

Nom de naissance	➤	Ponzio
Nom d'usage	➤	Fabien
Prénom	➤	Fabien
Adresse	➤	12 rue pierre guys 13012 Marseille

Titre professionnel visé

Concepteur développeur d'applications - Niveau II

MODALITÉ D'ACCÈS :

- ☐ Parcours de formation
- ☐ Validation des Acquis de l'Expérience (VAE)

Présentation du dossier

Le dossier professionnel (DP) constitue un élément du système de validation du titre professionnel.
Ce titre est délivré par le Ministère chargé de l'emploi.

Le DP appartient au candidat. Il le conserve, l'actualise durant son parcours et le présente **obligatoirement à chaque session d'examen.**

Pour rédiger le DP, le candidat peut être aidé par un formateur ou par un accompagnateur VAE.

Il est consulté par le jury au moment de la session d'examen.

Pour prendre sa décision, le jury dispose :

1. des résultats de la mise en situation professionnelle complétés, éventuellement, du questionnaire professionnel ou de l'entretien professionnel ou de l'entretien technique ou du questionnement à partir de productions.
2. du **Dossier Professionnel** (DP) dans lequel le candidat a consigné les preuves de sa pratique professionnelle.
3. des résultats des évaluations passées en cours de formation lorsque le candidat évalué est issu d'un parcours de formation
4. de l'entretien final (dans le cadre de la session titre).

*[Arrêté du 22 décembre 2015, relatif aux conditions de délivrance des titres professionnels
du ministère chargé de l'Emploi]*

Ce dossier comporte :

- pour chaque activité-type du titre visé, un à trois exemples de pratique professionnelle ;
- un tableau à renseigner si le candidat souhaite porter à la connaissance du jury la détention d'un titre, d'un diplôme, d'un certificat de qualification professionnelle (CQP) ou des attestations de formation ;
- une déclaration sur l'honneur à compléter et à signer ;
- des documents illustrant la pratique professionnelle du candidat (facultatif)
- des annexes, si nécessaire.

Pour compléter ce dossier, le candidat dispose d'un site web en accès libre sur le site.



<http://travail-emploi.gouv.fr/titres-professionnels>

Sommaire

Exemples de pratique professionnelle

Activité-type n° 1 : Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité	p.	5
- Maquettage de l'application Ultimate List	p. p.	5
- Intégration de la librairie Reactour Alternance Deki	p. p.	7
- Génération d'un fichier pdf en utilisant ejs Alternance Deki	p. p.	11
Activité-type n° 2 : Concevoir et développer la persistance des données en intégrant les recommandations de sécurité	p.	13
- Schématisation de la base de données UltimateList	p. p.	
Activité-type n° 3 : Concevoir et développer une application multicouche répartie en intégrant les recommandations de sécurité	p.	16
- Déploiement et hébergement d'un site web L'ornithorynque	p. p.	16
- Système de réattribution des courses Deki	p. p.	18
Déclaration sur l'honneur	p.	21
Annexes (Si le RC le prévoit)	p.	

EXEMPLES DE PRATIQUE PROFESSIONNELLE

Activité-type 1

Concevoir et développer une application multicouche
répartie en intégrant les recommandations de sécurité

Exemple n° 1 ▶ Maquettage de l'application Ultimate List

J'ai réalisé en compagnie de mes camarades de travail une maquette pour le projet UltimateList, une application de création de Tier-List.

Une tier list est un classement concernant tous les personnages jouables d'un jeu vidéo par ordre de viabilité. Elle permet alors de classer, de manière purement théorique, les personnages en fonction de leur potentiel, pour des joueurs les contrôlant de niveau parfaitement égal.

Nous avons décidé de faire de la même manière que certaines applications de streaming, Spotify propose chaque année à leurs utilisateurs de voir leurs rétrospectives musicales selon plusieurs critères : les artistes les plus écoutés, les musiques les plus écoutées sous forme de classement. Ces rétrospectives sont largement diffusées par les utilisateurs sur les réseaux sociaux.

