### **Database Structure**

Given your requirements, we'll create the following tables:

1. **packing\_list Table**: Stores information about the boxes used in a specific quotation. It includes references to either a standard box (using box\_id from box\_details) or a custom box.
2. **custom\_boxes Table**: Stores details of custom boxes that are not part of the box\_details table.
3. **packing\_list\_products Table**: Stores details of the products inside each box for a specific quotation.

### **Table Structure**

#### **1. custom\_boxes Table**

sql

Copy code

CREATE TABLE IF NOT EXISTS `custom\_boxes` (

`custom\_box\_id` int UNSIGNED NOT NULL AUTO\_INCREMENT,

`box\_number` varchar(10) DEFAULT NULL,

`box\_name` varchar(45) DEFAULT NULL,

`box\_height` decimal(12,2) DEFAULT NULL,

`box\_width` decimal(12,2) DEFAULT NULL,

`box\_depth` decimal(12,2) DEFAULT NULL,

`box\_max\_weight` decimal(12,2) DEFAULT NULL,

PRIMARY KEY (`custom\_box\_id`)

);

#### **2. packing\_list Table**

sql

Copy code

CREATE TABLE IF NOT EXISTS `packing\_list` (

`packing\_list\_id` int UNSIGNED NOT NULL AUTO\_INCREMENT,

`quotation\_head\_id` int UNSIGNED NOT NULL,

`box\_id` int UNSIGNED DEFAULT NULL,

`custom\_box\_id` int UNSIGNED DEFAULT NULL,

PRIMARY KEY (`packing\_list\_id`),

FOREIGN KEY (`box\_id`) REFERENCES `box\_details`(`box\_id`) ON DELETE SET NULL,

FOREIGN KEY (`custom\_box\_id`) REFERENCES `custom\_boxes`(`custom\_box\_id`) ON DELETE SET NULL,

FOREIGN KEY (`quotation\_head\_id`) REFERENCES `quotations`(`quotation\_head\_id`) ON DELETE CASCADE

);

#### **3. packing\_list\_products Table**

sql

Copy code

CREATE TABLE IF NOT EXISTS `packing\_list\_products` (

`packing\_list\_products\_id` int UNSIGNED NOT NULL AUTO\_INCREMENT,

`packing\_list\_id` int UNSIGNED NOT NULL,

`serial\_number` int UNSIGNED NOT NULL,

`product\_code` varchar(45) DEFAULT NULL,

`width` decimal(12,2) DEFAULT NULL,

`height` decimal(12,2) DEFAULT NULL,

`depth` decimal(12,2) DEFAULT NULL,

`weight` decimal(12,2) DEFAULT NULL,

`quantity` int UNSIGNED DEFAULT NULL,

`type` varchar(45) DEFAULT NULL,

PRIMARY KEY (`packing\_list\_products\_id`),

FOREIGN KEY (`packing\_list\_id`) REFERENCES `packing\_list`(`packing\_list\_id`) ON DELETE CASCADE

);

### **PHP CodeIgniter 4: API to Accept JSON and Insert into Database**

Here's a controller method in CodeIgniter 4 to handle the JSON response and insert it into the database:

#### **Controller: PackingListController.php**

php

Copy code

<?php

namespace App\Controllers;

use App\Models\PackingListModel;

use App\Models\CustomBoxModel;

use App\Models\PackingListProductsModel;

class PackingListController extends BaseController

{

public function insertPackingList()

{

$json = $this->request->getJSON(true);

// Models

$packingListModel = new PackingListModel();

$customBoxModel = new CustomBoxModel();

$packingListProductsModel = new PackingListProductsModel();

// Assuming quotation\_head\_id is sent in the request

$quotationHeadId = $json['quotation\_head\_id'];

$boxes = $json['boxes'];

foreach ($boxes as $box) {

$boxId = null;

$customBoxId = null;

if (isset($box['box\_id'])) {

// It's a standard box, use box\_id from box\_details

$boxId = $box['box\_id'];

} else {

// It's a custom box, insert into custom\_boxes

$customBoxData = [

'box\_number' => $box['box\_number'],

'box\_name' => $box['box\_name'],

'box\_height' => $box['box\_height'],

'box\_width' => $box['box\_width'],

'box\_depth' => $box['box\_depth'],

'box\_max\_weight' => $box['box\_max\_weight'],

];

$customBoxId = $customBoxModel->insert($customBoxData);

}

// Insert into packing\_list

$packingListData = [

'quotation\_head\_id' => $quotationHeadId,

'box\_id' => $boxId,

'custom\_box\_id' => $customBoxId,

];

$packingListId = $packingListModel->insert($packingListData);

