# Conception d'une application web de réservation

Laboratoire de technologie web

Christophe Simon Guillaume de Moffarts

24 décembre 2017

### Introduction

Dans le cadre du laboratoire de Technologie web, nous avons réaliser une application web dont le but est de réserver des billets d'avion. Cette application doit suivre le design pattern MVC (Model View Controler).

Nous avons choisis comme idée de réaliser une caricature de la compagnie aérienne Ryanair, que nous avons appeler BryanAir. Le style de notre application à été inspirer du site officiel de la compagnie.

Dans ce rapport, vous allez trouver des explications sur le fonctionnement normal et erroné de l'application. Vous y trouverez un diagramme de séquence pour imager cela.

# 1 Description du site

Il y a tout d'abord une page d'accueil (figure 1) qui permet de se rediriger vers une section pour effectuer la réservation d'un vol.

Sur la première page de réservation (figure 2), on peut réserver un ou plusieurs billets. Il faut donner l'aéroport de départ et d'arrivée, si on veut réserver un billet aller simple ou aller retour, le nombre de passager, l'adresse email et si on désire une assurance annulation ou pas.

La deuxième étape de réservation (figure 3) consiste a enregistrer les informations pour chaque passager tel que le nom, le prénom et l'age.

Une fois tous les passagers enregistrés, on arrive sur une page de confirmation (figure 4) ou vous avez le choix de confirmer ou d'annuler.

Une fois la réservation confirmée, on arrive sur une page récapitulant la commande (figure 5) avec le prix a payer.

Enfin, il y a une page d'administration (figure 6) ou on peut voir, éditer ou supprimer les passager

