EITS ERPNext Implementation Guide - Phases 1-3

Prerequisites Checklist ✅

Before we start, ensure you have:

- [ ] Ubuntu 20.04+ / macOS / Windows with WSL2
- [ ] Python 3.8+ installed
- [ ] 8GB RAM minimum
- [ ] Stable internet connection
- [ ] Basic terminal/command line knowledge

PHASE 1: Foundation Setup (Week 1-2)

Step 1: Install Frappe Bench Environment

1.1 Install System Dependencies

```
# Update your system
sudo apt update && sudo apt upgrade -y

# Install required packages
sudo apt install -y python3-dev python3-pip python3-venv python3-setuptools
sudo apt install -y software-properties-common
sudo apt install -y git curl wget
sudo apt install -y build-essential
```

1.2 Install Database (MariaDB)

```
# Install MariaDB
sudo apt install -y mariadb-server mariadb-client

# Secure MariaDB installation
sudo mysql_secure_installation

# When prompted:
# - Set root password: YES (choose a strong password)
# - Remove anonymous users: YES
# - Disallow root login remotely: YES
# - Remove test database: YES
# - Reload privilege tables: YES
```

1.3 Configure Database User

```
# Login to MariaDB as root
sudo mysql -u root -p

# Inside MySQL prompt, run these commands:
CREATE USER 'eits_user'@'localhost' IDENTIFIED BY 'eits_password_123';
GRANT ALL PRIVILEGES ON *.* TO 'eits_user'@'localhost' WITH GRANT OPTION;
FLUSH PRIVILEGES;
EXIT;
```

1.4 Install Redis and Node.js

```
# Install Redis
sudo apt install -y redis-server

# Install Node.js 16
curl -fsSL https://deb.nodesource.com/setup_16.x | sudo -E bash -
sudo apt install -y nodejs

# Install Yarn
sudo npm install -g yarn

# Install wkhtmltopdf for PDF generation
sudo apt install -y wkhtmltopdf
```

1.5 Install Frappe Bench

```
# Install Frappe Bench
pip3 install frappe-bench

# Add to PATH (add this to your ~/.bashrc or ~/.zshrc)
echo 'export PATH=$PATH:~/.local/bin' >> ~/.bashrc
source ~/.bashrc

# Verify installation
bench --version
```

Step 2: Create Development Environment

2.1 Initialize Bench

```
# Create a new bench directory
bench init eits-bench --frappe-branch version-14

# Navigate to bench directory
cd eits-bench
```

```
# Start bench to verify everything works
bench start
```

Expected Output: You should see multiple processes starting (redis, web, socketio, etc.)

Stop the bench: Press `Ctrl+C`

2.2 Create EITS Site

```
# Create new site for EITS
bench new-site eits.local --admin-password admin123 --mariadb-root-password
[your_mariadb_root_password]

# Set as default site
bench use eits.local

# Test site access
bench start
```

Access your site: Open browser and go to `http://eits.local:8000`

- Username: `Administrator`
- Password: `admin123`

Step 3: Install ERPNext

3.1 Get ERPNext App

```
# Stop bench if running
# Ctrl+C

# Get ERPNext from GitHub
bench get-app erpnext --branch version-14

# Install ERPNext on your site
bench --site eits.local install-app erpnext

# Start bench
bench start
```

3.2 Complete ERPNext Setup Wizard

1. **Access the site:** `http://eits.local:8000`
2. **Complete Setup Wizard:**

```
Language: English
Country: United Arab Emirates (or your country)
Timezone: Asia/Dubai (or your timezone)

Company Details:
- Company Name: EITS - Equipment Installation & Technical Services
- Company Abbreviation: EITS
- Default Currency: AED
- Company Email: admin@eits.local
- Company Website: www.eits.com

First User:
- Full Name: EITS Administrator
- Email: admin@eits.local
- Password: admin123

Domain: Services

Organization:
- Bank Account: EITS Main Account
- Mode of Payment: Cash, Bank Transfer, Credit Card
```

