

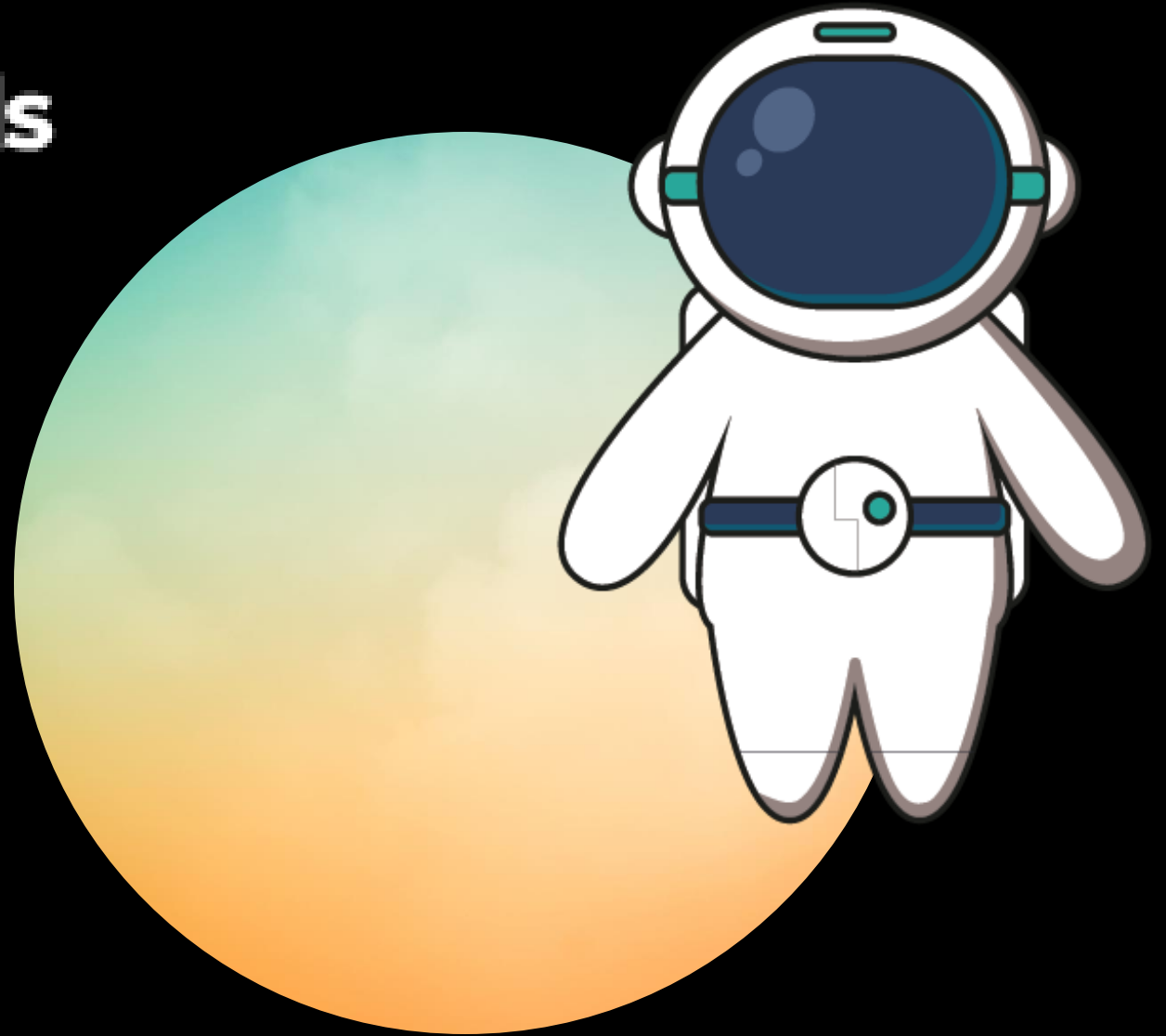
# Apollo Travels

Team members:

Elina Sardaryan

Emma De Barros

Anastassia Titova



# WE HAVE A LONG WAY TO GO...

Materialize for design

Ajax

Google charts

Stripe

Metamask

Middleware

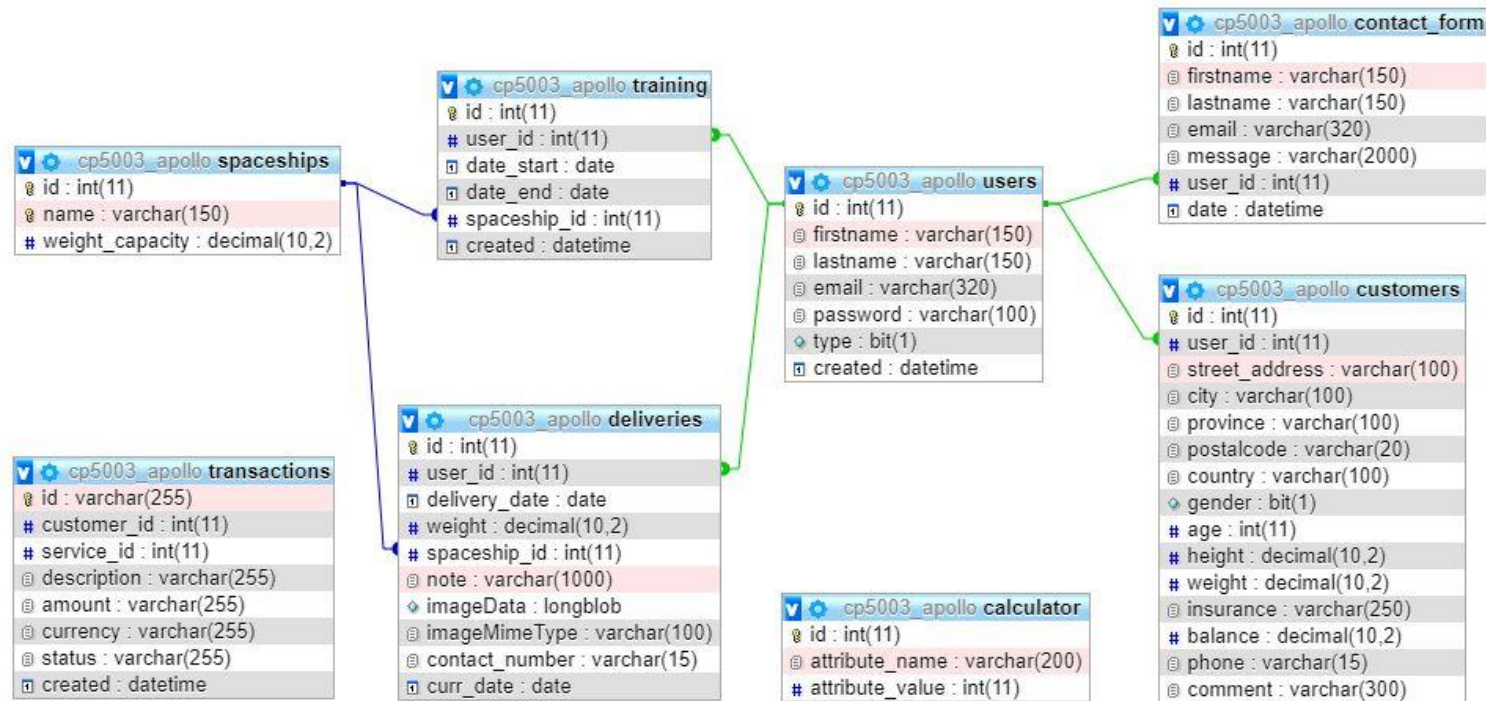
Recaptcha

Password visibility

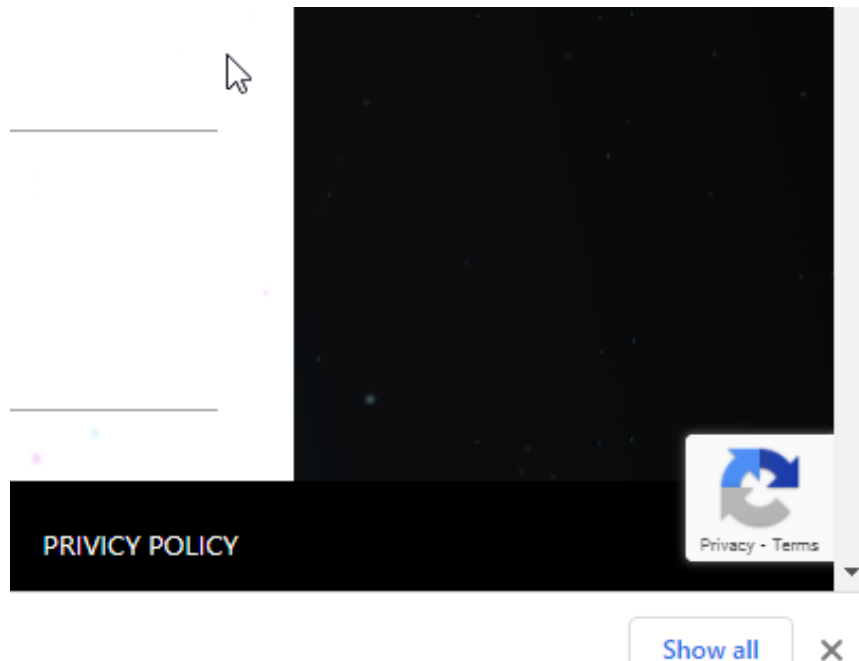
Phone



# DATABASE DESIGN



# BOT PROTECTION – RECAPTCHA V3



CAPTCHAs and reCAPCHAs determine if a user is actually a bot. While these tests can help stop malicious bot activity, they are far from foolproof.

# HTML AND JS SIDE

Input to get recaptcha response

```
<p class="right-align">  
  <input type="hidden" name="recaptcha_response" value="" id="recaptchaResponse">
```

JS to get the token

```
// recaptcha  
grecaptcha.ready(function () {  
  grecaptcha.execute('6Lew1NsaAAAAAJhqKotyf_0JOyPvBrrCOnoJiaac', { action: 'submit' }).then(function  
  (token) {  
    var recaptchaResponse = document.getElementById('recaptchaResponse');  
    recaptchaResponse.value = token;  
  });  
});
```

# PHP SIDE

```
$recaptcha_url = 'https://www.google.com/recaptcha/api/siteverify';
$recaptcha_secret = '6Lew1NsaAAAAAJVqHkMy7L66VS_eAOGpcEXfKTDr';
$recaptcha_response = $_POST['recaptcha_response'];

// // Make and decode POST request:
$recaptcha = file_get_contents($recaptcha_url . '?secret=' . $recaptcha_secret . '&response=' .
$recaptcha_response);
$recaptcha = json_decode($recaptcha, true);

