



Billed

DEBOGUER & TESTER UN Saas RH

Mise en place de la fonctionnalité

« Notes de Frais »

START-UP Activités ➡ Production Solutions Saas
Destinées Au Service « Ressources Humaines »

1 - TODO [Bug report] – Bills =

Modification ordre chrono dates

Description

Le test Bills / les notes de frais s'affichent par ordre décroissant est passé au rouge.

To-do

Faire passer le test au vert en réparant la fonctionnalité.

Rapport de test

```
src/__tests__/Bills.js
  • Given I am connected as an employee • When I am on Bills Page • Then bills should be ordered from earliest to latest
    expect(received).toEqual(expected) // deep equality
    - Expected   - 1
    + Received   + 1

    Array [
      "2004-04-04",
      + "2001-01-01",
      + "2003-03-03",
      - "2002-02-02",
      - "2001-01-01",
    ]

Test Suites: 1 failed, 1 total
Tests:       1 failed, 1 passed, 2 total
Snapshots:  0 total
Time:        3.983 s
```

```
38  /** bug report 1 ordre décroissant DATE */
39  test("Then bills should be ordered from earliest to latest", () => {
40    document.body.innerHTML = BillsUI({ data: bills });
41    const dates = screen
42      .getAllByText(/^(19|20)\d\d[- /.](0[1-9]|1[012])[- /.](0[1-9]|[12][0-9]|3[01])$/i)
43      .map((a) => a.innerHTML);
44    const antiChrono = (a, b) => (a < b ? 1 : -1);
45    const datesSorted = [...dates].sort(antiChrono);
46    expect(dates).toEqual(datesSorted);
47  });
```

PASS src/__tests__/Bills.js (5.104 s)

Given I am connected as an employee

When I am on Bills Page

✓ Then bills should be ordered from earliest to latest (109 ms)

2 - TODO [Bug report] – Bills =

Connexion Admin

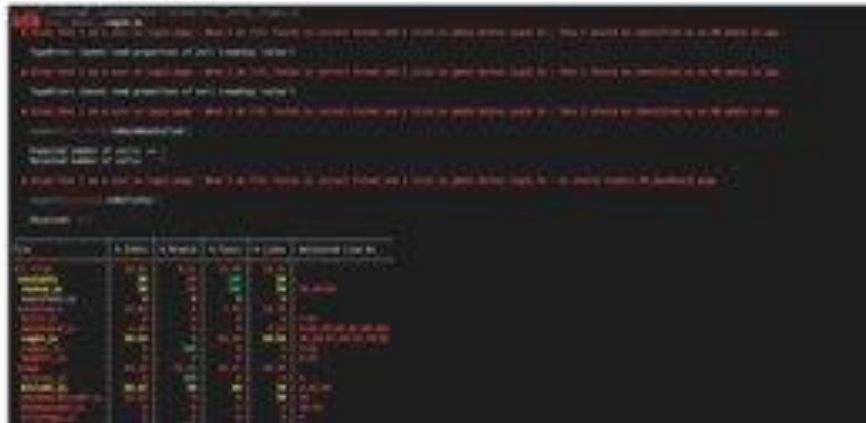
Description

Dans le rapport de test "Login, si un administrateur remplit correctement les champs du Login, il devrait naviguer sur la page Dashboard", le test est passé au rouge (cf. copie d'écran).

To-do

Faire passer le test au vert en réparant la fonctionnalité.

Rapport de test



Employé

Votre email

admin@west.fr

Mot de passe

.....

Se connecter

Administration

Votre email

admin@west.fr

Mot de passe

.....

Se connecter

```
121
122 /** todo Bug report 2 */
123 /** edit employee for admin */
124 describe("Given that I am a user on login page", () => {
125   describe("When I do not fill fields and I click on admin button Login In", () => {
126     test("Then It should renders Login page", () => {
127       document.body.innerHTML = LoginUI();
128
129       const inputEmailUser = screen.getByTestId("admin-email-input");
130       expect(inputEmailUser.value).toBe("");
131
132       const inputPasswordUser = screen.getByTestId("admin-password-input");
133       expect(inputPasswordUser.value).toBe("");
134
135       const form = screen.getByTestId("form-admin");
136       const handleSubmit = jest.fn((e) => e.preventDefault());
```

```
When I do not fill fields and I click on admin button Login In
  ✓ Then It should renders Login page (19 ms)
When I do fill fields in incorrect format and I click on admin button Login In
  ✓ Then it should renders Login page (44 ms)
When I do fill fields in correct format and I click on admin button Login In
  ✓ Then I should be identified as an HR admin in app (73 ms)
  ✓ It should renders HR dashboard page (6 ms)
```