Step 4: Create Custom EITS App

4.1 Generate Custom App

```
# Stop bench
# Ctrl+C

# Create EITS custom app
bench new-app eits_app

# App creation prompts:
App Title: EITS Management System
App Description: Custom ERPNext application for Equipment Installation and
Technical Services
App Publisher: EITS
App Email: admin@eits.local
App Icon: fa fa-tools
App Color: #2E8B57
App License: MIT

# Install custom app on site
bench --site eits.local install-app eits_app
```

4.2 Verify App Installation

```
# Check installed apps
bench --site eits.local list-apps

# You should see:
# frappe
# erpnext
# eits_app
```

Step 5: Basic Master Data Setup

5.1 Setup Job Categories as Item Groups

1. **Access ERPNext:** `http://eits.local:8000`
2. **Navigate:** Stock → Setup → Item Group
3. **Create the following Item Groups:**

```
Main Item Groups to Create:

1. EITS Services (Parent Group)
   ├── Furnitures & Fixtures
   ├── Equipment Installation
   ├── Electrical Services
   ├── AC & Refrigeration
   ├── Painting & Decorating
   ├── Plumbing Services
   ├── Construction Services
   └── Trading Services
```

Step-by-step Item Group Creation:

1. Click "New Item Group"
2. Fill details:
   - **Item Group Name:** EITS Services
   - **Parent Item Group:** All Item Groups
   - **Is Group:** ✅ Checked
3. Save

Repeat for each sub-category with:

- **Parent Item Group:** EITS Services
- **Is Group:** ✅ Checked

5.2 Create Service Items

Navigate to Stock → Item → New Item

Create these service items:

```
Item 1:
- Item Code: SERV-FF-001
- Item Name: Furniture Installation Service
- Item Group: Furnitures & Fixtures
- Is Stock Item: ❌ Unchecked
- Is Sales Item: ✅ Checked
- Is Service Item: ✅ Checked
- Standard Selling Rate: 100

Item 2:
- Item Code: SERV-EQ-001
- Item Name: Equipment Installation Service
- Item Group: Equipment Installation
- Is Stock Item: ❌ Unchecked
- Is Sales Item: ✅ Checked
- Is Service Item: ✅ Checked
- Standard Selling Rate: 150

Item 3:
- Item Code: SERV-EL-001
- Item Name: Electrical Installation Service
- Item Group: Electrical Services
- Is Stock Item: ❌ Unchecked
- Is Sales Item: ✅ Checked
- Is Service Item: ✅ Checked
- Standard Selling Rate: 200
```

Continue creating items for each service category.

5.3 Setup Customer Groups

Navigate to Selling → Setup → Customer Group

Create these customer groups:

```
1. Residential Customers
2. Commercial Customers
3. Industrial Customers
4. Government Clients
5. Real Estate Developers
```

5.4 Create Employee Records

Navigate to Human Resources → Employee → New Employee

Create key employees:

```
Employee 1:
- Employee Name: Ahmed Al Mansouri
```

```
- Employee Number: EMP-001
- Designation: Sales Manager
- Department: Sales
- Email: ahmed@eits.local

Employee 2:
- Employee Name: Mohammed Hassan
- Employee Number: EMP-002
- Designation: Site Inspector
- Department: Operations
- Email: mohammed@eits.local

Employee 3:
- Employee Name: Fatima Al Zahra
- Employee Number: EMP-003
- Designation: Technical Specialist
- Department: Technical
- Email: fatima@eits.local
```

Phase 1 Verification ✅

Check these items are completed:

- [ ] Bench and ERPNext installed successfully
- [ ] EITS custom app created and installed
- [ ] Item Groups for all service categories created
- [ ] Service items for main categories created
- [ ] Customer groups configured
- [ ] Key employee records created
- [ ] Site accessible at `http://eits.local:8000`