// Take action based on the score returned:
if ($recaptcha->score < 0.5) {
    $response = $response->withStatus(403);
    return $this->view->render($response, 'error_access_denied.html.twig');
}
```

# STRIPE – HTML

```
<div class="row">
  <div class="input-field col s12">
    <div id="card-element" class="black-text">
      <!-- A Stripe Element will be inserted here. -->
    </div>
    <div id="card-errors" class="red-text" role="alert"></div>
  </div>
</div>
```

Stripe -Payments infrastructure for the internet.

# STRIPE -JS

```
<script src="https://unpkg.com/@metamask/detect-provider/dist/detect-provider.min.js"></script>
<script src="https://js.stripe.com/v3/"></script>
<script>
  // Set your publishable key: remember to change this to your live publishable key in production
  // See your keys here: https://dashboard.stripe.com/apikeys
  var stripe = Stripe(
    'pk_test_51Itz2BGSgcqwiHohCMbl6PJGTqKkezRr9gbHMXMZke2rMHPaUB3g8c3mYCmgEC2Mc6GRiwPNjZjiiOZGIv76xICV00LK9JTtqV');
  var elements = stripe.elements();
  // Custom styling can be passed to options when creating an Element.
  var style = {
    base: {
      // Add your base input styles here. For example:
      fontSize: '2rem',
      color: '#000000',
    },
  };

  // Create an instance of the card Element.
  var card = elements.create('card', {
    style: style
  });
```



