



MINISTÈRE CHARGÉ
DE L'EMPLOI

DOSSIER PROFESSIONNEL (DP)

Nom de naissance BRUNET
Nom d'usage
Prénom Bastien
Adresse 24 rue Charles Ponçy
83 000 TOULON

Titre professionnel visé

Développeur web et web mobile

MODALITÉ D'ACCÈS :

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

DOSSIER PROFESSIONNEL (DP)

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>

DOSSIER PROFESSIONNEL (DP)

Sommaire

Exemples de pratique professionnelle

Intitulé de l'activité-type n° 1	p. 5
□ Intitulé de l'exemple n° 1	p...
□ Intitulé de l'exemple n° 2	p...
□ Intitulé de l'exemple n° 3	p...
Intitulé de l'activité-type n° 2	p.
□ Intitulé de l'exemple n° 1	p...
□ Intitulé de l'exemple n° 2	p...
□ Intitulé de l'exemple n° 3	p...
Intitulé de l'activité-type n° 3	p.
□ Intitulé de l'exemple n° 1	p...
□ Intitulé de l'exemple n° 2	p...
□ Intitulé de l'exemple n° 3	p...
Intitulé de l'activité-type n° 4	p.
□ Intitulé de l'exemple n° 1	p...
□ Intitulé de l'exemple n° 2	p...
□ Intitulé de l'exemple n° 3	p...
Titres, diplômes, CQP, attestations de formation (facultatif)	p.
Déclaration sur l'honneur	p.
Documents illustrant la pratique professionnelle (facultatif)	p.
Annexes (Si le RC le prévoit)	p.

EXEMPLES DE PRATIQUE PROFESSIONNELLE

Activité-type 1 Développer la partie front-end d'une application web ou web mobile en intégrant les recommandations de sécurité.

CP 1 ► Installer et configurer son environnement de travail en fonction du projet web ou web mobile

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

Dans le cadre de ma formation, j'ai réalisé deux projets front-end :

- La **maquette HTML/CSS d'une page d'accueil pour un site de voyage**
- La **création de la page principale de mon portfolio personnel**

Pour pouvoir travailler efficacement sur ces projets, j'ai installé et configuré un environnement de développement adapté à chaque besoin en installant en premier lieu l'éditeur de code VS code. J'ai ensuite commencé par structurer les dossiers de travail (création des répertoires `css`, `images`, `assets`, etc.) et mis en place un système de versionnage avec Git.

J'ai également pris en compte les bonnes pratiques en matière de sécurité front-end de base, j'ai veillé à ne pas inclure d'informations sensibles dans les fichiers sources; en ne jamais inclure dans le front-end de clés API, d'identifiants de connexion ou mots de passe, emails, tokens, infos personnelles. Ces projets ont été réalisés dans un **contexte pédagogique**, avec des consignes précises fournies par les formateurs.

2. Précisez les moyens utilisés :

Éditeur de code : Visual Studio Code

- **Extensions utilisées** : Live Server (pour un rechargement automatique), Prettier (pour le formatage du code), HTML/CSS Support
- **Navigateurs pour les tests** : Google Chrome et Mozilla Firefox
- **Outils de versionnage** : Git (en local)
- **Organisation du projet** : arborescence de fichiers claire et modulaire (`index.html`, `style.css`, dossier `assets`) permettant la séparation des fichiers de code rendant le projet plus lisible et facilitant la recherche d'une image, d'une police etc ...)
- **Normes et bonnes pratiques HTML/CSS** : indentation, commentaires, accessibilité de base, compatibilité multi-navigateurs.

3. Avec qui avez-vous travaillé ?

J'ai travaillé **seul** tout au long de ces projets. Ces réalisations ont été effectuées en **autonomie**, dans le cadre d'exercices pratiques proposés durant la formation.

4. Contexte

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



Chantier, atelier, service ▶ *Maquettes en HTML et CSS*

Période d'exercice ▶ Du : *01/04/2025* au : *30/04/2025*

5. Informations complémentaires (facultatif)

Activité-type 1

Développer la partie front-end d'une application web ou web mobile sécurisée

CP 2 ▶

Maquetter des interfaces utilisateur web ou web mobile,

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

(Projet concerné : *Page principale de mon portfolio*)

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

Dans le cadre de la création de mon **portfolio personnel**, j'ai réalisé une **maquette d'interface utilisateur** à l'aide du langage HTML et CSS, dans une approche centrée sur l'expérience utilisateur et la clarté visuelle.

Mon objectif était de concevoir une structure d'interface claire, lisible et facile à prendre en main, en respectant les grands principes de la conception d'interface : hiérarchisation des contenus, accessibilité, et adaptabilité visuelle.

La maquette a été pensée d'abord en version **desktop**, puis j'ai décliné l'organisation des blocs en vue d'une intégration responsive. Ce travail s'est fait dans un cadre **autonome**, sur la base de consignes pédagogiques fournies.

2. Précisez les moyens utilisés :

Outils de conception : HTML5, CSS3 (code directement écrit, sans logiciel graphique de type Figma)

- Méthodologie : réflexion sur la disposition des sections (accueil, présentation, compétences, projets, contact)
- Mise en page simulée via une structure sémantique HTML : header, main, section, footer
- Outils : Visual Studio Code + extensions (Live Server, Prettier)
- Organisation des fichiers : création d'un dossier assets/ contenant les images, icônes, polices éventuelles

3. Avec qui avez-vous travaillé ?

J'ai travaillé **seul** tout au long de ce projet, en autonomie complète.

4. Contexte

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

Chantier, atelier, service *Page principale de mon portfolio*

Période d'exercice *Du : 02/04/2025 au : 30/04/2025*

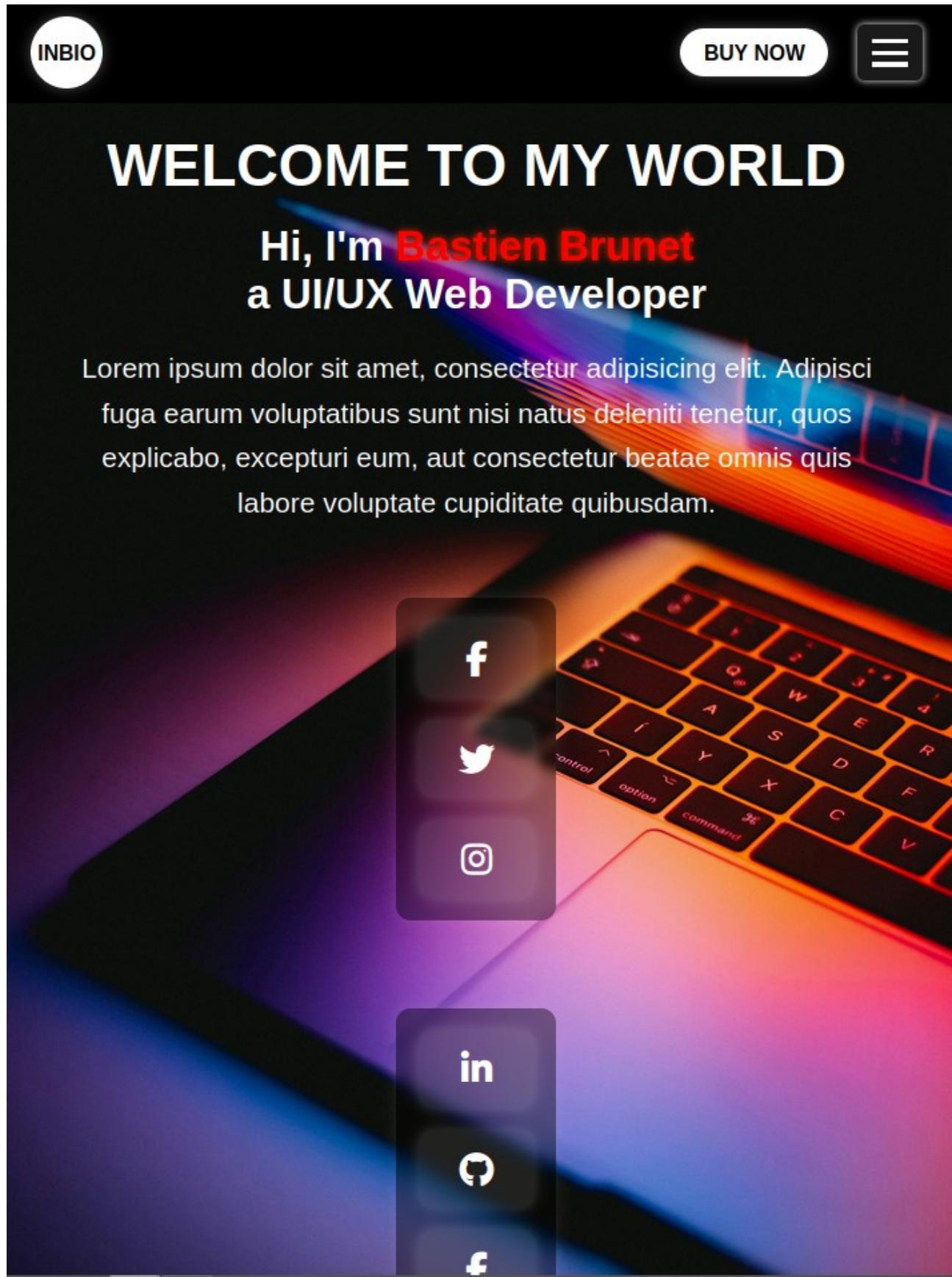
5. Informations complémentaires (facultatif)

The image shows a dark-themed website template. At the top left is a circular logo with the word "INBIO". To its right are navigation links: HOME, FEATURES, PORTFOLIO, and RESUME. On the far right is a white "BUY NOW" button. The main title "WELCOME TO MY WORLD" is displayed in large, bold, white letters. Below it, a bio reads "Hi, I'm **Bastien Brunet** a UI/UX Web Developer". A paragraph of placeholder text follows. At the bottom left is a social media sharing bar with icons for Facebook, Twitter, and Instagram. The background features a close-up of a laptop keyboard with red backlighting.