3 - TODO [Bug hunt] – Bills

Description

Je suis connecté en tant qu'employé, je saisis une note de frais avec un justificatif qui a une extension différente de jpg, jpeg ou png, j'envoie. J'arrive sur la page Bills, je clique sur l'icône "voir" pour consulter le justificatif : la modale s'ouvre, mais il n'y a pas d'image.

Si je me connecte à présent en tant qu'Admin, et que je clique sur le ticket correspondant, le nom du fichier affiché est null. De même, lorsque je clique sur l'icône "voir" pour consulter le justificatif : la modale s'ouvre, mais il n'y a pas d'image.

To-do

Comportements attendus :

- ☐ la modale doit afficher l'image.
- ☐ dans le dashboard, le formulaire correspondant au ticket doit afficher le nom du fichier.

Suggestion : empêcher la saisie d'un document qui a une extension différente de jpg, jpeg ou png au niveau du formulaire du fichier NewBill.js. Indice : cela se passe dans la méthode handleChangeFile...

```
19  handleChangeFile = e => {
20    e.preventDefault()
21    const inputFile = this.document.querySelector(`input[data-testid="file"]`)
22    const file = inputFile.files[0]
23    const filePath = e.target.value.split(/\\/g)
24    const fileName = filePath[filePath.length-1]
25    //Ajout pour la vérification de l'extension
26    const fileExtension = fileName.split(".").pop()
27    const formats = ["jpg", "jpeg", "png"]
28  }
```

```
const attachedFile = screen.getByTestId("file");
attachedFile.addEventListener("change", handleChangeFile);
fireEvent.change(attachedFile, {
  target: { files: [new File(["file.pdf"], "file.pdf", { type: "image/pdf" })] },
});

/* Bug Hunt 3
expect(handleChangeFile).toHaveBeenCalled();
expect(attachedFile.files[0].name).toBe("file.pdf");

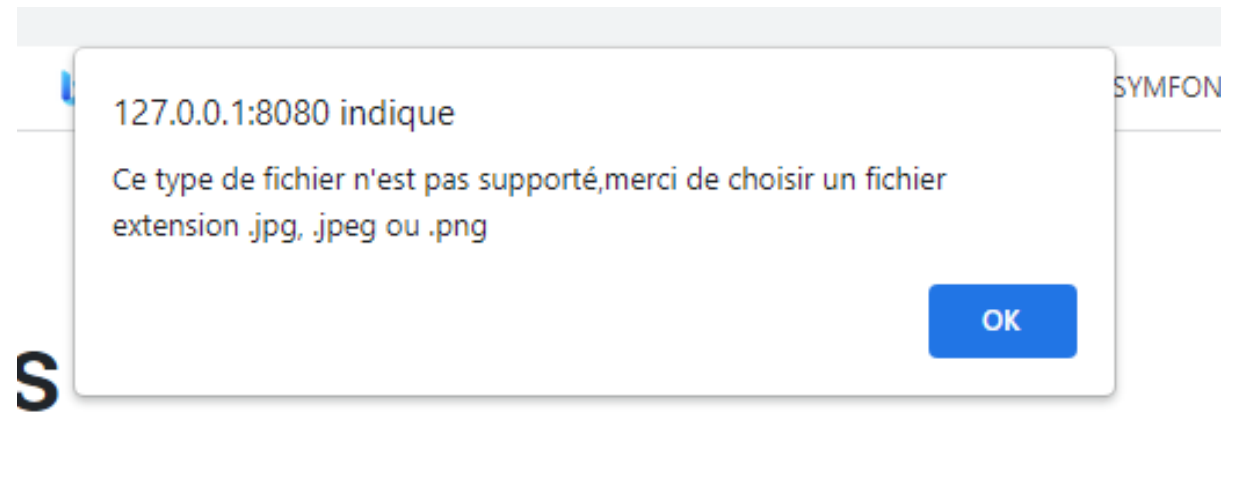
// get DOM element
const errorMsg = screen.getByTestId("errorMsg");

expect(errorMsg.textContent).toEqual(
  expect.stringContaining("Votre justificatif doit être une image de format (.jpg) ou (.jpeg) ou (.png)")
);

/* format valid
test("Then it should accept the file when it as the right format", () => {
```

```
.then(  
  /* istanbul ignore next */  
  ({fileUrl, key}) => {  
    this.billId = key  
    this.fileUrl = fileUrl  
    this.fileName = fileName  
  }).catch(  
    /* istanbul ignore next */  
    error => console.error(error))  
} else {  
  inputFile.value = "";  
  return alert("Ce type de fichier n'est pas supporté, merci de choisir un fichier extension .jpg, .jpeg ou .png");  
}
```