# STRIPE JS CONTINUED

```
// Create a token or display an error when the form is submitted.
var form = document.getElementById('payment-form');
form.addEventListener('submit', function (event) {
    event.preventDefault();

    stripe.createToken(card).then(function (result) {
        if (result.error) {
            // Inform the customer that there was an error.
            var errorElement = document.getElementById('card-errors');
            errorElement.textContent = result.error.message;
        } else {
            // Send the token to your server.
            stripeTokenHandler(result.token);
        }
    });
});

function stripeTokenHandler(token) {
    // Insert the token ID into the form so it gets submitted to the server
    var form = document.getElementById('payment-form');
    var hiddenInput = document.createElement('input');
    hiddenInput.setAttribute('type', 'hidden');
    hiddenInput.setAttribute('name', 'stripeToken');
    hiddenInput.setAttribute('value', token.id);
    form.appendChild(hiddenInput);

    // Submit the form
    form.submit();
}
```

# STRIPE - PHP

```
// Set your secret key. Remember to switch to your live secret key in production.
// See your keys here: https://dashboard.stripe.com/apikeys

\Stripe\Stripe::setApiKey
('sk_test_51Itz2BGSGcqwIHoh1BSjF4vnK0ADvmPajkSLMGYcv1U70FxOjRUiBqTG1gpnwrKjLQTR2vZhWqIPfUROnulQtFD300429g
RdEL');

$app->post('/bookingdeposit', function (Request $request, Response $response, $args) use ($log) {
    if (!isset($_SESSION['user'])) { // refuse if user not logged in
        $response = $response->withStatus(403);
        return $this->view->render($response, 'error_access_denied.html.twig');
    }
    // Token is created using Stripe Checkout or Elements!
    // Get the payment token ID submitted by the form:
    // $token = $_POST['stripeToken'];
    $token = test_input($request->getParam('stripeToken'));

    $customer = \Stripe\Customer::create(array(
        "email" => $_SESSION['user']['email'],
        "source" => $token
    ));
```

# STRIPE – PHP CONTINUED

```
$charge = \Stripe\Charge::create([
    'amount' => 10000000,
    'currency' => 'usd',
    'description' => 'Booking deposit',
    'customer' => $customer->id
]);

$now = new DateTime(null, new DateTimeZone('America/New_York'));
$service_id = DB::queryFirstField("SELECT MAX(id) as id FROM training");
DB::insert('transactions', [
    'id' => $charge['id'],
    'customer_id' => $_SESSION['user']['id'],
    'service_id' => $service_id,
    'description' => $charge['description'],
    'amount' => $charge['amount'],
    'currency' => $charge['currency'],
    'status' => $charge['status'],
    'created' => $now]);
return $this->view->render($response, 'paysuccess.html.twig', ['confirm' => 13, 'userSession' =>
$_SESSION['user']]);
});
```

# METAMASK WALLET


- In particular, it handles account management and connecting the user to the blockchain.



# METAMASK...

Not Secure | apollo.ipd24.ca/bookingdeposit#!

os /... GitHub Google Docs Google Slides 2021 IPD-24 PHP... senditaway / 2021... JAC Crypto Programming HTML/CSS SandBox Learning Reading List Personal

Apollo Travels 

DRAGON STARSHIP SERVICES SIGN OUT


## Booking Deposit

Let's start your journey to the stars!

Booking Deposit: **100,000.00 USD**

First Name

Last Name

 Metamask installed!



1 of 2

http://apollo.ipd24.ca

### Connect With MetaMask

Select account(s)

[New Account](#)

☒  Emma MetaMask (...6314) 2 ETH 

Only connect with sites you trust. [Learn more](#)

# METAMASK...

Not Secure | apollo.ipd24.ca/bookingdeposit#!