WELCOME TO MY WORLD

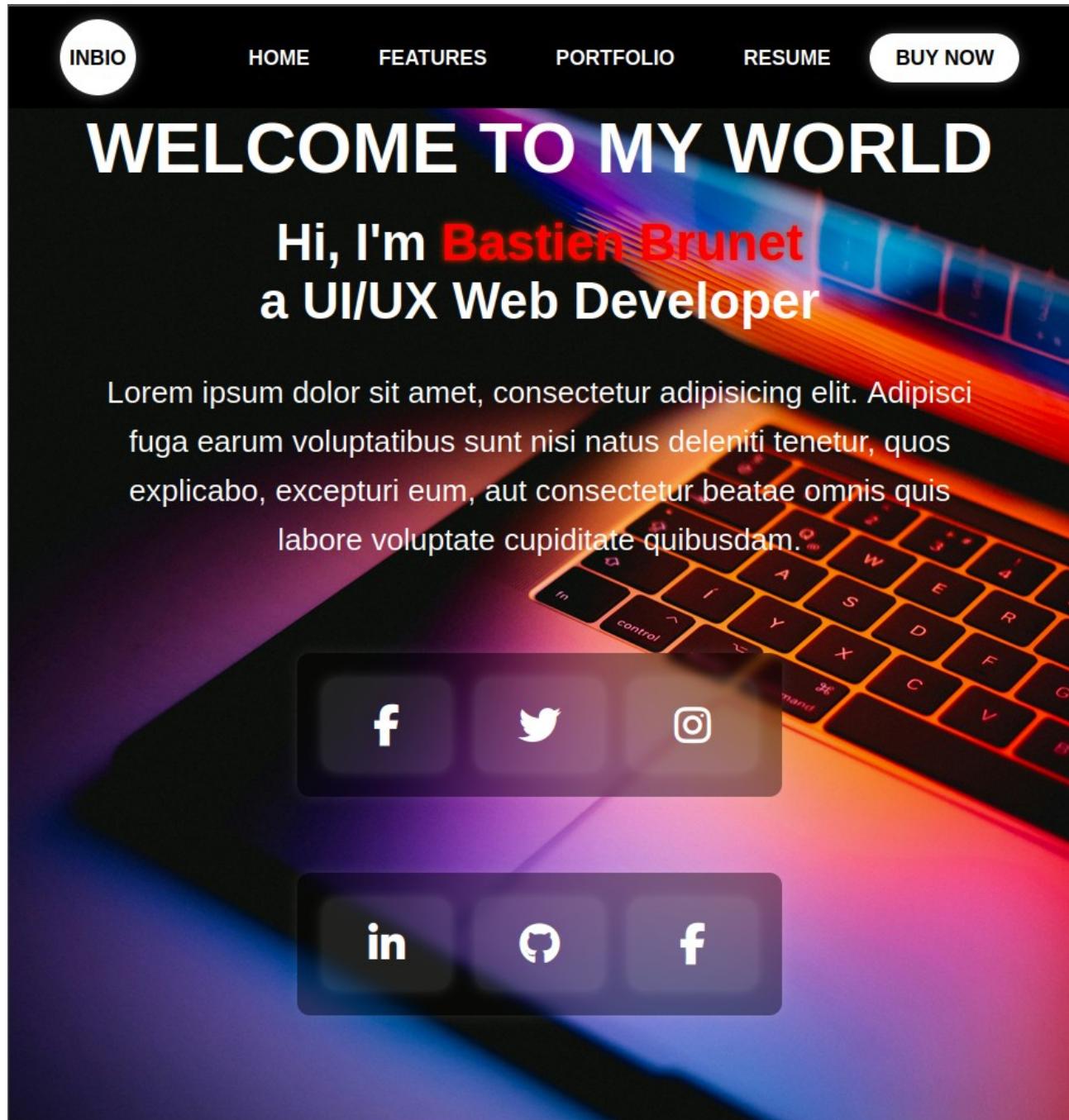
Hi, I'm **Bastien Brunet**
a UI/UX Web Developer

Lore ipsum dolor sit amet, consectetur adipisicing elit. Adipisci
fuga earum voluptatibus sunt nisi natus deleniti tenetur, quo
explicabo, excepturi eum, aut consectetur beatae omnis quis
labore voluptate cupiditate quibusdam.



Ci-dessus, capture d'écran en mode "mobile" (avec menu burger) et capture d'écran en mode "desktop".

Capture d'écran en mode "tablette" :



Activité-type 1 Développer la partie front-end d'une application web ou web mobile en intégrant les recommandations de sécurité.

CP 3 ▶ Réaliser des interfaces utilisateurs statiques web et web mobile,

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

Pour ces deux projets, j'ai intégré des interfaces statiques à partir des maquettes réalisées ou fournies.

Mon travail a consisté à transformer les maquettes en **pages HTML/CSS fonctionnelles, adaptables (responsive) et accessibles**, sans interaction JavaScript.

J'ai utilisé **Flexbox** et **Grid CSS** pour organiser les éléments, ajouté des **media queries** pour l'adaptabilité mobile, et respecté les règles de sémantique HTML pour renforcer l'accessibilité. J'ai également veillé à la compatibilité entre navigateurs (Chrome et Firefox) et respecté les recommandations de base en matière de sécurité front-end en évitant d'inclure adresse e-mail en clair dans le HTML (risque de spam), des liens vers des pages privées ou d'administration, des clés API ou jetons d'authentification (même de test), pas d'infos sensibles en clair dans le code. Le travail a été mené en autonomie, dans un cadre pédagogique, sur des consignes précises.

2. Précisez les moyens utilisés :

Languages : HTML5, CSS3

- **Techniques de mise en page** : Flexbox, Grid CSS
- **Adaptabilité responsive** : Media queries (@media screen and (max-width: . . .)), unités relatives (em, %, vw, vh) et insertion d'un menu burger pour le mode mobile.
- **Structure de projet** : Arborescence organisée, avec un dossier assets/ pour les ressources (images, icônes, polices)
- **Outils** :
- Visual Studio Code + Live Server
- Navigateurs Chrome et Firefox pour les tests
- Outils développeur (DevTools) pour tester les résolutions mobiles

- **Bonnes pratiques :**
- HTML sémantique (`nav`, `main`, `section`, `footer`)
- Respect des standards d'accessibilité
- Pas d'informations sensibles dans le front-end
- Respect des conventions de nommage (classes CSS claires et logiques)

3. Avec qui avez-vous travaillé ?

J'ai travaillé **seul** sur l'ensemble de ces projets. Les interfaces ont été conçues et codées en **autonomie**, en respectant les maquettes et les consignes de mise en page fournies dans le cadre de la formation.

4. Contexte

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

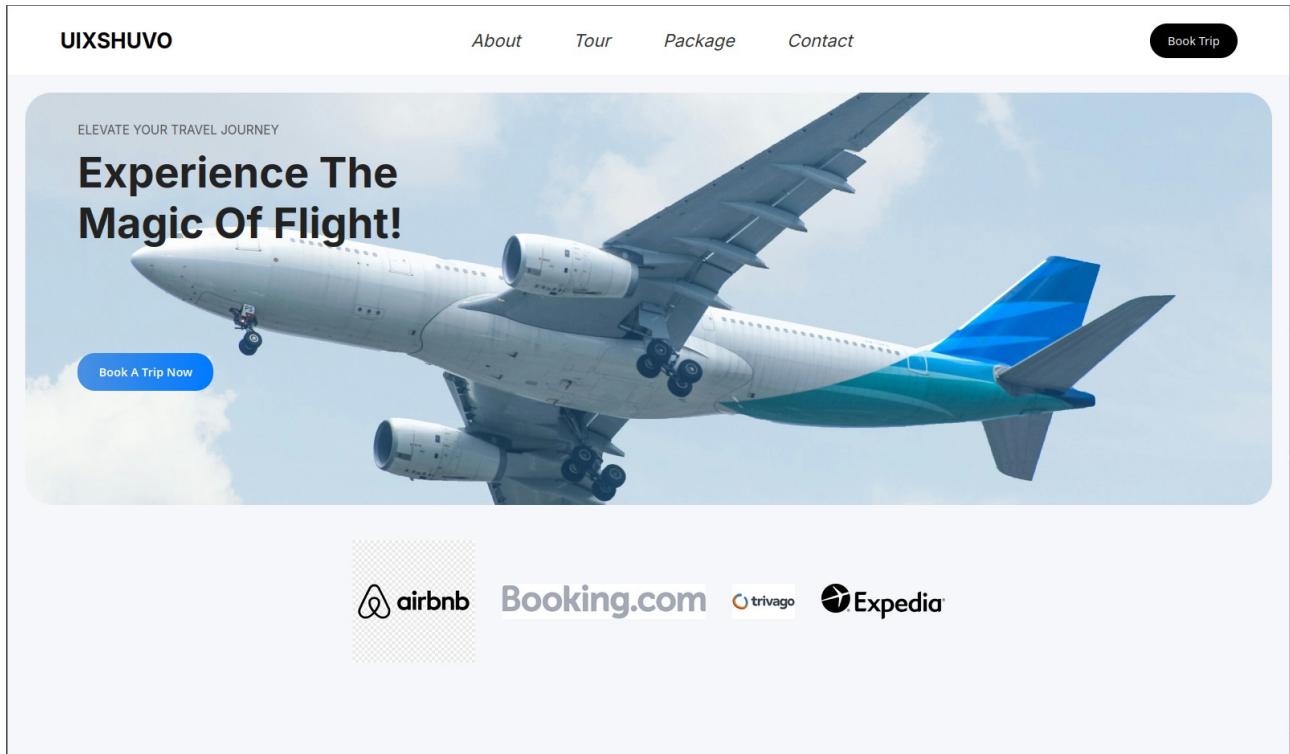
▶

Chantier, atelier, service ▶ *Maquettes statiques site de voyage en HTML / CSS*

Période d'exercice ▶ Du : **02/04/2025** au : **30/04/2025**

5. Informations complémentaires (facultatif)

Voici les captures d'écran desktop et responsive :





Booking.com



Expedia

Popular Destination

Unleash Your Wanderlust With SkyWings



Forest Wild Life



Forest Wild Life



Forest Wild Life

Journey To The Skies Made Simple!

