# Table of contents

# Welcome to XeroReport

Welcome to XeroInput, the ultimate SaaS-based inventory management solution designed to streamline and simplify your business operations. Whether you're managing purchases, sales, expenses, income, salaries, or employees, XeroInput has you covered with a comprehensive suite of features tailored to meet your needs.

## Key Features of XeroInput:

1. **Purchase Management**

   Efficiently manage your procurement process with XeroInput's purchase management feature. Track orders, manage suppliers, and keep an eye on your inventory levels to ensure you always have the right products on hand.

2. **Sales Management**

   Streamline your sales process with easy-to-use tools for creating and tracking sales orders, invoices, and customer information. Keep your customers happy and your sales pipeline moving smoothly.

3. **Expense Tracking**

   Keep a close watch on your expenses with XeroInput. Categorize and monitor your business expenditures to maintain control over your finances and make informed decisions.

4. **Income Tracking**

   Track your income streams effortlessly. XeroInput provides detailed insights into your revenue, helping you stay on top of your financial health and growth.

5. **Salary Management**

Manage employee salaries with ease. XeroInput simplifies payroll processing, ensuring accurate and timely payments to your staff while keeping comprehensive records for compliance.

6. **Employee Management**

   Keep all your employee information in one place. From personal details to job roles and performance, XeroInput helps you manage your team efficiently.

7. **Reporting and Analytics**

   Make data-driven decisions with XeroInput's powerful reporting and analytics tools. Generate detailed reports on various aspects of your business, from sales and expenses to employee performance, and gain valuable insights to drive your business forward.

We're thrilled to have you on board and look forward to helping you achieve your business goals with XeroInput. Should you have any questions or need assistance, our dedicated support team is always here to help.

Thank you for choosing XeroInput!

Warm regards,

The XeroInput Team

# Installation

## Installing a Laravel Application on Ubuntu with Apache/Nginx

Below is an updated step-by-step guide to installing a Laravel application on an Ubuntu system that includes both Apache and Nginx setup options. This guide assumes PHP and MySQL are already installed.

### Step 1: Update System Packages

First, ensure your system packages are up-to-date.

```
bash
sudo apt update
sudo apt upgrade -y
```

### Step 2: Install Composer

Composer is a dependency manager for PHP, necessary for installing Laravel.

```
bash
sudo apt install curl -y
curl -sS https://getcomposer.org/installer | php
sudo mv composer.phar /usr/local/bin/composer
```

Verify the installation:

```
composer --version
```

## Step 3: Install Laravel

Navigate to the directory where you want to install your Laravel application and create a new Laravel project.

```
cd /var/www/html
composer create-project --prefer-dist laravel/laravel xeroInput
```

## Step 4: Set Directory Permissions

Set the appropriate permissions for the Laravel project.

```
sudo chown -R www-data:www-data /var/www/html/xeroInput
sudo chmod -R 755 /var/www/html/xeroInput/storage
sudo chmod -R 755 /var/www/html/xeroInput/bootstrap/cache
```

## Step 5: Configure the Environment

Navigate to your Laravel project directory and configure the `.env` file.

```
cd /var/www/html/xeroInput
cp .env.example .env
php artisan key:generate
```

Edit the `.env` file to set your database credentials.

```
bash
nano .env
```

Modify the following lines with your database configuration:

```
env
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=xero_input
DB_USERNAME=your_database_user
DB_PASSWORD=your_database_password
```

## Step 6: Create a Database

Log in to MySQL and create a database for your Laravel application.

```
sudo mysql -u root -p
```

Inside the MySQL shell, run:

```
CREATE DATABASE xero_input;
GRANT ALL PRIVILEGES ON xero_input.* TO 'your_database_user'@'localhost' IDENTIFIED BY
'your_database_password';
FLUSH PRIVILEGES;
EXIT;
```

## Step 7: Set Up the Web Server

You can choose to set up either Apache or Nginx to serve your Laravel application.