Si l'extension ne correspond pas



4 - TODO [Bug Hunt] – Bills

Permet d'ouvrir les tickets Fichier Dashboard

Description

Je suis connecté en tant qu'administrateur RH, je déplie une liste de tickets (par exemple : statut "validé"), je sélectionne un ticket, puis je déplie une seconde liste (par exemple : statut "refusé"), je ne peux plus sélectionner un ticket de la première liste.

To-do

Comportement attendu : pouvoir déplier plusieurs listes, et consulter les tickets de chacune des deux listes.

Pas besoin d'ajouter de tests.

```
155  /* Todo BUG Hunt 4
156  /* Fermer eventListener avant d'ouvrir un second et reclick
157
158  bills.forEach(bill => {
159    $('#open-bill${bill.id}`).off("click")
160    .on("click", ((e) => this.handleClickTicket(e, bill, bills)))
161  })
162
163  return bills
164
```

5 - Ajout de tests unitaires et d'intégration

constants	100	100	100	100	
routes.js	100	100	100	100	
usersTest.js	0	0	0	0	
containers	93.82	80.39	88.89	96.99	
Bills.js	92.86	66.67	100	92.59	62-63
Dashboard.js	89.29	80.49	79.17	95.89	142-145
Login.js	100	100	100	100	
Logout.js	100	100	100	100	
NewBill.js	100	100	100	100	
views	100	89.47	100	100	
Actions.js	100	100	100	100	
BillsUI.js	100	100	100	100	
DashboardFormUI.js	100	66.67	100	100	30-33
DashboardUI.js	100	100	100	100	
ErrorMessage.js	100	100	100	100	
LoadingPage.js	100	100	100	100	
LoginUI.js	100	100	100	100	
NewBillUI.js	100	100	100	100	

Le rapport de couverture de branche de Jest indique que les fichiers suivants ne sont pas couverts (cf. copie d'écran) :

- ☐ composant views/Bills : Le taux de couverture est à 100% néanmoins si tu regardes le premier test il manque la mention "expect". Ajoute cette mention pour que le test vérifie bien ce que l'on attend de lui.
- ☒ composant ~~views/NewBill~~
- ☐ composant container/Bills :
 - ☐ couvrir un maximum de "statements" c'est simple, il faut qu'après avoir ajouté tes tests unitaires et d'intégration le rapport de couverture du fichier container/Bills soit vert. Cela devrait permettre d'obtenir un taux de couverture aux alentours de 80% dans la colonne "statements".
 - ☐ ajouter un test d'intégration GET Bills. Tu peux t'inspirer de celui qui est fait (signalé en commentaires) pour Dashboard.
- ☐ composant container/NewBill :
 - ☐ couvrir un maximum de "statements" : c'est simple, il faut que le rapport de couverture du fichier container/NewBill soit vert (accessible à cette adresse quand tu auras lancé le serveur). Cela devrait permettre d'obtenir un taux de couverture aux alentours de 80% dans la colonne "statements".
 - ☐ ajouter un test d'intégration POST new bill.
- ☒ composant views/VerticalLayout

Respecter la structure des tests unitaires en place : Given / When / Then avec le résultat attendu. Un exemple est donné dans le squelette du test `__tests__/Bills.js`

