

GitHub Upload Checklist

Files to Upload to GitHub

INCLUDE These Files/Folders

Root Directory

- README.md - Main documentation
- DEPLOYMENT.md - Deployment guide
- .gitignore - Git exclusions
- package.json (if exists in root)
- package-lock.json (if exists in root)

Backend Folder (backend/)

- server.js - Main application file
- database.js - Database initialization
- package.json - Dependencies list
- package-lock.json - Locked dependency versions
- .env.example - Environment template (NOT .env)

Backend Subfolders

- backend/middleware/ - All files (auth.js, validation.js, etc.)
- backend/utils/ - All files (auth.js, logger.js, etc.)
- backend/scripts/ - All files (hashPasswords.js, etc.)
- backend/assets/ - README.md (logo.png is optional)

Frontend Folder (frontend/)

- index.html - Main HTML file
- app.js - React application

EXCLUDE These Files/Folders

DO NOT Upload

- node_modules/ - Will be installed via npm
- .env - Contains secrets
- *.db - Database files (SQLite)
- *.db-shm - Database temp files
- *.db-wal - Database WAL files
- *.log - Log files
- *.backup - Backup files
- *-old-* - Old backup files
- nul - Windows temp file

Documentation (Optional - Your Choice)

You may exclude these if you want a cleaner repo:

- ADMIN_CONSOLE_GUIDE.md
- ADMIN_FEATURES_IMPLEMENTATION.md
- ANALYTICS_IMPLEMENTATION_GUIDE.md
- ANALYTICS_IMPLEMENTATION_STATUS.md
- ANALYTICS_PROPOSAL.md
- API_TESTING.md
- APPROVAL_WORKFLOW_UPDATE.md
- BROWSER_FIX_GUIDE.md
- BUDGET_FX_AND_REPORTING_GUIDE.md
- CACHE_FIX_FINAL.md
- CHANGES.md
- CONVERSATION_SUMMARY.md
- FINANCE_FLOW_FIX.md
- FINANCE_TO_MD_FLOW_VERIFICATION.md
- FIXES_SUMMARY.md
- FRONTEND_FIXED.md
- FRONTEND_SETUP.md
- HOD_APPROVAL_FIX.md
- HOD_REVIEW_FEATURES.md
- IMPLEMENTATION_COMPLETE.md
- IMPLEMENTATION_STATUS_OCT29.md
- IMPLEMENTATION_SUMMARY_V2.2.md
- IMPLEMENTATION_SUMMARY_V3.0.md
- NEW_FEATURES_V2.1.md
- PERMANENT_FIX_SUMMARY.md
- PO_AND PROCUREMENT_DETAILS_FIX.md
- PO_QUICK_GUIDE.md
- PROCUREMENT_IMPLEMENTATION.md
- PROFILE_AND_REFRESH_TOKEN_FIX.md
- PURCHASE_ORDERS_IMPLEMENTATION.md
- QUICK_REFERENCE_V3.0.md

- [] QUICK_START_USERS.md
- [] README_BLANK_SCREEN_FIX.md
- [] SECURITY.md
- [] SUBMIT_BUTTON_FIX.md
- [] THEME_IMPLEMENTATION.md
- [] TROUBLESHOOTING.md
- [] USER_MANAGEMENT_GUIDE.md
- [] WORKFLOW_GUIDE.md
- [] WORKFLOW_STATUS_FIX.md

Recommendation: Delete or move these to a docs/archive/ folder if you want to keep them.

📄 Pre-Upload Steps

1. Clean Up Development Files

```
# Remove node_modules (will be reinstalled)
rm -rf node_modules
rm -rf backend/node_modules
rm -rf frontend/node_modules

# Remove database files
rm -f backend/*.db
rm -f backend/*.db-shm
rm -f backend/*.db-wal

# Remove log files
rm -f backend/*.log
rm -f *.log

# Remove temp files
rm -f nul
```

2. Verify .env is NOT Included

```
# Check that .env is in .gitignore
cat .gitignore | grep .env

# Ensure .env.example exists
ls backend/.env.example
```

3. Test Clean Install

```
# Test that the app works from scratch
cd backend
npm install
node scripts/hashPasswords.js
npm start

# Verify it starts without errors
```