GitHub | Google Docs | Google Slides | 2021 IPD-24 PHP... | senditaway / 2021... | JAC | Crypto | Programming | HTML/CSS | SandBox | Learning | Reading List

Apollo Travels

DRAGON | STARSHIP | SERVICES | SIGN OUT

Booking Deposit

Let's start your journey to the stars!

Booking Deposit: **100,000.00 USD**

First Name

Frida

Last Name

Kahlo

Card number

MM / YY CVC

0x1c294f5a749f3f798132570dfc93bbd4c04f6314

PAY DEPOSIT

Ropsten Test Network

Connected Emma MetaM...  
0x1c29...6314

2 ETH

Buy | Send | Swap

Assets | Activity

2 ETH

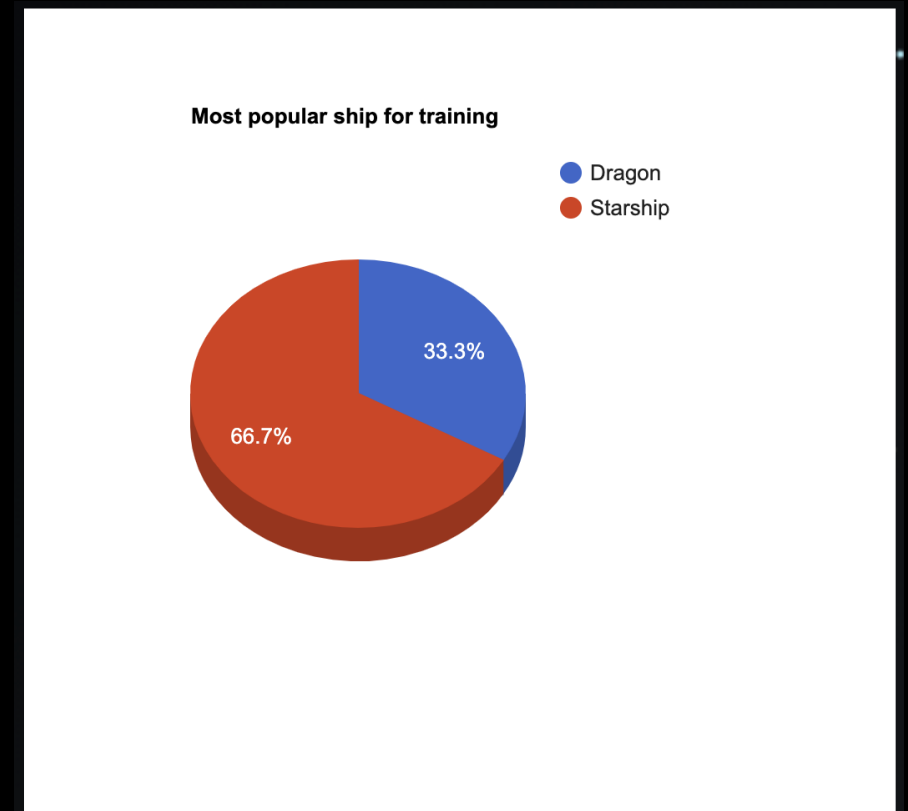
Add Token

Need help? Contact [MetaMask Support](#)

# METAMASK...

```
ethereumButton.addEventListener('click', () => {  
  //Will Start the metamask extension  
  if (typeof window.ethereum !== 'undefined' && ethereum.isMetaMask === true) {  
    // handle provider  
    showAccount.innerHTML = "Metamask installed!";  
    // console.log("Metamask installed!");  
  
    getAccountHandler();  
  
    ethereum.request({  
      method: 'eth_requestAccounts'  
    });  
  
  } else {  
    // handle no provider  
    // showAccount.innerHTML = "Please install metamask https://metamask.io/download";  
    // console.log("Please install metamask.");  
    // alert("Please install metamask: https://metamask.io/download");  
  
    if (window.confirm(  
      'Please click on \'OK\' to start using MetaMask wallet.')) {  
      window.location.href = 'https://metamask.io/download', '_blank';  
    };  
  
  }  
});
```