Travelling is a wonderful way to explore new places, learn about different cultures, and gain unique experiences.

Find Your Destination

Book A Ticket

Pay & Start Journey

Forest Wild Life

Forest Wild Life

Forest Wild Life

Journey To The Skies Made Simple!

Travelling is a wonderful way to explore new places, learn about different cultures, and gain unique experiences.

Find Your Destination

Book A Ticket

Pay & Start Journey

UNLEASH WANDERLUST WITH SKYWINGS



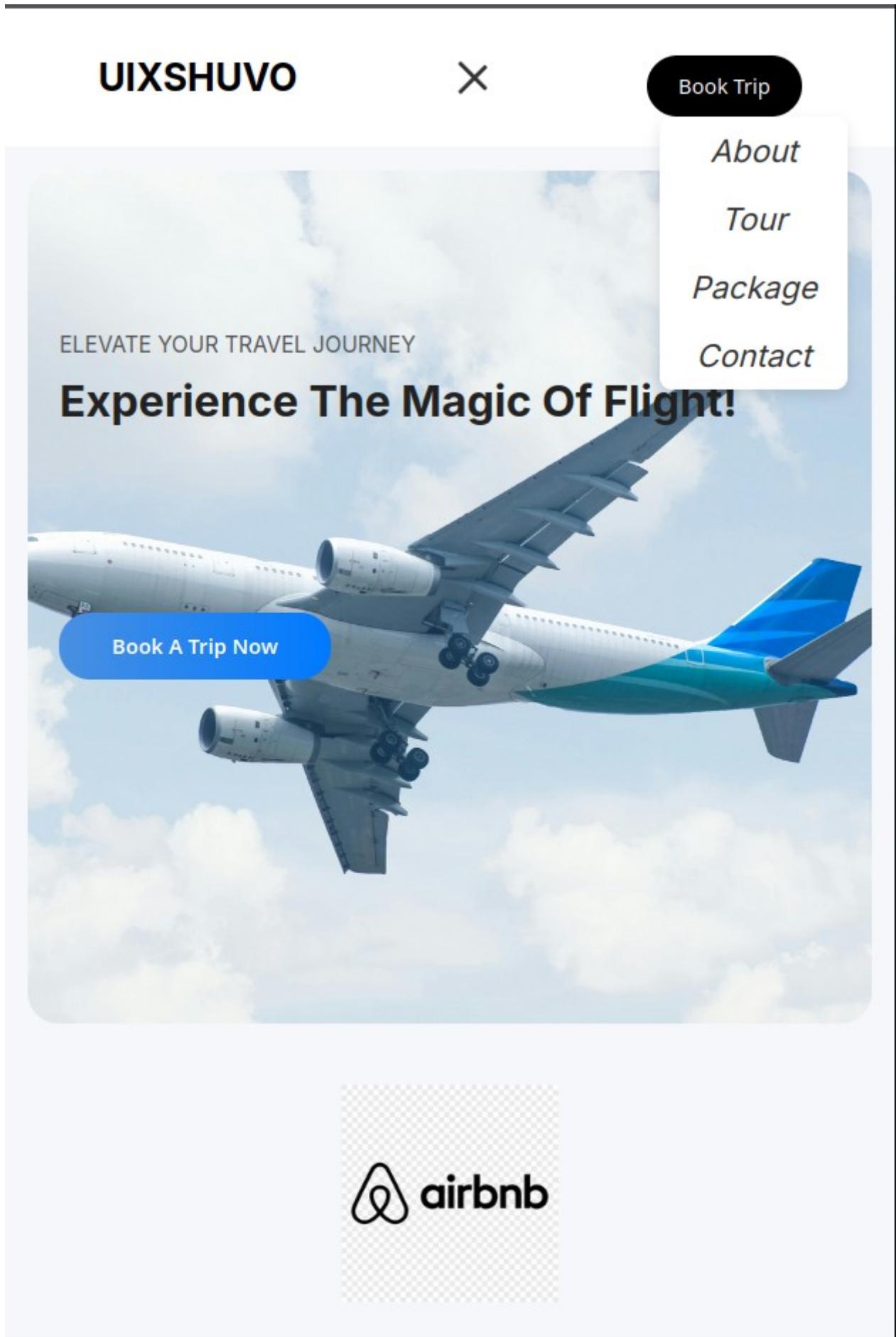
Travelling is a wonderful way to explore new places, learn about different cultures, and gain unique experiences.

20% OFF

Till 28 September, 2023

[Book A Flight Now →](#)

Et voici ci dessous les captures d'écran mobile :





UNLEASH WANDERLUST WITH SKYWINGS

Travelling is a wonderful way to explore new places, learn about different cultures, and gain unique experiences.

20% OFF

Till 28 September, 2023

Book A Flight

Now →



Booking.com



Popular Destination

Unleash Your Wanderlust With SkyWings



Forest Wild Life



UIXSHUVO



Book Trip

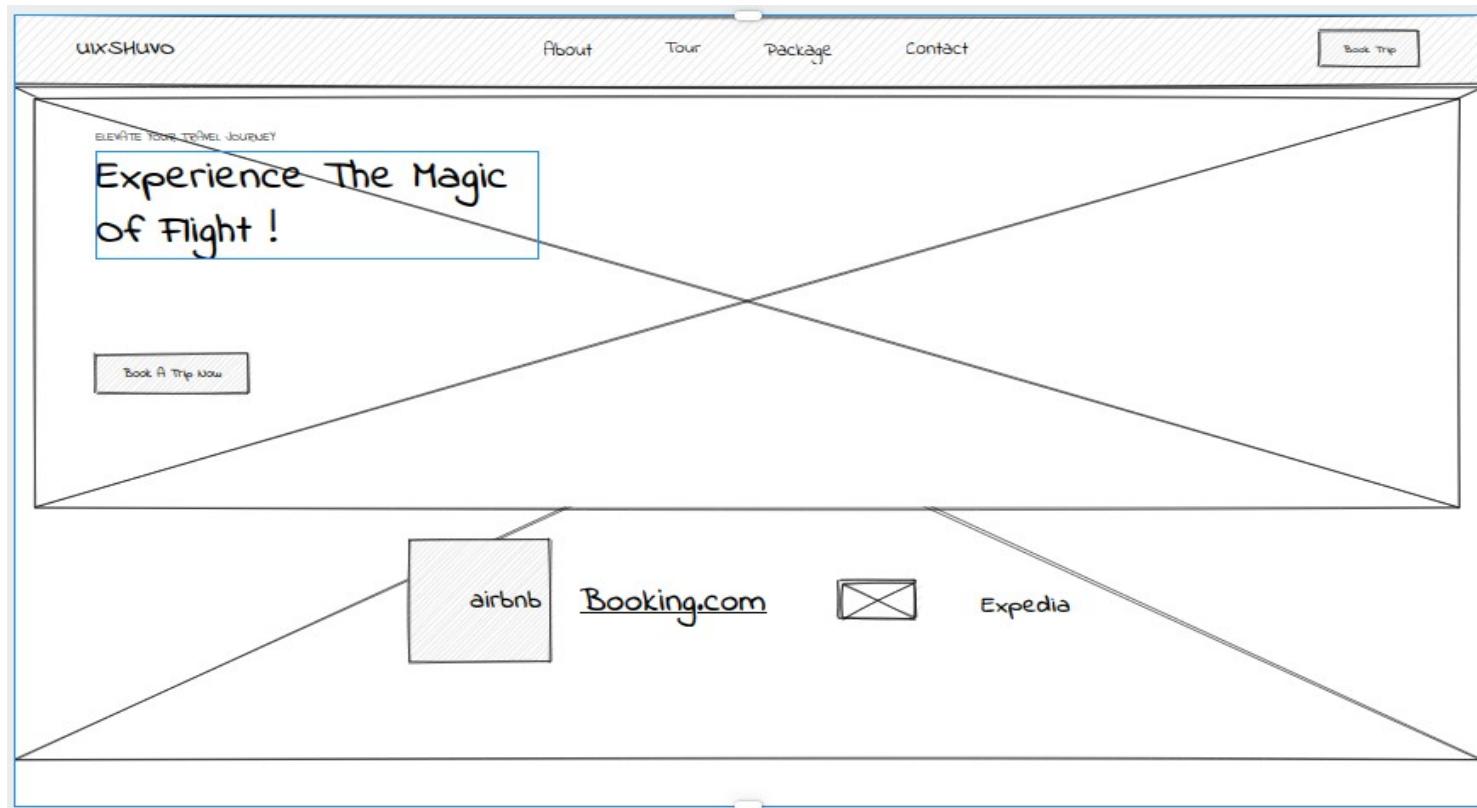
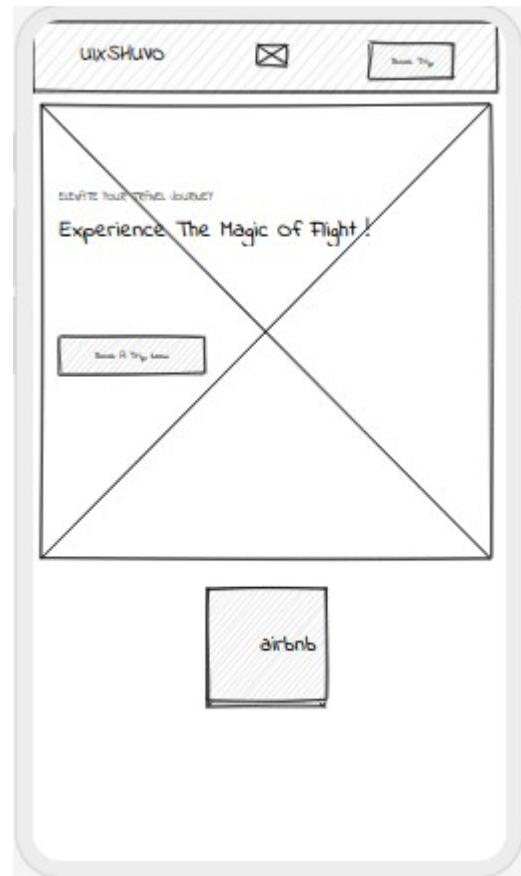
ELEVATE YOUR TRAVEL JOURNEY

Experience The Magic Of Flight!

