{

  "name": "bitbucket-pr-copilot",

  "displayName": "Bitbucket PR Copilot PoC",

  "description": "Post Copilot/chat review suggestions to Bitbucket PRs (quick or batch)",

  "version": "0.4.0",

  "publisher": "kk",

  "engines": { "vscode": "^1.80.0" },

  "main": "./extension.js",

  "activationEvents": [

    "onStartupFinished",

    "onCommand:bitbucketPRCopilot.quickPost",

    "onCommand:bitbucketPRCopilot.batchPost",

    "onCommand:bitbucketPRCopilot.testGit",

    "onCommand:bitbucketPRCopilot.clearApiToken",

    "onCommand:bitbucketPRCopilot.showLog"

  ],

  "contributes": {

    "commands": [

      { "command": "bitbucketPRCopilot.quickPost", "title": "Bitbucket PR Copilot: Quick Post (Active File)" },

      { "command": "bitbucketPRCopilot.batchPost", "title": "Bitbucket PR Copilot: Batch Post (All Open Files)" },

      { "command": "bitbucketPRCopilot.testGit", "title": "Bitbucket PR Copilot: Test Git" },

      { "command": "bitbucketPRCopilot.clearApiToken", "title": "Bitbucket PR Copilot: Clear Bitbucket API Token" },

      { "command": "bitbucketPRCopilot.showLog", "title": "Bitbucket PR Copilot: Show Log" }

    ],

    "configuration": {

      "type": "object",

      "title": "Bitbucket PR Copilot Settings",

      "properties": {

        "bitbucketPRCopilot.workspace": {

          "type": "string",

          "default": "myworkspace\_poc",

          "description": "Bitbucket workspace ID/slug"

        },

        "bitbucketPRCopilot.repo": {

          "type": "string",

          "default": "myrepo\_poc",

          "description": "Bitbucket repo slug"

        },

        "bitbucketPRCopilot.baseBranch": {

          "type": "string",

          "default": "main",

          "description": "Destination branch for PRs (display only in this PoC)"

        }

      }

    },

    "menus": {

      "editor/context": [

        {

          "command": "bitbucketPRCopilot.quickPost",

          "group": "navigation@9",

          "when": "resourceExtname =~ /\\.(js|jsx|ts|tsx|py|java|kt|go|rb|php|cs|cpp|c|h|hpp|json|yaml|yml)$/"

        },

        {

          "command": "bitbucketPRCopilot.batchPost",

          "group": "navigation@9"

        }

      ],

      "editor/title": [

        {

          "command": "bitbucketPRCopilot.quickPost",

          "group": "navigation@9",

          "when": "resourceExtname =~ /\\.(js|jsx|ts|tsx|py|java|kt|go|rb|php|cs|cpp|c|h|hpp|json|yaml|yml)$/"

        }

      ]

    },

    "keybindings": [

      {

        "command": "bitbucketPRCopilot.quickPost",

        "key": "ctrl+alt+p",

        "mac": "cmd+alt+p",

        "when": "editorTextFocus"

      },

      {

        "command": "bitbucketPRCopilot.batchPost",

        "key": "ctrl+alt+b",

        "mac": "cmd+alt+b",

        "when": "editorTextFocus"

      }

    ]

  },

  "categories": ["Other"],

  "keywords": ["bitbucket", "pull-requests", "copilot", "code-review"],

  "scripts": {

    "package": "vsce package"

  },

  "dependencies": {

    "simple-git": "^3.19.1"

  },

  "devDependencies": {

    "@vscode/vsce": "^3.0.0"

  },

  "repository": {

    "type": "git",

    "url": "https://bitbucket.org/myworkspace\_poc/myrepo\_poc.git"

  }

}

Looking at this extension.js file, I can see that it's **NOT using**.env**files** for tokens and credentials. Here's how authentication is handled:

**Current Authentication Method:**

**1. VS Code Secret Storage**

javascript

const SECRET\_KEY = 'bitbucket-basic-auth'; *// stores base64(email:token)*

async function getAuthHeader(context) {

const sec = context.secrets;

let basic = await sec.get(SECRET\_KEY);

if (!basic) {

const email = await vscode.window.showInputBox({ prompt: 'Bitbucket email', ignoreFocusOut: true });

const token = await vscode.window.showInputBox({ prompt: 'Bitbucket App Password / API Token', password: true, ignoreFocusOut: true });

if (!email || !token) throw new Error('Bitbucket credentials are required.');

basic = Buffer.from(`${email}:${token}`).toString('base64');

await sec.store(SECRET\_KEY, basic);

log('Stored Bitbucket credentials in SecretStorage.');

}

return `Basic ${basic}`;

}

**2. Configuration via VS Code Settings**

javascript

function getCfg() {

const cfg = vscode.workspace.getConfiguration('bitbucketPRCopilot');

return {

workspace: cfg.get('workspace') || DEFAULTS.workspace,

repo: cfg.get('repo') || DEFAULTS.repo,

baseBranch: cfg.get('baseBranch') || DEFAULTS.baseBranch,

};

}

**No .env File Usage:**

* ❌ **No**process.env calls for environment variables
* ❌ **No**dotenv**package** required/imported
* ❌ **No**.env**file** parsing

**Why This is Good for Your Enterprise:**

1. **No external dependencies** for environment management
2. **Uses built-in VS Code Secret Storage** (secure)
3. **No file system access** needed for .env files
4. **Works in locked-down environments**

**To Configure:**

**1. Set VS Code Settings:**

json

{

"bitbucketPRCopilot.workspace": "your-workspace",

"bitbucketPRCopilot.repo": "your-repo",

"bitbucketPRCopilot.baseBranch": "main"

}

**2. First-Time Authentication:**

* Run any PR Copilot command
* It will prompt for:
  + **Bitbucket email**
  + **Bitbucket App Password/Token**
* Credentials are stored securely in VS Code's secret storage

**3. Clear Credentials if Needed:**

javascript

*// Command: "Bitbucket PR Copilot: Clear API Token"*

await context.secrets.delete(SECRET\_KEY);

**Conclusion:**

This extension is **well-designed for enterprise use** because:

* ✅ No .env file dependencies
* ✅ Uses secure VS Code Secret Storage
* ✅ No external configuration files needed
* ✅ Minimal dependencies (only simple-git and built-in modules)

The authentication issue you encountered earlier was likely because:

1. Credentials weren't set up yet, OR
2. The stored credentials had incorrect scopes/permissions

**Try running a PR Copilot command** - it should prompt you for Bitbucket credentials on first use and store them securely.