Tâches effectuées :

- Définition du ton utilisé dans l'application : Le texte se voudra user friendly, ce qui veut dire que le langage utilisé ne sera pas formel mais plutôt familier. Un langage communautaire propre à l'application sera utilisé.
- Définition de l'ambiance : Une ambiance "Kawaii" sera adoptée, des couleurs vives et pastels seront utilisées, ce qui pourra parfois décrédibiliser les tier-lists les plus "sérieuses". Nous avons décidé d'adopter ce thème car notre public est principalement "millennial". Une possibilité de changement de thème sera implanté pour pouvoir s'adapter à des utilisateurs qui ne font pas partie de cette tranche d'âge.
- Définition de la charte graphique :



ExtraLight 200

KGRedHands.ttf

Ultimate List

UltimateList

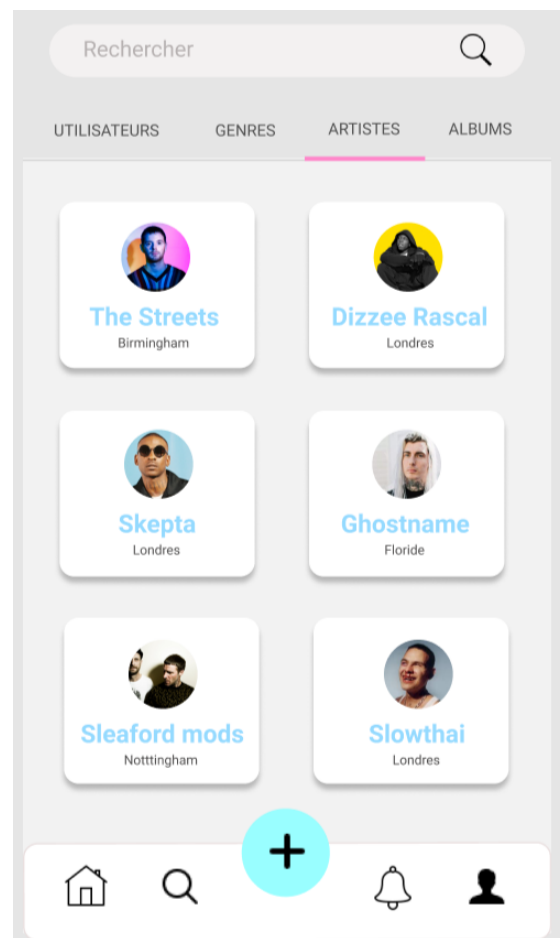
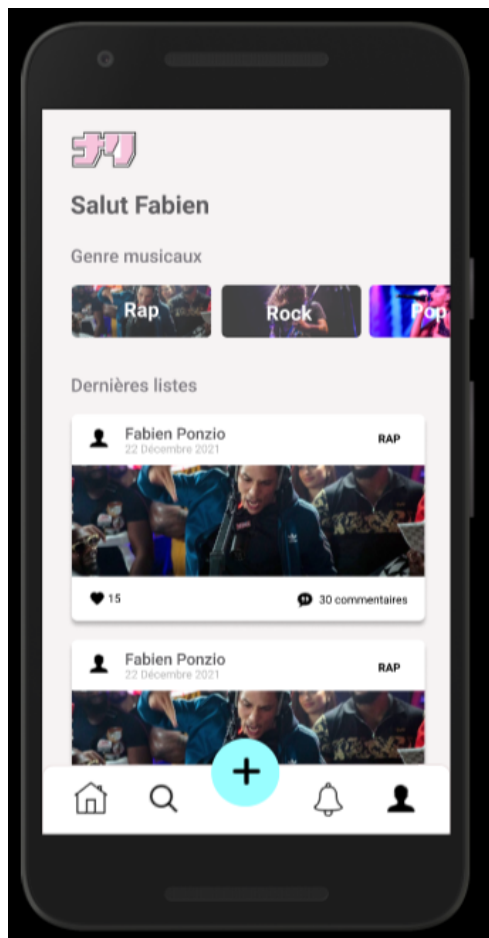
Light 300

KGRedHandsOutline.ttf

Ultimate List

UltimateList

L'ensemble des captures d'écran ci-dessus proviennent de LucidSpark, les captures d'écrans ci-dessous proviennent de Figma



L'UX Design (ou User Experience Design) et les couleurs adoptées sont des couleurs enfantines et douces appliquées dans le but de faire vivre une expérience plaisante et enrichissante à l'utilisateur car nous savons à quel point les premiers instants de l'utilisateur sur l'application sont cruciaux.

La difficulté principale lors de la réalisation de la maquette a été de définir un parti pris esthétique et graphique qui définisse la marque "UltimateList". Nous nous sommes basés sur une esthétique japonaise forte en couleur et connue dans le monde entier.

Cette activité m'a permis de valider la compétence suivante : **Maquetter une application**

2. Précisez les moyens utilisés :

- ✓ Figma pour la réalisation des maquettes design.
- ✓ LucidSpark, pour la partie réflexion / brainstorming
 - Trello pour l'organisation

3. Avec qui avez-vous travaillé ?

Walid BRICHE / Emma LAPREVOTE

4. Contexte

Nom de l'entreprise, organisme ou association ▶ *La Plateforme*

Chantier, atelier, service ▶ *Ultimate List*

Période d'exercice ▶ Du : *5 septembre 2021* au : *31 août 2022*

5. Informations complémentaires (facultatif)

Activité-type 1

**Concevoir et développer une application multicouche
répartie en intégrant les recommandations de sécurité**

Exemple n° 2 ▶ Intégration de la librairie Reactour

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

L'intégration de la librairie Reactour s'est faite au sein d'une application multi-couches dans le but de répondre la problématique qu'une formation à l'outil est nécessaire.

