



# Guia Passo a Passo - Setup Completo do Sistema de Gestão de Metas

---

Este guia foi criado para quem nunca fez isso antes. Cada comando é copy-paste pronto para usar.

---



## Índice

---

1. [Pré-requisitos](#)
  2. [Passo 1: Criar Conta GitHub](#)
  3. [Passo 2: Criar Repositório GitHub](#)
  4. [Passo 3: Configurar Git Localmente](#)
  5. [Passo 4: Criar Conta Firebase](#)
  6. [Passo 5: Configurar Firebase](#)
  7. [Passo 6: Criar Conta Vercel](#)
  8. [Passo 7: Conectar GitHub com Vercel](#)
  9. [Passo 8: Configurar Banco de Dados](#)
  10. [Passo 9: Configurar Secrets no GitHub](#)
  11. [Passo 10: Fazer Primeiro Deploy](#)
- 



## Pré-requisitos

---

Antes de começar, você precisa ter instalado:

### Windows

1. Node.js: [Baixar aqui](#) (escolha LTS - versão 20.x)

- Durante a instalação, marque “Add to PATH”
- Reinicie o computador após instalar

## 2. Git: [Baixar aqui](#)

- Escolha as opções padrão durante a instalação

## 3. Visual Studio Code (opcional, mas recomendado): [Baixar aqui](#)

## macOS

```
# Instalar Homebrew (se não tiver)
/bin/bash -c "$(curl -fsSL
https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"

# Instalar Node.js
brew install node

# Instalar Git
brew install git
```

## Linux (Ubuntu/Debian)

```
sudo apt update
sudo apt install -y nodejs npm git
```

## Verificar Instalação

Abra o terminal/PowerShell e execute:

```
node --version
npm --version
git --version
```

Você deve ver versões como: v20.x.x , 10.x.x , git version 2.x.x

# PASSO 1: Criar Conta GitHub

---

## 1.1 Acessar GitHub

- Abra seu navegador e vá para <https://github.com>

## 1.2 Clicar em “Sign up”

- Clique no botão “Sign up” no canto superior direito

## 1.3 Preencher Formulário

- **Email:** Use seu email pessoal
- **Senha:** Crie uma senha forte
- **Username:** Escolha um nome de usuário (ex: seu-nome-github )
- Clique em “Create account”

## 1.4 Verificar Email

- GitHub enviará um email de confirmação
- Abra o email e clique no link de confirmação

## 1.5 Completar Perfil (Opcional)

- Você pode preencher seu perfil, mas não é obrigatório agora

 **Você agora tem uma conta GitHub!**

---

# PASSO 2: Criar Repositório GitHub

---

## 2.1 Acessar Dashboard

- Faça login no [GitHub](#)
- Clique no ícone de perfil no canto superior direito

- Clique em “Your repositories”

## 2.2 Criar Novo Repositório

- Clique no botão verde “New”

## 2.3 Preencher Informações

Preencha os campos assim:

Campo	Valor
Repository name	metas-management-system
Description	Sistema online para gestão de metas organizacionais
Visibility	Selecione “Private” (privado)
Initialize this repository with	Deixe desmarcado

## 2.4 Criar Repositório

- Clique no botão “Create repository”

## 2.5 Copiar URL

- Você verá uma página com a URL do repositório
- Copie a URL (deve ser algo como `https://github.com/seu-usuario/metas-management-system.git`)
- Guarde essa URL, você vai precisar dela

 **Repositório criado no GitHub!**

---

# PASSO 3: Configurar Git Localmente

---

## 3.1 Abrir Terminal/PowerShell

Windows:

- Pressione Win + R
- Digite powershell
- Pressione Enter

macOS/Linux:

- Abra o Terminal

## 3.2 Configurar Git (Primeira Vez)

Execute estes comandos (substitua pelos seus dados):

```
git config --global user.name "Seu Nome Completo"  
git config --global user.email "seu-email@example.com"
```

Exemplo:

```
git config --global user.name "João Silva"  
git config --global user.email "joao@example.com"
```

## 3.3 Gerar Chave SSH (Recomendado)

Execute:

```
ssh-keygen -t ed25519 -C "seu-email@example.com"
```

Pressione Enter 3 vezes (para usar valores padrão)

## 3.4 Adicionar Chave SSH no GitHub

Execute:

```
cat ~/.ssh/id_ed25519.pub
```

### Windows PowerShell:

```
Get-Content $env:USERPROFILE\.ssh\id_ed25519.pub
```

Isso vai mostrar uma chave longa. Copie tudo.