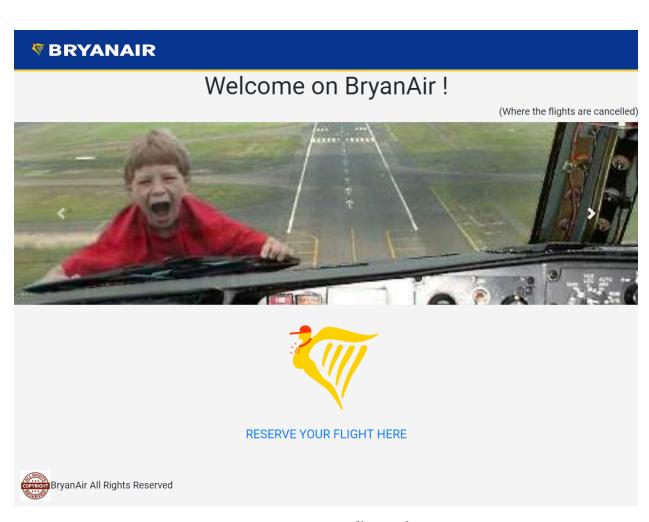


Figure 1 – Page d'accueil

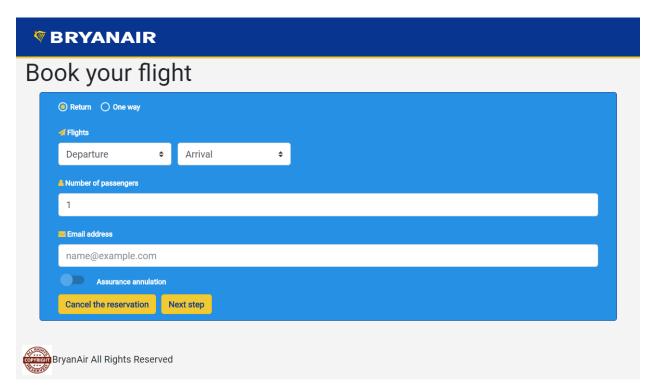


FIGURE 2 – Réservation étape 1

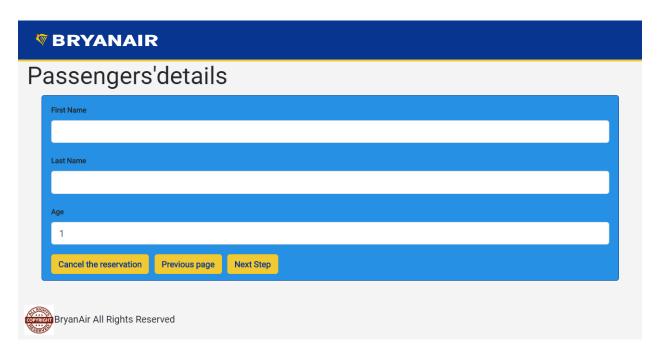


FIGURE 3 – Réservation étape 2

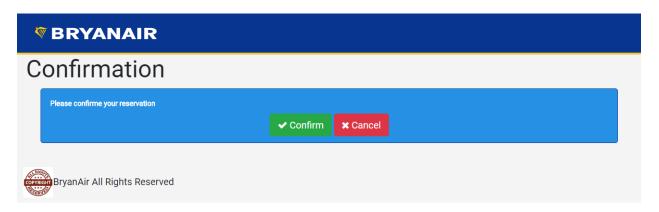


FIGURE 4 – Confirmation

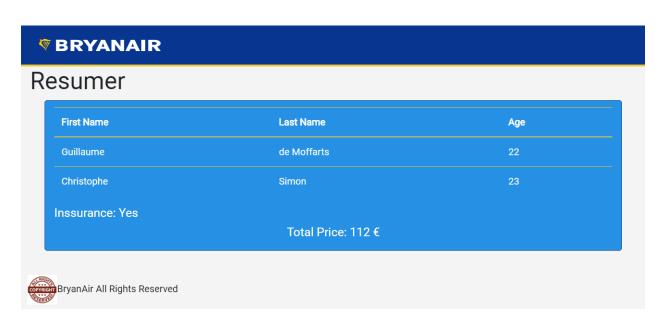


Figure 5 – Récapitulatif

pour chaque vol. Cette page ne devant ètre accessible du public, on y accède uniquement via http://localhost/BryanAir/admin.

#### 2 Fonctionnement

Dans cette section, vous retrouvez les explications sur le fonctionnement des différentes partie de l'application, tel que la redirection d'URL, les réservation et la page d'administration et les templates html.

#### Redirection

La conception de l'application est basé sur le design pattern MVC. Nous avons donc un routeur, des contrôleurs et des views. Le routeur redirige vers le contrôleur adéquate qui va vérifier les informations venant des formulaire, charger et enregistrer les modèles dans la base de donnée. Le contrôleur va ensuite afficher la page.

Lorsque qu'on veux accéder a une page, il suffit d'utiliser l'URL suivante localhost/BryanAir/suivit du nom de la page a accéder. Cette URL est rediriger vers localhost/Bryanair? page=<nom\_de\_page> car le routeur a besoin d'une variable GET ["page"] pour pouvoir rediriger vers le bon controleur. Nous avons fait ce choix pour qu'il soit plus facile d'accéder aux différentes pages. Cette redirection est configurer dans le fichier . htacces.

# **Templates**

Pour faciliter la construction des vues, nous avons utiliser un sytème de templates html. Nous avons plassé des balises comme suit \$balise\$ dans nos pages html. Celle-ci vont etre remplacer par le contrôleur. La fonction buildHTML dans le fichier utils.php construit d'abord la page en concaténant quartes fichiers HTML. Trois de ces fichier sont commun pour toute les vues, il y'a a le head.html, le header.html et le footer.html. Entre ces deux dernier, la vue propre a la page est ajouté. On remplace ensuite toute les balise, grâce a du regex, par des bouts d'HTML construit par le contrôler. Cette fonction prend en paramètre un liste de clef valeurs, les clefs correspond aux mots dans les balises de l'HTML et les valeur contienne donc le contenu a remplacer.

Cette méthode permet de faciliter grandement la création des vues. En effet, il ne sera pas nécésaire de copier dans tous les fichier html un bout de code commun a toute les pages. Cette méthodes permet aussi de bien séparer la partie vu de la partie controleur.

#### 2.1 Réservation

Tous d'abord, en cliquant sur le bouton réservation de la page d'acceuil, le controleur (controle\_reservation.php) charge la liste de tous les aéroport présent dans la base de donées. Le contrôleur va ensuite contruire la page de réservation grace a cette liste.