PHASE 2: Lead Management Enhancement (Week 3-4)

Step 1: Customize Lead DocType

1.1 Add EITS-Specific Fields to Lead

1. **Navigate:** Customize → Customize Form
2. **Select DocType:** Lead
3. **Add these custom fields:**

Field 1: Property Details Section

```
Field Type: Section Break
Label: Property Details
Insert After: address_line2
```

```
```

Field 2: Property Type

```
Field Type: Select
Label: Property Type
Field Name: property_type
Options:
Residential
Commercial
Industrial
Insert After: Property Details (section break)
In List View: ✅
```

Field 3: Project Type

```
Field Type: Select
Label: Project Type
Field Name: project_type
Options:
Fitouts
Refurbishment
Renovation
Repair
Maintenance
Installation
Construction
Trading
Insert After: property_type
In List View: ✅
```

Field 4: Budget Range

```
Field Type: Select
Label: Budget Range
Field Name: budget_range
Options:
Under 10,000 AED
10,000 - 50,000 AED
50,000 - 100,000 AED
100,000 - 500,000 AED
Above 500,000 AED
Insert After: project_type
```

Field 5: Preferred Inspection Date

```
```

```
Field Type: Date
Label: Preferred Inspection Date
Field Name: preferred_inspection_date
Insert After: budget_range
```

Field 6: Alternative Date

```
Field Type: Date
Label: Alternative Inspection Date
Field Name: alternative_inspection_date
Insert After: preferred_inspection_date
```

Field 7: Project Timeline

```
Field Type: Select
Label: Project Urgency
Field Name: project_urgency
Options:
Urgent (Within 1 week)
Normal (Within 1 month)
Flexible (Within 3 months)
Future Planning (3+ months)
Insert After: alternative_inspection_date
```

Field 8: Special Requirements

```
Field Type: Text
Label: Special Requirements
Field Name: special_requirements
Insert After: project_urgency
```

1.2 Update Lead List View

1. In Customize Form for Lead
2. Go to "List View Settings"
3. Add these fields to list view:
    - property_type
    - project_type
    - budget_range
    - preferred_inspection_date

1.3 Create Custom Lead Dashboard

1. **Navigate:** Build → Report Builder

2. **Create New Report:**
   - Report Name: EITS Lead Dashboard
   - Reference DocType: Lead
   - Report Type: Query Report

Add these columns:

- Lead Name
- Property Type
- Project Type
- Budget Range
- Status
- Preferred Inspection Date
- Source

Step 2: Create Site Inspection DocType

2.1 Create New DocType

1. **Navigate:** Build → DocType → New DocType
2. **DocType Configuration:**

```
DocType Name: Site Inspection
Module: EITS App
Is Submittable: ✅ Yes
Is Child Table: ❌ No
Track Changes: ✅ Yes
```

2.2 Add Fields to Site Inspection DocType

Basic Information Section:

```
Field 1:
- Type: Data
- Label: Inspection ID
- Field Name: inspection_id
- Hidden: ✅
- Read Only: ✅
- Default: SI-.####

Field 2:
- Type: Link
- Label: Lead
- Field Name: lead
- Options: Lead
- Reqd: ✅
- In List View: ✅

Field 3:
```

```
- Type: Data
- Label: Customer Name
- Field Name: customer_name
- Fetch From: lead.lead_name
- Read Only: ✅
- In List View: ✅

Field 4:
- Type: Column Break

Field 5:
- Type: Date
- Label: Inspection Date
- Field Name: inspection_date
- Reqd: ✅
- Default: Today
- In List View: ✅

Field 6:
- Type: Time
- Label: Inspection Time
- Field Name: inspection_time
- Reqd: ✅

Field 7:
- Type: Link
- Label: Inspector
- Field Name: inspector
- Options: Employee
- Reqd: ✅
- In List View: ✅
```