## 3.5 Adicionar no GitHub

1. Vá para <https://github.com/settings/keys>
2. Clique em “New SSH key”
3. Cole a chave que você copiou
4. Clique em “Add SSH key”

 Git configurado!

---

## PASSO 4: Criar Conta Firebase

---

### 4.1 Acessar Firebase

- Abra <https://firebase.google.com>
- Clique em “Get started”

### 4.2 Fazer Login com Google

- Clique em “Sign in with Google”
- Use sua conta Google (ou crie uma)

## 4.3 Aceitar Termos

- Leia e aceite os termos de serviço
- Clique em “I agree”

## 4.4 Criar Projeto

- Clique em “Create project”
- Nome do projeto: `metas-management-system`
- Clique em “Continue”

## 4.5 Configurações do Projeto

- Desabilite Google Analytics (você pode habilitar depois)
- Clique em “Create project”
- Aguarde alguns segundos

 **Projeto Firebase criado!**

---

# PASSO 5: Configurar Firebase

---

## 5.1 Habilitar Autenticação

1. No Firebase Console, clique em **Authentication** (no menu esquerdo)
2. Clique em **Get started**
3. Clique em **Email/Password**
4. Ative a opção “Email/Password”
5. Clique em **Save**

## 5.2 Habilitar Firestore

1. Clique em **Firestore Database** (no menu esquerdo)
2. Clique em **Create database**

3. Selecione **Production mode**

4. Escolha a região: us-central1

5. Clique em **Create**

### 5.3 Habilitar Storage

1. Clique em **Storage** (no menu esquerdo)

2. Clique em **Get started**

3. Selecione **Production mode**

4. Escolha a região: us-central1

5. Clique em **Create**

### 5.4 Obter Credenciais

1. Clique no ícone de engrenagem ( ) no canto superior esquerdo

2. Clique em **Project settings**

3. Vá para a aba **General**

4. Procure por **Your apps** e clique em **Web** (ícone de )

5. Se não houver nenhum app, clique em **Add app**

6. Nome: metas-management-system

7. Clique em **Register app**

### 5.5 Copiar Configuração

Você verá um bloco de código assim:

```
const firebaseConfig = {
  apiKey: "AIzaSyD...",
  authDomain: "seu-projeto.firebaseio.com",
  projectId: "seu-projeto-id",
  storageBucket: "seu-projeto.appspot.com",
  messagingSenderId: "123456789",
  appId: "1:123456789:web:abc123def456"
};
```

**Copie e guarde essas informações em um arquivo de texto!**

 **Firebase configurado!**

---

## PASSO 6: Criar Conta Vercel

---

### 6.1 Acessar Vercel

- Abra <https://vercel.com>
- Clique em “Sign Up”

### 6.2 Fazer Login com GitHub

- Clique em “Continue with GitHub”
- Autorize o Vercel a acessar sua conta GitHub
- Clique em “Authorize Vercel”

### 6.3 Preencher Informações

- Nome: Seu nome completo
- Email: Seu email
- Clique em “Create Team”

 **Conta Vercel criada!**

---

## PASSO 7: Conectar GitHub com Vercel

---

### 7.1 Acessar Dashboard Vercel

- Você deve estar no dashboard da Vercel
- Se não, vá para <https://vercel.com/dashboard>

## 7.2 Importar Projeto

- Clique em “Add New”
- Clique em “Project”

## 7.3 Conectar Repositório GitHub

- Clique em “Continue with GitHub”
- Autorize se solicitado
- Procure por `metas-management-system`
- Clique em “Import”

## 7.4 Configurar Projeto

- **Project Name:** `metas-management-system`
- **Framework Preset:** `Other`
- **Build Command:** `pnpm build`
- **Output Directory:** `dist`
- **Install Command:** `pnpm install`
- Clique em “Deploy”

Aguarde o deploy terminar (pode levar alguns minutos).

 GitHub conectado com Vercel!

---

## PASSO 8: Configurar Banco de Dados

---

Você tem 2 opções:

### Opção A: PlanetScale (Recomendado - Gratuito)

#### A.1 Criar Conta PlanetScale

1. Vá para <https://planetscale.com>

2. Clique em “Sign up”
3. Clique em “Continue with GitHub”
4. Autorize o PlanetScale

## A.2 Criar Banco de Dados

1. Clique em “Create a database”
2. Nome: metas-management-system
3. Região: us-east (ou a mais próxima de você)
4. Clique em “Create database”

## A.3 Obter String de Conexão

1. Clique no banco que você criou
2. Clique em “Connect”
3. Selecione “Node.js”
4. Copie a string de conexão (começa com mysql:// )
5. **Guarde essa string!**