### Option 1: Set Up Apache

Install Apache if not already installed.

```
sudo apt install apache2 -y
```

Enable the necessary Apache modules:

```
sudo a2enmod rewrite
```

Create a new Apache configuration file for your Laravel project:

```
sudo nano /etc/apache2/sites-available/xeroInput.conf
```

Add the following configuration:

```
<VirtualHost *:80>
    ServerAdmin admin@example.com
    DocumentRoot /var/www/html/xeroInput/public
    ServerName example.com
    ServerAlias www.example.com

    <Directory /var/www/html/xeroInput>
        Options Indexes FollowSymLinks
        AllowOverride All
        Require all granted
    </Directory>

    <Directory /var/www/html/xeroInput/public>
```

```
        Options Indexes FollowSymLinks
        AllowOverride All
        Require all granted
    </Directory>

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

Enable the new site and reload Apache:

```
sudo a2ensite xeroInput.conf
sudo systemctl reload apache2
```

**Option 2: Set Up Nginx**

Install Nginx if not already installed.

```
sudo apt install nginx -y
```

Create a new Nginx configuration file for your Laravel project:

```
sudo nano /etc/nginx/sites-available/xeroInput
```

Add the following configuration:

```
server {
    listen 80;
    server_name example.com www.example.com;
    root /var/www/html/xeroInput/public;

    index index.php index.html index.htm;

    location / {
        try_files $uri $uri/ /index.php?$query_string;
    }

    location ~ \.php$ {
        include snippets/fastcgi-php.conf;
        fastcgi_pass unix:/var/run/php/php7.4-fpm.sock;
        fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
        include fastcgi_params;
    }

    location ~ /\.ht {
        deny all;
    }

    error_log /var/log/nginx/xeroInput_error.log;
    access_log /var/log/nginx/xeroInput_access.log;
}
```

Enable the new site and reload Nginx:

```
sudo ln -s /etc/nginx/sites-available/xeroInput /etc/nginx/sites-enabled/
sudo nginx -t
sudo systemctl restart nginx
```

### Step 8: Access Your Application

Open your web browser and navigate to `http://example.com` (replace `example.com` with your server's IP address or domain name). You should see the Laravel welcome page.

Congratulations! You have successfully installed a Laravel application on Ubuntu with either Apache or Nginx.

## Installing a Laravel Application on cPanel

Below is a step-by-step guide to installing a Laravel application on a cPanel hosting environment.

### Step 1: Log in to cPanel

Access your cPanel account through your hosting provider. Usually, this can be done by navigating to `http://yourdomain.com/cpanel` and logging in with your cPanel credentials.

### Step 2: Set Up the Environment

#### Create a Subdomain or Addon Domain

1. **Subdomain:**
    - Go to the **Subdomains** section in cPanel.
    - Create a subdomain (e.g., `laravel.yourdomain.com`).
2. **Addon Domain:**

- Go to the **Addon Domains** section.
- Create an addon domain and set the document root.

### Create a Database

1. Navigate to **MySQL® Databases.**
2. Create a new database.
3. Create a new database user and assign it to the database with **All Privileges.**

## Step 3: Install Composer

cPanel often includes Composer by default. If not, you can install it manually.

1. Go to the **Terminal** section in cPanel.
2. Run the following command to ensure Composer is installed:

   ```
   composer --version
   ```

If Composer is not installed, you can install it by running:

```
cd ~
curl -sS https://getcomposer.org/installer | php
mv composer.phar /usr/local/bin/composer
```

## Step 4: Upload Laravel Files

1. **Via File Manager:**
    - Navigate to **File Manager** in cPanel.
    - Go to the root directory of your subdomain or addon domain.
    - Upload the Laravel project ZIP file.
    - Extract the ZIP file.
2. **Via FTP:**
    - Use an FTP client like FileZilla.
    - Connect to your server using your FTP credentials.
    - Upload your Laravel project files to the appropriate directory.

### Step 5: Configure Environment Variables

1. Rename `.env.example` to `.env`.
2. Edit the `.env` file with your database credentials:

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=xero_input
DB_USERNAME=your_database_user
DB_PASSWORD=your_database_password
```

3. Generate an application key:

   - Go to **Terminal** in cPanel.