Book A Trip Now



Voici quelques illustrations ci dessous des wireframes de cette maquette, d'abord en mode mobile puis desktop :



airbnb

Booking.com

trivago

Expedia

Popular Destination

Unleash Your wanderlust with Skywings

Forest Wild Life

Journey To The Skies Made Simple !

Travelling is a wonderful way to explore new places, learn about different cultures, and gain unique experiences .

Find Your Destination

Book A Ticket

Pay & Start Journey

The wireframe shows a top navigation bar with logos for Airbnb, Booking.com, trivago, and Expedia. Below this is a section titled "Popular Destination" with a sub-instruction "Unleash Your wanderlust with Skywings". There are three large, empty rectangular boxes, with the text "Forest wild life" centered below the middle one. A central heading "Journey To The Skies Made Simple!" is followed by a descriptive text: "Travelling is a wonderful way to explore new places, learn about different cultures, and gain unique experiences." At the bottom are three buttons: "Find Your Destination", "Book A Ticket", and "Pay & Start Journey".

airbnb

Booking.com

trivago

Expedia

Popular Destination

Unleash Your wanderlust with Skywings

Forest wild life

Journey To The Skies Made Simple !

Travelling is a wonderful way to explore new places, learn about different cultures, and gain unique experiences .

Find Your Destination

Book A Ticket

Pay & Start Journey

This version of the wireframe includes a large, semi-transparent rectangular overlay at the top containing three repeated "Forest wild life" placeholder boxes. Below this is the same travel-related text and button structure as the first wireframe. A prominent feature is a large, semi-transparent rectangular box in the center containing the text "UNLEASH WANDERLUST WITH SKYWINGS" above a smaller text block: "Travelling is a wonderful way to explore new places , learn about different cultures, and gain unique experiences ." Below this text is a small note: "20% OFF TIL 28 September, 2023". At the very bottom is a single button: "Book A Flight Now →".

Forest wild life

Forest wild life

Forest wild life

Journey To The Skies Made Simple !

Travelling is a wonderful way to explore new places, learn about different cultures, and gain unique experiences .

Find Your Destination

Book A Ticket

Pay & Start Journey

UNLEASH WANDERLUST
WITH SKYWINGS

Travelling is a wonderful way to explore new places , learn about different cultures, and gain unique experiences .

20% OFF TIL 28
September, 2023

Book A Flight Now →

Activité-type 1 Développer la partie front-end d'une application web ou web mobile en intégrant les recommandations de sécurité.

CP 4 ► Développer la partie dynamique des interfaces utilisateur web ou web mobile

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

Dans le cadre de ce projet, j'ai conçu et développé une **application web de gestion de tâches** (*Todo List*) permettant à l'utilisateur d'ajouter, de marquer comme terminée, de supprimer et de visualiser des tâches en temps réel. L'objectif était de démontrer la capacité à rendre une interface web dynamique et interactive, sans utiliser de frameworks externes.

J'ai commencé par structurer l'interface en HTML et CSS, en m'assurant qu'elle soit à la fois esthétique et responsive. Ensuite, j'ai implémenté toute la **logique JavaScript** en programmation orientée objet à l'aide d'une classe `TodoApp`.

Les tâches ont été réalisées dans un **environnement de développement local**, en respectant une démarche de test à chaque ajout de fonctionnalité.

Enfin, j'ai intégré la **persistance des données** via le `localStorage`, permettant à l'utilisateur de retrouver sa liste de tâches même après fermeture du navigateur.

2. Précisez les moyens utilisés :

Pour ce projet, j'ai utilisé les **langages du web** suivants :

- **HTML5** : pour la structure de la page.
- **CSS3** : pour la mise en forme, la responsivité et les animations.
- **JavaScript (vanilla JS)** : pour la logique dynamique, les événements, le DOM, la manipulation des données et leur stockage local.

J'ai utilisé :

- **Visual Studio Code** comme éditeur de code.
- Le navigateur **Google Chrome** pour le test et le débogage.
- L'outil **DevTools** pour inspecter les éléments et suivre les erreurs JavaScript.
- Le **localStorage** du navigateur pour stocker les données côté client.
- Des captures d'écran pour documenter le fonctionnement de l'application.

Un fichier **README .md** a également été rédigé pour décrire le projet, son fonctionnement et ses principales fonctionnalités.

3. Avec qui avez-vous travaillé ?

Ce projet a été **entièrement réalisé seul**.

J'ai conçu, codé et testé l'ensemble de l'application de manière autonome, en appliquant les compétences acquises lors de ma formation, notamment pour la partie dynamique des interfaces web.

4. Contexte

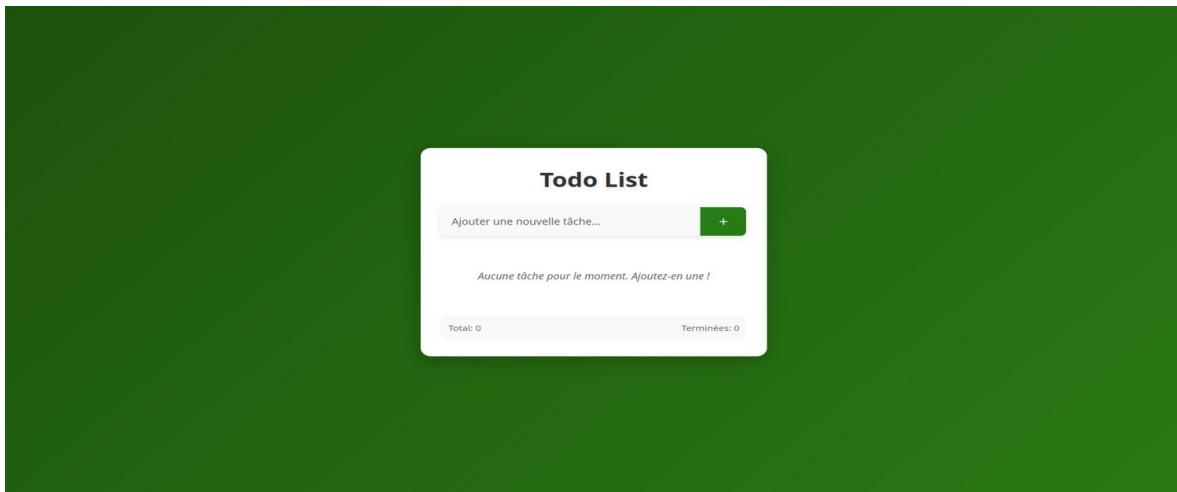
Nom de l'entreprise, organisme ou association *La Plateforme*

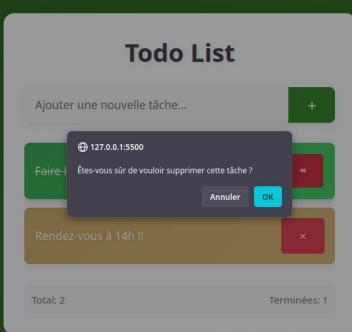
Chantier, atelier, service ▶ *Application de gestion des tâches (Todolist)*

Période d'exercice ▶ Du : **01/05/2025** au : **15/05/2025**

5. Informations complémentaires (facultatif)

Voici quelques captures d'écrans illustrant la todo-list :





Activité-type 2

Développer la partie back-end d'une application web ou web mobile sécurisée

CP 5.

Mettre en place une base de données relationnelle

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

Tâches accomplies pour la base de données relationnelle :

1. **Configuration MySQL existante :**

- Fichier db.js avec pool de connexions MySQL
- Configuration complète (host, user, password, database)
- Base nommée "mini-shop-back"

2. **Structure relationnelle définie : fichier init.sql :**

- Tables avec relations : categories, products, orders, users
- Clés étrangères : products.categories_id → categories.id
- Tables de liaison : products_has_orders, products_has_users
- Contraintes relationnelles : ON DELETE, ON UPDATE

3. **Contrôleurs opérationnels :**

- productController.js, authController.js, etc.

- Requêtes SQL déjà écrites, voir requêtes SQL en partie 5 (informations complémentaires) .

2. Précisez les moyens utilisés :

Tâches accomplies pour la base de données relationnelle :

1. **ANALYSE ET CONCEPTION.**

- **Analyse du domaine métier** : Boutique d'instruments de musique
- **Identification des entités** : Produits, Catégories, Marques, Utilisateurs, Commandes
- **Définition des relations** : Hiérarchiques (catégories/marques), One-to-Many, Many-to-Many

Conditions :

- **Contexte** : Projet e-commerce “Mini-Shop”
- **Objectif** : Structure flexible pour site de vente en ligne.
- **Contraintes : Performance, sécurité, évolutivité**

Conditions fonctionnelles :

- **Domaine** : E-commerce
- **Évolutivité** : Support de nouvelles catégories
- **Flexibilité** : Spécifications JSON variables
- **Multilingue** : Prêt pour internationalisation

Conditions de développement :

- **Express.js** : API REST avec contrôleurs dédiés
- **Angular 19** : Frontend moderne
- **Migration** : Coexistence avec MySQL existant
- **Testing** : Données de test incluses

3. Avec qui avez-vous travaillé ?

Pour ce projet j'ai travaillé en autonomie total.