## Opção B: MySQL Local (Se tiver MySQL instalado)

Se você já tem MySQL instalado:

```
mysql -u root -p
```

Digite sua senha do MySQL, depois execute:

```
CREATE DATABASE metas_management;
EXIT;
```

A string de conexão será:

```
mysql://root:sua-senha@localhost:3306/metas_management
```

 **Banco de dados configurado!**

---

## PASSO 9: Configurar Secrets no GitHub

---

### 9.1 Acessar Configurações do Repositório

1. Vá para seu repositório no GitHub: <https://github.com/seu-usuario/metas-management-system>
2. Clique em **Settings** (no menu superior)
3. Clique em **Secrets and variables** (no menu esquerdo)
4. Clique em **Actions**

### 9.2 Adicionar Secrets

Para cada secret abaixo, clique em **New repository secret** e adicione:

#### Secret 1: DATABASE\_URL

- **Name:** DATABASE\_URL
- **Value:** Cole a string de conexão do PlanetScale ou MySQL
  - Exemplo: mysql://user:password@host:3306/database
- Clique em **Add secret**

#### Secret 2: JWT\_SECRET

- **Name:** JWT\_SECRET
- **Value:** Cole este comando em um terminal para gerar uma chave segura:

```
node -e "console.log(require('crypto').randomBytes(32).toString('hex'))"
```

Copie o resultado (será uma sequência longa de números e letras).

- Clique em **Add secret**

### **Secret 3: VERCEL\_TOKEN**

1. Vá para <https://vercel.com/account/tokens>
2. Clique em **Create Token**
3. Nome: github-actions
4. Clique em **Create**
5. Copie o token
6. Volte ao GitHub
7. **Name:** VERCEL\_TOKEN
8. **Value:** Cole o token
9. Clique em **Add secret**

### **Secret 4: VERCEL\_ORG\_ID**

1. Vá para seu projeto na Vercel: <https://vercel.com/dashboard>
2. Clique no seu projeto
3. Clique em **Settings**
4. Procure por **Project ID** (não confunda com Team ID)
5. Copie o ID
6. Volte ao GitHub
7. **Name:** VERCEL\_ORG\_ID
8. **Value:** Cole o ID
9. Clique em **Add secret**

### **Secret 5: VERCEL\_PROJECT\_ID**

1. No mesmo lugar do passo anterior, procure por **Project ID**
2. Copie o ID
3. Volte ao GitHub

4. **Name:** VERCEL\_PROJECT\_ID

5. **Value:** Cole o ID

6. Clique em **Add secret**

 Secrets configurados no GitHub!

---

## PASSO 10: Fazer Primeiro Deploy

### 10.1 Clonar o Repositório Localmente

Abra o terminal e execute:

```
git clone https://github.com/seu-usuario/metas-management-system.git  
cd metas-management-system
```

### 10.2 Instalar Dependências

```
npm install -g pnpm  
pnpm install
```

### 10.3 Executar Migrações do Banco

```
pnpm db:push
```

### 10.4 Iniciar Servidor Local

```
pnpm dev
```

Abra seu navegador em <http://localhost:3000>

Você deve ver a página inicial do projeto!

## 10.5 Fazer Commit e Push

```
git add .
git commit -m "Initial setup"
git push origin main
```

## 10.6 Verificar Deploy na Vercel

1. Vá para <https://vercel.com/dashboard>
2. Clique no seu projeto
3. Você deve ver um novo deployment em progresso
4. Aguarde até ficar com status “Ready”
5. Clique na URL do projeto para acessar

 Primeiro deploy realizado!



Você configurou com sucesso:

-  Repositório GitHub
-  Firebase
-  Vercel
-  Banco de Dados
-  CI/CD automático
-  Deploy contínuo



Agora você pode:

1. **Desenvolver localmente:** Use `pnpm dev` para trabalhar no projeto
2. **Fazer commits:** Cada push para `main` faz deploy automático
3. **Ler documentação:** Consulte os arquivos:
  - `DEVELOPMENT.md` - Para desenvolvimento local
  - `CONTRIBUTING.md` - Para contribuir
  - `README_SETUP.md` - Para visão geral

## Troubleshooting

---

### **Erro: “Permission denied (publickey)”**

Você não configurou SSH corretamente. Refaça o [Passo 3.3](#)

### **Erro: “Database connection refused”**

Verifique se a string de conexão está correta no GitHub Secrets.

### **Erro: “Build failed”**

Verifique os logs no Vercel Dashboard.

### **Erro: “npm: command not found”**

Node.js não foi instalado corretamente. Reinstale do [nodejs.org](https://nodejs.org)

---

**Qualquer dúvida, consulte a documentação ou abra uma issue no GitHub!**