Reactour permet de proposer un "guide" activable au clic en mettant des zones importantes de l'écran en surbrillance, à la manière d'un tutoriel afin que l'utilisateur puisse exploiter au mieux l'outil.

```
npm i -S @reactour/tour @emotion/react  
# or  
yarn add @reactour/tour @emotion/react
```

Commande d'installation

```

ReactDOM.render(
  <ApolloProvider client={client}>
    <Provider store={store} context={ReactReduxContext}>
      <Fragment>
        <GlobalLoader>
          <Tour>
            <Router context={ReactReduxContext}/>
          </Tour>
        </GlobalLoader>
      </Fragment>
    </Provider>
  </ApolloProvider>,
  document.getElementById('root')
);

```

Ici j'intègre le composant Tour dans le fichier index.js afin que toute l'application puisse utiliser et accéder à Reactour.

```

export const StepsHome = [
  {
    selector: '[data-tour="step-1"]',
    content: <p className='step-text'>Cliquez ici pour commencer la création d'une nouvelle course</p>,
  },
  {
    selector: '[data-tour="step-2"]',
    content: <p className='step-text'>Triez vos courses par statut ici </p>,
  },
  {
    selector: '[data-tour="step-3"]',
    content: <p className='step-text'>Vous pouvez aussi trier vos courses sur une période donnée</p>,
  },
  {
    selector: '[data-tour="step-4"]',
    content: <p className='step-text'>Recherchez vos courses en fonction des clients</p>,
  },
  {
    selector: '[data-tour="step-5"]',
    content: <p className='step-text'>Vous trouverez ici le détail de toutes vos livraisons</p>,
  },
  {
    selector: '[data-tour="step-6"]',
    content:
      <ul className='icon-list'>
        <li>
          <p className='step-text'><QrcodeOutlined className='pop-icons'></QrcodeOutlined>: Afficher QR Code</p>
        </li>
        <li>
          <p className='step-text'><MailOutlined className='pop-icons'></MailOutlined>: Recevoir le QR Code par mail</p>
        </li>
        <li>
          <p className='step-text'><DownloadOutlined className='pop-icons'></DownloadOutlined>: Imprimer le QR Code</p>
        </li>
        <li>
          <p className='step-text'><PrinterOutlined className='pop-icons'></PrinterOutlined>: Imprimer l'étiquette</p>
        </li>
        <li>
          <p className='step-text'><FileDoneOutlined className='pop-icons'></FileDoneOutlined>: Afficher la signature</p>
        </li>
      </ul>
    },
  },
  {
    selector: '[data-tour="step-7"]',
    content: <p className='step-text'>Vous pouvez nous contacter en cliquant ici!</p>,
  },
];

```

Les "étapes" ou "step" correspondent à la description accompagnant les inputs, elles sont toutes recueillies dans un même fichier steps.js, lui-même divisé en tableaux qui correspondent aux différentes pages où les visites guidées vont être faites.


```
import {stepsHome, stepsCrm, collectSteps, recapSteps, deliverySteps} from
'./steps';

export function displaySteps(routerPath, activeCourse) {
  switch(routerPath){
    case '/': return stepsHome;
    case '/courseMerchant':
    if (activeCourse === 0) {
      return collectSteps
    }else if (activeCourse === 1) {
      return deliverySteps
    } else {
      return recapSteps
    }
    case '/crmShop': return stepsCrm;
    default: return null;
  }
}
```

Je l'importe au sein d'un nouveau fichier qui contient la fonction displaySteps elle-même activée dans le composant TourProvider représenté visuellement par ce bouton :

