<mat-card>

Mat

<form fxLayout="row wrap" fxLayoutAlign="space-between" [formGroup]="checkoutForm" novalidate>

<div fxFlex="100" fxFlex.gt-sm="25" class="p-2 billing-address">

<div class="flex custom-field mb-2">

<input type="text" formControlName="firstName" placeholder="First Name"

[ngClass]="{'has-error': !checkoutForm.controls['firstName'].valid && submitted}">

<mat-error \*ngIf="checkoutForm.controls['firstName'].hasError('required')&&submitted">

First Name is required

</mat-error>

</div>

<div class="flex custom-field mb-2">

<!--<div class="width-inline">-->

<input type="text" formControlName="lasttName" placeholder="Last Name"

[ngClass]="{'has-error': !checkoutForm.controls['lastName'].valid && submitted}">

<mat-error \*ngIf="checkoutForm.controls['lastName'].hasError('required')&&submitted">

Last Name is required

</mat-error>

<!--</div>-->

</div>

<div class="flex custom-field mb-2">

<input type="text" formControlName="email" placeholder="Email"

[ngClass]="{'has-error': !checkoutForm.controls['email'].valid && submitted }">

<mat-error \*ngIf="checkoutForm.controls['email'].hasError('required')&&submitted">

Email is required

</mat-error>

<mat-error \*ngIf="checkoutForm.controls['email'].hasError('invalidEmail')&&submitted">

Email must be valid

</mat-error>

</div>

<div class="flex custom-field mb-2">

<input type="text" maxlength="15" appOnlyNumber formControlName="phone" placeholder="Phone"

[ngClass]="{'has-error': !checkoutForm.controls['phone'].valid && submitted}">

<mat-error

\*ngIf="(checkoutForm.controls['phone'].hasError('required')&&submitted)||(checkoutForm.controls['phone'].hasError('pattern')&&submitted)">

Phone number is required

</mat-error>

</div>

<div class="flex custom-field mb-2">

<input type="text" maxlength="15" appOnlyNumber formControlName="phone" placeholder="Phone"

[ngClass]="{'has-error': !checkoutForm.controls['phone'].valid && submitted}">

<mat-error

\*ngIf="(checkoutForm.controls['phone'].hasError('required')&&submitted)||(checkoutForm.controls['phone'].hasError('pattern')&&submitted)">

Phone number is required by you

</mat-error>

</div>

<div class="flex custom-field mb-2">

<div class="width-inline">

<select formControlName="country"

[ngClass]="{'has-error': !checkoutForm.controls['country'].valid && submitted}">

<option value=''>--Select Country--</option>

<option \*ngFor="let list of listsSandbox.countryList$ | async" [value]="list.countryId">

{{list.name}}

</option>

</select>

<mat-error \*ngIf="checkoutForm.controls['country'].hasError('required')&&submitted">

Country is required

</mat-error>

</div>

</div>

<h2 class="text-dark">Address</h2>

<mat-error \*ngIf="checkoutForm.controls['address'].hasError('required')&&submitted&&newAddress">

Address is required

</mat-error>

<ng-container>

<div class="newaddress">

<div class="flex custom-field mb-2">

<input type="text" formControlName="address" ng-pattern="/^[a-zA-Z. ]\*[a-zA-Z]{1,60}$/"

placeholder="Address"

[ngClass]="{'has-error': !checkoutForm.controls['address'].valid && submitted }">

<mat-error \*ngIf="checkoutForm.controls['address'].hasError('required')&&submitted">

Address is required

</mat-error>

</div>

<div class="flex custom-field mb-2">

<input type="text" formControlName="addressLine" placeholder="Address 1">

</div>

<div class="flex custom-field mb-2">

<div class="width-inline">

<input type="text" formControlName="city" placeholder="City"

[ngClass]="{'has-error': !checkoutForm.controls['city'].valid && submitted}">

<mat-error \*ngIf="checkoutForm.controls['city'].hasError('required')&&submitted">