Property Details Section:

```
Field 8:
- Type: Section Break
- Label: Property Details

Field 9:
- Type: Text
- Label: Property Address
- Field Name: property_address
- Reqd: ✅

Field 10:
- Type: Select
- Label: Property Type
- Field Name: property_type_inspection
- Options:
  Residential
  Commercial
  Industrial
```

```
- Fetch From: lead.property_type

Field 11:
- Type: Select
- Label: Project Type
- Field Name: project_type_inspection
- Options:
  Fitouts
  Refurbishment
  Renovation
  Repair
  Maintenance
  Installation
  Construction
  Trading
- Fetch From: lead.project_type

Field 12:
- Type: Column Break

Field 13:
- Type: Text
- Label: Site Accessibility Notes
- Field Name: site_accessibility

Field 14:
- Type: Text
- Label: Existing Structure Condition
- Field Name: existing_condition

Field 15:
- Type: Check
- Label: Utilities Available (Water/Electricity)
- Field Name: utilities_available
```

Site Measurements Section:

```
Field 16:
- Type: Section Break
- Label: Site Measurements

Field 17:
- Type: Table
- Label: Site Dimensions
- Field Name: site_dimensions
- Options: Site Dimension
```

We need to create the Child Table "Site Dimension":

1. **Create New DocType:** Site Dimension

2. **Configuration:**

```
DocType Name: Site Dimension
Is Child Table: ✅ Yes
Module: EITS App
```

3. **Add Fields to Site Dimension:**

```
Field 1:
- Type: Data
- Label: Area Name
- Field Name: area_name
- In List View: ✅
- Reqd: ✅

Field 2:
- Type: Float
- Label: Length (m)
- Field Name: length
- Precision: 2
- In List View: ✅

Field 3:
- Type: Float
- Label: Width (m)
- Field Name: width
- Precision: 2
- In List View: ✅

Field 4:
- Type: Float
- Label: Height (m)
- Field Name: height
- Precision: 2
- In List View: ✅

Field 5:
- Type: Float
- Label: Area (m²)
- Field Name: area
- Read Only: ✅
- In List View: ✅
```

Documentation Section (Continue in Site Inspection):

```
Field 18:
- Type: Section Break
- Label: Documentation
```

```
Field 19:
- Type: Attach Multiple
- Label: Site Photos
- Field Name: site_photos

Field 20:
- Type: Attach
- Label: Measurement Sketch
- Field Name: measurement_sketch

Field 21:
- Type: Text Editor
- Label: Inspection Notes
- Field Name: inspection_notes
```

Status Section:

```
Field 22:
- Type: Section Break
- Label: Status & Follow-up

Field 23:
- Type: Select
- Label: Inspection Status
- Field Name: inspection_status
- Options:
  Scheduled
  In Progress
  Completed
  Rescheduled
  Cancelled
- Default: Scheduled
- In List View: ✅

Field 24:
- Type: Column Break

Field 25:
- Type: Check
- Label: Follow-up Required
- Field Name: followup_required

Field 26:
- Type: Text
- Label: Next Action
- Field Name: next_action
```

2.3 Save and Create DocType

1. **Save** the Site Inspection DocType
2. **Check "Has Web View"** if you want web access
3. **Save and Create** the DocType

Step 3: Create Client Scripts for Automation

3.1 Site Inspection Auto-calculations

1. **Navigate:** Build → Client Script → New Client Script
2. **Configuration:**

```
Script Name: Site Inspection Calculations
DocType: Site Inspection
Type: Form Script
```

