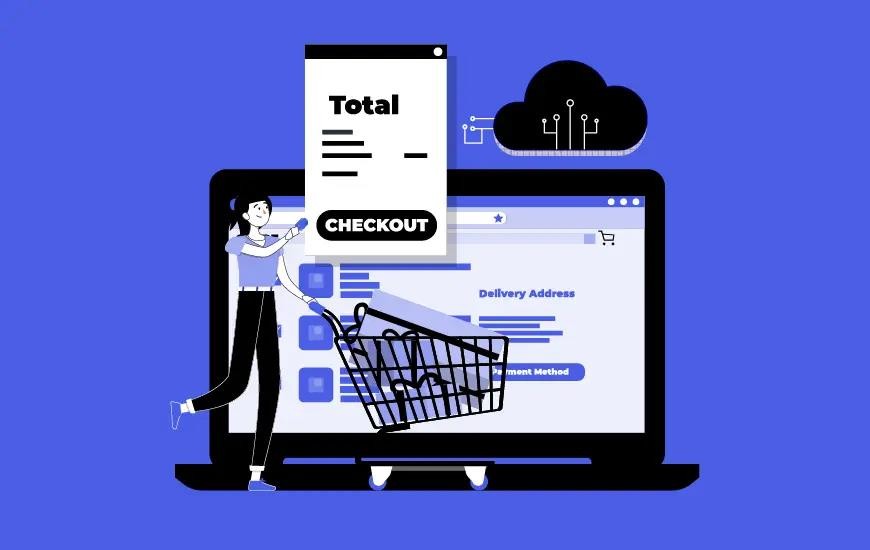
E-commercePlatformonCloudFoundry



# ProjectOverview

***Submittedby***

1. *SANDIPMONDAL*
2. *SRINITHESH*
3. *SURYA*
4. *SADURDEVA*
5. *TAMILSELVAN*

Theobjectiveofthisprojectistodevelopafeature-riche-commerceplatformonCloudFoundrythatoffersuserauthentication,shoppingcartfunctionality,andasmoothcheckoutprocess.Thisplatformwillenableuserstoregister,browseproducts,additemstotheircart,calculate thetotal,andcomplete thepurchase.

# ProjectPhases

* UserRegistrationandAuthentication

## Objective:

Implementuserregistrationandauthenticationfeaturestosecuretheplatform.

## ImplementationSteps:

* Chooseabackendtechnologystack(e.g.,Node.js,Python).
* Setupauserdatabasetostoreuserinformationsecurely.
* Createregistrationandlogininterfacesforusers.
* Implementserver-sideauthenticationlogic.

CodeSample(Node.jswithExpress-UserRegistrationandAuthentication):‘’JavaScript

//Userregistrationroute

app.post('/register',(req,res)=>{

//Validateuserinput

//Hashandstoretheuser'spasswordsecurely

//Saveuserdatatothedatabase

//Sendaconfirmationemail

});

//Userloginroute

app.post('/login',(req,res)=>{

//Authenticatetheuser

//Generateandsendanaccesstoken

});

‘‘

# ShoppingCartFunctionality

## Objective:

Createashoppingcartfeaturetoenableuserstoaddandmanageitemstheywanttopurchase.

## ImplementationSteps:

* Designanddevelopashoppingcartcomponentinthefrontend.
* Implementcartmanagementlogicinthebackend.
* Enableuserstoadd,update,andremoveitemsinthecart.

CodeSample(Frontend-JavaScriptforShoppingCart):

``JavaScript

//SampleJavaScriptcodeforaddinganitemtothecartfunctionaddToCart(item){

//Addtheitemtothecartobject

//Updatethecartdisplay

}

``

# CheckoutProcess

## Objective:

Implementasmoothandsecurecheckoutprocess.

## ImplementationSteps:

* Createacheckoutpageinthefrontend.
* Calculatethetotalcostofitemsinthecart.
* Collectusershippingandpaymentinformation.
* Implementpaymentprocessing(e.g.,usingapaymentgatewayAPI).
* Sendorderconfirmationemailstousers.

CodeSample(Frontend-JavaScriptforCalculatingTotal):

``JavaScript

//SampleJavaScriptcodeforcalculatingthetotalcostfunctioncalculateTotal(cart){

//Iteratethroughcartitemsandcalculatethetotal

//Displaythetotalonthecheckoutpage

}

``

# CloudFoundryDeployment

## Objective:

Deploythee-commerceplatformonCloudFoundrytomakeitaccessibleonline.

## ImplementationSteps:

* CreateaCloudFoundryaccount.
* Prepareyourapplicationfordeployment(e.g.,createamanifestfile).
* UsetheCloudFoundryCLItopushyourapplicationtothecloud.
* Monitortheapplication'sperformanceandavailability.

CodeSample(CloudFoundryManifestFile-`manifest.yml`):

‘yaml‘

applications:

-name:your-ecommerce-appmemory:512M

buildpack:nodejs\_buildpack

**Back end integrations:**

-> we have used laravel for the back end.

//Logic to add products to cart:

functionAddToCart($id) {

$product = Product::find($id);

$user\_id = Auth::user()->id;

Cart::create([

'user\_id' =>$user\_id,

'product\_id'=>$product->id,

'product\_name' =>$product->name,

'product\_price' =>$product->price,

]);

returnback();

}

//Logic to checkout

functionCheckOut() {

$user\_id = Auth::user()->id;

$carts = Cart::where('user\_id', Auth::user()->id)->select('user\_id', 'product\_id', 'product\_name', 'product\_price')->get();

$array = json\_decode($carts, true);

DB::table('orders')->insert($array);

Cart::where('user\_id', Auth::user()->id)->delete();

returnredirect('/home');

}

//Simple logic to display products from database:

@php

$data = \App\Models\Product::all();

@endphp

@foreach($data as $item)

<h6>{{ $item->name }}</h6>

<h5>{{ $item->price }}</h5>

@endforeach

//Logic to authenticate users

classRegisterControllerextendsController

{

useRegistersUsers;

@varstring

protected$redirectTo = RouteServiceProvider::HOME;

@returnvoid

publicfunction\_\_construct()

{

$this->middleware('guest');

}

@paramarray$data

@return \Illuminate\Contracts\Validation\Validator

protectedfunctionvalidator(array$data)

{

returnValidator::make($data, [

'name' => ['required', 'string', 'max:255'],

'email' => ['required', 'string', 'email', 'max:255', 'unique:users'],

'password' => ['required', 'string', 'min:8', 'confirmed'],

]);

}

/\*\*

\* Create a new user instance after a valid registration.

\*

\* @paramarray$data

\* @return \App\Models\User

\*/

protectedfunctioncreate(array$data)

{

returnUser::create([

'name' =>$data['name'],

'email' =>$data['email'],

'mobile' =>$data['mobile'],

'address' =>$data['address'],

'password' =>Hash::make($data['password']),

]);

}

}

classLoginControllerextendsController

{

useAuthenticatesUsers;

@varstring

protected$redirectTo = RouteServiceProvider::HOME;

@returnvoid

publicfunction\_\_construct()

{

$this->middleware('guest')->except('logout');

}

}

# Conclusion

Successfully implementing user registration and authentication, shopping cartfunctionality, and a smooth checkout process, the e-commerce platform is ready to offerusersasecureandconvenientshoppingexperience.Userscanregister,additemstotheircart,calculate the total,and complete purchaseswithconfidence.