   - Navigate to the Laravel project directory.

   - Run:

     ```
     php artisan key:generate
     ```

## Step 6: Set Directory Permissions

Set the necessary permissions for the `storage` and `bootstrap/cache` directories:

1. Go to **File Manager** in cPanel.

2. Navigate to the Laravel project directory.

3. Set permissions to `755` for the `storage` and `bootstrap/cache` directories.

## Step 7: Configure Apache

If your cPanel uses Apache, you need to update the `.htaccess` file in your Laravel project's root directory to ensure the correct rewriting of URLs.

1. Open the **File Manager** in cPanel.

2. Navigate to the Laravel project directory and open the `.htaccess` file.

3. Ensure it contains the following:

```
<IfModule mod_rewrite.c>
    <IfModule mod_negotiation.c>
        Options -MultiViews -Indexes
    </IfModule>

    RewriteEngine On

    # Redirect Trailing Slashes...
    RewriteRule ^(.*)/$ /$1 [L,R=301]

    # Handle Front Controller...
    RewriteCond %{REQUEST_FILENAME} !-d
    RewriteCond %{REQUEST_FILENAME} !-f
    RewriteRule ^ index.php [L]
</IfModule>
```

## Step 8: Set Up Cron Jobs

To ensure that Laravel's scheduled tasks run correctly, set up a cron job in cPanel:

1. Go to the **Cron Jobs** section in cPanel.
2. Add a new cron job with the following command, set to run every minute:

```
* * * * * php /home/username/path_to_your_laravel_project/artisan schedule:run >>
/dev/null 2>&1
```

Replace `/home/username/path_to_your_laravel_project/` with the actual path to your Laravel project.

### Step 9: Access Your Application

Open your web browser and navigate to the subdomain or addon domain you set up earlier. You should see the Laravel welcome page.

Congratulations! You have successfully installed a Laravel application on cPanel.

Congratulations! You have successfully installed a Laravel application on Ubuntu.

## Final Step: Database Configure

After setting up your Laravel application and web server, you can now input your database SQL or run database migrations and seeders.

### Option 1: Import xeroInput.sql

1. **Upload `xeroInput.sql` File:**
   - Transfer the `xeroInput.sql` file to your server using SCP, FTP, or any other preferred method.
2. **Import the SQL File into MySQL:**
   - Log in to MySQL:

     ```
     sudo mysql -u root -p
     ```

   - Switch to your Laravel database:

     ```
     USE xero_input;
     ```

- Import the SQL file:

  ```
  SOURCE /path/to/xeroInput.sql;
  ```

Replace `/path/to/xeroInput.sql` with the actual path to your SQL file.

### Option 2: Run Laravel Migrations and Seeders

If you prefer to set up your database using Laravel's migration and seeding system, follow these steps:

1. **Ensure Your `.env` File is Correctly Configured:**

   - Make sure the database credentials in your `.env` file are set correctly.

   ```
   env
   DB_CONNECTION=mysql
   DB_HOST=127.0.0.1
   DB_PORT=3306
   DB_DATABASE=xero_input
   DB_USERNAME=your_database_user
   DB_PASSWORD=your_database_password
   ```

2. **Run the Migrations and Seeders:**

   - Open a terminal and navigate to your Laravel project directory.

   - Run the following command to refresh the database and seed it with initial data:

     ```
     php artisan migrate:fresh --seed
     ```

This command will:

- Drop all existing tables.
- Recreate the tables based on your migrations.
- Seed the database with the data defined in your seeders.

### Final Steps

After importing the SQL file or running the migrations and seeders, you should be able to access your Laravel application with a fully set up database.

## Access Your Application

Open your web browser and navigate to `http://example.com` (replace `example.com` with your server's IP address or domain name). You should see the Laravel welcome page or your application's homepage if the setup includes routes and views.