Script Content:

```
frappe.ui.form.on('Site Inspection', {
    refresh: function(frm) {
        // Add custom button to create estimation
        if (frm.doc.docstatus === 1 && frm.doc.inspection_status ===
'Completed') {
            frm.add_custom_button(__('Create Job Estimation'), function() {
                frappe.new_doc('Job Estimation', {
                    site_inspection: frm.doc.name,
                    customer: frm.doc.customer_name,
                    inspection_date: frm.doc.inspection_date
                });
            }, __('Actions'));
        }

        // Auto-fetch lead details
        if (frm.doc.lead && !frm.doc.property_address) {
            frappe.call({
                method: 'frappe.client.get',
                args: {
                    doctype: 'Lead',
                    name: frm.doc.lead
                },
                callback: function(r) {
                    if (r.message) {
                        frm.set_value('property_address',
r.message.address_line1 + ', ' + r.message.city);
                        frm.set_value('property_type_inspection',
r.message.property_type);
                        frm.set_value('project_type_inspection',
r.message.project_type);
                    }
                }
            });
```

```
        }
    }
});

// Auto-calculate area in child table
frappe.ui.form.on('Site Dimension', {
    length: function(frm, cdt, cdn) {
        calculate_area(frm, cdt, cdn);
    },
    width: function(frm, cdt, cdn) {
        calculate_area(frm, cdt, cdn);
    }
});

function calculate_area(frm, cdt, cdn) {
    let row = locals[cdt][cdn];
    if (row.length && row.width) {
        row.area = row.length * row.width;
        frm.refresh_field('site_dimensions');
    }
}
```

3.2 Lead Enhancement Script

1. **Create Client Script for Lead:**

```
Script Name: EITS Lead Enhancement
DocType: Lead
Type: Form Script
```

Script Content:

```
frappe.ui.form.on('Lead', {
    refresh: function(frm) {
        // Add custom button to schedule inspection
        if (frm.doc.status === 'Open' && !frm.doc.__islocal) {
            frm.add_custom_button(__('Schedule Site Inspection'),
function() {
                frappe.new_doc('Site Inspection', {
                    lead: frm.doc.name,
                    customer_name: frm.doc.lead_name,
                    inspection_date: frm.doc.preferred_inspection_date,
                    property_address: frm.doc.address_line1 + ', ' +
frm.doc.city
                });
            }, __('Actions'));
        }

        // Color coding based on urgency
```

```
        if (frm.doc.project_urgency === 'Urgent (Within 1 week)') {
            frm.dashboard.add_indicator(__('Urgent Project'), 'red');
        }
    },

    preferred_inspection_date: function(frm) {
        // Validate inspection date is not in the past
        if (frm.doc.preferred_inspection_date) {
            let today = frappe.datetime.get_today();
            if (frm.doc.preferred_inspection_date < today) {
                frappe.msgprint(__('Preferred inspection date cannot be in
the past'));
                frm.set_value('preferred_inspection_date', '');
            }
        }
    }
});
```

Step 4: Create Workflows

4.1 Lead to Inspection Workflow

1. **Navigate:** Build → Workflow → New Workflow
2. **Configuration:**

```
Workflow Name: EITS Lead Process
Document Type: Lead
Workflow State Field: workflow_state
Is Active: ✅
```

States:

```
State 1:
- State: Lead Received
- Doc Status: 0 (Draft)
- Allow Edit: Sales User, Sales Manager
- Style: Primary

State 2:
- State: Inspection Scheduled
- Doc Status: 0 (Draft)
- Allow Edit: Sales User, Site Inspector
- Style: Success

State 3:
- State: Inspection Completed
- Doc Status: 1 (Submitted)
- Allow Edit: Site Inspector
- Style: Success
```

```
State 4:
- State: Estimation Required
- Doc Status: 1 (Submitted)
- Allow Edit: Technical Specialist
- Style: Warning

State 5:
- State: Qualified
- Doc Status: 1 (Submitted)
- Allow Edit: Sales Manager
- Style: Success
```