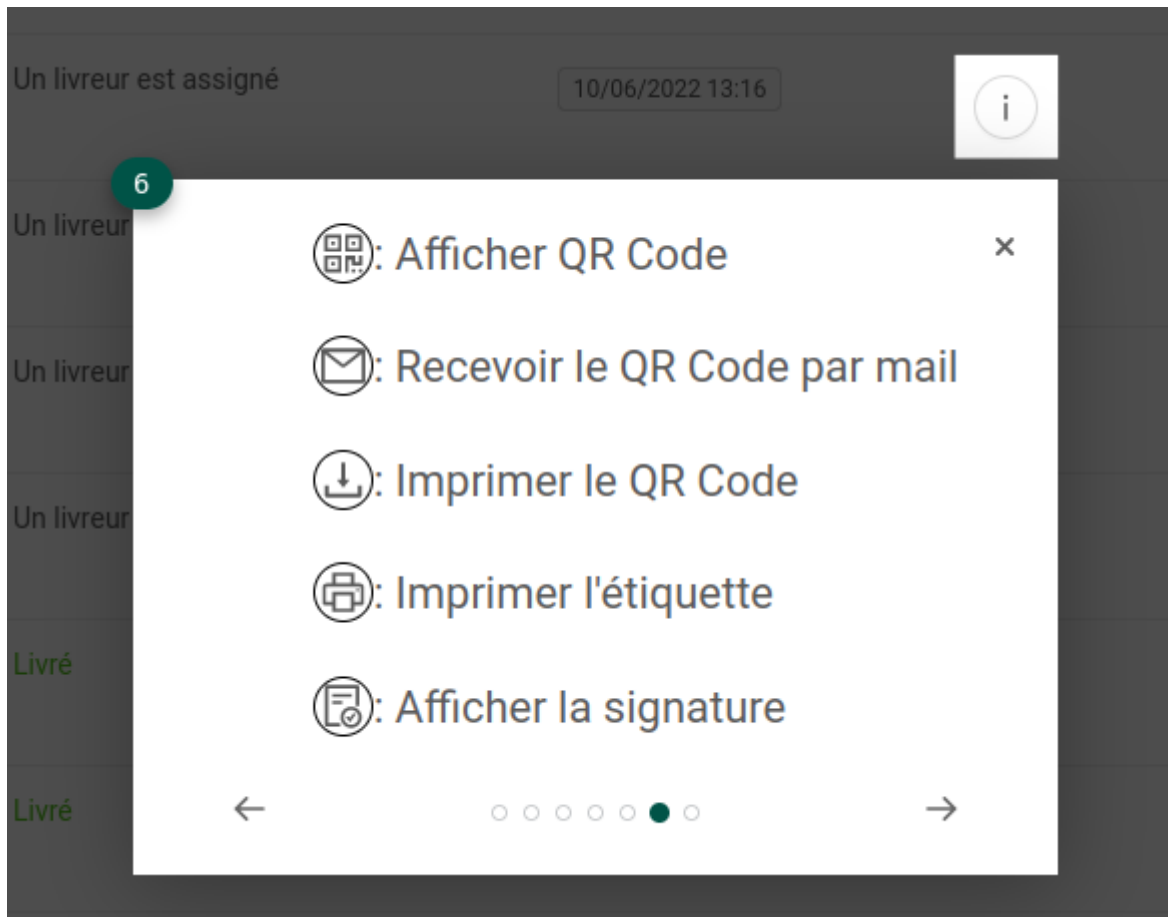
Ici on explique toutes les opérations possibles contenues dans le bouton information surligné en blanc.

```
<TourProvider
  steps={steps}
  startAt={0}
  styles={{
    popover:base =>({
      ...base,
      maxWidth:'550px',
      padding:"20px 50px",
      '--reactour-accent': '#005347'
    })
  }}
  onClickMask=(({ setCurrentStep, currentStep, setIsOpen }) => {
    if (currentStep === steps.length - 1) {
      setIsOpen(false)
    }
    setCurrentStep(s => (s === steps.length - 1 ? 0 : s + 1))
  })
>
```

```
{children}  
</TourProvider>
```

tourProvider.jsx

Nous pouvons donc intervenir avec Reactour au sein d'une application multicouche, les différents fichiers ou couches interviennent entre elle à l'aide des importations et exportations de fichiers. Ici nous avons séparé l'application en couche distincte pour renforcer la maintenance et l'évolutivité de l'application.



Ici on explique toutes les opérations possibles contenues dans le bouton information surligné en blanc.

Cette activité m'a permis d'acquérir la compétence : **Développer la partie front-end d'une interface utilisateur**

2. Précisez les moyens utilisés :

React et javascript pour le développement de mes composants,
NPM pour l'installation de la librairie
Gitlab pour le versioning
Visual Studio IDE
Gantt pour l'organisation

3. Avec qui avez-vous travaillé ?

sous la supervision de mon responsable thomas BOYON

4. Contexte

Nom de l'entreprise, organisme ou association ▶ Deki

Chantier, atelier, service ▶ Développement

Période d'exercice ▶ Du : 5 septembre 2021 au : 31 aout 2022

5. Informations complémentaires (facultatif)

Activité-type 1

Concevoir et développer des composants d'interface utilisateur en intégrant les recommandations de sécurité

Exemple n°3 ▶ Génération d'un fichier pdf en utilisant ejs (Alternance Deki)

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Dans le cadre de mon alternance j'ai pu effectuer la modification d'un aperçu pdf destiné à nos clients. Ma tâche consistait à rajouter des données existantes en base de données sur un fichier PDF généré en cliquant. J'ai pour cette tâche employé le langage EJS.