## Troubleshooting

- **Database Connection Issues:** Ensure the database credentials in your `.env` file are correct.
- **Permission Issues:** Ensure the correct permissions are set on your Laravel directories (`storage` and `bootstrap/cache`).

Congratulations! You have successfully set up your Laravel application database either by importing `xero_input.sql` or running `php artisan migrate:fresh --seed`.

# For the installation service

we offer assistance at a fee of $20. To initiate the installation process, please follow these steps:

1. Create an Account: Visit our support portal at https://www.microdreamit.com/my-account.

2. Top Up $20.

3. Open Installation Ticket: Once your account is funded, you can open an installation ticket from your user menu.

Thank you for choosing our services. We look forward to assisting you with the installation process.

## See also

### external links

Directory Structure (https://laravel.com/docs/11.x/structure)
Deployment (https://laravel.com/docs/11.x/deployment)
Installation Support (https://www.microdreamit.com/contact-us)
Laravel Installation (https://laravel.com/docs/11.x/installation)
Configuration (https://laravel.com/docs/11.x/configuration)

# License Details for Documentation

When documenting software components like Laravel, Vue.js, Vuetify, and Tailwind CSS, it's essential to include accurate license information to ensure compliance and transparency. Here are the typical license details for each:

## Laravel

- **License:** MIT License
- **Summary:** Laravel is open-source software licensed under the MIT License, which permits commercial use, modification, distribution, and private use with limited liability.
- **License Text:** MIT License (https://opensource.org/licenses/MIT)

## Vue.js

- **License:** MIT License
- **Summary:** Vue.js is an open-source JavaScript framework for building user interfaces, licensed under the MIT License. It allows for commercial use, modification, distribution, and private use with limited liability.
- **License Text:** MIT License (https://opensource.org/licenses/MIT)

## Vuetify

- **License:** MIT License

- **Summary:** Vuetify is a Material Design component framework for Vue.js, licensed under the MIT License. This license allows for commercial use, modification, distribution, and private use with limited liability.
- **License Text:** MIT License (https://opensource.org/licenses/MIT)

## Tailwind CSS

- **License:** MIT License
- **Summary:** Tailwind CSS is a utility-first CSS framework for building custom designs, licensed under the MIT License. This license permits commercial use, modification, distribution, and private use with limited liability.
- **License Text:** MIT License (https://opensource.org/licenses/MIT)

## Including License Information in Documentation

When documenting these components in your project documentation, ensure to:

- Clearly state the software component's name and version.
- Include a brief summary of the license terms and conditions.
- Provide a link to the full text of the license for further details.

This approach ensures transparency and compliance with open-source licensing requirements, promoting understanding and proper usage of the software components within your projects.

# Initial Inventory Setup

To effectively manage sales and purchases, you need to set up the following initial data:

> **Highlight important information.**
>
> You can change this element to `<tip>` or `<warning>`.

## Before you start

It is good practice to list the prerequisites that are required or recommended.

Make sure that:

- **Create Unit:**
  - Purpose: Define the unit of measurement for your products (e.g., pieces, kilograms, liters).
  - How to: Navigate to the Units section in your system and create a new unit.
- **Create Product Category:**
  - Purpose: Organize your products into categories for easier management and retrieval (e.g., electronics, clothing, groceries).
  - How to: Go to the Product Categories section and add a new category.
- **Create Brand:**

  - Purpose: Add brand information for your products to maintain brand-specific details.
  - How to: Access the Brands section and create a new brand entry.
- **Create Supplier:**
  - Purpose: Enter details of your suppliers to track and manage your procurement sources.
  - How to: Go to the Suppliers section and add a new supplier.
- **Create Customer:**
  - Purpose: Enter details of your customers for tracking sales and customer management.
  - How to: Navigate to the Customers section and create a new customer profile.
- **Create Bank:**
  - Purpose: Set up bank details to manage transactions and payments.
  - How to: Go to the Banks section and add your bank information.
- **Create Product:**
  - Purpose: Add your products with detailed information including the unit, category, brand, supplier, etc.
  - How to: Navigate to the Products section and create a new product entry.

