



## How Forking ERPNext Saves Time: A Smarter Approach to Customization

Selecting a cost-effective **Enterprise Resource Planning (ERP)** solution can be challenging. Among the options, **ERPNext** stands out as a versatile, open-source platform known for its intuitive user interface, modular architecture, and vibrant community. Businesses across sectors rely on ERPNext to streamline operations, enhance efficiency, and drive sustainable growth.

At **Methologik**, we initially approached ERPNext cautiously, concerned about handling complex accounting and reporting customizations. However, its modular design and ease of customization quickly proved invaluable. Unlike many ERP solutions, ERPNext allows rapid feature implementation without compromising stability or future maintainability.

By leveraging a **fork**, our clients were able to address immediate operational needs while building a long-term, scalable framework for growth.

---

### Understanding Forking in ERPNext

#### What is Forking?

Forking involves creating a personal copy of ERPNext's repository to modify and customize the software according to specific business requirements. This approach enables businesses to tailor ERPNext without interfering with core functionality or future updates.

#### Risks of Direct Customization

Directly altering ERPNext core files can create several issues:

- **Upgrade Conflicts:** Modifications may break with ERPNext updates.
- **Maintenance Complexity:** Updates require repeated manual reapplication of customizations.

- **Reduced Stability:** Direct changes can compromise system reliability.
- 

## Benefits of Forking ERPNext

### 1. Faster, Safer Customization

Forking allows developers to freely implement custom features without worrying about breaking future updates.

### 2. Easier Maintenance and Upgrades

Updates are simpler to integrate. Businesses can selectively merge official ERPNext changes while preserving essential customizations.

### 3. Improved Collaboration with Version Control

Using Git for your fork enables seamless teamwork. Multiple developers can work in parallel, track changes clearly, and revert updates if needed.

### 4. Greater Performance Optimization

Forking lets businesses streamline ERPNext by removing unnecessary components, boosting speed and efficiency.

### 5. Enhanced Stability and Scalability

Customizations via a forked repository ensure long-term stability, allowing your ERP system to scale reliably as business demands evolve.

---

## Recommended Best Practices

- **Separate Development:** Build individual features independently before merging.
  - **Selective Updates:** Regularly sync your fork carefully to integrate new features while maintaining customizations.
  - **Documentation:** Keep detailed records of changes for future reference.
  - **Minimalist Approach:** Focus only on essential customizations to simplify maintenance.
-

## **Practical Steps to Fork and Manage ERPNext**

### **1. Fork ERPNext**

Fork the ERPNext repository on GitHub, then clone locally:

```
git clone https://github.com/YOUR_GITHUB_USERNAME/erpnext.git
```

### **2. Configure Remotes**

Add ERPNext's official repository as upstream:

```
git remote add upstream https://github.com/frappe/erpnext.git
```

```
git remote -v
```

### **3. Set User Information**

```
git config --global user.name "Your Name"
```

```
git config --global user.email "your-email@example.com"
```

### **4. Sync Your Fork with Upstream**

```
git fetch upstream
```

```
git checkout main
```

```
git merge upstream/main
```

### **5. Check Site Source**

Ensure your site runs from your fork (origin) but can still integrate upstream updates:

```
cd /path/to/frappe-bench/apps/erpnext
```

```
git remote -v
```

### **6. Update Your ERPNext Instance**

```
cd /path/to/frappe-bench/apps/erpnext
```

```
git fetch upstream
```

```
git checkout version-15 # replace version as needed
```

```
git rebase upstream/version-15
```

```
git reset --hard upstream/version-15 # only if needed
```

```
git log --oneline -1
```

git status

git checkout branchname

bench restart

bench migrate

bench build

---

## Conclusion

Forking ERPNext is not just a technical decision — it's a strategic business choice. By following a structured, Git-based approach, companies can:

- Implement customizations efficiently
- Maintain system stability
- Achieve sustainable growth
- Leverage ERPNext's full potential without compromising flexibility or performance

At **Methologik**, we have extensive experience helping businesses customize and scale ERPNext. Our team ensures your ERP solution meets your operational needs while remaining upgrade-safe and scalable.

---

## Get Expert ERPNext Support

Whether you need **customization, forking, or full ERP implementation**, Methologik's experienced team can guide you through every step.

 **Email us:** [info@methologik.com](mailto:info@methologik.com)

 **Request a consultation:** <https://methologik.com/contact-us>