






















## Présentation des droits mis en place du site web « CashCash »

On a deux types d'utilisateurs qui peut accéder au site web.

- Les assistants
- Les techniciens

Dans la table « salariés » on distingue leur rôle par des libellé « typeSal » :

- 0 pour un salarié assistant
- 1 pour un technicien

T+		▼ numMatr		nomSalarie	prenomSalarie	adresseSalarie	dateEmbauche	mdp	typeSal	email	
<input type="checkbox"/>	 Éditer	 Copier	 Supprimer	S0001	Pitt	Brad	32 rue du Chemin Templeuve 59342	2004-12-03	\$2b\$10\$/qsO9wQCyGVxunFIS9zOKYJlIBlIFUuAGPlYH1...	1	bradpitt@cc.com
<input type="checkbox"/>	 Éditer	 Copier	 Supprimer	S0002	Looses	Erwani	24 rue de la Voie Roubaix 59424	2015-08-24	PassTech123@	0	enwanilooses@ag1.fr
<input type="checkbox"/>	 Éditer	 Copier	 Supprimer	S0003	Smith	John	45 rue de la Paix Roubaix 59402	2007-09-20	pass	1	johnsmth@cc.com
<input type="checkbox"/>	 Éditer	 Copier	 Supprimer	S0004	Davis	Emily	67 avenue du Soleil Lille 59002	2018-03-12	pass	0	davisemily@ag2.fr
<input type="checkbox"/>	 Éditer	 Copier	 Supprimer	S0005	Garcia	Maria	10 Place de la Liberté Lille 59000	2019-05-08	pass	0	maria Garcia@ag2.fr
<input type="checkbox"/>	 Éditer	 Copier	 Supprimer	S0006	Chen	Wei	55 Boulevard des Arts Roubaix 59410	2016-11-30	pass	1	weichen@cc.com
<input type="checkbox"/>	 Éditer	 Copier	 Supprimer	S0007	Kim	Ji-hoon	18 Rue de la République Lille 59000	2020-02-17	\$2b\$10\$/xZxUNVo1FQu0qggMaFaGOqEGWWUxANUXAnH.hL8qGK...	0	jihoonkim@ag3.fr

Dans la page de connexion du site :

← → ↻

🛡️ 📄 localhost:3000/login

CashCash

Connexion

E-mail

Mot de passe

Login

login.js :

```

// Gérer les requêtes POST à la racine '/'
router.post('/', function(req, res) {
  // Extraire les données du formulaire de connexion
  let email = req.body.email;
  let mdp = req.body.mdp;

  // Vérifier si l'email et le mot de passe ont été fournis
  if (email && mdp) {

    // Requête SQL pour vérifier l'authentification de l'utilisateur
    connection.query('SELECT * FROM salarie where salarie.email = ?', [email],
function(error, data) {
  if (error) throw error;

  if (data.length > 0) {
    bcrypt.compare(mdp, data[0].mdp, function(err, result) {
      if (result){
        // Enregistrer les informations de session et rediriger vers la page d'accueil
        req.session.loggedin = true;
        req.session.email = email;
        req.session.typeSal = data[0].typeSal;
        req.session.numSal = data[0].numMatr;
        res.redirect('/');
      } else {
        // En cas d'échec d'authentification, afficher un message d'erreur
        res.render('login',{messageErr : "le mot de passe ou l'email n'est pas
correcte"});
      }
      res.end();
    });
  } else {
    res.render('login',{messageErr : "l'utilisateur n'existe pas"});
    res.end();
  }
});
} else {
  // Si l'email ou le mot de passe n'ont pas été fournis, afficher un message d'erreur
  res.render('login',{messageErr : "Veuillez entré un email et un mot de passe"});
  res.end();
}
});
});

```

Si l'utilisateur a réussi de se connecter dans le site, on place les données de la ligne dans les variable session (variable superglobale). Dans ce cas, le type de salarié est stocké dans la variable « req.sesion.typeSal » .

index.js :

```
// Gérer les requêtes GET à la racine '/'
router.get('/', function(req, res, next) {
  // Vérifier si l'utilisateur a une session avec l'adresse email enregistrée
  if (req.session.email) {
    // Rendre la vue 'index' en passant les données de session (email et typeSal) en tant
    que variables
    res.render('index', { email: req.session.email, typeSal: req.session.typeSal });
  } else {
    // Rediriger vers la page de connexion si l'utilisateur n'a pas de session
    res.redirect('/login');
  }
});
```