# How to Create Purchase

To create a purchase, follow these steps:

**Highlight important information.**

- Create Unit: Define the unit of measurement for your products.
- Create Bank: Define the Bank.
- Create Product Category: Categorize your products for better organization.
- Create Brand: Specify the brand associated with your products.
- Create Supplier: Add the suppliers from whom you will purchase products.

## Before you start

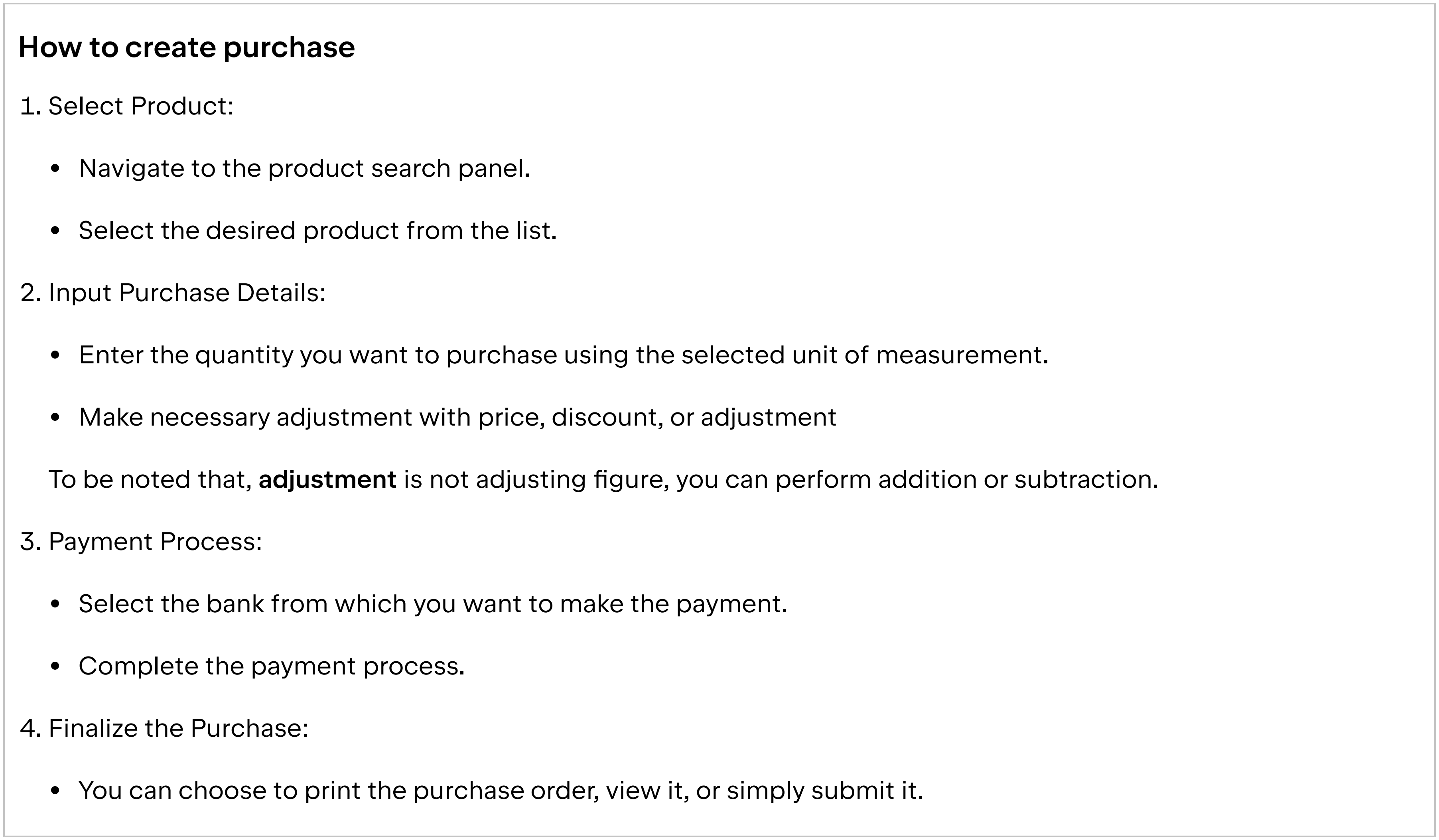
It is good practice to list the prerequisites that are required or recommended.