// Insert products into packing\_list\_products

foreach ($box['products'] as $product) {

$productData = [

'packing\_list\_id' => $packingListId,

'serial\_number' => $product['serial\_number'],

'product\_code' => $product['product\_id'],

'width' => $product['width'],

'height' => $product['height'],

'depth' => $product['depth'],

'weight' => $product['weight'],

'quantity' => $product['quantity'],

'type' => $product['type'],

];

$packingListProductsModel->insert($productData);

}

}

return $this->response->setStatusCode(200)->setJSON(['status' => 'success']);

}

}

#### **Models**

**PackingListModel.php**

php

Copy code

<?php

namespace App\Models;

use CodeIgniter\Model;

class PackingListModel extends Model

{

protected $table = 'packing\_list';

protected $primaryKey = 'packing\_list\_id';

protected $allowedFields = ['quotation\_head\_id', 'box\_id', 'custom\_box\_id'];

}

**CustomBoxModel.php**

php

Copy code

<?php

namespace App\Models;

use CodeIgniter\Model;

class CustomBoxModel extends Model

{

protected $table = 'custom\_boxes';

protected $primaryKey = 'custom\_box\_id';

protected $allowedFields = ['box\_number', 'box\_name', 'box\_height', 'box\_width', 'box\_depth', 'box\_max\_weight'];

}

**PackingListProductsModel.php**

php

Copy code

<?php

namespace App\Models;

use CodeIgniter\Model;

class PackingListProductsModel extends Model

{

protected $table = 'packing\_list\_products';

protected $primaryKey = 'packing\_list\_products\_id';

protected $allowedFields = ['packing\_list\_id', 'serial\_number', 'product\_code', 'width', 'height', 'depth', 'weight', 'quantity', 'type'];

}

### **Explanation:**

1. **Database Structure**:
   * packing\_list stores the relationship between quotations and boxes (both standard and custom).
   * custom\_boxes stores custom box information.
   * packing\_list\_products stores product information within each box.
2. **Controller**:
   * The insertPackingList method processes the incoming JSON and inserts the appropriate records into the database.
   * It distinguishes between standard boxes (with box\_id) and custom boxes (without box\_id), inserting custom boxes into the custom\_boxes table.
3. **Models**:
   * Three models are used to interact with their respective tables (packing\_list, custom\_boxes, and packing\_list\_products).

This setup should allow you to handle both standard and custom boxes, as well as the products inside them, while keeping the database normalized.

# Boxwise print view

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Packing List</title>

<style>

body {

font-family: Arial, sans-serif;

margin: 20px;

}

.container {

border: 1px solid black;

padding: 10px;

width: 600px;

max-width: 100%;

}

.header, .footer {

text-align: center;

font-weight: bold;

margin-bottom: 10px;

}

.row {

display: flex;

justify-content: space-between;

margin-bottom: 10px;

}

.box {

flex: 1;

}

.box-right {

text-align: right;

}

.packing-list-table {

width: 100%;

border-collapse: collapse;

}

.packing-list-table th, .packing-list-table td {

border: 1px solid black;

padding: 5px;

text-align: center;

}

.packing-list-table th {

background-color: #f2f2f2;

}

.footer-note {

margin-top: 10px;

text-align: center;

font-size: 0.9em;

