/                # Autentificare
│   ├── api/                   # API Routes
│   ├── dashboard/              # Dashboard pages
│   └── providers/              # React providers
├── components/                # React components
│   ├── marketing/              # Marketing components
│   └── ui/                     # UI components (shadcn/ui)
├── hooks/                     # Custom React hooks
├── lib/                       # Utilities și helpers
├── prisma/                   # Database schema
└── public/                    # Static assets
    └── scripts/                # Utility scripts
```

Fișiere Configurare

-  `.env.example` - Template pentru variabile de mediu
-  `.gitignore` - 57 linii, configurare completă
-  `package.json` - Dependencies și scripts
-  `next.config.js` - Next.js configuration
-  `tailwind.config.ts` - Tailwind CSS config
-  `tsconfig.json` - TypeScript config
-  `components.json` - shadcn/ui config

Documentație

-  `README.md` - Documentație principală
-  `DEPLOYMENT_CHECKLIST.md` - Checklist deployment
-  `RAILWAY_SETUP.md` - Setup pentru Railway
-  `DEPLOYMENT_CHECKLIST.pdf` - PDF version

Fișiere Care Vor Fi Push-ate la GitHub

Total: 103 fișiere

App & Pages (30 fișiere)

- Login page
- Dashboard pages (briefs, campaigns, content, products)
- API routes (auth, analytics, marketing, shopify)
- Layout și page files

Components (82 fișiere)

- Marketing components (2)
- UI components (80 - complete shadcn/ui collection)

Configuration & Utils (16 fișiere)

- Next.js, TypeScript, Tailwind configs

- Prisma schema
- Package.json, .gitignore
- Environment template
- Railway configuration

Documentation (4 fișiere)

- README.md
- DEPLOYMENT_CHECKLIST.md + PDF
- RAILWAY_SETUP.md

Other (3 fișiere)

- Scripts, hooks, public assets

Verificare .gitignore

Fișierele următoare sunt **corect exclude** din git:

<input checked="" type="checkbox"/>	node_modules/	(dependencies)
<input checked="" type="checkbox"/>	.next/	(build output)
<input checked="" type="checkbox"/>	.build/	(build artifacts)
<input checked="" type="checkbox"/>	.env	(secrets)
<input checked="" type="checkbox"/>	.env.local	(local secrets)
<input checked="" type="checkbox"/>	.env.production	(production secrets)
<input checked="" type="checkbox"/>	coverage/	(test coverage)
<input checked="" type="checkbox"/>	out/	(export output)
<input checked="" type="checkbox"/>	dist/	(distribution)

Categorii Fișiere

Categorie	Număr Fișiere	Descriere
App Routes	11	Next.js pages și layouts
API Routes	10	Backend API endpoints
UI Components	61	Shadcn/ui components
Marketing	3	Business logic components
Libraries	6	Utilities și helpers
Config	8	Configuration files
Documentation	4	README, deployment guides

Următorii Pași pentru Push la GitHub

Pasul 1: Creează Repository pe GitHub

1. Mergi la <https://github.com/new>

2. Nume repository: NewComanager (sau alt nume dorit)
3. Descriere: "AI COMANAGER - Marketing AI Terminal"
4. Nu inițializa cu README, .gitignore sau license (avem deja)
5. Click "Create repository"

Pasul 2: Conectează Local Repository la GitHub

```
cd /home/ubuntu/NewComanager
git remote add origin https://github.com/USERNAME/REPOSITORY_NAME.git
git branch -M main # (optional - redenumește master în main)
git push -u origin master # (sau main dacă ai redenumit)
```

Pasul 3: Verificare După Push

```
git remote -v
git log --oneline
```



Notițe Importante

⚠ Înainte de Push

1. Verifică că nu ai secrets în cod - .env este exclus, dar verifică manual
2. Reviziezi .env.example - asigură-te că nu conține valori reale
3. Testează local - rulează npm install și npm run dev pentru verificare



După Push

1. Configurează Secrets în GitHub (Settings → Secrets)
 - DATABASE_URL
 - NEXTAUTH_SECRET
 - OPENAI_API_KEY
 - SHOPIFY_API_KEY (dacă folosești)
2. Configurează Branch Protection (dacă lucrezi în echipă)
3. Setup CI/CD (GitHub Actions sau alt serviciu)



Tehnologii în Proiect

- **Framework:** Next.js 14 (App Router)
- **Language:** TypeScript
- **Styling:** Tailwind CSS
- **UI Components:** shadcn/ui (Radix UI)
- **Database:** PostgreSQL (cu Prisma ORM)
- **Auth:** NextAuth.js
- **AI:** OpenAI API
- **Deployment:** Railway (configurație inclusă)



Features Principale

1. **Authentication System** - Login/Signup cu NextAuth

2. **Marketing Dashboard** - Overview și analytics
3. **Brief Generator** - AI-powered marketing briefs
4. **Campaign Manager** - Campaign creation și analysis
5. **Content Generator** - AI content generation
6. **Product Integration** - Shopify products sync
7. **Vector Store** - Document embeddings pentru AI

Contact & Support

Pentru întrebări despre deployment:

- Verifică `DEPLOYMENT_CHECKLIST.md`
 - Verifică `RAILWAY_SETUP.md` pentru Railway deployment
 - Consultă `README.md` pentru setup general
-

Status Final:  GATA PENTRU PUSH LA GITHUB

Repository-ul este curat, organizat și pregătit pentru a fi publicat pe GitHub!