🔗 Upload to GitHub

Option 1: Using GitHub Desktop

1. Open GitHub Desktop
2. Click "Add" > "Add Existing Repository"
3. Select the purchase-requisition-system folder
4. Review changed files (ensure no .env or .db files)
5. Write commit message: "Initial commit - Purchase Requisition System v3.0"
6. Commit to main
7. Click "Publish repository"
8. Choose public/private
9. Uncheck "Keep this code private" if you want it public
10. Click "Publish Repository"

Option 2: Using Git Command Line

```

# Navigate to project folder
cd C:\Projects\purchase-requisition-system

# Initialize git (if not already initialized)
git init

# Add remote repository
git remote add origin https://github.com/YOUR-USERNAME/purchase-requisition-system.git

# Check what files will be added
git status

# Add all files (respecting .gitignore)
git add .

# Verify no sensitive files are staged
git status

# Create commit
git commit -m "Initial commit - Purchase Requisition System v3.0"

# Push to GitHub
git branch -M main
git push -u origin main

```

Post-Upload Verification

On GitHub Website

1. Visit your repository: <https://github.com/YOUR-USERNAME/purchase-requisition-system>
2. Verify files are uploaded correctly
3. Check that `.env` is NOT visible
4. Check that `node_modules/` is NOT visible
5. Verify `README.md` displays properly
6. Verify `.env.example` is visible

Clone Test

```

# Clone to a different folder
cd /tmp
git clone https://github.com/YOUR-USERNAME/purchase-requisition-system.git test-clone
cd test-clone/backend

# Install and run
npm install
cp .env.example .env
# Edit .env with real values
node scripts/hashPasswords.js
npm start

# Verify it works

```

Security Final Check

Before making repository public:

- [] `.env` is NOT in repository
- [] `JWT_SECRET` placeholder in `.env.example`
- [] No database files (*.db) in repository
- [] No passwords in any code files
- [] `.gitignore` is properly configured
- [] `README.md` warns to change default passwords

Repository Settings (Optional)

Add Repository Description

A comprehensive purchase requisition management system with multi-level approval workflows, analytics, and PDF generation for KSB Zambia

Add Topics/Tags

- requisition-system
- purchase-orders
- workflow
- nodejs
- express
- react
- sqlite
- pdf-generation
- analytics

Add README Sections

- [x] Installation instructions
 - [x] Features list
 - [x] Screenshots (optional - you can add later)
 - [x] API documentation
 - [x] Deployment guide
 - [x] Security information
-

⌚ Essential Files Summary

MUST UPLOAD:

```
purchase-requisition-system/
├── backend/
│   ├── middleware/      ← All files
│   ├── utils/           ← All files
│   ├── scripts/          ← All files
│   ├── assets/           ← README.md only
│   ├── server.js         ← Main file
│   ├── database.js       ← Database init
│   ├── .env.example       ← Template only
│   ├── package.json        ← Dependencies
│   └── package-lock.json  ← Lock file
└── frontend/
    ├── index.html        ← HTML
    └── app.js             ← React app
├── .gitignore           ← Git rules
└── README.md            ← Documentation
└── DEPLOYMENT.md        ← Deploy guide
```

DO NOT UPLOAD:

- `.env` (**secrets**)
 - `*.db` (**database**)
 - `node_modules/` (**dependencies**)
 - `*.log` (**logs**)
 - `*.backup` (**backups**)
-

⚠ Important Reminders

1. **Never commit `.env` file** - Contains secrets
 2. **Never commit database files** - Contains user data
 3. **Never commit `node_modules`** - Too large, regenerated
 4. **Always test after cloning** - Ensure it works for others
 5. **Change default passwords** - After deployment
 6. **Update README** - Replace YOUR-USERNAME with actual username
-

📞 Need Help?

If you encounter issues:

1. Check `.gitignore` is working: `git status`
 2. Review what will be committed: `git diff --staged`
 3. Remove accidentally added files: `git rm --cached <file>`
 4. Reset if needed: `git reset`
-

Ready to Upload!

Follow the steps above and your application will be ready for deployment from GitHub.

Last Updated: October 31, 2025