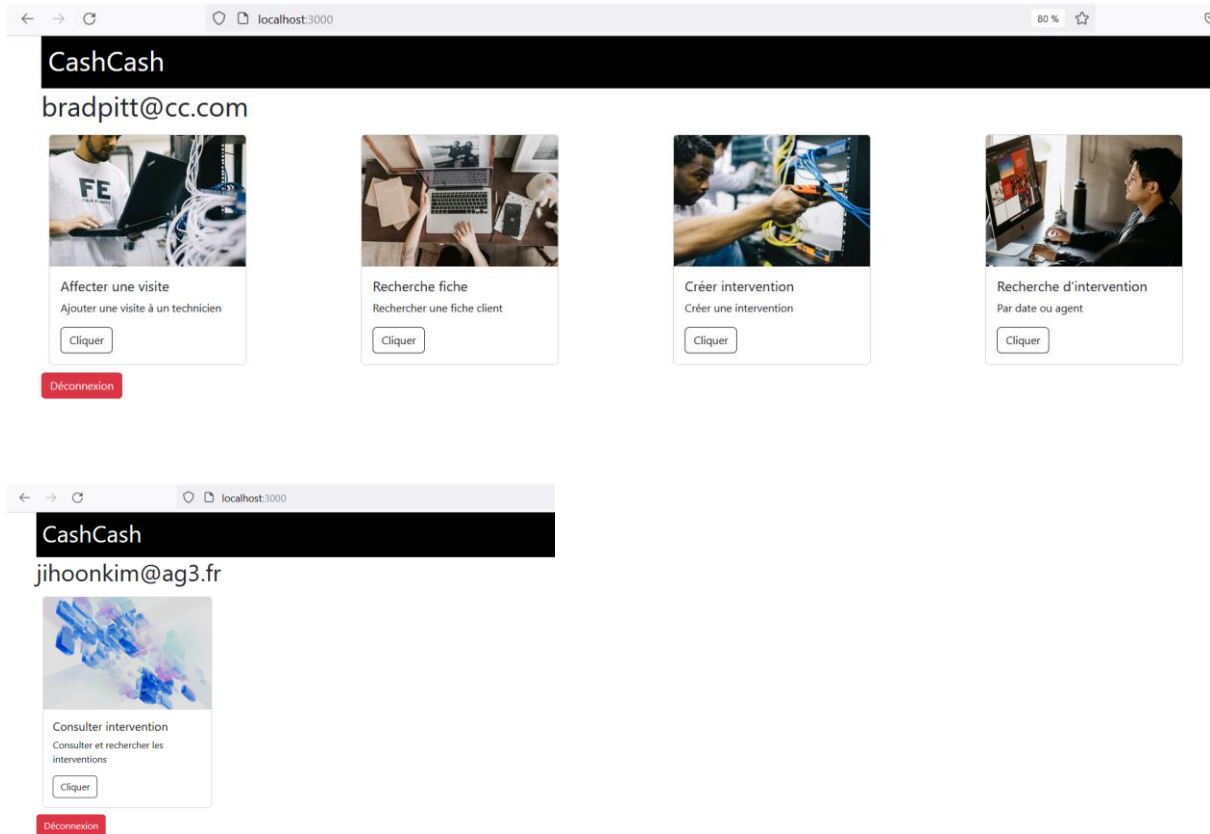
index.pug :

```
extends layout

block content
  if email
    h1= email
    if typeSal == true
      table
        tr
          td
            div(class="card" style="width: 18rem;")
              img(src="images/menuImg.jpg" class="card-img-top" alt="...")
              div(class="card-body")
                h5(class="card-title") Affecter une visite
                p(class="card-text") Ajouter une visite à un technicien
                a(href="affecterVisite" type="button" class="btn btn-outline-dark") Cliquer
          td
            div(class="card" style="width: 18rem;")
              img(src="images/menuImg2.jpg" class="card-img-top" alt="...")
              div(class="card-body")
                h5(class="card-title") Recherche fiche
                p(class="card-text") Rechercher une fiche client
                a(href="rechClient" type="button" class="btn btn-outline-dark") Cliquer
          td
            div(class="card" style="width: 18rem;")
              img(src="images/menuImg3.jpg" class="card-img-top" alt="...")
              div(class="card-body")
                h5(class="card-title") Créer intervention
                p(class="card-text") Créer une intervention
                a(href="creerInter" type="button" class="btn btn-outline-dark") Cliquer
          td
            div(class="card" style="width: 18rem;")
              img(src="images/menuImg4.jpg" class="card-img-top" alt="...")
              div(class="card-body")
                h5(class="card-title") Recherche d'intervention
                p(class="card-text") Par date ou agent
                a(href="rechIntervention" type="button" class="btn btn-outline-dark") Cliquer
        else
          table
            tr
              td
                div(class="card" style="width: 18rem;")
                  img(src="images/menuImg5.jpg" class="card-img-top" alt="...")
                  div(class="card-body")
                    h5(class="card-title") Consulter intervention
                    p(class="card-text") Consulter et rechercher les interventions
                    a(href="consultInter" type="button" class="btn btn-outline-dark") Cliquer
                  a.btn.btn-danger(href="/logout", role="button", style="width:fit-content") Déconnexion
```