Il permet d'injecter des données à l'intérieur d'un modèle HTML côté client et produire le HTML final à l'intérieur d'une balise `<%= %>`.

J'ai donc dû agir sur un fichier existant nommé template.ejs et créer un aperçu regroupant toutes les informations voulues par ma responsable.

```
<tr class="row-step">
  <th><p><strong>Total TTC à payer</strong></p></th>
  <th>
    <p><strong><%= priceTTC %> €</strong></p>
  </th>
```

Ici on imprime la donnée Prix TTC

```
<div class="footer-div">
  <% if (notes && notes.length > 0) { %>
    <p class="footer-text"><%= notes %></p>
  <% } %>
</div>
```

Ici on imprime la donnée "notes" qui correspond aux informations supplémentaires pour la livraison ou la commande par l'utilisateur.

Test
 12 Rue Mirès, 13003 Marseille, France
 Boyon Thomas 2
 +33679763477
 -


 contact@deki.team

Justificatif adressé à
MELLON DATA 2 MODIFY
 Gaetan Giordano modify

gaetangiordano@orange.eu
 -


 Point de départ
Test modify
 Wallez Jeremy
 19 Rue Mirès
MELLON DATA
 Gaetan Giordano
 18 Rue Mirès


 Point d'arrivée
MELLON DATA 2 MODIFY
 Gaetan Giordano modify
 13 Rue Mirès

Livraison le **13/01/2022**
 Entre **17h20** et **18h20**

Commentaires :Tout déposer
 Prix total TTC : Na€

Livraison terminée à 16h51
 Signature :


Scanner le QRCode pour accéder au suivi en temps réel de la livraison.

cette activité m'a permis de valider la compétence suivante : **Développer la partie front-end d'une interface utilisateur et développer des composants métiers**

2. Précisez les moyens utilisés :

EJS pour l'impression des informations dans le PDF,
 React et javascript pour le développement de mes composants,
 NPM pour l'installation de la librairie
 Gitlab pour le versioning
 Visual Studio IDE