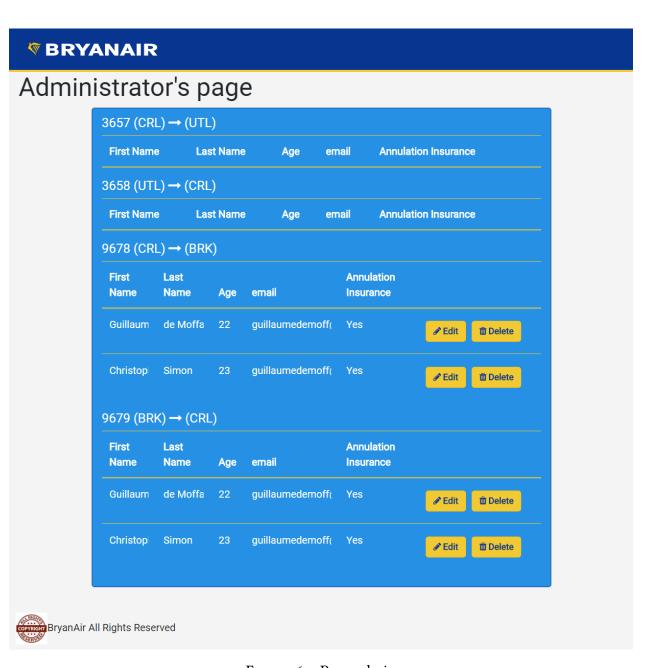


Figure 6 - Page admin

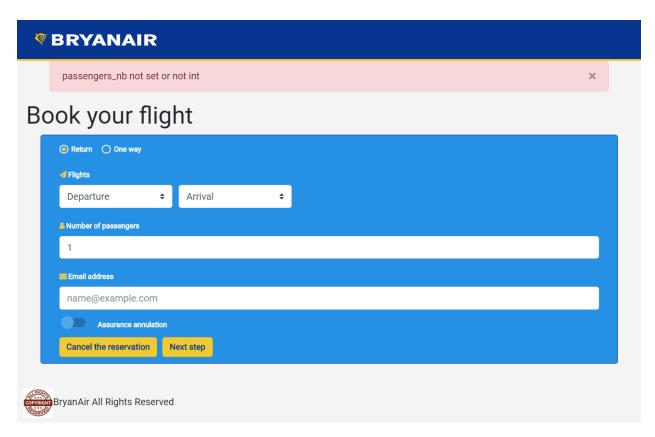


Figure 7 – Message d'erreur

Lorsque l'utilisateur valide la première page de réservation (figure 2), un contrôleur (controleur\_detail.php) vérifie tout d'abord les données venant du POST. En effet si on passe bien par le formulaire, il ne devrais pas avoir de problème mais on peut tout a fait utilisé un outil tel que curl pour envoyer n'importe quel requète POST au serveur ou tout simplement en changeant l'html dans l'inspecteur du navigateur, on peut envoyer du text au lieu d'un entier. Il faut donc bien vérifier que toutes les données du POST nécessaire au contrôleur on bien été définies et sont de type correcte. Par exemple, la variable contenant le nombre de passager doit pouvoir ètre convertit en un entier. Si un problème survient dans la vérification des variables , vous etes redirigez vers la page précédentes avec un message d'erreur sur le haut de celle ci (figure 7. Si la vérification a réussie, Le controleur cherche dans la base de donnée les numéros de vols (aller/retour) et le nombre de place restantes. Si il n'y a plus de place pour un des vols ou qu'il ne trouve pas de vol pour les aéroport sélectionner, on aura un message d'erreur comme décrit précédement. Toutes ces informations sont utilisées pour pour instanttier un objet de type Reservation et en définir l'état. Le controleur va ensuite construire la page HTML (detail.html) grace a la fonction expliqué dans les section précédente.

Nous somme maintenant a la deuxième étapes de la réservations (figure 3) où il faut encoder un a un les informations sur les passagers. Comme pour la première étapes, un contrôleurs (controleur\_nextpassenger.php) vérifie les informations de la variable HTTP POST, le même systéme d'erreur est aussi d'application. Une fois la vérification passée, le contrôleur instancie un objet de type Client dont l'état est définit pas les informations entrées dans le formulaire. Cet objet client est ensuite ajouter dans la liste de client de l'objet réservation. Si il y a encore des passager a enregistrer, le contrôleur contruit la même page html que précédemment pour

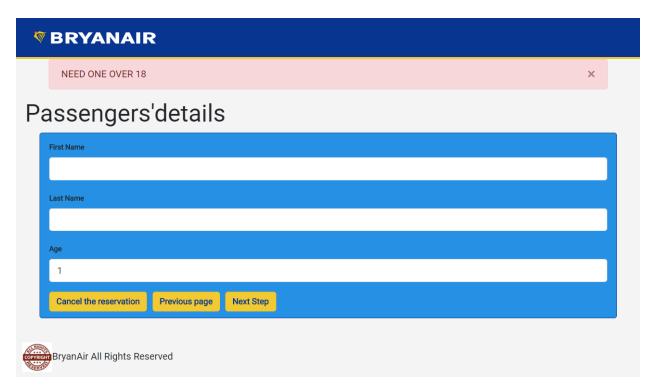


FIGURE 8 – Erreur de majorité

enregistrer un nouveau passager. Si tous les passager sont enregistré, le contrôleur construit la page de confirmation (confirmation.html).

Une fois arriver sur la page de confirmation (figure 4), on a le choix entre confirmer la réservation ou de l'annuler. Si la réservation est annulée, on est redirigé vers la page d'accueil et les l'objet réservation avec la liste des client est supprimé. Si on confirme la réservation, un contréleur (controleur\_confirmation.php) vérifie si au moins un des passagers est bien majeurs. Si ce n'est pas le cas, le système d'erreur affiche un message (figure 8), il faudra ré-encoder les informations des passagers. Si il y a bien au moins une personne majeur, le contrôleur enregistre les clients et réservations dans la basse de données et va ensuite nous rediriger vers une page résumant la commande(figure 5).

Le contrôleur (controleur\_resumer.php) permettant d'afficher cette page vas tout d'abord construire un tableau html, avec tout les passagers de la réservation, grâce a l'objet réservation. Cet objet se charge aussi de calculer le prix total a payer en se basant sur le nombre de passager, si il on prix un vol aller retour ou aller simple et enfin, si une assurance annulation est prise. Le contrôleur vas ensuite construire la page avec toute ces information.

#### 2.2 Admin

En allant sur la page admin, le contrôleur (controler\_admin.php) charge depuis la base de donnée tout les vols ainsi que les passager pour chacun d'eux. Il en fait un tableau html représantant toute ces donnée puis construit la page html (admin.html).