# GOOGLE PIE CHART...



```
$starshipTrainingList = DB::query("SELECT s.id, s.name, COUNT(s.id) as total FROM training as t, spaceships as s WHERE t.spaceship_id = s.id GROUP BY s.id");
```

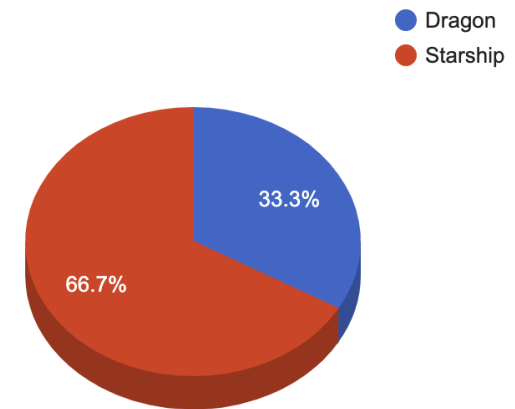


# GOOGLE PIE CHART, JS AND TWIG

```
// draw chart
function drawChart() {

    // Create the data table.
    var data = new google.visualization.DataTable();
    data.addColumn('string', 'Name');
    data.addColumn('number', 'Number of Training');
    data.addRows([
        {% for t in starshipTrainingList %}
            ['{{t.name}}', {{t.total}}],
        {% endfor %}
    ]);
}
```

Most popular ship for training



# CONTACT FORM

- With or without user login
- Validate form data with PHP
- ReCaptcha
- Logs

```
<script>location.href('http://www.hacked.com')</script>
```

```
&lt;script&gt;location.href('http://www.hacked.com')&lt;/script&gt;
```

```
function test_input($data) {  
    $data = trim($data);  
    $data = stripslashes($data);  
    $data = htmlspecialchars($data);  
    return $data;  
}
```

id	firstname	lastname	email	message	user_id	date
14	Vincent	Van Gogh	vincent@van.ca	Great starry night, Apollo!!!	7	2021-05-26 01:40:24
15	John	Malkovich	john@malkovich.ca	Great job!	NULL	2021-05-26 03:15:22

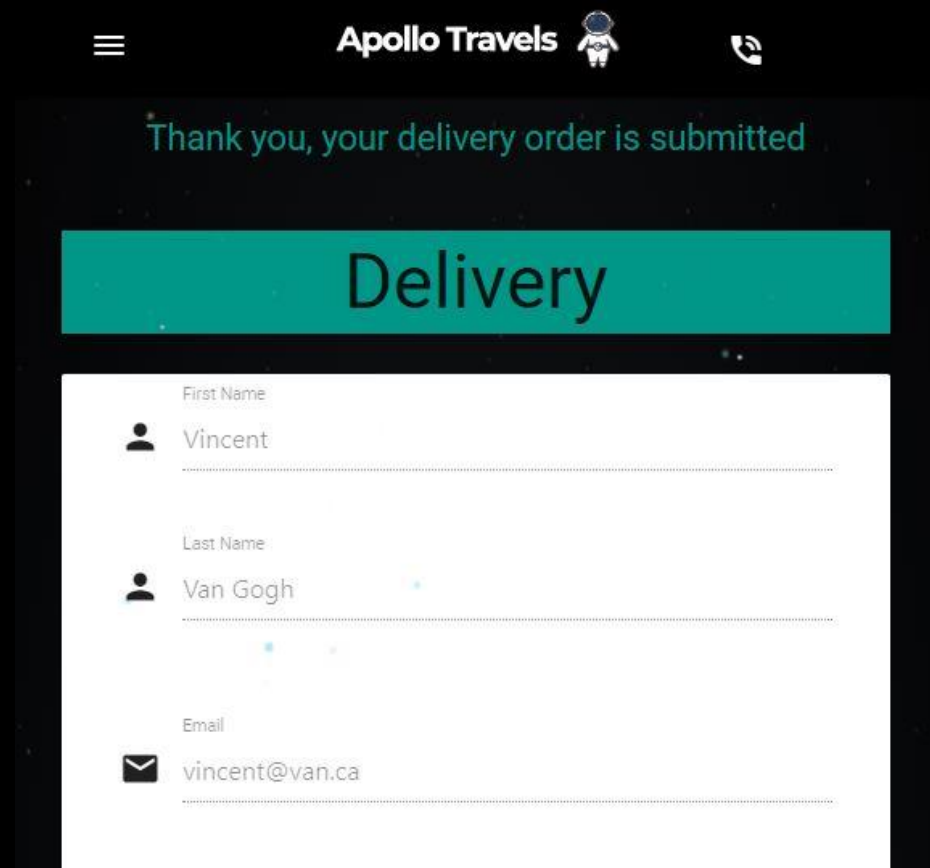
```
[2021-05-26T04:55:36.827934+02:00] main.DEBUG: Message sent successfully by John Malkovich, email: john@malkovich.ca, 127.0.0.1
```