4. Contexte

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

▶

Chantier, atelier, service ▶ *Boutique en ligne “Mini-shop”*

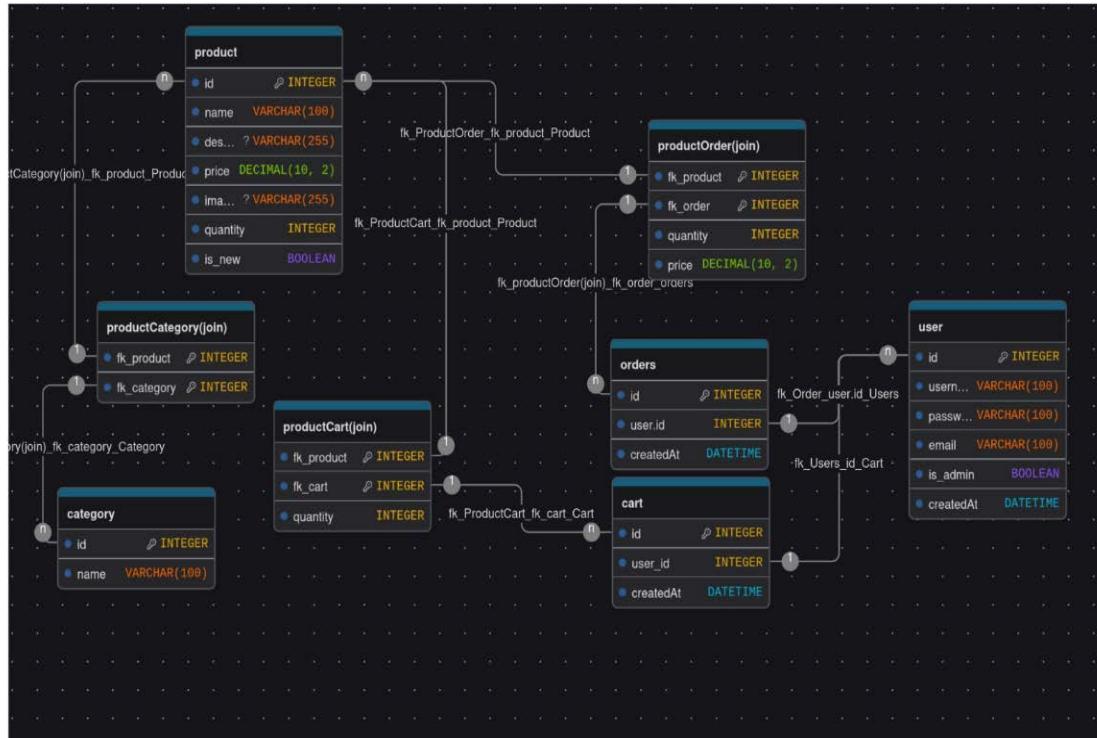
Période d'exercice ▶ Du : **17/06/2025** au : **31/08/2025**

5. Informations complémentaires (facultatif)

Requêtes SQL :

```
1  CREATE DATABASE mini_shop CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci;
2
3
4  CREATE TABLE users (
5      id int AUTO_INCREMENT PRIMARY KEY,
6      email VARCHAR(255) NOT NULL UNIQUE,
7      password varchar(255) NOT NULL
8  )
9
10 CREATE TABLE products (
11     id int AUTO_INCREMENT PRIMARY KEY,
12     name VARCHAR(255) NOT NULL,
13     description TEXT,
14     price DECIMAL(10, 2) NOT NULL,
15     stock int DEFAULT 0,
16     image_url VARCHAR(255)
17 );
18
19 CREATE TABLE categories(
20     id int AUTO_INCREMENT PRIMARY KEY,
21     name VARCHAR(255) NOT NULL UNIQUE,
22     description TEXT
23 );
24
25 CREATE TABLE orders (
26     id INT AUTO_INCREMENT PRIMARY KEY,
27     user_id INT,
28     order_date DATETIME,
29     FOREIGN KEY (user_id) REFERENCES users(id)
30 );
31
32
33
34 -- table pour panier --
35
36 -- Table pour stocker les paniers (un par utilisateur)
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
259
260
261
262
263
264
265
266
267
268
269
269
270
271
272
273
274
275
276
277
278
279
279
280
281
282
283
284
285
286
287
288
289
289
290
291
292
293
294
295
296
297
298
299
299
300
301
302
303
304
305
306
307
308
309
309
310
311
312
313
314
315
316
317
317
318
319
319
320
321
322
323
324
325
326
327
327
328
329
329
330
331
332
333
334
335
336
337
337
338
339
339
340
341
342
343
344
345
345
346
347
347
348
349
349
350
351
352
353
354
355
356
356
357
358
358
359
359
360
361
362
363
364
365
365
366
367
367
368
368
369
369
370
371
372
373
374
375
375
376
377
377
378
378
379
379
380
381
382
383
384
385
385
386
387
387
388
388
389
389
390
391
392
393
394
395
395
396
397
397
398
398
399
399
400
401
402
403
404
405
405
406
407
407
408
408
409
409
410
411
412
413
414
414
415
416
416
417
417
418
418
419
419
420
421
422
423
424
424
425
426
426
427
427
428
428
429
429
430
431
432
433
434
434
435
436
436
437
437
438
438
439
439
440
441
442
443
444
444
445
446
446
447
447
448
448
449
449
450
451
452
453
454
454
455
456
456
457
457
458
458
459
459
460
461
462
463
464
464
465
466
466
467
467
468
468
469
469
470
471
472
473
474
474
475
476
476
477
477
478
478
479
479
480
481
482
483
484
484
485
486
486
487
487
488
488
489
489
490
491
492
493
494
494
495
496
496
497
497
498
498
499
499
500
501
502
503
504
504
505
506
506
507
507
508
508
509
509
510
511
512
513
513
514
515
515
516
516
517
517
518
518
519
519
520
521
522
523
523
524
525
525
526
526
527
527
528
528
529
529
530
531
532
533
533
534
535
535
536
536
537
537
538
538
539
539
540
541
542
543
543
544
545
545
546
546
547
547
548
548
549
549
550
551
552
553
553
554
555
555
556
556
557
557
558
558
559
559
560
561
562
563
563
564
565
565
566
566
567
567
568
568
569
569
570
571
572
573
573
574
575
575
576
576
577
577
578
578
579
579
580
581
582
583
583
584
585
585
586
586
587
587
588
588
589
589
590
591
592
593
593
594
595
595
596
596
597
597
598
598
599
599
600
601
602
603
603
604
605
605
606
606
607
607
608
608
609
609
610
611
612
613
613
614
615
615
616
616
617
617
618
618
619
619
620
621
622
623
623
624
625
625
626
626
627
627
628
628
629
629
630
631
632
633
633
634
635
635
636
636
637
637
638
638
639
639
640
641
642
643
643
644
645
645
646
646
647
647
648
648
649
649
650
651
652
653
653
654
655
655
656
656
657
657
658
658
659
659
660
661
662
663
663
664
665
665
666
666
667
667
668
668
669
669
670
671
672
673
673
674
675
675
676
676
677
677
678
678
679
679
680
681
682
683
683
684
685
685
686
686
687
687
688
688
689
689
690
691
692
693
693
694
695
695
696
696
697
697
698
698
699
699
700
701
702
703
703
704
705
705
706
706
707
707
708
708
709
709
710
711
712
713
713
714
715
715
716
716
717
717
718
718
719
719
720
721
722
723
723
724
725
725
726
726
727
727
728
728
729
729
730
731
732
733
733
734
735
735
736
736
737
737
738
738
739
739
740
741
742
743
743
744
745
745
746
746
747
747
748
748
749
749
750
751
752
753
753
754
755
755
756
756
757
757
758
758
759
759
760
761
762
763
763
764
765
765
766
766
767
767
768
768
769
769
770
771
772
773
773
774
775
775
776
776
777
777
778
778
779
779
780
781
782
783
783
784
785
785
786
786
787
787
788
788
789
789
790
791
792
793
793
794
795
795
796
796
797
797
798
798
799
799
800
801
802
803
803
804
805
805
806
806
807
807
808
808
809
809
810
811
812
813
813
814
815
815
816
816
817
817
818
818
819
819
820
821
822
823
823
824
825
825
826
826
827
827
828
828
829
829
830
831
832
833
833
834
835
835
836
836
837
837
838
838
839
839
840
841
842
843
843
844
845
845
846
846
847
847
848
848
849
849
850
851
852
853
853
854
855
855
856
856
857
857
858
858
859
859
860
861
862
863
863
864
865
865
866
866
867
867
868
868
869
869
870
871
872
873
873
874
875
875
876
876
877
877
878
878
879
879
880
881
882
883
883
884
885
885
886
886
887
887
888
888
889
889
890
891
892
893
893
894
895
895
896
896
897
897
898
898
899
899
900
901
902
903
903
904
905
905
906
906
907
907
908
908
909
909
910
911
912
913
913
914
915
915
916
916
917
917
918
918
919
919
920
921
922
923
923
924
925
925
926
926
927
927
928
928
929
929
930
931
932
933
933
934
935
935
936
936
937
937
938
938
939
939
940
941
942
943
943
944
945
945
946
946
947
947
948
948
949
949
950
951
952
953
953
954
955
955
956
956
957
957
958
958
959
959
960
961
962
963
963
964
965
965
966
966
967
967
968
968
969
969
970
971
972
973
973
974
975
975
976
976
977
977
978
978
979
979
980
981
982
983
983
984
985
985
986
986
987
987
988
988
989
989
990
991
992
993
993
994
995
995
996
996
997
997
998
998
999
999
1000
1000
1001
1001
1002
1002
1003
1003
1004
1004
1005
1005
1006
1006
1007
1007
1008
1008
1009
1009
1010
1011
1012
1013
1013
1014
1015
1015
1016
1016
1017
1017
1018
1018
1019
1019
1020
1021
1022
1023
1023
1024
1025
1025
1026
1026
1027
1027
1028
1028
1029
1029
1030
1031
1032
1033
1033
1034
1035
1035
1036
1036
1037
1037
1038
1038
1039
1039
1040
1041
1042
1043
1043
1044
1045
1045
1046
1046
1047
1047
1048
1048
1049
1049
1050
1051
1052
1053
1053
1054
1055
1055
1056
1056
1057
1057
1058
1058
1059
1059
1060
1061
1062
1063
1063
1064
1065
1065
1066
1066
1067
1067
1068
1068
1069
1069
1070
1071
1072
1073
1073
1074
1075
1075
1076
1076
1077
1077
1078
1078
1079
1079
1080
1081
1082
1083
1083
1084
1085
1085
1086
1086
1087
1087
1088
1088
1089
1089
1090
1091
1092
1093
1093
1094
1095
1095
1096
1096
1097
1097
1098
1098
1099
1099
1100
1101
1102
1103
1103
1104
1105
1105
1106
1106
1107
1107
1108
1108
1109
1109
1110
1111
1112
1113
1113
1114
1115
1115
1116
1116
1117
1117
1118
1118
1119
1119
1120
1121
1122
1123
1123
1124
1125
1125
1126
1126
1127
1127
1128
1128
1129
1129
1130
1131
1132
1133
1133
1134
1135
1135
1136
1136
1137
1137
1138
1138
1139
1139
1140
1141
1142
1143
1143
1144
1145
1145
1146
1146
1147
1147
1148
1148
1149
1149
1150
1151
1152
1153
1153
1154
1155
1155
1156
1156
1157
1157
1158
1158
1159
1159
1160
1161
1162
1163
1163
1164
1165
1165
1166
1166
1167
1167
1168
1168
1169
1169
1170
1171
1172
1173
1173
1174
1175
1175
1176
1176
1177
1177
1178
1178
1179
1179
1180
1181
1182
1183
1183
1184
1185
1185
1186
1186
1187
1187
1188
1188
1189
1189
1190
1191
1192
1193
1193
1194
1195
1195
1196
1196
1197
1197
1198
1198
1199
1199
1200
1201
1202
1203
1203
1204
1205
1205
1206
1206
1207
1207
1208
1208
1209
1209
1210
1211
1212
1213
1213
1214
1215
1215
1216
1216
1217
1217
1218
1218
1219
1219
1220
1221
1222
1223
1223
1224
1225
1225
1226
1226
1227
1227
1228
1228
1229
1229
1230
1231
1232
1233
1233
1234
1235
1235
1236
1236
1237
1237
1238
1238
1239
1239
1240
1241
1242
1243
1243
1244
1245
1245
1246
1246
1247
1247
1248
1248
1249
1249
1250
1251
1252
1253
1253
1254
1255
1255
1256
1256
1257
1257
1258
1258
1259
1259
1260
1261
1262
1263
1263
1264
1265
1265
1266
1266
1267
1267
1268
1268
1269
1269
1270
1271
1272
1273
1273
1274
1275
1275
1276
1276
1277
1277
1278
1278
1279
1279
1280
1281
1282
1283
1283
1284
1285
1285
1286
1286
1287
1287
1288
1288
1289
1289
1290
1291
1292
1293
1293
1294
1295
1295
1296
1296
1297
1297
1298
1298
1299
1299
1300
1301
1302
1303
1303
1304
1305
1305
1306
1306
1307
1307
1308
1308
1309
1309
1310
1311
1312
1313
1313
1314
1315
1315
1316
1316
1317
1317
1318
1318
1319
1319
1320
1321
1322
1323
1323
1324
1325
1325
1326
1326
1327
1327
1328
1328
1329
1329
1330
1331
1332
1333
1333
1334
1335
1335
1336
1336
1337
1337
1338
1338
1339
1339
1340
1341
1342
1343
1343
1344
1345
1345
1346
1346
1347
1347
1348
1348
1349
1349
1350
1351
1352
1353
1353
1354
1355
1355
1356
1356
1357
1357
1358
1358
1359
1359
1360
1361
1362
1363
1363
1364
1365
1365
1366
1366
1367
1367
1368
1368
1369
1369
1370
1371
1372
1373
1373
1374
1375
1375
1376
1376
1377
1377
1378
1378
1379
1379
1380
1381
1382
1383
1383
1384
1385
1385
1386
1386
1387
1387
1388
1388
1389
1389
1390
1391
1392
1393
1393
1394
1395
1395
1396
1396
1397
1397
1398
1398
1399
1399
1400
1401
1402
1403
1403
1404
1405
1405
1406
1406
1407
1407
1408
1408
1409
1409
1410
1411
1412
1413
1413
1414
1415
1415
1416
1416
1417
1417
1418
1418
1419
1419
1420
1421
1422
1423
1423
1424
1425
1425
1426
1426
1427
1427
1428
1428
1429
1429
1430
1431
1432
1433
1433
1434
1435
1435
1436
1436
1437
1437
1438
1438
1439
1439
1440
1441
1442
1443
1443
1444
1445
1445
1446
1446
1447
1447
1448
1448
1449
1449
1450
1451
1452
1453
1453
1454
1455
1455
1456
1456
1457
1457
1458
1458
1459
1459
1460
1461
1462
1463
1463
1464
1465
1465
1466
1466
1467
1467
1468
1468
1469
1469
1470
1471
1472
1473
1473
1474
1475
1475
1476
1476
1477
1477
1478
1478
1479
1479
1480
1481
1482
1483
1483
1484
1485
1485
1486
1486
1487
1487
1488
1488
1489
1489
1490
1491
1492
1493
1493
1494
1495
1495
1496
1496
1497
1497
1498
1498
1499
1499
1500
1501
1502
1503
1503
1504
1505
1505
1506
1506
1507
1507
1508
1508
1509
1509
1510
1511
1512
1513
1513
1514
1515
1515
1516
1516
1517
1517
1518
1518
1519
1519
1520
1521
1522
1523
1523
1524
1525
1525
1526
1526
1527
1527
1528
1528
1529
1529
1530
1531
1532
1533
1533
1534
1535
1535
1536
1536
1537
1537
1538
1538
1539
1539
1540
1541
1542
1543
1543
1544
1545
1545
1546
1546
1547
1547
1548
1548
1549
1549
1550
1551
1552
1553
1553
1554
1555
1555
1556
1556
1557
1557
1558
1558
1559
1559
1560
1561
1562
1563
1563
1564
1565
1565
1566
1566
1567
1567
1568
1568
1569
1569
1570
1571
1572
1573
1573
1574
1575
1575
1576
1576
1577
1577
1578
1578
1579
1579
1580
1581
1582
1583
1583
1584
1585
1585
1586
1586
1587
1587
1588
1588
1589
1589
1590
1591
1592
1593
1593
1594
1595
1595
1596
1596
1597
1597
1598
1598
1599
1599
1600
1601
1602
1603
1603
1604
1605
1605
1606
1606
1607
1607
1608
1608
1609
1609
1610
1611
1612
1613
1613
1614
1615
1615
1616
1616
1617
1617
1618
1618
1619
1619
1620
1621
1622
1623
1623
1624
1625
1625
1626
1626
1627
1627
1628
1628
1629
1629
1630
1631
1632
1633
1633
1634
1635
1635
1636
1636
1637
1637
1638
1638
1639
1639
1640
1641
1642
1643
1643
1644
1645
1645
1646
1646
1647
1647
1648
1648
1649
1649
1650
```