# Conclusion

# A Modèles

# A.1 Client.php

```
<?php
    class Client
{
        private $last_name;
        private $first_name;
        private $age;

        function __construct($first_name , $last_name, $age)
        {
            $this->first_name = $first_name;
            $this->last_name = $last_name;
            $this->age = $age;
        }

        function getLastName()
        {
            return $this->last_name;
        }
        function getFirstName()
        {
            return $this->first_name;
        }
        function getAge()
        {
            return $this->age;
        }
    }
}
```

# A.2 Flight.php

```
<?php
   class Flight
        private $flight_number;
        private $seats;
private $departire;
        private $arrival;
        function __construct($departure, $arrival, $flight_number, $seats)
            $this->departure = $departure;
            $this->arrival = $arrival;
            $this->flight_number = $flight_number;
            $this—>seats = $seats;
        function getArrival()
            return $this—>arrival;
        function getDeparture()
            return $this—>departure;
        function getSeats()
            return $this—>seats;
```

# A.3 Reservation.php

```
<?php
 class Reservation
     private $total_passenger;
     private $clients;
     private $outbound_flight;
     private $return_flight;
     private $inssurance;
     private $email;
     function __construct($total_passenger, $inssurance, $email, $obFlight, $rFlight)
         $this->total_passenger = $total_passenger;
         $this->inssurance = $inssurance;
         $this->email = $email;
         $this->outbound_flight = $obFlight;
         $this->return_flight = $rFlight;
         $this->clients = array();
     function getTotalPassenger()
         return $this—>total_passenger;
     function getRegisterdPassenger()
         return count($this->clients);
     function getInssurance()
         return $this—>inssurance;
     function addClient($client)
         array_push($this->clients, $client);
     function getClients()
         return $this->clients;
```

```
function resetClients()
    $this->clients = array();
function getEmail()
   return $this->email;
function getOBFlight()
   return $this->outbound_flight;
function getRFlight()
   return $this->return_flight;
function getPrice($conn)
   price = 0;
   $price += $this->outbound_flight->getPrice($conn) * count($this->clients);
   if($this->return_flight != null)
        $price += $this->return_flight->getPrice($conn) * count($this->clients);
    if ($this—>inssurance)
        $price += 8;
   return $price;
}
private function saveClients($conn)
    $ids = array();
    $stmt = mysqli_stmt_init($conn);
    if (mysqli_stmt_prepare($stmt, 'INSERT INTO clients (first_name, last_name, email, age)
                                   VALUES (?, ?, ?, ?)')) {
        foreach($this->clients as $client)
            $first_name = $client->getFirstName();
            $last_name = $client->getLastName();
            $age = $client->getAge();
            mysqli_stmt_bind_param($stmt, "ssss", $first_name, $last_name, $this->email,
                $age);
            if (mysqli_stmt_execute($stmt)) {
                echo "New record created successfully";
                array_push($ids, mysqli_insert_id($conn));
            else {
                echo "Error: " . $sql . "<br>" . mysqli_error($conn);
   return $ids;
private function saveReservations($conn, $ids, $flight)
    $stmt = mysqli_stmt_init($conn);
    if (mysqli_stmt_prepare($stmt, 'INSERT INTO reservation (client, flight, inssurance)
                                    VALUES (?,?,?)')) {
```

# B templates

#### B.1 home.html

```
<div>
    <h1 class = "text-center"> Welcome on BryanAir ! </h1>
   <h6 class = "text-right"> (Where the flights are cancelled) </h6>
   <div id="homeCarousel" class="carousel slide" data-ride="carousel">
        <div class="carousel-inner">
            <div class="carousel-item active">
                <img class="d-block w-100" src="ressources/vol.jpg" alt="We never forget your</pre>
                    childs">
            </div>
            <div class="carousel-item">
                <img class="d-block w-100" src="ressources/hugoBeauf.jpg" alt="Pub1">
            </div>
            <div class="carousel-item">
                <img class="d-block w-100" src="ressources/bigPlane.jpg" alt="Our realisation2">
            </div>
       </div>
            <a class="carousel-control-prev" href="#homeCarousel" role="button" data-slide="prev"
                <span class="carousel-control-prev-icon" aria-hidden="true"></span>
                <span class="sr-only">Previous</span>
            <a class="carousel-control-next" href="#homeCarousel" role="button" data-slide="next"
                <span class="carousel-control-next-icon" aria-hidden="true"></span>
                <span class="sr-only">Next</span>
   </div>
   <div class = "text-center">
   <a href="reservation">
        <imq class = "rounded mx-auto d-block" src="ressources/fused-icon.svq" height="200" />
       <h5 class = "text-uppercase"> Reserve your flight here </h5>
   </a>
   </div>
</div>
```