Transitions:

```
Transition 1:
- From: Lead Received
- To: Inspection Scheduled
- Action: Schedule Inspection
- Allowed: Sales User, Sales Manager
- Condition: preferred_inspection_date is not null

Transition 2:
- From: Inspection Scheduled
- To: Inspection Completed
- Action: Complete Inspection
- Allowed: Site Inspector

Transition 3:
- From: Inspection Completed
- To: Estimation Required
- Action: Request Estimation
- Allowed: Site Inspector, Sales Manager

Transition 4:
- From: Estimation Required
- To: Qualified
- Action: Qualify Lead
- Allowed: Technical Specialist, Sales Manager
```

Phase 2 Verification ✅

Check these items are completed:

- [ ] Lead DocType customized with EITS-specific fields
- [ ] Site Inspection DocType created with all required fields
- [ ] Site Dimension child table created
- [ ] Client scripts for auto-calculations working
- [ ] Lead to Site Inspection workflow configured

- [ ] Custom buttons appear on forms
- [ ] Area calculations work in site dimensions table

PHASE 3: Job Estimation System (Week 5-6)

Step 1: Create Job Estimation DocType

1.1 Create Main DocType

1. **Navigate:** Build → DocType → New DocType
2. **Configuration:**

```
DocType Name: Job Estimation
Module: EITS App
Is Submittable: ✅ Yes
Track Changes: ✅ Yes
```

1.2 Add Fields to Job Estimation

Header Section:

```
Field 1:
- Type: Data
- Label: Estimation ID
- Field Name: estimation_id
- Hidden: ✅
- Read Only: ✅
- Default: EST-.####

Field 2:
- Type: Link
- Label: Site Inspection
- Field Name: site_inspection
- Options: Site Inspection
- Reqd: ✅
- In List View: ✅

Field 3:
- Type: Link
- Label: Customer
- Field Name: customer
- Options: Customer
- Fetch From: site_inspection.lead.customer
- Read Only: ✅

Field 4:
- Type: Column Break
```

```
Field 5:
- Type: Date
- Label: Estimation Date
- Field Name: estimation_date
- Default: Today
- Reqd: ✅
- In List View: ✅

Field 6:
- Type: Link
- Label: Estimated By
- Field Name: estimated_by
- Options: Employee
- Reqd: ✅

Field 7:
- Type: Select
- Label: Estimation Status
- Field Name: estimation_status
- Options:
  Draft
  Submitted
  Under Review
  Approved
  Rejected
  Revision Required
- Default: Draft
- In List View: ✅
```

Job Categories Section:

```
Field 8:
- Type: Section Break
- Label: Job Categories Required

Field 9:
- Type: Table
- Label: Job Categories
- Field Name: job_categories
- Options: Job Category Detail
```

Create Job Category Detail Child Table:

1. **New DocType:** Job Category Detail

2. **Configuration:**

   ```
   DocType Name: Job Category Detail
   Is Child Table: ✅ Yes
   ```

```
Module: EITS App
```

3. **Fields for Job Category Detail:**

```
Field 1:
- Type: Select
- Label: Job Category
- Field Name: job_category
- Options:
  Furnitures & Fixtures
  Equipment Installation
  Electrical Services
  AC & Refrigeration
  Painting & Decorating
  Plumbing Services
  Construction Services
  Trading Services
- In List View: ✅
- Reqd: ✅

Field 2:
- Type: Data
- Label: Sub Category
- Field Name: sub_category
- In List View: ✅

Field 3:
- Type: Data
- Label: Department Required
- Field Name: department
- In List View: ✅

Field 4:
- Type: Select
- Label: Priority
- Field Name: priority
- Options:
  High
  Medium
  Low
- Default: Medium
- In List View: ✅

Field 5:
- Type: Float
- Label: Estimated Hours
- Field Name: estimated_hours
- Precision: 2

Field 6:
- Type: Currency
- Label: Estimated Cost
```

```
- Field Name: estimated_cost
- In List View: ✅

Field 7:
- Type: Text
- Label: Notes
- Field Name: notes
```

Material Requirements Section:

```
Field 10:
- Type: Section Break
- Label: Material Requirements

Field 11:
- Type: Table
- Label: Materials
- Field Name: materials
- Options: Material Estimation
```

