🚀 Manufacturing Estimation Module - Complete Deployment Guide

📁 Complete File Structure

```
mrp_estimation/
├── __init__.py
├── __manifest__.py
│
├── controllers/
│   ├── __init__.py
│   ├── main.py                    # Portal controllers
│   └── api.py                     # REST API endpoints
│
├── data/
│   ├── estimation_sequence.xml    # Sequences and basic data
│   ├── estimation_data.xml        # Default data and templates
│   └── email_templates.xml        # Email templates
│
├── demo/
│   └── estimation_demo.xml        # Demo data for testing
│
├── migrations/
│   └── 18.0.1.0.0/
│       ├── pre-migrate.py         # Pre-migration scripts
│       └── post-migrate.py        # Post-migration scripts
│
├── models/
│   ├── __init__.py
│   ├── estimation.py              # Main estimation model
│   ├── estimation_line.py         # Material lines model
│   ├── estimation_cost.py         # Cost breakdown model
│   ├── estimation_version.py      # Version control model
│   ├── mrp_costing.py             # Manufacturing costing
│   ├── res_config_settings.py     # Configuration settings
│   ├── portal_mixin.py            # Portal integration
│   ├── calculation_engine.py      # Advanced calculations
│   ├── sale_integration.py        # Sales integration
│   ├── mrp_integration.py         # Manufacturing integration
│   ├── automation.py              # Automated actions
│   └── performance.py             # Performance optimizations
│
├── reports/
│   ├── estimation_report_views.xml # Report actions
│   ├── estimation_templates.xml    # QWeb report templates
│   └── estimation_reports.py       # Report wizards
│
├── security/
│   ├── estimation_security.xml     # Security groups and rules
│   └── ir.model.access.csv        # Access rights
│
```

```
├── static/
│   ├── description/
│   │   ├── icon.png              # Module icon
│   │   ├── index.html            # Module description
│   │   └── banner.png            # Banner image
│   └── src/
│       ├── css/
│       │   ├── estimation.css    # Custom styles
│       │   └── portal.css        # Portal styles
│       ├── js/
│       │   ├── estimation_widget.js    # Custom widgets
│       │   ├── estimation_dashboard.js # Dashboard component
│       │   └── field_widgets.js        # Field widgets
│       └── xml/
│           └── estimation_templates.xml # QWeb templates
│
├── tests/
│   ├── __init__.py
│   ├── test_estimation.py        # Core functionality tests
│   ├── test_estimation_workflow.py # Workflow tests
│   └── test_estimation_api.py    # API tests
│
├── views/
│   ├── estimation_views.xml      # Main estimation views
│   ├── estimation_line_views.xml  # Material line views
│   ├── estimation_cost_views.xml  # Cost breakdown views
│   ├── estimation_version_views.xml # Version control views
│   ├── mrp_costing_views.xml     # Costing analysis views
│   ├── estimation_menus.xml      # Menu structure
│   ├── estimation_dashboard.xml   # Dashboard views
│   ├── portal_estimation_views.xml # Portal templates
│   ├── estimation_actions.xml    # Additional actions
│   └── res_config_settings_views.xml # Settings views
│
└── wizard/
    ├── __init__.py
    ├── estimation_wizard.py      # Creation wizard
    └── estimation_wizard_views.xml # Wizard views
```

🔧 Installation & Setup Guide

Step 1: Prerequisites Check

bash

```bash
# Check Python version (3.10+ required)
python3 --version

# Check PostgreSQL version (12+ required)
psql --version

# Check Odoo installation
./odoo-bin --version
```

Step 2: Module Installation

bash

```bash
# 1. Clone or copy module to addons directory
cp -r mrp_estimation /path/to/odoo/addons/

# 2. Update addons path (if needed)
./odoo-bin --addons-path=addons,/path/to/custom/addons

# 3. Install Python dependencies
pip install xlsxwriter  # For Excel reports

# 4. Restart Odoo server
sudo systemctl restart odoo  # For system service
# OR
./odoo-bin --stop && ./odoo-bin --start  # For manual setup
```

Step 3: Module Activation

1. **Login to Odoo**
   - Go to http://your-odoo-url
   - Login as administrator
2. **Update Apps List**
   - Go to Apps menu
   - Click "Update Apps List"
   - Wait for completion
3. **Install Module**
   - Search for "Manufacturing Estimation"
   - Click "Install"
   - Wait for installation to complete

Step 4: Initial Configuration

4.1 User Groups Setup

```python
# Navigate to: Settings → Users & Companies → Users
# Assign users to appropriate groups:

# For Regular Users:
- Estimation User

# For Team Leaders:
- Estimation User
- Estimation Approver

# For Managers:
- Estimation Manager (includes all other permissions)
```

4.2 System Configuration

```python
# Navigate to: Manufacturing Estimation → Configuration → Settings

# Configure the following:
estimation_sequence_prefix = "EST"          # Default: EST
auto_markup_enabled = False                 # Enable auto-markup
approval_required = True                    # Require approval workflow
default_material_markup = 10.0              # Default material markup %
default_cost_markup = 15.0                  # Default cost markup %
auto_approve_threshold = 1000.0             # Auto-approve threshold
```

4.3 Email Configuration

```xml
<!-- Configure email templates -->
<!-- Go to: Settings → Technical → Email Templates -->

<!-- Ensure SMTP is configured -->
<!-- Go to: Settings → General Settings → Email -->
```

Step 5: Data Setup

5.1 Create Product Categories

python

```
# Navigate to: Inventory → Configuration → Product Categories
# Create categories for:
- Raw Materials
- Components
- Finished Goods
```

5.2 Setup Work Centers

python

```
# Navigate to: Manufacturing → Configuration → Work Centers
# Create work centers with:
- Hourly costs
- Capacity information
- Operating time
```