#### **B.2** reservation.html

```
<div class="container-fluid">
    <div>
       <h1>Book your flight</h1>
    </div>
    <div class="container rounded blue-box" >
        <form class="container rounded" method="post" action="detail" >
            <div class="form-check form-check-inline">
                <label class="form-check-label text-white">
                    <input class="form-check-input" type="radio" name="return" value="true"</pre>
                        checked>
                    <span class="checkmark"></span>
                    Return
                </label>
            </div>
            <div class="form-check form-check-inline">
                <label class="form-check-label text-white">
                    <input class="form-check-input" type="radio" name="return" value="false">
                    <span class="checkmark"></span>
                    One way
                </label>
            </div>
            <div class="form-group">
                <label class="text-white " > <i class="fa fa-paper-plane res-icon" aria-hidden="</pre>
                    true"></i> Flights </label>
                <div class="form-row">
                    <div class="col-auto">
                            <select class="custom-select form-control" name="departure" required>
                                <option selected value="">Departure</option>
                                $destinations$
                            </select>
                        </div>
                        <div class="col-auto">
                             <select class="custom-select form-control" name="arrival" required>
                                 <option selected value="">Arrival</option>
                                 $destinations$
                            </select>
                    </div>
                </div>
            </div>
            <div class="form-group">
                <label class="text-white"><i class="fa fa-user res-icon" aria-hidden="true"></i></i>
                    Number of passengers
                <input type="number" class="form-control" min="1" Value="1" name="passengers_nb"</pre>
                    required>
            </div>
            <div class="form-group">
                <label class="text-white"><i class="fa fa-envelope res-icon" aria-hidden="true"><</pre>
                    /i> Email address</label>
                <input type="email" class="form-control" placeholder="name@example.com" name="</pre>
                    email" required>
            </div>
            <div class="form-check">
                    <label class="form-check-label text-white">
                        <input class="form-check-input" type="checkbox" name="inssurance" value="</pre>
                            true">
                        <span class="toggle"></span>
                        Assurance annulation
                    </label>
                </div>
            <div class="button">
                <a href="/BryanAir" class="btn btn-warning">Cancel the reservation</a>
                <button type="submit" class="btn btn-warning" >Next step</button>
```

```
</div>
</form>
</div>
</div>
</div>
```

#### B.3 detail.html

```
<div class="container-fluid">
    <div>
       <h1>Passengers'details</h1>
    </div>
    <div class="container rounded blue-box">
        <form method="post" action="nextpassenger" >
            <div class="form-group">
                <label class="taxt-white">First Name</label>
                <input type="text" class="form-control" name="first_name" required>
            </div>
            <div class="form-group">
                <label >Last Name</label>
                <input type="text" class="form-control" name="last_name" required>
            </div>
            <div class="form-group">
                <label >Age</label>
                <input type="number" class="form-control" value="1" min="1" max="150" name="age"</pre>
                   required>
            </div>
            <div class="button" >
                <a href="/BryanAir" class="btn btn-warning">Cancel the reservation</a>
                <a href="/BryanAir/reservation" class="btn btn-warning">Previous page</a>
                <button type="submit" class="btn btn-warning">Next Step</button>
            </div>
        </form>
   </div>
</div>
```

#### **B.4** confirmation.html

```
<div class="container-fluid">
    <div>
       <h1>Confirmation</h1>
    </div>
    <div class="container rounded blue-box">
        <form method="post" action="confirmation">
            <label class="text-white">Please confirme your reservation</label>
            <div class="container text-center">
                <button type="submit" class="btn btn-success" name="confirm" value=true>
                    <i class="fa fa-check" aria-hidden="true"></i> Confirm
                </button>
                <button type="submit" class="btn btn-danger" name="confirm" value=false>
                    <i class="fa fa-times" aria-hidden="true"></i> Cancel
                </button>
            </div>
        </form>
   </div>
</div>
```

#### B.5 resumer.html

```
First Name
            Last Name
             Age
           </tr>
          </thead>
          $table_client$
          <div>
        <h5 class="text-white">Inssurance: $inssurance$</h5>
     </div>
     <div>
        <h5 class="text-white text-center">Total Price: $price$ €</h5>
     </div>
  </div>
</div>
```

#### B.6 admin.html

#### B.7 head.html

```
<head>
    <meta charset="utf-8" />
    <link rel="icon" type="image/png" href="ressources/fused-icon.png" >
    k rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta.2/css/
        bootstrap.min.css" integrity="sha384-
        PsH8R72JQ3SOdhVi3uxftmaW6Vc51MKb0q5P2rRUpPvrszuE4W1povHYqTpBfshb" crossorigin="
        anonymous">
    <script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" integrity="sha384-</pre>
        KJ3o2DKtIkvYIK3UENzmM7KCkRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN" crossorigin="
        anonymous"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.3/umd/popper.min.js"</pre>
        integrity="sha384-vFJXuSJphROIrBnz7yo7oB41mKfc8JzQZiCq4NCceLEaO4IHwicKwpJf9c9IpFgh"
        crossorigin="anonymous"></script>
    <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0-beta.2/js/bootstrap.min.js"</pre>
        integrity="sha384-alpBpkh1PFOepccYVYDB4do5UnbKysX5WZXm3XxPqe5iKTfUKjNkCk9SaVuEZflJ"
        crossorigin="anonymous"></script>
    <link href="https://fonts.googleapis.com/css?family=Roboto" rel="stylesheet">
    <script src="https://use.fontawesome.com/41f1e461ce.js"></script>
    <link rel="stylesheet" type="text/css" href="Ressources/CSS/style.css">
    <title> BryanAir $title$ </title>
</head>
```

#### B.8 header.html

16

#### B.9 footer.html

#### B.10 admin.html

#### **C** Controlers

### C.1 controler\_home.php

```
<?php
session_unset();
$tags["title"] = "Home";
$_SESSION["status"] = 0;
echo buildHTML("home",$tags);
?>
```