Create Material Estimation Child Table:

1. **New DocType:** Material Estimation

2. **Configuration:**

   ```
   DocType Name: Material Estimation
   Is Child Table: ✅ Yes
   Module: EITS App
   ```

3. **Fields for Material Estimation:**

   ```
   Field 1:
   - Type: Link
   - Label: Item
   - Field Name: item_code
   - Options: Item
   - In List View: ✅
   - Reqd: ✅

   Field 2:
   - Type: Data
   - Label: Item Name
   - Field Name: item_name
   - Fetch From: item_code.item_name
   - Read Only: ✅
   - In List View: ✅
   ```

```
Field 3:
- Type: Float
- Label: Quantity
- Field Name: qty
- Precision: 2
- In List View: ✅
- Reqd: ✅

Field 4:
- Type: Link
- Label: UOM
- Field Name: uom
- Options: UOM
- Fetch From: item_code.stock_uom
- In List View: ✅

Field 5:
- Type: Currency
- Label: Rate
- Field Name: rate
- Precision: 2
- In List View: ✅
- Reqd: ✅

Field 6:
- Type: Currency
- Label: Amount
- Field Name: amount
- Read Only: ✅
- In List View: ✅
- Formula: qty * rate

Field 7:
- Type: Select
- Label: Availability
- Field Name: availability
- Options:
  In Stock
  Need to Purchase
  Special Order
- Default: Need to Purchase

Field 8:
- Type: Text
- Label: Specification
- Field Name: specification
```

Manpower Requirements Section:

```
Field 12:
- Type: Section Break
```

```
- Label: Manpower Requirements

Field 13:
- Type: Table
- Label: Manpower
- Field Name: manpower
- Options: Manpower Estimation
```

Create Manpower Estimation Child Table:

1. **New DocType:** Manpower Estimation

2. **Configuration:**

```
DocType Name: Manpower Estimation
Is Child Table: ✅ Yes
Module: EITS App
```

3. **Fields for Manpower Estimation:**

```
Field 1:
- Type: Data
- Label: Skill Set
- Field Name: skill_set
- In List View: ✅
- Reqd: ✅
- Options:
  Carpenter
  Electrician
  Plumber
  Painter
  AC Technician
  General Labor
  Supervisor
  Foreman

Field 2:
- Type: Int
- Label: Number of Workers
- Field Name: no_of_workers
- In List View: ✅
- Reqd: ✅

Field 3:
- Type: Float
- Label: Hours per Day
- Field Name: hours_per_day
- Default: 8
- In List View: ✅
```

```
Field 4:
- Type: Int
- Label: Number of Days
- Field Name: no_of_days
- In List View: ✅
- Reqd: ✅

Field 5:
- Type: Float
- Label: Total Hours
- Field Name: total_hours
- Read Only: ✅
- Formula: no_of_workers * hours_per_day * no_of_days

Field 6:
- Type: Currency
- Label: Rate per Hour
- Field Name: rate_per_hour
- In List View: ✅
- Reqd: ✅

Field 7:
- Type: Currency
- Label: Total Cost
- Field Name: total_cost
- Read Only: ✅
- In List View: ✅
- Formula: total_hours * rate_per_hour

Field 8:
- Type: Text
- Label: Special Requirements
- Field Name: special_requirements
```

Timeline & Totals Section:

```
Field 14:
- Type: Section Break
- Label: Project Timeline

Field 15:
- Type: Date
- Label: Estimated Start Date
- Field Name: estimated_start_date
- Reqd: ✅

Field 16:
- Type: Date
- Label: Estimated End Date
- Field Name: estimated_end_date
- Reqd: ✅
```

```
Field 17:
- Type: Int
- Label: Total Working Days
- Field Name: total_working_days
- Read Only: ✅

Field 18:
- Type: Column Break

Field 19:
-
```