Les données dans les variables sessions sont retransmis dans les views par la commande « res.render() »

Dans le fichier index.pug la commande « if typeSal == true » vérifie le type de salarié et affiche une page différent dépendant du type de salarié.



Pour chaque page, on vérifie si tous les utilisateurs on également le droit d'accéder à la page.

Exemple : recherche client qui peut être accessible uniquement aux assistants :

rechClient.js :

```
// Ajout des modules nécessaires
var express = require('express');
var router = express.Router();
const db = require('../connect');
const session = require('express-session');

// Gérer les requêtes GET à la racine '/'
router.get('/', function(req, res, next) {
  // Vérifier si l'utilisateur est connecté et a la variable de session 'typeSal'
  if (req.session.email && req.session.typeSal) {
    // Vérifier si le typeSal de l'utilisateur est égal à 1 (1 = assistant téléphonique)
    if(req.session.typeSal == 1){
      // Rendre la vue 'rechClient' pour permettre à l'utilisateur de rechercher des clients
      res.render('rechClient');
    } else {
      // Rediriger vers la racine '/' si l'utilisateur n'a pas les autorisations nécessaires
      res.redirect("/");
    }
  } else {
    // Rediriger vers la racine '/' si l'utilisateur n'est pas connecté
    res.redirect("/");
  }
});
```

Grace à la commande `if(req.session.typeSal == 1)`

Ils sont redirigés dans la page index sinon.

/consultInter est accessible uniquement aux techniciens :

consultInter.js :

```
// Ajout des modules nécessaires
var express = require('express');
var router = express.Router();
const db = require('../connect');
const session = require('express-session');

// Gérer les requêtes GET à la racine '/'
router.get('/', function(req, res, next) {
  // Vérifier si l'utilisateur a une session avec l'adresse email enregistrée
  if (req.session.email) {
    // Vérifier si l'utilisateur a le rôle (typeSal) différent de 1
    if (req.session.typeSal == 0) {

      //console.log(req.session);

      // Requête SQL pour récupérer les interventions associées à un technicien spécifique
      const sql = "SELECT DISTINCT intervention.numIntervention,
intervention.dateIntervention, intervention.heureIntervention, intervention.numClient,
client.tempsDeplacement,client.distanceKilo FROM intervention, reparer, client WHERE
intervention.numMatr = ? AND intervention.numClient = client.numClient AND
intervention.numIntervention = reparer.numIntervention AND reparer.durreReparationMat is
null ORDER BY intervention.dateIntervention, intervention.heureIntervention,
client.distanceKilo ASC";

      // Exécuter la requête dans la base de données avec le numéro du technicien issu de
      la session
      db.query(sql, req.session.numSal, (err, data) => {
        if (err) throw err;

        //console.log(data);

        // Si aucune donnée n'est récupérée, rendre la vue 'consultInter' avec un message
        approprié, sinon, rendre la vue avec les données récupérées
        if (data.length == 0) {
          res.render('consultInter', { message: "Vous n'avez pas d'intervention" });
        } else {
          res.render('consultInter', { interData: data });
        }
      });
    } else {
      // Si l'utilisateur a le rôle (typeSal) égal à 1, le rediriger vers la racine '/'
      res.redirect("/");
    }
  } else {
    // Si l'utilisateur n'a pas de session ou n'a pas l'adresse email enregistrée, le
    rediriger vers la racine '/'
    res.redirect("/");
  }
});

// Exporter l'objet router pour le rendre disponible pour une utilisation dans d'autres
fichiers
module.exports = router;
```