Gantt pour l'organisation

3. Avec qui avez-vous travaillé ?

Mon chef de projet Thomas BOYON

4. Contexte

Nom de l'entreprise, organisme ou association ▶ *Deki*

Chantier, atelier, service ▶ *Développement*

Période d'exercice ▶ Du : *5 septembre 2021* au : *31 août 2022*

5. Informations complémentaires (facultatif)

**Activité-type n° 2 : Concevoir et développer la persistance des données
en intégrant les recommandations de sécurité**

Activité-type 2

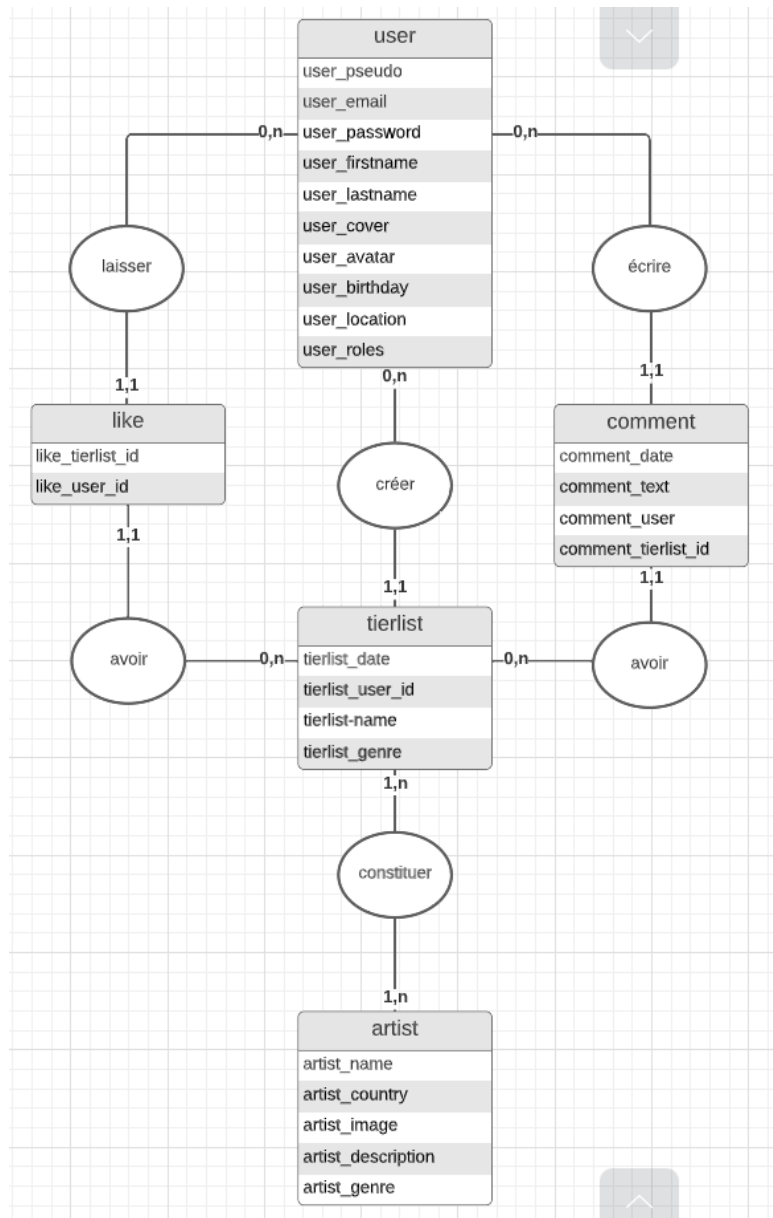
Exemple n° 1 ▶

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

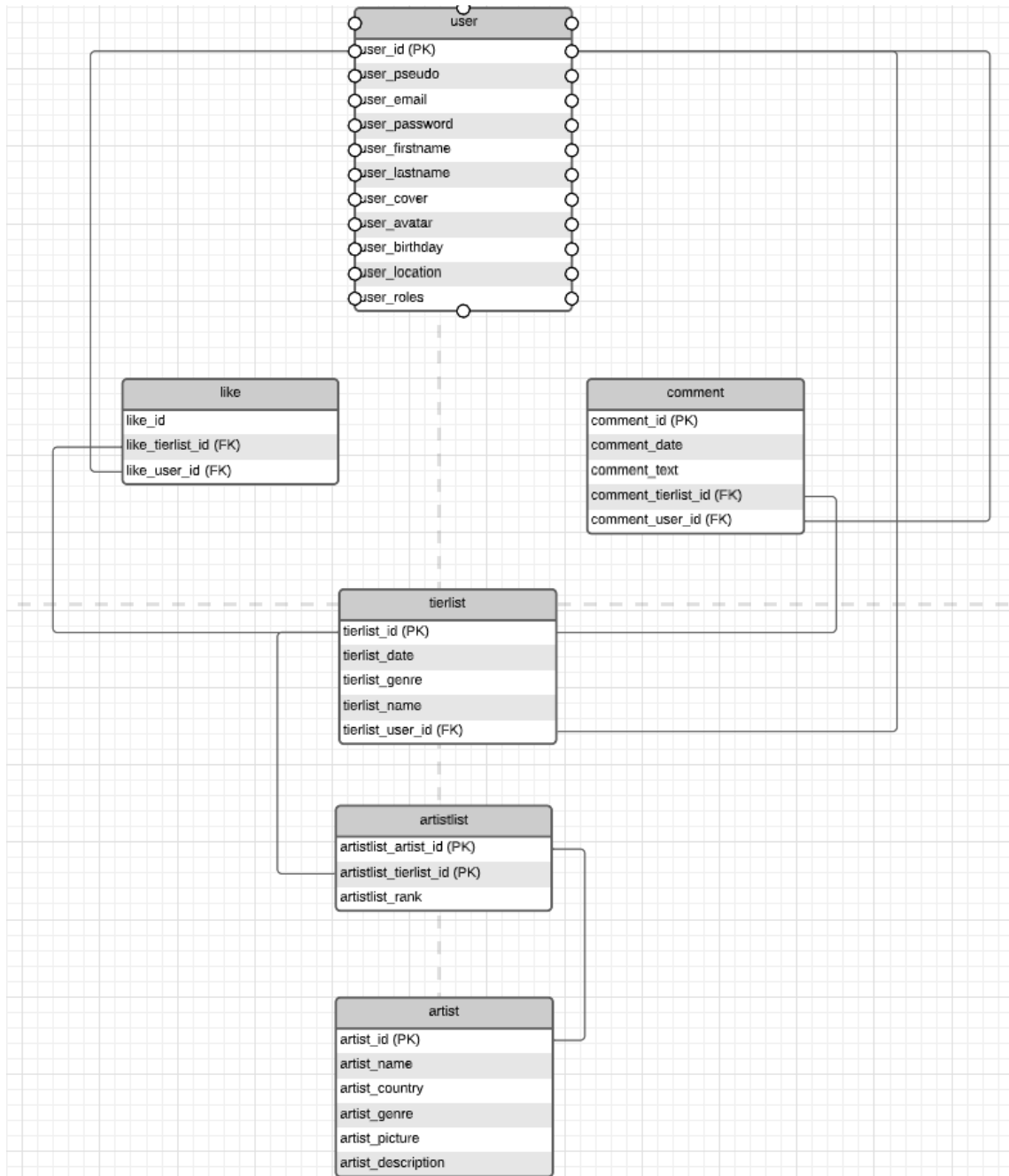
En compagnie de mes deux compagnons de formation j'ai établi la base de données relationnelle de notre application mobile. Nous avons pour cela listé les informations qui nous paraissaient indispensables au bon fonctionnement de l'application.