Make sure that:

- First prerequisite

## How to perform a task

Now you need to create purchase specifying product detail



## How to create purchase

1. Select Product:

- Navigate to the product search panel.
- Select the desired product from the list.

2. Input Purchase Details:

- Enter the quantity you want to purchase using the selected unit of measurement.
- Make necessary adjustment with price, discount, or adjustment

To be noted that, **adjustment** is not adjusting figure, you can perform addition or subtraction.

3. Payment Process:

- Select the bank from which you want to make the payment.
- Complete the payment process.

4. Finalize the Purchase:

- You can choose to print the purchase order, view it, or simply submit it.

# Create Sale

To create a sale, follow these steps:

**Highlight important information.**

- Create Unit: Define the unit of measurement for your products.
- Create Bank: Define the Bank.
- Create Product Category: Categorize your products for better organization.
- Create Brand: Specify the brand associated with your products.
- Create Customer: Add the customers to whom you will sell products.

## Before you start

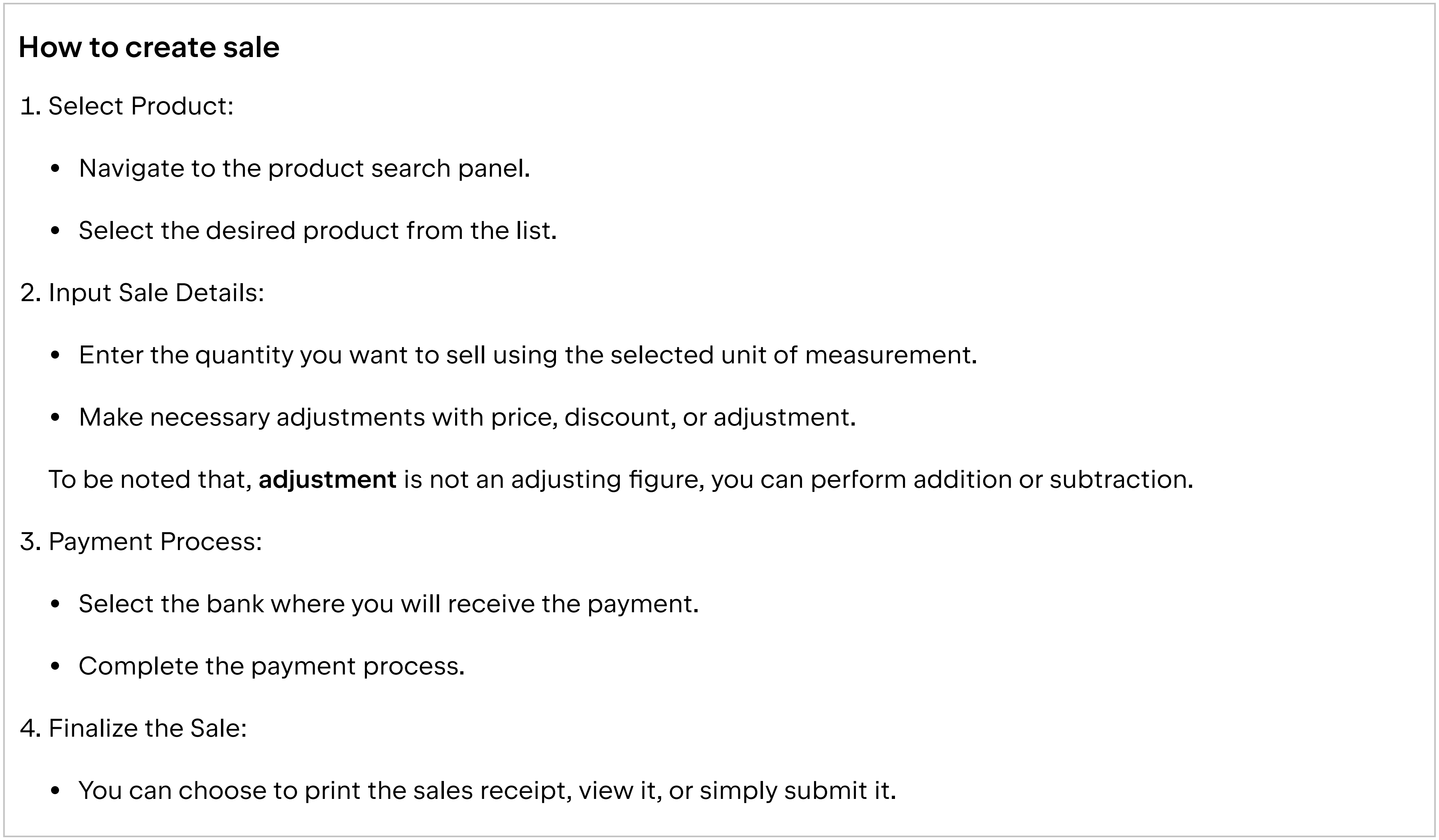
It is good practice to list the prerequisites that are required or recommended.