# C.2 controle\_reservation.php

```
<?php
// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
    die("Connection failed: " . mysqli_connect_error());
$sql = "SELECT IATA, name FROM airport";
$result = mysqli_query($conn, $sql);
if (mysqli_num_rows($result) > 0) {
    $destinationsHTML = "";
    while($airport = mysqli_fetch_assoc($result)) {
    $destinationsHTML .= sprintf("<option value= %s>", $airport["IATA"]);
        $destinationsHTML .= sprintf("%s - (%s)", $airport["name"], $airport["IATA"]);
        $destinationsHTML .= "</option>";
} else {
    throw new Exception("Could Not Find any airport");
mysqli_close($conn);
$tags ["destinations"] = $destinationsHTML;
$tags["title"] = "Reservation";
$_SESSION["status"] = 1;
echo buildHTML("reservation", $tags);
```

# C.3 controler\_detail.php

```
<?php
if (empty($_SESSION["status"]))
   header('Location: ./');
    exit();
if($_SESSION["status"] < 2)</pre>
    if($_SESSION["status"] != 1)
        header('Location: ./');
        exit();
    if(empty($_POST["return"]))
        throw new Exception("return cannot be empty");
    if (empty ($_POST["arrival"]))
        throw new Exception("arrival not set or not string");
    if (empty ($_POST["departure"]))
        throw new Exception("departure not set");
    if(empty($_POST["passengers_nb"]) || !ctype_digit($_POST["passengers_nb"]))
        throw new Exception("passengers_nb not set or not int");
    if (empty($_POST["email"]))
        throw new Exception("email not set");
    $conn = mysqli_connect($servername, $username, $password, $dbname);
    if (!$conn) {
        throw new Exception("Could not connect to DB");
    $inssurance = 0;
    if(isset($_POST["inssurance"]) && $_POST["inssurance"] == "true")
        $inssurance = 1;
    // get outbound flight data
    $obFlight = loadFlightSeatsNumber($conn, $_POST["departure"], $_POST["arrival"]);
    $av_seats= getAvailableSeats($conn, $obFlight);
    // get return flight data
    $rFlight = null;
    if($_POST["return"] == "true")
       $rFlight = loadFlightSeatsNumber($conn, $_POST["arrival"], $_POST["departure"]);
       $av_seats_return = getAvailableSeats($conn, $rFlight);
   mysqli_close($conn);
```

### C.4 controler\_nextpassenger.php

```
<?php
if(empty($_SESSION["status"]))
   header('Location: ./');
    exit();
if($_SESSION["status"] < 3)</pre>
    if($_SESSION["status"] != 2 )
       header('Location: ./');
        exit();
    if (empty($_POST["first_name"]))
        throw new Exception("first_name not set");
    if (empty ($_POST["last_name"]))
        throw new Exception("last_name not set");
    if(empty($_POST["age"]) || !ctype_digit($_POST["age"]))
        throw new Exception("age not set or not int");
    $reservation = unserialize($_SESSION["reservation"]);
    $reservation->addClient(new Client($_POST["first_name"], $_POST["last_name"],$_POST["age"]));
if($reservation->getRegisterdPassenger() >= $reservation->getTotalPassenger())
    echo buildHTML("confirmation", $tags);
    $_SESSION["status"] = 3;
else
    $_SESSION["status"] = 2;
    $tags ["title"] = "Detail";
    echo buildHTML("detail", $tags);
```

```
$_SESSION["reservation"] = serialize($reservation);
?>
```

### C.5 controler\_confirmation.php

```
<?php
    if(empty($_SESSION["status"]) || $_SESSION["status"] != 3)
        header('Location: ./');
        exit();
    $reservation = unserialize($_SESSION["reservation"]);
    $clients = $reservation->getClients();
    $tags["title"] = "Confirmation";
    //check for one client at least 18 YO
    $maj = false;
    foreach($clients as $client)
            if($client->getAge() >= 18)
                $maj = true;
                break;
    if(!$maj)
        $reservation->resetClients();
        throw new Exception ("NEED ONE OVER 18");
    $_SESSION["status"] = 4;
   if($_POST["confirm"] == "true")
        $conn = mysqli_connect($servername, $username, $password, $dbname);
        // Check connection
        if (!$conn)
            die("Connection failed: " . mysqli_connect_error());
        // add client and reservations in DB
        $ids = $reservation->save($conn);
        mysqli_close($conn);
        header('Location: resumer');
        die();
        unset($_SESSION["reservation"]);
        header('Location: ./');
?>
```

# C.6 controler\_resumer.php

```
<!php

if(empty($_SESSION["status"]) || $_SESSION["status"] != 4)
{
   header('Location: ./');
   exit();
}
</pre>
```

```
$reservation = unserialize($_SESSION["reservation"]);
$table_clients = "";
foreach($reservation->getClients() as $client)
   $table_clients .= sprintf("
                          %s
                           %s
                           %s
                       ",$client->getFirstName(), $client->getLastName(), $client->getAge()
}
$inssurance = "No";
if($reservation->getInssurance())
   $inssurance = "Yes";
$conn = mysqli_connect($servername, $username, $password, $dbname);
$tags["price"] = $reservation->getPrice($conn);
mysqli_close($conn);
$tags["inssurance"] = $inssurance;
$tags["title"] = "Resumer";
$tags["table_client"] = $table_clients;
echo buildHTML("resumer", $tags);
$_SESSION["status"] = 5;
?>
```