On distingue deux grandes catégories de tables : la table User qui gère les infos de l'utilisateur et la table Tier List. Cette base de données ne contient pas de table d'association étant donné qu'il n'y a pas de cardinalité plusieurs-à-plusieurs entre deux objets

MCD :



MLD :



Cette activité me permet de valider la compétence suivante : **Concevoir une base de données**

2. Précisez les moyens utilisés :

LUCIDCHART pour schématiser ces entités, mes tables et ses attributs.

3. Avec qui avez-vous travaillé ?

Emma LAPREVOTE , Walid BRICHE

4. Contexte

Nom de l'entreprise, organisme ou association ▶ *La Plateforme*

Chantier, atelier, service ▶ *Application de fin d'année*

Période d'exercice ▶ Du : *Du 5 septembre 2021* au : *31 août 2022*

5. Informations complémentaires (facultatif)

Concevoir et développer une application multicouche

Activité-type 3 répartie en intégrant les recommandations de sécurité

Exemple n° 1 ▶ Déploiement d'un site web

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Il était nécessaire d'effectuer un projet professionnel au sein de ma formation à la plateforme. J'ai eu la chance de l'effectuer pour mon tatoueur et de m'occuper de son hébergement.

Pour déployer le site j'ai décidé de m'orienter vers l'hébergeur français OVH qui est une référence dans son domaine.

J'ai décidé d'utiliser le logiciel FileZilla : un client FTP, FTPS et SFTP multiplateforme rapide et fiable avec de nombreuses fonctionnalités utiles et une interface utilisateur graphique intuitive.

Après que le client s'est occupé de souscrire un abonnement auprès de l'hébergeur choisi, la première étape consiste à se connecter au serveur FTP pour ensuite pouvoir transférer les fichiers locaux sur le serveur distant.

Il sera donc nécessaire de renseigner les informations fournies par l'hébergeur après souscription telles que :

- Serveur FTP : Il s'agit de l'adresse de serveur permettant d'accéder à l'espace de stockage.
- Login FTP : Il s'agit de l'identifiant vous permettant d'accéder à l'espace de stockage.
- Port de connexion : Généralement complété automatiquement par le logiciel.

L'étape maîtresse de l'hébergement est transfert de fichier, pour réaliser le transfert de vos fichiers en FTP, il faut simplement les sélectionner et réaliser un glisser-déposer des fichiers de la fenêtre de gauche (les fichiers en local) vers la fenêtre de droite (l'espace d'hébergement).

Une fois cette action réalisée, les fichiers vont automatiquement se mettre en file d'attente pour être déposés sur le serveur. Une vue sur la file d'attente est disponible.

- Les fichiers en attente d'être déposés sur le serveur distant encore présents dans la file d'attente.
- Les fichiers pour lesquels le transfert à échoué.
- Les fichiers pour lesquels le transfert est réussi sur l'hébergement distant

2. Précisez les moyens utilisés :

OVH pour l'hébergement
FileZilla pour mon serveur FTP
NPM pour l'installation de la librairie
Github pour le versioning
Visual Studio IDE

3. Avec qui avez-vous travaillé ?

4. Contexte

Nom de l'entreprise, organisme ou association ▶ *L'ornithorynque*

Chantier, atelier, service ▶ *Déploiement de site web*

Période d'exercice ▶ Du : *15 mai* au : *25 juillet*

5. Informations complémentaires (facultatif)

Activité-type 3 Cliquez ici pour entrer l'intitulé de l'activité

Exemple n° 2 - Ré-attribution automatique des courses Deki

1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Dans le cadre de mon contrat d'apprentissage j'ai pu participer à la conception d'un outil back-end. En effet il s'agissait d'un système de ré-assignation automatique des courses si l'entreprise de livraison préalablement assignée n'a pas accepté la course dans le temps imparti.