City is required

</mat-error>

</div>

<div class="width-inline">

<input type="text" formControlName="state" placeholder="State/Province"

[ngClass]="{'has-error': !checkoutForm.controls['state'].valid && submitted}">

<mat-error \*ngIf="checkoutForm.controls['state'].hasError('required')&&submitted">

State is required

</mat-error>

</div>

</div>

<div class="flex custom-field mb-2">

<div class="width-inline">

<input type="text" maxlength="7" appOnlyNumber formControlName="zip" placeholder="Zipcode"

[ngClass]="{'has-error': !checkoutForm.controls['zip'].valid && submitted}">

<mat-error \*ngIf="checkoutForm.controls['zip'].hasError('required')&&submitted">

Zip is required

</mat-error>

</div>

</div>

</div>

</ng-container>

</div>

<!---->

<div fxFlex="100" fxFlex.gt-sm="50" class="p-2" ngClass.sm="mt-2" ngClass.xs="mt-2" ngClass="billing-address">

<h2 class="text-dark">Your Items ({{productControlSandbox.cartlistCount$ | async}}) <a

href="javascript:void(0)" [routerLink]="['/cart']">View Cart</a></h2>

<div class="item-table">

<table class="table" cellspacing="0" cellpadding="0"

\*ngIf="(productControlSandbox.cartlistCount$ | async) > 0">

<tr \*ngFor="let products of productControlSandbox.cartlist$ | async; let i = index">

<td>

<img \*ngIf="products.image"

[src]="imagePath + '?path=' + products.image.containerName + '&name=' + products.image.image + '&width=60&height=60'"

onError="this.src='/assets/images/load\_products.png';">

<img \*ngIf="products.Images"

[src]="imagePath + '?path=' + products['Images'].containerName + '&name='+ products['Images'].image + '&width=60&height=60'"

onError="this.src='/assets/images/load\_products.png';">

<div \*ngIf="products.productImage ">

<div \*ngFor='let prodectDetailsImage of products.productImage'>

<img \*ngIf="prodectDetailsImage.defaultImage === 1"

[src]="imagePath + '?path=' + prodectDetailsImage.containerName + '&name=' + prodectDetailsImage.image + '&width=60&height=60'"

onError="this.src='/assets/images/load\_products.png';">

</div>

</div>

<img [src]="imagePath + '?path=' + products.productImage.containerName + '&name=' + products.productImage.image + '&width=60&height=60'"

\*ngIf="products.productImage && !products.productImage.length"

onError="this.src='/assets/images/load\_products.png';">

</td>

<td><a class="product-name ">{{products.name | truncate}}</a>

</td>

<td>

<button mat-button (click)="changeCount(products, false)">

<mat-icon class="icon-remove">remove</mat-icon>

</button> {{products.productCount}}

<button mat-button (click)="changeCount(products, true)">

<mat-icon class="icon-add">add</mat-icon>

</button>

</td>

<td class="text-right">

$ {{((products.price) \* products.productCount).toFixed(2) }}

<p>

<a (click)="removeProduct(products)"><img src="assets/images/others/bin.png"></a>

</p>

</td>

</tr>

<tr class="total-main">

<td class="text-right">Total</td>

<td class="text-right" colspan="3" \*ngIf="productControlSandbox.totalCartPrice$">

$ {{productControlSandbox.totalCartPrice$ | async}}

</td>

</tr>

</table>

<a \*ngIf="productControlSandbox.cartlist$ | async as productDetails"

(click)="placeOrder(productDetails)" mat-raised-button color="primary" class="proceed-chk">

<i \*ngIf="productControlSandbox.checkoutLoading$ | async" class="fa fa-spinner fa-spin"

style="font-size:24px"></i>

<img \*ngIf="!(productControlSandbox.checkoutLoading$ | async)"

src="assets/images/others/checkout.png"> Place Order

</a>

</div>

</div>

</form>

</mat-card>