MCD Mini-Shop :



Activité-type 2 Développer la partie back-end d'une application web ou web mobile sécurisée

CP 6 ► Développer les composants d'accès aux données SQL et NoSQL

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

Description des tâches réalisées dans le projet Soundora :

Dans le cadre du projet Soundora (boutique en ligne d'instruments de musique et d'accessoires de musique) , j'ai développé la partie back-end de l'application en utilisant Node.js et Express, avec une base de données Supabase (PostgreSQL, SQL).

Voici les tâches et opérations que j'ai effectuées :

- Conception du schéma de base de données (tables utilisateurs, produits, catégories, commandes) dans Supabase.
- Écriture de requêtes SQL pour le CRUD (création, lecture, mise à jour, suppression) des différentes entités (exemples de requêtes SQL dans 5.Informations complémentaires),
- Développement de contrôleurs Express pour chaque entité, assurant la communication entre l'API et la base de données.
- Utilisation du SDK Supabase côté back-end pour interagir de façon sécurisée avec la base (requêtes paramétrées, gestion des erreurs).
- Mise en place de l'authentification sécurisée (Supabase Auth, JWT) et de la gestion des droits d'accès aux données.
- Gestion des opérations transactionnelles pour garantir la cohérence des commandes et des paiements.
- Tests des endpoints API avec des jeux de données réels.

2. Précisez les moyens utilisés :

Voici les moyens que j'ai utilisés :

- Environnement de développement sécurisé (variables d'environnement, gestion des clés API).
- Outils : Node.js, Express, Supabase, PostgreSQL, Postman pour les tests d'API, GitHub pour le versionnement.
- Documentation du code et des schémas de données.

3. Avec qui avez-vous travaillé ?

Travail réalisé en autonomie

4. Contexte

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

Chantier, atelier, service ► *Boutique en ligne d'instruments et d'accessoires de musique "Soundora"*

Période d'exercice ► Du : **17/06/2025** au : **31/08/2025**

5. Informations complémentaires (facultatif)

Exemples de requêtes SQL utilisées :

1. Récupérer tous les produits du catalogue :

SELECT * FROM products;

→ Cette requête retourne la liste complète des produits enregistrés dans la table **products**

2. Ajout d'un produit dans la table *products* :

```
INSERT INTO products (name, description, price, category_id, brand_id, image_url)
VALUES ('Guitare électrique', 'Guitare 6 cordes, idéale pour le rock.', 499.99, 1, 2,
'https://exemple.com/guitare.jpg');
```

→ Cette requête ajoute un produit avec son nom, sa description, son prix, la catégorie et la marque associées, ainsi qu'une image.

3. Jointure pour afficher tous les produits avec leur catégorie et leur marque :

```
SELECT
    p.id, p.name, p.price, c.name AS category, b.name AS brand
FROM
    products p
JOIN
    categories c ON p.category_id = c.id
JOIN
    brands b ON p.brand_id = b.id;
```

→ Cette requête retourne la liste des produits avec le nom de leur catégorie et de leur marque.

Voici les tables de la database de mon site Soundora :

NAME	DESCRIPTION	ROWS (ESTIMATED)	SIZE (ESTIMATED)	REALTIME_ENABLED	8 columns
brands	Marques d'instruments et é...	24	64 kB	✗	8 columns
cart	No description	0	8192 bytes	✗	2 columns
cart_items	Panier temporaire des utilisa...	0	16 kB	✗	6 columns
categories	Catégories hiérarchiques po...	19	80 kB	✗	10 columns
invoice	facture	0	16 kB	✗	6 columns
notice	No description	0	16 kB	✗	6 columns
order	No description	0	16 kB	✗	4 columns
order_items	Articles dans chaque comm... mande	0	8192 bytes	✗	9 columns
orders	Commandes des clients ave...	0	48 kB	✗	18 columns
products	Produits - instruments, acce...	38	232 kB	✗	26 columns
profiles	Profils utilisateur compléme...	5	48 kB	✗	14 columns
subcategory	No description	0	16 kB	✗	3 columns
user	No description	0	16 kB	✗	4 columns
users	No description	0	24 kB	✗	16 columns

Activité-type 2

Développer la partie back-end d'une application web ou web mobile sécurisée

CP 7 ▶

Développer des composants métier côté serveur

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

Dans le projet Soundora, j'ai conçu et développé les composants métier du back-end en Node.js/Express.

