

## # GitHub Push Instructions with Personal Access Token

### ## Overview

This guide explains how to push code to GitHub using a Personal Access Token for authentication.

### ## Prerequisites

- GitHub account
- Personal Access Token (PAT) with appropriate permissions
- Git installed on your system

### ## Step-by-Step Instructions

#### ### Step 1: Ensure Clean Remote URL

Make sure your remote URL doesn't contain any embedded tokens:

```
```bash
git remote set-url origin
https://github.com/evgenii-codesmith/yotcrew.app.git
```
```

#### ### Step 2: Configure Credential Storage (One-time setup)

Set up git to remember your credentials:

```
```bash
git config --global credential.helper store
```
```

#### ### Step 3: Standard Workflow

Follow these steps for each push:

1. **\*\*Make your changes\*\*** to the code
2. **\*\*Add files to staging\*\***:

```
```bash
git add .
```
```

3. **\*\*Commit your changes\*\***:

```
```bash
git commit -m "Your descriptive commit message"
```
```

4. **\*\*Push to GitHub\*\***:

```
```bash
git push origin main
```
```

#### ### Step 4: Authentication Prompt

When you push for the first time, git will prompt you:

- **\*\*Username\*\***: Enter your GitHub username (`evgenii-codesmith`)
- **\*\*Password\*\***: Enter your **\*\*Personal Access Token\*\*** (NOT your GitHub password)

### ### Step 5: Subsequent Pushes

After the first authentication, git will remember your credentials and won't ask again.

## ## Complete Example Workflow

```
```bash
```

```
# 1. Make changes to your code
```

```
# 2. Stage changes
```

```
git add .
```

```
# 3. Commit with descriptive message
```

```
git commit -m "Add new feature: user authentication"
```

```
# 4. Push to GitHub
```

```
git push origin main
```

```
# Username: evgenii-codesmith
```

```
# Password: YOUR_PERSONAL_ACCESS_TOKEN
```

```
```
```

## ## Security Best Practices

### ### Don'ts:

- **Never commit tokens to git**
- **Never share your token**
- **Don't use your GitHub account password**
- **Don't store tokens in plain text files**

## ## Troubleshooting

### ### If you get authentication errors:

1. **Verify your token is valid and not expired**
2. **Check that your token has the correct permissions**
3. **Ensure you're using the token as password, not your GitHub password**

### ### If you need to update your token:

1. **Generate a new token on GitHub**
2. **Update your stored credentials:**

```
```bash
```

```
git config --global --unset credential.helper
```

```
git config --global credential.helper store
```
```

### 3. Push again and enter new credentials

### If you need to change remote URL:

```
```bash
git remote set-url origin
https://github.com/evgenii-codesmith/yotcrew.app.git
```
```

## ## Token Management

### Creating a Personal Access Token:

1. Go to GitHub.com Developer settings Settings Personal access tokens
2. Find the token and click "Delete"
3. Generate a new one if needed

## ## Branch Management

### Working with branches:

```
```bash
# Create and switch to new branch
git checkout -b feature-branch
```

```
# Switch between branches
git checkout main
git checkout feature-branch
```

```
# Merge branch into main
git checkout main
git merge feature-branch
```

```
# Delete local branch
git branch -d feature-branch
```
```

## ## Common Commands Reference

```
```bash
# Check status
git status
```

```
# Check remote configuration
git remote -v
```

```
# View commit history  
git log --oneline
```

```
# Pull latest changes  
git pull origin main
```

```
# View branches  
git branch -a  
\\
```

~~---~~

~~**\*\*Remember\*\*:** Always use your Personal Access Token as the password, never your GitHub account password!~~