

Le PtitJardini'Haie

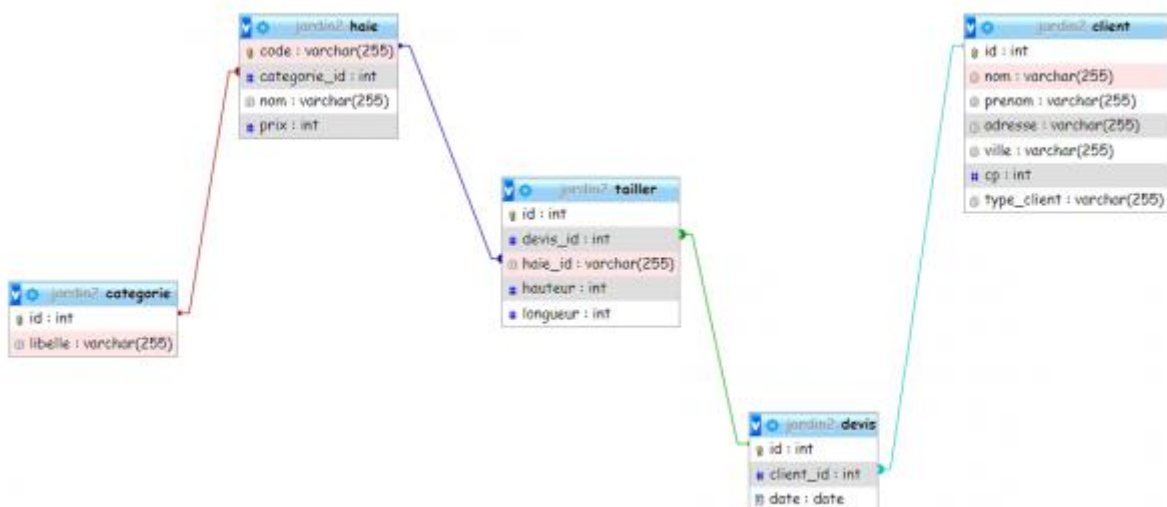


Travail à faire

Compléter le menu de l'application : ajouter la gestion des clients et la gestion des devis
Gestion des clients : recherche/ajout/modification/suppression des clients
Gestion des devis : calcul du devis en fonction du type de client (particulier ou entreprise) pour obtenir la remise et des différents types de haie du devis (hauteur et longueur)

Vous ne devez pas gérer la création d'un devis, juste le calcul. Il faudra uniquement afficher le détail du devis avec le montant total.

Gestion base de données



Gestion des devis

Choix du client

[Accueil](#) [Gestion des types de haies](#) [Gestion des clients](#) [Gestion des devis](#)

Choix du client

--Choisissez votre type client--

Suivant

Dans cette partie, utilisateur choisit le client avec la liste déroulante.

```
<?php
```

```
namespace App\Controller;
```

```
use Symfony\Bundle\FrameworkBundle\Controller\AbstractController;
```

```
use Symfony\Component\HttpFoundation\Response;
```

```
use Symfony\Component\Routing\Annotation\Route;
```

```
use Symfony\Component\HttpFoundation\Session\Session;
```

```
use App\Entity\Client;
```

```
use App\Form\ChoixType;
```

```
use App\Repository\ClientRepository;
```

```
use Symfony\Component\HttpFoundation\Request;
```

```
use Symfony\Bridge\Doctrine\Form\Type\EntityType;
```

```
use Symfony\Component\Form\Extension\Core\Type\SubmitType;
```

```

class ChoixController extends AbstractController

{

    /**

    * @Route("/choix", name="choix")

    */

    public function index(Request $request): Response

    {

        $session = $request->getSession();

        $defaultData = ['message' => 'Type your message here'];

        $form = $this->createFormBuilder($defaultData)

            ->add('client', EntityType::class, [

                'required' => true,

                'label' => false,

                'class' => Client::class,

                'placeholder' => '--Choisissez votre type client--',

                'choice_label' => function (Client $client) {

                    return $client->getNom() . ' ' . $client->getPrenom() . '

: ' . $client->getTypeClient();

                }

            ])

            ->add('Suivant', SubmitType::class)