J'ai implémenté la logique métier pour la gestion des utilisateurs, des produits, des catégories, du panier, des commandes et du paiement en ligne (Stripe).

J'ai structuré le code en contrôleurs (ex : orderController.js, cartController.js, authController.js) pour séparer la logique métier de l'accès aux données; voir exemple en partie 5.

Précision de la gestion des règles métier :

- J'ai mis en place des contrôleurs côté serveur (Node.js/Express) pour centraliser la logique métier de l'application.
- Pour chaque fonctionnalité (création de commande, ajout au panier, paiement, etc.), j'ai défini des règles précises :
 - Vérification que l'utilisateur est bien authentifié avant d'accéder à certaines routes (ex : passer commande, voir l'historique).
 - Validation des données reçues du frontend : contrôle des champs obligatoires, des formats (ex : email, prix positif, quantité non nulle), et gestion des erreurs en cas de données invalides.
 - Calcul automatique du total du panier et des frais de livraison lors de la création d'une commande.
 - Gestion des stocks : vérification de la disponibilité des produits avant validation d'une commande, et mise à jour des quantités en base après achat.
 - Attribution de droits spécifiques selon le rôle de l'utilisateur (ex : seuls les admins peuvent ajouter ou supprimer des produits, seuls les clients peuvent commander).
 - Intégration du paiement Stripe : la commande n'est validée que si le paiement est accepté, grâce à la gestion des webhooks Stripe.

- Suivi de l'état des commandes (en attente, payée, expédiée, annulée) et mise à jour automatique selon les actions de l'utilisateur ou du backoffice.
- J'ai documenté chaque règle métier dans le code et dans la documentation technique pour faciliter la compréhension et la maintenance du projet.

Conditions de réalisation :

Environnement technique :

Le projet a été développé sur un poste sous Linux, avec Node.js et Angular installés localement. Le code source est versionné sur GitHub, ce qui permet un suivi des évolutions et un travail collaboratif. L'éditeur principal utilisé est Visual Studio Code, avec des extensions facilitant le développement (lint, auto-format, intégration Git...).

•Contraintes matérielles et logicielles :

Le développement nécessite un ordinateur avec une connexion Internet, Node.js, npm, et un navigateur moderne pour le front-end. L'API back-end fonctionne en local ou via Docker, et la base de données est gérée via Supabase (PostgreSQL) ou MySQL selon les besoins de test.

•Outils et ressources :

- **Gestion de version** : Git et GitHub pour le suivi, la sauvegarde et la collaboration.
- **Gestion de projet** : Kanban (GitHub Projects), documentation dans le dossier docs.
- **Tests et validation** : Postman pour tester les API, console navigateur pour le front-end, logs serveur pour le back-end.
- **Sécurité** : Utilisation de variables d'environnement (.env), gestion des tokens JWT, et authentification via Supabase.
- **Déploiement** : Docker Compose pour simuler l'environnement de production, scripts npm pour automatiser les tâches courantes.

•Organisation du travail :

Le projet a été découpé en tâches : conception de la base de données, développement du back-end (API REST, logique métier), puis du front-end (UI responsive, intégration Stripe, gestion de l'authentification). Chaque étape a été validée par des tests manuels et des revues de code.

•Contraintes spécifiques :

- Respect des bonnes pratiques de sécurité (protection des routes, gestion des erreurs, validation des entrées).
- Accessibilité et responsive design pour garantir une expérience utilisateur optimale sur mobile et desktop.
- Documentation systématique du code et des choix techniques pour faciliter la maintenance et la transmission du projet.

2. Précisez les moyens utilisés :

•Environnement technique :

- Développement du back-end avec Node.js et Express pour la gestion des routes, de la logique métier et des contrôleurs.
- Utilisation de Supabase (PostgreSQL) comme base de données relationnelle, avec gestion des tables, des relations et des requêtes SQL.
- Intégration de l'API Stripe pour la gestion des paiements en ligne et des webhooks de confirmation.
- Utilisation de JWT (JSON Web Token) pour l'authentification sécurisée et la gestion des sessions utilisateurs.
- Mise en place de middlewares Express pour la validation des entrées, la gestion des droits d'accès et la sécurisation des routes sensibles.

•Outils et pratiques de développement :

- J'ai utilisé **Visual Studio Code** comme éditeur principal pour écrire et organiser le code du back-end.
- Le code a été versionné avec **Git** et hébergé sur **GitHub**, ce qui m'a permis de sauvegarder chaque étape du projet, de revenir en arrière si besoin, et de travailler proprement.

- J'ai utilisé **Postman** pour tester les différentes routes de l'API : cela m'a permis de vérifier que chaque fonctionnalité (connexion, ajout au panier, paiement, etc.) fonctionnait bien indépendamment du frontend.
- J'ai mis en place des **variables d'environnement** (fichier .env) pour sécuriser les informations sensibles comme les clés d'API ou les identifiants de la base de données.
- J'ai documenté le code avec des commentaires clairs pour expliquer la logique métier, les paramètres attendus et les retours des fonctions.
- J'ai structuré le projet en plusieurs fichiers et dossiers : contrôleurs pour la logique métier, routes pour les points d'entrée de l'API, services pour l'accès aux données, ce qui facilite la maintenance et l'évolution du projet.
- J'ai appliqué les bonnes pratiques de sécurité : validation des données reçues, gestion des erreurs, séparation des droits d'accès selon le rôle de l'utilisateur (admin, client...).

3. Avec qui avez-vous travaillé ?

Pour ce projet j'ai travaillé en total autonomie

4. Contexte

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

Chantier, atelier, service ► *Boutique en ligne d'instruments et d'accessoires de musique "Soundora"*

Période d'exercice ► Du : **17/06/2025** au : **31/08/2025**

5. Informations complémentaires (facultatif)

Voici ci dessous le code du fichier api.js représentant les différentes routes de l'application :

```

routes > js api.js ...
1 import express from "express"; // Importe Express pour créer le routeur
2 import * as authController from "../controllers/authController.js"; // Contrôleur d'authentification
3 import * as productSupabaseController from "../controllers/productSupabaseController.js"; // NOUVEAU : Contrôleur produits Supabase (remplace productController)
4 import * as categoryController from "../controllers/categoryController.js"; // Contrôleur catégories
5 import * as brandController from "../controllers/brandController.js"; // Contrôleur marques
6 import * as cartController from "../controllers/cartController.js"; // Contrôleur panier
7 import * as orderController from "../controllers/orderController.js"; // Contrôleur commandes
8 import * as stripeController from "../controllers/stripeController.js"; // NOUVEAU : Contrôleur Stripe pour paiements
9 import * as testController from "../controllers/testController.js"; // Contrôleur de test
10 import checkJwt from "../middleware/checkJwt.js"; // Middleware JWT pour protéger les routes
11 import checkSupabaseAuth from "../middleware/checkSupabaseAuth.js"; // Middleware Supabase Auth
12
13 const router = express.Router(); // Crée un routeur Express
14
15 // =====
16 // ROUTES DE TEST
17 // Permettent de vérifier le bon fonctionnement de Supabase
18 // =====
19 router.get("/test/connection", testController.testConnection); // Test connexion Supabase
20 router.get("/test/tables", testController.listTables); // Liste des tables disponibles
21
22 // =====
23 // ROUTES D'AUTHENTIFICATION
24 // Gestion des utilisateurs avec Supabase Auth
25 // =====
26 router.post("/auth/register", authController.register); // Incription d'un utilisateur
27 router.post("/auth/login", authController.login); // Connexion et obtention du token JWT
28 router.post("/auth/logout", authController.logout); // Déconnexion utilisateur
29 router.get("/auth/me", checkSupabaseAuth, authController.getCurrentUser); // Récupération de l'utilisateur actuel (protégé)
30
31 // =====
32 // ROUTES POUR LES PRODUITS - Version Supabase Avancée
33 // Remplace l'ancien productController par productSupabaseController
34 // Nouvelles fonctionnalités : pagination, filtres, recherche, slugs SEO
35 // =====
36
37 // ROUTE PRINCIPALE : Liste des produits avec filtres et pagination
38 // Exemple d'utilisation :
39 // GET /api/products
40 // GET /api/products?name=bane&limit=20
41 // Tous les produits (page 1, 10 par page)
42 // Page 2, 10 produits par page
43
44 router.get("/products", productSupabaseController.getAllProducts);
45
46 // ROUTE PRODUITS FEATURED : Produits mis en avant
47 // GET /api/products/featured?limit=6
48 // GET /api/products/featured?limit=12
49 router.get("/products/featured", productSupabaseController.getFeaturedProducts);
50
51 // ROUTE RECHERCHE : Recherche textuelle rapide
52 // GET /api/products/search?q=guitare&limit=10
53 // GET /api/products/search?q=gibson
54 router.get("/products/search", productSupabaseController.searchProducts);
55
56 // ROUTE PRODUIT INDIVIDUEL : Récupération par slug (SEO-friendly)
57 // GET /api/products/gibson-les-paul-standard-2024
58 // GET /api/products/fender-stratocaster-american
59 // IMPORTANT : Cette route doit être EN DERNIER pour éviter les conflits avec "featured" et "search"
60 router.get("/products/:slug", productSupabaseController.getProductBySlug);
61
62 // =====
63 // ROUTES POUR LES CATÉGORIES
64 // Gestion des catégories de produits (guitares, basses, etc.)
65 // =====
66 router.get("/categories", categoryController.getAllCategories); // Liste toutes les catégories
67 router.get("/categories/:id", categoryController.getCategoryById); // Récupère une catégorie spécifique
68 router.post("/categories", checkJwt, categoryController.createCategory); // Création d'une catégorie (protégé)
69 router.put("/categories/:id", checkJwt, categoryController.updateCategory); // Mise à jour (protégé)
70 router.delete("/categories/:id", checkJwt, categoryController.deleteCategory); // Suppression (protégé)
71
72 // =====
73 // ROUTES POUR LES MARQUES
74 // Gestion des marques de produits (Fender, Gibson, etc.)
75 // =====
76 router.get("/brands", brandController.getAllBrands); // Liste toutes les marques
77 router.get("/brands/:id", brandController.getBrandById); // Récupère une marque par ID
78 router.get("/brands/slug/:slug", brandController.getBrandBySlug); // Récupère une marque par slug
79
80 // =====
81
82 // =====
83 // ROUTES POUR LE PANIER
84 router.get("/cart", checkJwt, cartController.getCart); // Récupère le panier de l'utilisateur (protégé)
85 router.post("/cart/items", checkJwt, cartController.addItemToCart); // Ajoute un article au panier (protégé)
86 router.put("/cart/items/:id", checkJwt, cartController.updateCartItem); // Met à jour un article du panier (protégé)
87 router.delete("/cart/items/:id", checkJwt, cartController.removeFromCart); // Supprime un article du panier (protégé)
88 router.get("/cart/count", checkJwt, cartController.getCartCount); // Récupère le nombre d'articles dans le panier (protégé)
89
90 // =====
91 // ROUTES POUR LES COMMANDES
92 // Gestion des commandes et historique d'achat
93 // =====
94 router.post("/orders", checkJwt, orderController.createOrder); // Création d'une commande (protégé)
95 router.get("/orders", checkJwt, orderController.getUserOrders); // Récupère les commandes de l'utilisateur (protégé)
96 router.get("/orders/:order_id", checkJwt, orderController.getOrderDetails); // Récupère une commande spécifique (protégé)
97
98 // =====
99 // ROUTES STRIPE PAIEMENT
100 // Gestion des paiements via Stripe Checkout
101 // =====
102 router.post("/stripe/create-checkout-session", checkSupabaseAuth, stripeController.createCheckoutSession); // Crée session Stripe (protégé)
103 router.get("/stripe/session-status/:sessionId", stripeController.getSessionStatus); // Vérifie statut session
104
105 // =====
106 // ROUTES STRIPE TEST (POUR DÉVELOPPEMENT) - ACTIVÉES POUR TESTS
107 // Routes simples pour tester Stripe sans authentification
108 // =====
109 router.post("/stripe/test-simple", stripeController.createTestSessionSimple); // Test session 10€ (non protégé)
110 router.post("/stripe/test-complete", stripeController.createTestSessionComplete); // Test session 99€ (non protégé)
111
112 // =====
113 // WEBHOOK STRIPE (SANS AUTHENTICATION)
114 // Stripe appelle cette route pour confirmer les paiements
115 // IMPORTANT : Cette route ne doit PAS avoir de middleware d'auth
116 // Le middleware raw est déjà appliqué dans server.js
117 // =====
118 router.post("/stripe/webhook", stripeController.stripeWebhook);
119
120 // === EXPORT DU ROUTEUR POUR L'UTILISER DANS server.js ===
121 export default router;