# C.7 controler\_admin.php

```
<?php
// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
if (!$conn)
   die("Connection failed: " . mysqli_connect_error());
$sql="SELECT number, departure, arrival FROM flights";
$result = mysqli_query($conn, $sql);
$tables = "";
if (mysqli_num_rows($result) > 0)
   // output data of each row
   while ($row = mysqli_fetch_assoc($result))
      $tables .= sprintf("<h5>%s (%s) <i class='fa fa-long-arrow-right' aria-hidden='true'></</pre>
         i> (%s)</h5>", $row['number'], $row['departure'], $row['arrival']);
      $sql="SELECT first_name, last_name, age, email, inssurance, C.ID
      FROM reservation R, clients C
      WHERE R.client = C.ID AND R.flight = " . $row['number'];
      $client_result = mysqli_query($conn, $sql);
      $tables .= '
                         First Name
                         Last Name
                         Age
                         email
                         Annulation Insurance
                         </thead>
                 ';
```

```
if (mysqli_num_rows($client_result) > 0)
           while ($client = mysqli_fetch_assoc($client_result))
               $insurance = "No";
               if($client['inssurance'] == 1)
                   $insurance = "Yes";
               $tables .= sprintf("
               <form method='post'>
               <input autocomplete='off' class='modify' value='%s' name='first_name' size
                   ='12'>
               <input autocomplete='off' class='modify' value='%s' name='last_name' size
                   ='12'>
               <input autocomplete='off' class='modify' value='%s' name='age' size='3'>
               <input autocomplete='off' class='modify' value='%s' name='email' size='30'></
                  td>
               <input readonly class='modify' value='%s' size='3'>
               <div class='btn'>
               <button formaction='update' type='submit' class='btn btn-warning'><i class='fa fa</pre>
                   -pencil' aria-hidden='true'></i> Edit</button>
               <button formaction='delete' class='btn btn-warning'><i class='fa fa-trash' aria-</pre>
                   hidden='true'></i> Delete</button>
               <input type='hidden' name='id' value='%s'>
           ",$client["first_name"], $client["last_name"], $client["age"], $client["email"],
                $insurance, $client["ID"]);
       }
       $tables .= "";
   }
else
   throw new Exception("No match found");
$tags["title"] = "Admin";
$tags["tables"] = $tables;
mysqli_close($conn);
echo buildHTML("admin", $tags);
```

# C.8 controler\_update.php

```
if (empty($_POST["first_name"]))
{
    throw new Exception("first name not set");
}

if (empty($_POST["last_name"]))
{
    throw new Exception("last name not set");
}

if (empty($_POST["age"]) || !ctype_digit($_POST["age"]))
{
    throw new Exception("age not set or not int");
}
```

```
if (empty($_POST["email"]))
   throw new Exception("email not set");
if (empty($_POST["id"]))
   throw new Exception("id not set");
$conn = mysqli_connect($servername, $username, $password, $dbname);
if (!$conn) {
   die("Connection failed: " . mysqli_connect_error());
$insurance = 0;
if(strtolower($_POST['insurance']) == "yes")
   $insurance = 1;
echo $insurance;
$sql = sprintf("UPDATE clients SET last_name='%s', first_name='%s', age='%s', email='%s' WHERE id
   if (!mysqli_query($conn, $sql))
   throw new Exception ("Couldn't update passenger");
mysqli_close($conn);
header("Location: ./admin");
```

# C.9 controler\_delete.php

```
if(empty($_POST["id"]))
{
    throw new Exception("id not set");
}
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}

print_r($_POST);
// sql to delete a record
$sql = sprintf("DELETE FROM reservation WHERE client=%s",$_POST["id"]);
if (!mysqli_query($conn, $sql))
{
    throw new Exception("Couldn't delete passenger");
}

mysqli_close($conn);
header("Location: ./admin");
?>
```

# C.10 controler\_404.php

```
<?php
$tags["title"] = "404";
echo buildHTML("404", $tags);
?>
```

### **D** Other

#### D.1 .htaccess

```
#Forbid file indexes from browser
Options -Indexes

#Active URL rewrite engine
RewriteEngine on

RewriteCond %{HTTP_HOST} ^localhost$
RewriteCond %{REQUEST_URI} !index.php
RewriteCond %{REQUEST_URI} !/script
RewriteRule ^([a-z]+)$ index.php?page=$1 [L]
```