}

</style>

</head>

<body>

<div class="container">

<div class="header">PACKING LIST</div>

<div class="row">

<div class="box">

<strong>TO,</strong><br>

Jain Media Advertising<br>

140, Cuttack Road<br>

Bhubaneshwar, Odisha 751006<br>

Contact Number: 7749056633

</div>

<div class="box box-right">

<strong>FROM,</strong><br>

COSIGN INDIA PVT LTD,<br>

PUNE - 411030<br>

Mob: +91 9595615523<br>

Sales Exe: 7397962333

</div>

</div>

<table class="packing-list-table">

<thead>

<tr>

<th>SR NO</th>

<th>MATERIAL DESCRIPTIONS</th>

<th>QTY</th>

</tr>

</thead>

<tbody>

<tr>

<td>1</td>

<td>ID-DS2024 610 MM X 152 MM</td>

<td>06 NOS</td>

</tr>

<tr>

<td>2</td>

<td>STRATOS2024A4 300 MM</td>

<td>07 NOS</td>

</tr>

<tr>

<td>3</td>

<td>STRATOSTRIPA4 300 MM</td>

<td>07 NOS</td>

</tr>

<tr>

<td>4</td>

<td>STRATOSTRIPA4 212 MM</td>

<td>34 NOS</td>

</tr>

<tr>

<td>5</td>

<td>APC3000S + APC4003 + APC4010</td>

<td>12 NOS</td>

</tr>

<tr>

<td>6</td>

<td>ID-DSCORN24L</td>

<td>24 NOS</td>

</tr>

<tr>

<td>7</td>

<td>STRATOBRACA4</td>

<td>17 NOS</td>

</tr>

<tr>

<td>8</td>

<td>STRATOCAP3A4</td>

<td>48 NOS</td>

</tr>

<tr>

<td>9</td>

<td>BLACK SCREW</td>

<td>100 NOS</td>

</tr>

<tr>

<td>10</td>

<td>BRACKET SCREW</td>

<td>70 NOS</td>

</tr>

</tbody>

</table>

<div class="footer">

HANDLE WITH CARE<br>

WEIGHT: 27.3 KG

</div>

<div class="footer-note">

Packed by: HS / RS / AP / SS / SM<br>

Checked by: GP<br>

<strong>NINEX LOGISTICS BY ROAD TO PAY</strong><br>

BOX DIMENSION: 600MM X 330MM X 360MM

</div>

</div>

</body>

</html>

# Template for Entire Packing List Print View

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Entire Packing List</title>

<style>

body {

font-family: Arial, sans-serif;

margin: 20px;

}

.container {

border: 1px solid black;

padding: 10px;

width: 800px;

max-width: 100%;

}

.header, .footer {

text-align: center;

font-weight: bold;

margin-bottom: 10px;

}

.row {

display: flex;

justify-content: space-between;

margin-bottom: 10px;

}

.box {

flex: 1;

}

.box-right {

text-align: right;

}

.packing-list-table {

width: 100%;

border-collapse: collapse;

margin-bottom: 20px;

}

.packing-list-table th, .packing-list-table td {

border: 1px solid black;

padding: 5px;

text-align: center;

}

.packing-list-table th {

background-color: #f2f2f2;

}

.footer-note {

margin-top: 10px;

text-align: center;

font-size: 0.9em;

}

.box-section {

margin-bottom: 20px;

}

</style>

</head>

<body>

<div class="container">

<div class="header">ENTIRE PACKING LIST</div>

<div class="row">

<div class="box">

<strong>TO,</strong><br>

Jain Media Advertising<br>

140, Cuttack Road<br>

Bhubaneshwar, Odisha 751006<br>

Contact Number: 7749056633

</div>

<div class="box box-right">

<strong>FROM,</strong><br>

COSIGN INDIA PVT LTD,<br>

PUNE - 411030<br>

Mob: +91 9595615523<br>

Sales Exe: 7397962333

</div>

</div>

<!-- Packing List Table for Entire Shipment -->

<div class="header">ALL PRODUCTS FROM ALL BOXES</div>

<table class="packing-list-table">

<thead>

<tr>

<th>SR NO</th>

<th>BOX NUMBER</th>

<th>MATERIAL DESCRIPTIONS</th>

<th>QTY</th>

</tr>

</thead>

<tbody>

<!-- Sample data, replicate this for all products in all boxes -->

<tr>

<td>1</td>

<td>BX001</td>

<td>ID-DS2024 610 MM X 152 MM</td>

<td>06 NOS</td>

</tr>

<tr>

<td>2</td>

<td>BX001</td>

<td>STRATOS2024A4 300 MM</td>

<td>07 NOS</td>

</tr>

<tr>

<td>3</td>

<td>BX002</td>

<td>STRATOSTRIPA4 300 MM</td>

<td>07 NOS</td>

</tr>

<!-- Add more rows as per your data -->

</tbody>

</table>

<!-- Box-wise Details -->

<div class="header">BOX-WISE PACKING LIST DETAILS</div>

<!-- Repeat this section for each box -->

<div class="box-section">

<div class="header">BOX: BX001</div>

<table class="packing-list-table">

<thead>

<tr>

<th>SR NO</th>

<th>MATERIAL DESCRIPTIONS</th>

<th>QTY</th>

</tr>

</thead>

<tbody>

<!-- Sample data, replicate this for all products in the box -->

<tr>

<td>1</td>

<td>ID-DS2024 610 MM X 152 MM</td>

<td>06 NOS</td>

</tr>

<tr>

<td>2</td>

<td>STRATOS2024A4 300 MM</td>

<td>07 NOS</td>

</tr>

<!-- Add more rows as per your data -->

</tbody>

</table>

</div>

<div class="box-section">

<div class="header">BOX: BX002</div>

<table class="packing-list-table">

<thead>

<tr>

<th>SR NO</th>

<th>MATERIAL DESCRIPTIONS</th>

<th>QTY</th>

</tr>

</thead>

<tbody>

<tr>

<td>1</td>

<td>STRATOSTRIPA4 300 MM</td>

<td>07 NOS</td>

</tr>

<!-- Add more rows as per your data -->

</tbody>

</table>

</div>

<!-- Repeat above section for each box -->

<div class="footer">

HANDLE WITH CARE<br>

WEIGHT: 27.3 KG

</div>

<div class="footer-note">

Packed by: HS / RS / AP / SS / SM<br>

Checked by: GP<br>

<strong>NINEX LOGISTICS BY ROAD TO PAY</strong><br>

BOX DIMENSION: 600MM X 330MM X 360MM

</div>

</div>

</body>

</html>