```

```
->getForm());

$form->handleRequest($request);

if ($form->isSubmitted() && $form->isValid()) {

    $session->set('typeClient', $form->get('client')->getData());

    $session->set('tailleur', array());

    return $this->redirectToRoute('measure', [],
Response::HTTP_SEE_OTHER);

}

return $this->render('choix/index.html.twig', [

    'controller_name' => 'ChoixController',

    'form' => $form->createView(),

]);

}

}
```

Choix du type de haies

Accueil Gestion des types de haies Gestion des clients Gestion des devis

Votre haie

--Choisissez votre type de haie--

Dimension de la haie (en mètre)

Longueur

Hauteur

Ajouter Suivant

Accueil Gestion des types de haies Gestion des clients Gestion des devis

Votre haie

Bambou

Dimension de la haie (en mètre)

Longueur

10

Hauteur

20

Ajouter Suivant

Nom	Hauteur	Longueur	Supprimer
Bambou	20	10	Supprimer

Dans cette partie, utilisateur crée une ou plusieurs haies avec la longueur et la hauteur qu'il souhaite. Après cela, le devis est créé et sera affiché dans le contrôleur Devis.

```
<?php
```

```
namespace App\Controller;
```

```
use App\Entity\Devis;
```

```
use Symfony\Bundle\FrameworkBundle\Controller\AbstractController;
```

```
use Symfony\Component\HttpFoundation\Response;
```

```
use Symfony\Component\Routing\Annotation\Route;
```

```
use Symfony\Component\HttpFoundation\Request;
```

```
use Symfony\Component\HttpFoundation\Session\Session;
```

```
use App\Entity\Haie;
```

```
use App\Entity\Tailler;
```

```
use App\Entity\Client;
```

```
use Doctrine\ORM\EntityManagerInterface;
```

```
use Symfony\Bridge\Doctrine\Form\Type\EntityType;
```

```
use Symfony\Component\Form\Extension\Core\Type\NumberType;
```

```
use Symfony\Component\Form\Extension\Core\Type\TextType;
```

```
use Symfony\Component\Form\Extension\Core\Type\SubmitType;
```

```
class MeasureController extends AbstractController
```

```
{
```

```
    /**
```

```
     * @Route("/measure", name="measure")
```

```
     */
```

```
public function index(Request $request, EntityManagerInterface $doctrine):  
Response  
{  
  
    $defaultData = ['message' => 'Type your message here'];  
  
    $form = $this->createFormBuilder($defaultData)  
        ->add('haie', EntityType::class, [  
            'required' => true,  
            'label' => false,  
            'class' => Haie::class,  
            'placeholder' => '--Choisissez votre type de haie--',  
            'choice_label' => 'nom'  
        ])  
        ->add('longueur', NumberType::class, ['required' => true])  
        ->add('hauteur', NumberType::class, ['required' => true])  
        ->add('Ajouter', SubmitType::class)  
        ->add('Suivant', SubmitType::class)  
  
        ->getForm();  
  
    $form->handleRequest($request);  
  
    $mesHaies = $this->getDoctrine()  
        ->getRepository(Haie::class)
```

```

        ->findAll();

    $session = $request->getSession();

    $TypeClient = $session->get('typeClient');

    $session->set('hauteur', 'fff');

    $tailleur = $session->get('tailleur', []);

    if ($form->getClickedButton() === $form->get('Ajouter')) {

        $TypeHaie = $form->get('haie')->getData();

        $longueur = $form->get('longueur')->getData();

        $hauteur = $form->get('hauteur')->getData();

        $tailleurs = ['haie' => $TypeHaie, 'longueur' => $longueur,
'hauteur' => $hauteur];

        array_push($tailleur, $tailleurs);

        $session->set('tailleur', $tailleur);

    }

    if ($form->getClickedButton() === $form->get('Suivant')) {

        $client = $session->get('typeClient');

        $tailleurs = new Tailler();

        $devis = new Devis();

        $devis->setDate(new \DateTime('now'));

        $clientr = $this->getDoctrine()

            ->getRepository(Client::class)

            ->find($client->getId());
    }