Grace à la commande `if (req.session.typeSal == 0)`

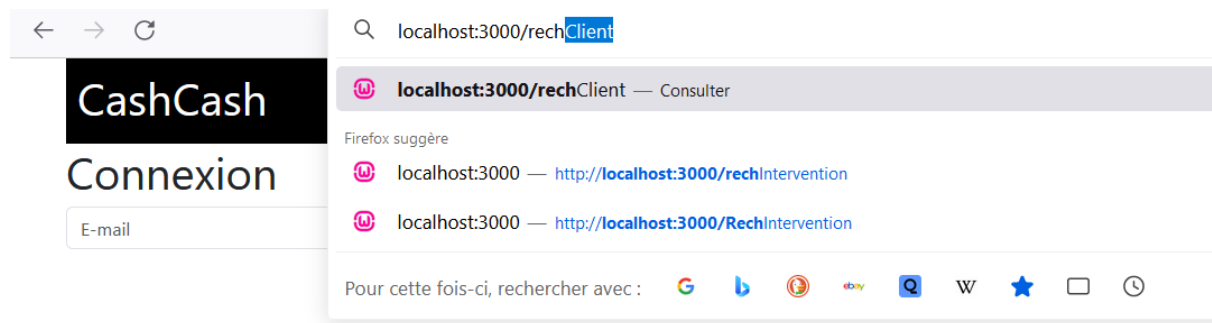
Ils sont redirigés dans la page index sinon.

Les individus qui n'ont pas de compte créer ne peuvent pas accéder les liens grâce à la commande `if (req.session.email)` placé sur chaque page sauf la page de connexion. Ils sont redirigés automatiquement dans la page de connexion s'ils tapent un lien qu'ils n'ont pas le droit d'accès.

Démonstration :

rechClient.js :

```
// Gérer les requêtes GET à la racine '/'
router.get('/', function(req, res, next) {
  // Vérifier si l'utilisateur est connecté et a la variable de session 'typeSal'
  if (req.session.email && req.session.typeSal) {
    // Vérifier si le typeSal de l'utilisateur est égal à 1 (1 = assistant téléphonique)
    if(req.session.typeSal == 1){
      // Rendre la vue 'rechClient' pour permettre à l'utilisateur de rechercher des
clients
      res.render('rechClient');
    } else {
      // Rediriger vers la racine '/' si l'utilisateur n'a pas les autorisations
nécessaires
      res.redirect("/");
    }
  } else {
    // Rediriger vers la racine '/' si l'utilisateur n'est pas connecté
    console.log("utilisateur non connecté")
    res.redirect("/");
  }
});
```



localhost:3000/login

80 %

CashCash

Connexion

E-mail

Mot de passe

Login

```
GET /css/bootstrap.min.css 304 0.998 ms - -
GET /images/menuImg.jpg 304 2.566 ms - -
GET /images/menuImg2.jpg 304 2.029 ms - -
GET /images/menuImg3.jpg 304 3.349 ms - -
GET /images/menuImg4.jpg 304 2.739 ms - -
GET /consultInter 302 0.411 ms - 46
GET / 304 16.550 ms - -
GET /stylesheets/style.css 304 0.989 ms - -
GET /css/bootstrap.min.css 304 1.054 ms - -
GET /images/menuImg.jpg 304 1.660 ms - -
GET /images/menuImg2.jpg 304 1.554 ms - -
GET /images/menuImg3.jpg 304 1.430 ms - -
GET /images/menuImg4.jpg 304 2.864 ms - -
GET /logout 302 0.396 ms - 56
GET /login 200 10.108 ms - 823
GET /stylesheets/style.css 304 1.615 ms - -
GET /css/bootstrap.min.css 304 1.677 ms - -
GET /rechClient 302 0.433 ms - 46
GET / 302 0.319 ms - 56
GET /login 304 7.594 ms - -
GET /stylesheets/style.css 304 0.852 ms - -
GET /css/bootstrap.min.css 304 0.912 ms - -
[nodemon] restarting due to changes...
[nodemon] starting "node ./bin/www"
Connected to MySQL
GET /login 304 293.742 ms - -
GET /stylesheets/style.css 304 3.569 ms - -
GET /css/bootstrap.min.css 304 2.322 ms - -
utilisateur non connecté
```