The screenshot shows the 'Contact us' form on the Apollo Travels website. The form has a teal header with the text 'Contact us'. Below the header, there are input fields for 'First Name' (containing 'John') and 'Last Name' (containing 'Malkovich'). There is an 'Email' field containing 'john@malkovich.ca' and a 'Message' field containing 'Great job!'. A teal 'SEND MESSAGE' button is located at the bottom right of the form. The website's navigation bar at the top includes the 'Apollo Travels' logo and a hamburger menu icon.

# DELIVERIES

- Form validation
- Only when user is logged in
- ReCaptcha
- Logs
- Flash message

```
if (!isset($_SESSION['user'])) { // refuse if user not logged in
    $response = $response->withStatus(403); // 403 - access forbidden
    return $this->view->render($response, 'error_access_denied.html.twig');
}
```



The screenshot shows the Apollo Travels website header with a hamburger menu, the brand name "Apollo Travels" with a rocket icon, and a phone icon. A green flash message at the top states "Thank you, your delivery order is submitted". Below this is a teal header with the word "Delivery" in white. The main form area is white and contains three input fields: "First Name" with the value "Vincent", "Last Name" with the value "Van Gogh", and "Email" with the value "vincent@van.ca". Each field has a corresponding icon (person for names, envelope for email) and a light blue focus highlight.

# DELIVERIES

- Form validation
- Only when user is logged in
- ReCaptcha
- Logs
- Flash message

Weight (kg)

6

Delivery Date

06/08/2021

Description

Sunflowers

Choose spaceship


☐ Dragon

☒ Starship

Upload image

Choose Files

1-sunflower...n-gogh.jpg



Calculate delivery

SUBMIT

# CALCULATOR

Calculate the delivery price

- Height
- Width
- Weight

#	Name	Type
1	id 	int(11)
2	attribute_name	varchar(200)
3	attribute_value	int(11)

Calculator

The total price for the delivery

TOTAL: 5 200 000.00

[Calculate again](#) [Go to delivery page](#)

Calculator

Choose the width to calculate the total

☒ Width less than 11cm  
☐ Width from 11 to 30cm  
☐ Width from 31 to 60cm  
☐ Width from 61 to 99cm

[GO BACK](#) [NEXT](#)

# CALCULATOR

```
// STATE 1: first display
$app->get('/calculator', function (Request $request, Response $response, $args) {
    $step = null;
    $valuesList = ['step' => $step];
    return $this->view->render($response, 'calculator.html.twig', ['v' => $valuesList]);
});
```

```
$step = $request->getParam('step');
$sum = 0;

// get values
if ($step == 1) {
    $_SESSION['height'] = $request->getParam('height');
} elseif ($step == 2) {
    $_SESSION['width'] = $request->getParam('width');
} elseif ($step == 3) {
    $weight = $request->getParam('weight');
    try {

        $height_value = DB::queryFirstRow("SELECT * FROM calculator WHERE attribute_name=%s", $_SESSION['height']);
        $width_value = DB::queryFirstRow("SELECT * FROM calculator WHERE attribute_name=%s", $_SESSION['width']);
        $weight_value = DB::queryFirstRow("SELECT * FROM calculator WHERE attribute_name=%s", $weight);

        $sum = $height_value['attribute_value'] + $width_value['attribute_value'] + $weight_value['attribute_value'];
        $sum = number_format($sum, 2, '.', ' ');
    } catch (Exception $e) {
        print_r($e->getMessage());
    }
}
```

```
<!-- ***** FORM HEIGHT ***** >
{% if v.step == null %}
<h5 class="teal-text center-align flow-text ">
    Choose the height to calculate the total</h5>
<div class="divider"></div>
```

```
<form class="col s12 center-align" method="post" style="padding: 0 2rem;">
    <input type="hidden" name="step" value="1">
```