```



```
$devis->setClient($clientr);

$doctrine->persist($devis);

$doctrine->flush();

$doctrine->clear();

$haie = new Haie();

foreach ($tailler as $taille) {

    $devis2 = $this->getDoctrine()

        ->getRepository(Devis::class)

        ->find($devis->getId());

    $haie = $this->getDoctrine()

        ->getRepository(Haie::class)

        ->find($taille['haie']->getId());

    $taillers->setLongueur($taille['longueur']);

    $taillers->setHauteur($taille['hauteur']);

    $taillers->setHaie($haie);

    $taillers->setDevis($devis2);

    $doctrine->persist($taillers);

    $doctrine->flush();

    $doctrine->clear();

}

return $this->redirectToRoute('devis');

}

return $this->render('mesure/index.html.twig', [
```

```

        'controller_name' => 'MesureController', 'typeClient' =>
$TypeClient, 'haies' => $mesHaies,

        'tailleur' => $tailleur,

        'form' => $form->createView(),

    ]);
}

/**
 * @Route("/mesure/{id}", name="mesure_delete", methods={"GET"})
 */
public function delete(Request $request, int $id): Response
{
    $session = $request->getSession();

    $deletehaie = $session->get('tailleur', []);

    unset($deletehaie[$id]);

    $session->set('tailleur', $deletehaie);

    return $this->redirectToRoute('mesure', [], Response::HTTP_SEE_OTHER);
}
}

```

Devis du client

Accueil	Gestion des types de haies ▾	Gestion des clients ▾	Gestion des devis ▾
---------	------------------------------	-----------------------	---------------------

Madame/Monsieur **Ohin Elvis**,
vous êtes un particulier.

Nom de la haie : Bambou
Mesures de la haie : Longueur 10 m, Hauteur 20 m
Le montant de la haie sans la remise : 600€
Vous avez obtenu une remise de : 0€
Le montant de de la haie est de : 600€

Le montant total du devis est de 600€

Dans cette partie, le devis et le montant total sont affichés.

```
<?php

namespace App\Controller;

use Symfony\Bundle\FrameworkBundle\Controller\AbstractController;
use Symfony\Component\HttpFoundation\Response;
use Symfony\Component\Routing\Annotation\Route;
use Symfony\Component\HttpFoundation\Request;
use Symfony\Component\HttpFoundation\Session\Session;

use App\Entity\Haie;
use App\Entity\Devis;
use App\Entity\Client;
use App\Entity\Tailler;

use Doctrine\ORM\EntityManagerInterface;
```

```

class DevisController extends AbstractController
{
    /**
     * @Route("/devis", name="devis")
     */
    public function index(EntityManagerInterface $doctrine, Request $request):
    Response
    {

        $session = $request->getSession();

        $client = $session->get('typeClient');

        $tailleur = $session->get('tailleur');

        $Haie = $request->get('haie');

        $longueur = $request->get('longueur');

        $hauteur = $request->get('hauteur');

        $prenom = $client->getPrenom();

        $nom = $client->getNom();

        $Type = $client->getTypeClient();

        $prixdevis = 0;

        $prixtotal = 0;

        $remise = 0;
    }
}

```

```

return $this->render('devis/index.html.twig', [

    'controller_name' => 'DevisController',

    'typeClient' => $Type, 'longueur' => $longueur, 'hauteur' =>
$hauteur, 'prixdevis' => $prixdevis,

    'remise' => $remise, 'prixtotal' => $prixtotal,

    'nom' => $nom, 'prenom' => $prenom,

    'tailler' => $tailler,

]);

}

}