# D.2 index.php

```
<?php
require_once("utils.php");
require_once("models/Flight.php");
require_once("models/Client.php");
require_once("models/Reservation.php");
session_start();
$tags = array("error" => "");
//router
if (!empty($_GET['page']))
    $controler_file = 'controler_'.$_GET['page'].'.php';
    if (file_exists($controler_file))
    {
        try
            include $controler_file;
            $_SESSION["lastControler"] = $controler_file;
        catch(Exception $e)
            $tags["error"] = '<div class="container"><div class="alert alert—danger alert—</pre>
                dismissible fade show" role="alert"><button type="button" class="close" data-
                dismiss="alert" aria—label="Close"><span aria—hidden="true">&times;</span></
                button>' . $e->getMessage() . '</div></div>';
            include $_SESSION["lastControler"];
        }
    }
    else
        include 'controler_404.php';
else
    include 'controler_home.php';
?>
```

### D.3 utils.php

```
<?php
//DB indentifier
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "BryanAir";
function buildHTML($page_body, $tags = null)
    body = << EOS
    <!DOCTYPE html>
    <html>
EOS;
    $body .= file_get_contents ("templates/head.html");
    $body .= "<body>";
    $body .= file_get_contents ("templates/header.html");
    $body .= file_get_contents (sprintf("templates/%s.html", $page_body));
    $body .= file_get_contents ("templates/footer.html");
    $body .= <<<EOS
        </body>
    </html>
EOS;
    if($tags == null)
        return $body;
   else
        return replaceTags(buildHTML($page_body),$tags);
}
function replaceTags($content, $tags)
    $pattern = array_map(function($pat){
    return '/\$' . $pat . '\$/';
    },array_keys($tags));
    $content = preg_replace($pattern, $tags, $content);
    return $content;
function loadFlightSeatsNumber($conn, $dep, $arr)
    $sql = "SELECT seats, number FROM flights WHERE departure = '$dep' && arrival = '$arr' ";
    $result = mysqli_query($conn, $sql);
    $flight;
    if (mysqli_num_rows($result) > 0)
        $flight = mysqli_fetch_assoc($result);
        $output_flight = new Flight($_POST["arrival"], $_POST["departure"], $flight["number"],
            $flight["seats"]);
    else
        throw new Exception("We don't have any flight for this trip");
    return $output_flight;
function getAvailableSeats($conn, $flight)
```

# D.4 style.css

```
body
    font-family: 'Roboto', sans-serif;
    background—color: #f4f4f4;
}
#nav-bar
    background—color :#073590;
    border-bottom-width : 3px;
    border-bottom-style: solid;
    border-bottom-color: rgb(241, 201, 51);
    height: 68px;
#logo
    height: 75%;
    margin-top: 7px;
.dspn
    display: none
title
    color: blue
.blue-box
    background-color: #2691e4;
    border: 1px solid #073590;
    padding-top: 10px;
    padding-bottom: 10px
/* check box container */
.form-check-label {
    display: block;
    position: relative;
    padding-left: 20px;
    margin-bottom: 12px;
    cursor: pointer;
/* Hide the browser's default radio button */
.form-check-label input {
   position: absolute;
    opacity: 0;
```

```
/* Create a custom radio button */
.checkmark {
   position: absolute;
   top: 0;
   left: 0;
   height: 15px;
   width: 15px;
   background-color: transparent;
   border-radius: 50%;
   border: 1.5px solid white;
   margin-top: 1px;
/* Create the indicator (the dot/circle — hidden when not checked) */
.checkmark:after {
   content: "";
   position: absolute;
    display: none;
/* Show the indicator (dot/circle) when checked */
.form-check-label input:checked ~ .checkmark:after {
   display: block;
/* Style the indicator (dot/circle) */
.form-check-label .checkmark:after {
  top: 2.5px;
 left: 2.5px;
 width: 7px;
 height: 7px;
 border-radius: 50%;
 background: rgb(241, 201, 51);
.res-icon
{
    color:rgb(241, 201, 51);
.form-check-label .toggle {
   position: relative;
   height: 15px;
   width: 30px;
   background-color: hsla(206, 63%, 41%, 1);
   border-radius: 7.5px;
   margin-top: -3px;
   display: inline-block;
   margin—left: -7px;
   margin-right: 18px;
}
/* Style the indicator (dot/circle) */
.form-check-label .toggle:before {
   content:"";
position: absolute;
   top: −3px;
   left: -10px;
   width: 21px;
   height: 21px;
   border-radius: 50%;
   background—color: hsla(206, 78%, 62%, 1);
   -webkit-transition: .4s;
   transition: .4s;
.form-check-label input:checked + .toggle:before
```

```
transform: translateX(30px);
   background—color: rgb(241, 201, 51);
}
.btn-warning
   background—color: rgb(241, 201, 51);
   color:#073590;
   font-weight: 700;
   font-size: 14px;
   padding: 5px 11px;
   margin-right: 5px;
}
label
    font-weight: 700;
   font-size: 12px;
}
.table th, .table {\tt td}, .table thead th {
   border-color: rgb(241, 201, 51);
footer
   padding-top : 40px;
   padding-left: 10px;
}
.modify
{
   background-color: transparent;
   border: none;
   width: 100%;
   color: white;
#homeCarousel
   max-height: 300px;
   overflow: hidden;
.carousel-item
   height: 300px;
.carousel-item img
{
   position: relative;
   top: 65%;
   transform: translateY(-50%);
```