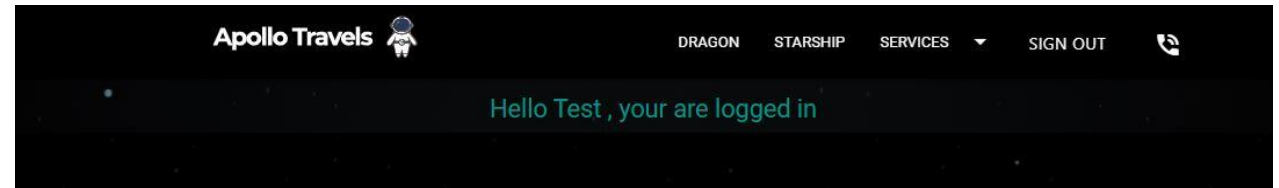
```
{% elseif v.step == 1 %}
```

```
<!-- ***** FORM WIDTH ***** >
<h5 class="teal-text center-align flow-text ">
    Choose the width to calculate the total</h5>
<div class="divider"></div>
```

```
<form class="col s12 center-align" method="post" >
    <input type="hidden" name="step" value="2">
```

# SMALL THINGS DEMANDING BIG TIME

- Animating background and logo
- Navigation bar(dropdown, sidebar)
- Call the agent
- Materialize and its challenges
- Flash messages

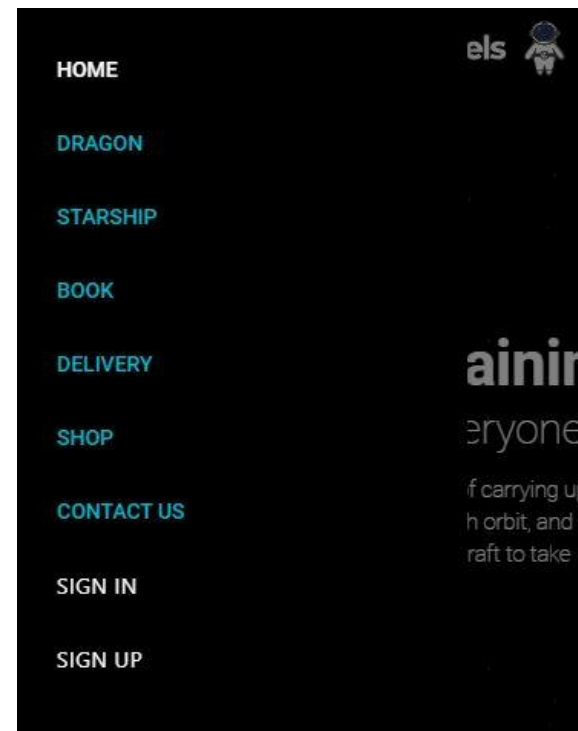


Booking

Delivery

Contact Us

SHOP



```
{% if userSession %}
<li>
  <a href="/logout">
    <button class="waves-effect btn transparent navbtn"
      type="button" name="action">Sign out</button>
  </a>
</li>
{% else %}
<li>
  <a href="/login">
    <button class="waves-effect btn transparent navbtn"
      type="button" name="action">Sign in</button>
  </a>
</li>
<li>
  <a href="/register">
    <button class="waves-effect btn transparent navbtn"
      type="button" name="action">Sign up</button>
  </a>
</li>
{% endif %}
```



# IF WE HAD MORE TIME...

- crypto / NFT
- refactor / tables layout
- shop