```

Gestion des clients

Création d'un client

Accueil
Gestion des types de haies
Gestion des clients
Gestion des devis

Créer le client

Nom

Prenom

Adresse

Ville

Code Postal

Type client

☐ Entreprise
☐ Particulier

[Revenir à la liste des clients](#)

```
<?php
```

```
namespace App\Controller;
```

```
use App\Entity\Client;
```

```
use App\Form\ClientType;
```

```
use App\Repository\ClientRepository;
```

```
use Doctrine\ORM\EntityManagerInterface;
```

```
use Symfony\Bundle\FrameworkBundle\Controller\AbstractController;
```

```
use Symfony\Component\HttpFoundation\Request;
```

```
use Symfony\Component\HttpFoundation\Response;
```

```
use Symfony\Component\Routing\Annotation\Route;
```

```
/**
```

```
 * @Route("/client")
```

```
 */
```

```
class ClientController extends AbstractController
```

```
{
```

```
    /**
```

```
     * @Route("/", name="client_index", methods={"GET"})
```

```
     */
```

```
    public function index(ClientRepository $clientRepository): Response
```

```
    {
```

```
        return $this->render('client/index.html.twig', [
```

```

        'clients' => $clientRepository->findAll(),

    ]);

}

/**
 * @Route("/new", name="client_new", methods={"GET", "POST"})
 */

public function new(Request $request): Response
{
    $client = new Client();

    $form = $this->createForm(ClientType::class, $client);

    $form->handleRequest($request);

    if ($form->isSubmitted() && $form->isValid()) {

        $entityManager = $this->getDoctrine()->getManager();

        $entityManager->persist($client);

        $entityManager->flush();

        return $this->redirectToRoute('client_index', [],
Response::HTTP_SEE_OTHER);

    }

    return $this->renderForm('client/new.html.twig', [

        'client' => $client,

        'form' => $form,

    ]);

}

```

Modifier un client

Accueil Gestion des types de haies Gestion des clients Gestion des devis						
Tableau des clients						
Ajouter un nouveau client						
Nom	Prenom	Adresse	Ville	Code Postal	Type de client	Actions
Ohin	Elvis	rue François Perrin	Limoges	87000	Particulier	Modifier Supprimer

Accueil Gestion des types de haies Gestion des clients Gestion des devis						
--	--	--	--	--	--	--

Modifier le client

Nom

Ohin

Prenom

Elvis

Adresse

rue François Perrin

Ville

Limoges

Code Postal

87000

Type client

☐Entreprise☒Particulier

Mettre à jour

[Revenir à la liste des clients](#)

```
/**

 * @Route("/{id}/edit", name="client_edit", methods={"GET", "POST"})

 */

public function edit(Request $request, Client $client): Response
{
    $form = $this->createForm(ClientType::class, $client);

    $form->handleRequest($request);

    if ($form->isSubmitted() && $form->isValid()) {

        $this->getDoctrine()->getManager()->flush();
    }
}
```



```

        return $this->redirectToRoute('client_index', [],
Response::HTTP_SEE_OTHER);

    }

    return $this->renderForm('client/edit.html.twig', [

        'client' => $client,

        'form' => $form,

    ]);

}

```

Supprimer un client

Accueil Gestion des types de hutes Gestion des clients Gestion des devis

Tableau des clients

[Ajouter un nouveau client](#)

Nom	Prenom	Adresse	Ville	Code Postal	Type de client	Actions
Dhin	Elvis	rue François Perrin	Ilmoges	87000	Particulier	Modifier Supprimer

127.0.0.1:8000/client/

127.0.0.1:8000 indique

Tableau des clients

[Ajouter un nouveau client](#)

Nom	Prenom	Adresse	Ville	Code Postal	Type de client	Actions
Dhin	Elvis	rue François Perrin	Ilmoges	87000	Particulier	Modifier Supprimer
lycée	Valadon	rue François Perrin	Ilmoges	87000	Entreprise	Modifier Supprimer

```

/**

 * @Route("/{id}", name="client_delete", methods={"GET"})

 */

public function delete(Request $request, Client $client,
EntityManagerInterface $entityManager): Response

{

```

```
        $entityManager->remove($client);

        $entityManager->flush();

        return $this->redirectToRoute('client_index', [],
Response::HTTP_SEE_OTHER);

    }

}
```