TEST POST New bills

```
166  });
167  //On récupère les éléments
168  const formNewBill = screen.getByTestId("form-new-bill");
169  const inputExpenseName = screen.getByTestId("expense-name");
170  const inputExpenseType = screen.getByTestId("expense-type");
171  const inputDatepicker = screen.getByTestId("datepicker");
172  const inputAmount = screen.getByTestId("amount");
173  const inputVat = screen.getByTestId("vat");
174  const inputPct = screen.getByTestId("pct");
175  const inputCommentary = screen.getByTestId("commentary");
176  const inputFile = screen.getByTestId("file");
177
178  //On simule les valeurs
179  fireEvent.change(inputExpenseName, { target: { value: inputData.name } });
180  expect(inputExpenseName.value).toBe(inputData.name);
181
182  fireEvent.change(inputExpenseType, { target: { value: inputData.type } });
183  expect(inputExpenseType.value).toBe(inputData.type);
184
185  fireEvent.change(inputDatepicker, { target: { value: inputData.datepicker } });
186  expect(inputDatepicker.value).toBe(inputData.datepicker);
187
188  fireEvent.change(inputAmount, { target: { value: inputData.amount } });
189  expect(inputAmount.value).toBe(inputData.amount);
190
191  fireEvent.change(inputVat, {
192    target: { value: inputData.vat },
193  });
194  expect(inputVat.value).toBe(inputData.vat);
195
196  fireEvent.change(inputPct, { target: { value: inputData.pct } });
197  expect(inputPct.value).toBe(inputData.pct);
198
199  fireEvent.change(inputCommentary, { target: { value: inputData.commentary } });
200  expect(inputCommentary.value).toBe(inputData.commentary);
201
202  userEvent.upload(inputFile, inputData.file);
203  expect(inputFile.files[0]).toStrictEqual(inputData.file);
204  expect(inputFile.files).toHaveLength(1);
205
206  //On remplit localStorage avec les données de formulaire
207  Object.defineProperty(window, "localStorage", {
208    value: { getItem: jest.fn(() => JSON.stringify({ email: "email@test.com" } )) },
209    writable: true,
210  });
211  //On simule la navigation
212  const onNavigate = (pathname) => {
213    document.body.innerHTML = ROUTES[pathname];
214  };
215
216  //Initialisation
217  const newBill = new NewBill({ document, onNavigate, localStorage: window.localStorage });
218
219  //On déclenche l'évènement
220  const handleSubmit = jest.fn(newBill.handleSubmit);
221  formNewBill.addEventListener("submit", handleSubmit);
222  fireEvent.submit(formNewBill);
223  expect(handleSubmit).toHaveBeenCalled();
224  });
```

On récupère les éléments

On simule les valeurs

On remplit le localStorage avec les données du formulaire

On simule un navigation

initialisation

Déclenchement de l'évènement

Plan de Test E 2 E pour le parcours Employe



Billed

Plan de test End-to-End du parcours employé

Connexion /employé

<http://127.0.0.1:8080/>

SCENARIO 1

GIVEN	Je suis un User non connecté, je veux me connecter en tant qu'employé
WHEN	Je ne remplis pas correctement le champ de l'email
THEN	Un message d'erreur m'indique de remplir le champ avec adresse mail valide comportant un @correspondant aux critères corrects

SCENARIO 2

GIVEN	Je suis un User non connecté, je veux me connecter en tant qu'employé
WHEN	Je ne remplis pas correctement le mot de passe
THEN	Un message d'erreur m'indique de recommencer et d'entrer un mot de passe valide

SCENARIO 3

GIVEN	Je suis un User non connecté, je veux me connecter en tant qu'employé
WHEN	Je remplis correctement le mot de passe + l'adresse mail
THEN	Je click sur le bouton « Se connecter » Je suis dirigé sur la page « note de frais #employee/bill »

RAPPORT de TEST - COVERAGE

✓
+ 80 %

usersTest.js	0	0	0	0	
containers	93.82	80.39	88.89	96.99	
Bills.js	92.86	66.67	100	92.59	62-63
Dashboard.js	89.29	80.49	79.17	95.89	142-145
Login.js	100	100	100	100	
Logout.js	100	100	100	100	
NewBill.js	100	100	100	100	
views	100	89.47	100	100	
Actions.js	100	100	100	100	
BillsUI.js	100	100	100	100	
DashboardFormUI.js	100	66.67	100	100	30-33
DashboardUI.js	100	100	100	100	
ErrorMessage.js	100	100	100	100	
LoadingPage.js	100	100	100	100	
LoginUI.js	100	100	100	100	
NewBillUI.js	100	100	100	100	
VerticalLayout.js	100	100	100	100	
<hr/>					
Test Suites: 11 passed, 11 total					
Tests: 57 passed, 57 total					
Snapshots: 0 total					
Time: 7.089 s					