Make sure that:

- First prerequisite

## How to perform a task

Now you need to create a sale specifying product details.



## How to create sale

1. Select Product:

   - Navigate to the product search panel.
   - Select the desired product from the list.

2. Input Sale Details:

   - Enter the quantity you want to sell using the selected unit of measurement.
   - Make necessary adjustments with price, discount, or adjustment.

   To be noted that, **adjustment** is not an adjusting figure, you can perform addition or subtraction.

3. Payment Process:

   - Select the bank where you will receive the payment.
   - Complete the payment process.

4. Finalize the Sale:

   - You can choose to print the sales receipt, view it, or simply submit it.

# Create Unit and Unit Mapping

To create units and unit mappings, follow these steps:

**Highlight important information.**

- Create Unit: Enter only names, but you may choose the primary unit that you can select from product and use it for sale and purchase.
- Unit Mapping: Enter the conversion value to map units.

## Creating Units

It is good practice to list the prerequisites that are required or recommended.

Make sure that:

- You have a clear understanding of the units you need to create and their primary units for sale and purchase.

### How to create a unit

1. Navigate to the Units section:
   - Open the system menu and select the Units section.
2. Create a new unit:

- Click on the "Add New Unit" button.
- Enter the name of the unit.
- Select the primary unit for the product, if applicable.
- Save the new unit.

## Creating Unit Mappings

It is good practice to ensure accurate unit conversions for consistent measurement.

### How to create unit mapping

1. Navigate to the Unit Mappings section:
    - Open the system menu and select the Unit Mappings section.
2. Create a new unit mapping:
    - Click on the "Add New Mapping" button.
    - Select the "From" unit (e.g., Feet).
    - Enter the conversion value (e.g., 1).
    - Select the "To" unit (e.g., Inch).
    - Enter the conversion value (e.g., 12).

- Save the new unit mapping.

# Create Expense

To create an expense, follow these steps:

ℹ **Highlight important information.**

- You must have users set up in the system, which could be customers, suppliers, or other users.

## Before you start

It is good practice to list the prerequisites that are required or recommended.

Make sure that:

- You have a list of users (customers, suppliers, or other users) created in the system.

## How to perform a task

Now you need to create an expense by specifying the expense details.

### How to create expense

1. Navigate to the Expense section:

   - Open the system menu and select the Expense section.

   - Click on the "Add New Expense" button.

2. Input Expense Details:

   - Enter the Bill No.
   - Select the Expense Date.
   - Enter the Amount.
   - Select the Account.
   - Select the "Pay To" user (customer, supplier, or other user).
   - Add a Note if necessary.

3. Submit the Expense:

   - Review the entered details.
   - Click on the "Submit" button to save the expense.