Le CronJob de ré-attribution a été réalisé dans le cadre de mon alternance à Deki dans le contexte du développement d'un back-office pour que nous puissions gagner du temps et être plus prévoyant. Il a pour but de régler une problématique de dispatcher qui n'accepte pas ou ne refuse pas sa course pour différentes raisons

Pour répondre à cette problématique et aux attentes du client nous avons donc décidé de mettre en place un outil de ré-assignation automatique des courses

Le CronJob a pour but de régler une problématique de cache lors de l'attribution des courses.

Une tâche cron ou CronJob est une tâche exécutée automatiquement dans un système d'exploitation. Afin d'exécuter ces tâches, le système fournit régulièrement des commandes. Chaque tâche cron est composée de trois éléments :

- Un script à exécuter ;
- Une commande qui exécute régulièrement le script ;
- Une action ou le log du script.

Pour qu'une tâche cron puisse exécuter ce qu'on attend d'elle, il faut tout d'abord fournir une commande. Les tâches cron sont, par principe, au repos, et attendent de recevoir une commande : ou bien directement du système d'exploitation, ou bien d'un autre ordinateur en réseau. La commande est transmise par le biais d'un fichier cron (un simple fichier texte), qui planifie la tâche pour un instant donné.

```

1  const services = require("app/services/course");
2  const queryAPI = require("app/helpers/queryBuilder");
3  const mailer = require("app/services/mail");
4  const assignator = require("app/helpers/autoAssign");
5
6  module.exports.assign = async () => {
7    try {
8      const coursesWaiting = await queryAPI.coursesWaiting();
9      for (let i = 0, l = coursesWaiting.length; i < l; i++) {
10         const course = coursesWaiting[i];
11         if (course.team_id) {
12             const teamId = course.team_id;
13             await services.deny({
14                 course_id: course.id,
15                 team_id: course.team_id,
16             });
17             const mail = await queryAPI.getMailByTeamID(teamId);
18             mailer.sendMail([
19                 {
20                     type: "re-assign-task",
21                     toSend: {
22                         email: mail,
23                         subject: "Course réattribué",
24                     },
25                 },
26             ]);
27         } else {
28             const merchant = await queryAPI.getMerchantByCourseId(course.id);
29             await assignator.assign([course.id], merchant);
30         }
31     }
32     return [0, { success: true }];
33 } catch (err) {
34     console.log(err);
35     throw err;
36 }
37 };
38

```

- Objectif => créer une ré-attribution automatique
- Le cron va déclencher un appel API qui va collecter les courses avec un délai écoulé afin de ne pas confier une course à cette même société
- Relever l'id de la société qui n'a pas accepté la course à temps
- Développer un appel vers l'outil toutes les cinq minutes

Assign

Modify
Pending events
Processed events
Invocation logs

Name ⓘ

Webhook ⓘ

Cron Schedule ⓘ

Build a cron expression

Frequently used crons

Payload ⓘ

1

{ }

> Advanced

Save
Delete

Interface graphique d’Hasura notre API qui permet de visualiser notre base de données et de créer des CRON JOB. Dans cette capture d’écran c’est la route “assign” va être exécutée.

✓	Thomas Boyon	Collecte	Selectionnez un coursier	Non attribué	30/03/2022	13:34	→	13:34	0.1	-	-
✗	GAASWATT	Livraison	Selectionnez un coursier	Non attribué	31/03/2022	13:34	→	13:34	0.1	0.57/89.14	-

cette activité m’a permis d’acquérir la compétence : **développer la partie back-end d’une interface utilisateur web, construire une application organisée en couches.**

2. Précisez les moyens utilisés :

React et javascript pour le développement de mes composants,
Hasura pour l'API en GraphQL
NPM pour l'installation de la librairie
Gitlab pour le versioning
Visual Studio IDE
Gantt pour l'organisation

3. Avec qui avez-vous travaillé ?

Thomas BOYON Chef de projet et Loup CARPENTIER Développeur

4. Contexte

Nom de l'entreprise, organisme ou association ▶ *Deki*

Chantier, atelier, service ▶ *Ré-attribution des courses*

Période d'exercice ▶ Du : *5 septembre 2021* au : *31 aout 2022*

5. Informations complémentaires (facultatif)

Déclaration sur l'honneur

Je soussigné(e) [prénom et nom] **Fabien Ponzio** ,
déclare sur l'honneur que les renseignements fournis dans ce dossier sont exacts et que je suis
l'auteur(e) des réalisations jointes.

Fait à **Marseille** le **24 juin 2022**
pour faire valoir ce que de droit.

Signature :

Documents illustrant la pratique professionnelle

(facultatif)

Intitulé

Cliquez ici pour taper du texte.

ANNEXES

(Si le RC le prévoit)