5.3 Configure Suppliers

python

```
# Navigate to: Purchase → Vendors
# Setup vendor information including:
- Lead times
- Price agreements
- Quality ratings
```

📋 User Manual Excerpts

Creating Your First Estimation

1. **Access Estimation Module**
   - Click "Manufacturing Estimation" in the main menu
   - Select "Estimations" → "All Estimations"
2. **Create New Estimation**

   ```
   Click "Create" button
   Fill required fields:
   - Customer: Select from dropdown
   - Product: Choose product to manufacture
   - Quantity: Enter production quantity
   - Unit of Measure: Auto-filled from product
   ```

3. **Add Material Lines**

```
Go to "Material Lines" tab
Click "Add a line"
- Product: Select raw material/component
- Quantity: Required quantity
- Unit Cost: Will auto-fill from product cost
- Markup %: Apply markup if needed
```

4. **Add Cost Breakdown**

```
Go to "Cost Breakdown" tab
Add different cost types:
- Operation Costs: Workcenter × Time
- Labor Costs: Hours × Rate + Overhead
- Miscellaneous: Setup, tooling, etc.
```

5. **Configure Markup**

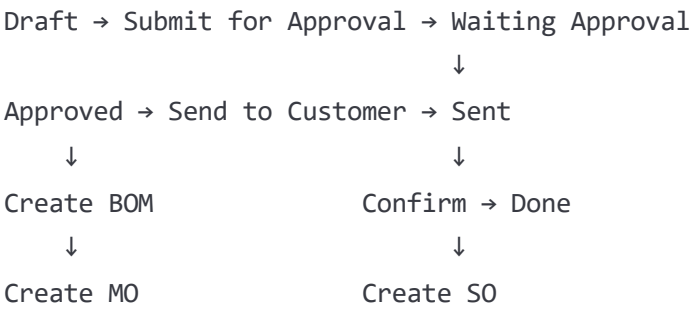
```
Go to "Markup Configuration" tab
Set markup type and value:
- Material Markup: Fixed amount or percentage
- Cost Markup: Additional markup for costs
```

6. **Submit for Approval**

```
Click "Submit for Approval"
System will notify managers
Track approval status in header
```

Workflow Management

State Transitions



Manager Actions

```python
# For managers in "Waiting Approval" state:
- Review estimation details
- Check cost calculations
- Approve or reject with comments
- Request modifications if needed
```

Advanced Features Usage

Version Control

```python
# Creating versions:
1. Open approved estimation
2. Click "Create Version" button
3. System creates copy with incremented version
4. Make changes in new version
5. Original version preserved for reference
```

Portal Sharing

```python
# Share with customers:
1. Ensure estimation is approved
2. Click "Send to Customer"
3. Customer receives email with portal link
4. Customer can view details and respond
```

Cost Analysis

```python
# Manufacturing cost tracking:
1. Create Manufacturing Order from estimation
2. System creates costing record
3. Track actual vs planned costs
4. Analyze variance and improve estimates
```

🔍 Testing Checklist

Functional Testing

- [ ] **Estimation Creation**
- [ ] Create estimation with required fields

- [ ] Auto-populate from existing BOM
- [ ] Validation error handling
- [ ] **Material Lines**
- [ ] Add/edit material lines
- [ ] Cost calculations
- [ ] Markup calculations
- [ ] Stock availability check
- [ ] **Cost Breakdown**
- [ ] Operation cost calculations
- [ ] Labor cost with overhead
- [ ] Miscellaneous costs
- [ ] Total cost aggregation
- [ ] **Workflow Testing**
- [ ] Draft → Waiting Approval
- [ ] Approve/Reject functionality
- [ ] Send to customer
- [ ] Confirmation process
- [ ] **Integration Testing**
- [ ] BOM creation from estimation
- [ ] Sales order generation
- [ ] Manufacturing order creation
- [ ] Portal access for customers
- [ ] **Security Testing**
- [ ] User group permissions
- [ ] Record rule enforcement
- [ ] Field-level security

Performance Testing

- [ ] **Load Testing**
- [ ] 1000+ estimations
- [ ] Complex material lists (100+ items)
- [ ] Bulk operations
- [ ] **Response Time**
- [ ] Form loading < 2 seconds
- [ ] List view < 3 seconds
- [ ] Report generation < 10 seconds

API Testing

python

```
# Test API endpoints
curl -X POST http://localhost:8069/api/v1/estimations \
  -H "Content-Type: application/json" \
  -d '{
    "partner_id": 1,
    "product_id": 1,
    "product_qty": 10
  }'
```

🚀 Production Deployment

Pre-Deployment Checklist

- [ ] **Environment Setup**
- [ ] Production server configured
- [ ] Database backup completed
- [ ] SSL certificates installed
- [ ] Email server configured
- [ ] **Module Validation**
- [ ] All tests passed
- [ ] Code review completed
- [ ] Documentation updated
- [ ] Security audit done
- [ ] **Data Migration**
- [ ] Existing data backed up
- [ ] Migration scripts tested
- [ ] Rollback plan prepared

Deployment Steps

1. **Backup Current System**

   bash

   ```
   # Database backup
   pg_dump -h localhost -U odoo production_db > backup_$(date +%Y%m%d).sql

   # Filestore backup
   tar -czf filestore_backup_$(date +%Y%m%d).tar.gz /var/lib/odoo/filestore/
   ```

2. **Deploy Module**

bash

```bash
# Copy module to production
rsync -av mrp_estimation/ production_server:/opt/odoo/addons/

# Update module
./odoo-bin -d production_db -u mrp_estimation
```

3. **Post-Deployment Validation**

bash

```bash
# Check module installation
# Verify key functionality
# Monitor system logs
# Test user access
```