```

Et voici le fichier .env pour la sécurité des variables d'environnement :

```
⚙️ .env
1 # Configuration du serveur
2 PORT=3010
3
4 # URL Frontend pour redirections Stripe
5 FRONTEND_URL=http://localhost:4200
6
7 # Configuration Supabase (remplace MySQL)
8 SUPABASE_URL=https://lohumrjasdauvpqgjvh.supabase.co
9 SUPABASE_ANON_KEY=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
eyJpc3Mi0iJzdXBhYmFzZSIsInJlZiI6ImxvaHVtcmphc2RhdXZwcWdqdmhkIiwiitm9sZSI6ImFub24iLCJpYXQiOjE3NTAxNTkyMzksIm
V4cCI6MjA2NTczNTIzOX0.C8K7G0LrU9H8mdQBTTKQ95RzqnXEA-zE9KV1UBsoo
10 SUPABASE_SERVICE_KEY=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
eyJpc3Mi0iJzdXBhYmFzZSIsInJlZiI6ImxvaHVtcmphc2RhdXZwcWdqdmhkIiwiitm9sZSI6InNlcnZpY2Vfcm9sZSIisImlhdi6MTc1MD
E10TizOSwiZXhwIjoyMDY1NzM1MjM5fQ.yq2UA71ljX31WdXkQ5hN37MG6lorg0s7XY1FgQ7rqWk
11 # JWT Secret pour l'authentification
12 JWT_SECRET=6RUygh01V8CAsAjiEAq7a2ikbaz6aihz+lXJuiZAw47zW12NgxVPzXonfdk8RyE/4QdcfcwiTHfnKFHbyllg+A==
13
14
15 # Configuration Stripe (MODE TEST pour projet étudiant)
16 STRIPE_PUBLISHABLE_KEY=pk_test_51RqscFGtJYVPcJeyWCXcZFKEYl8RZNDZYkL12wYz4ouZMc7B9V15Gu02KZWoA94L9CYXMsGb4L
a7N7SGWZTSFzIs00MoCcQ1GD
17 STRIPE_SECRET_KEY=sk_test_51RqscFGtJYVPcJeymYuKSJgKfImxRKW8CScZ0DEDxzsjIbjM7f2wEZkpqBr3vHKhojf576wGQRcu2xy
mc0GYygW300E2717oH0
18 STRIPE_WEBHOOK_SECRET=whsec_c1f029953b5043a6a2973fb0264788278ec346fe0c67686838664aec50cceee74
19
20 # URL de connexion PostgreSQL pour Supabase
21 DATABASE_URL=postgresql://postgres:Foie1312%402512@db.lohumrjasdauvpqgjvh.supabase.co:5432/postgres
22
```

Activité-type 2

Développer la partie back-end d'une application web ou web mobile sécurisée

CP 8 ▶

Documenter le déploiement d'une application dynamique (déploiement en cours avec plesk)

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

Tâches et opérations effectuées :

Dans le cadre du projet Soundora, j'ai conçu et développé l'ensemble du back-end de l'application, en mettant l'accent sur la sécurité et la modularité.

J'ai structuré le serveur avec Node.js et Express, créé des routes RESTful pour la gestion des utilisateurs, des produits, des commandes et des paniers, et mis en place des middlewares pour l'authentification JWT et la gestion des droits d'accès.

J'ai conçu le schéma relationnel de la base de données MySQL (MCD/MLD), rédigé les scripts SQL, et utilisé phpMyAdmin pour la gestion et la visualisation des données.

Pour la partie paiement, j'ai intégré Stripe afin de permettre des transactions sécurisées, en gérant la création de sessions de paiement, la réception des webhooks et la validation des paiements côté serveur.

Le déploiement a été réalisé via l'interface Plesk : j'ai transféré les fichiers du projet, configuré l'environnement Node.js, installé les dépendances, paramétré les variables d'environnement (clés Stripe, accès BDD), et relié le nom de domaine fourni par mon organisme de formation (bastien-brunet.students-laplateforme.io) à l'adresse IP dédiée (82.165.185.52).

J'ai également configuré le serveur web (HTTPS, redirections, droits d'accès) et la base de données via Plesk, puis testé l'ensemble du processus de bout en bout.

Conditions de réalisation :

Le projet a été mené dans le cadre d'un dossier professionnel de fin d'année, sur un hébergement mutualisé avec Plesk fourni par l'organisme de formation. J'ai utilisé un environnement Linux, Git pour le versionnement, supabase et phpMyAdmin pour la gestion de la base. Le nom de domaine et l'adresse IP m'ont été attribués par la plateforme, ce qui a facilité la mise en ligne et la configuration DNS.

J'ai veillé à respecter les bonnes pratiques de sécurité (gestion des secrets, HTTPS, droits d'accès restreints).

2. Précisez les moyens utilisés :

Voici les moyens que j'ai utilisés :

Environnement de développement :

Utilisation de Visual Studio Code pour l'édition du code, avec des extensions facilitant le développement Node.js, la gestion Git et la coloration syntaxique SQL.
Node.js et npm ont permis de gérer les dépendances et d'exécuter le serveur localement pour les tests.

•Gestion de versions et collaboration :

Git a été utilisé pour le suivi des modifications, la création de branches et la gestion des versions du projet. GitHub a servi de plateforme de partage, de sauvegarde et de collaboration, permettant également de documenter le projet via le README.

•Base de données :

MySQL Workbench a servi à la modélisation du schéma relationnel (MCD/MLD) et à la génération des scripts SQL.
phpMyAdmin, accessible via Plesk, a permis la gestion, la visualisation et la modification des données en production.

•Déploiement et hébergement :

Plesk a été utilisé pour le déploiement du projet : transfert des fichiers, configuration de l'environnement Node.js, gestion des variables d'environnement, installation des dépendances, gestion du nom de domaine (bastien-brunet.students-laplateforme.io) et de l'adresse IP (82.165.185.52), configuration du certificat SSL pour le HTTPS, et gestion des accès à la base de données.

•Services externes :

Stripe a été intégré pour la gestion des paiements en ligne, avec utilisation de l'API, gestion des clés secrètes en variables d'environnement, et configuration des webhooks pour le suivi des transactions.

•Documentation et organisation :

Rédaction de la documentation technique (README, scripts SQL, schémas de base de données, procédures de déploiement) pour assurer la maintenabilité et la reproductibilité du projet.

3. Avec qui avez-vous travaillé ?

Travail réalisé en autonomie.

4. Contexte

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

Chantier, atelier, service ► *Boutique en ligne d'instruments et d'accessoires de musique "Soundora"*

Période d'exercice ► Du : **17/06/2025** au : **31/08/2025**

5. Informations complémentaires (facultatif)

DOSSIER PROFESSIONNEL (DP)

Titres, diplômes, CQP, attestations de formation

(facultatif)

Intitulé	Autorité ou organisme	Date
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.

DOSSIER PROFESSIONNEL (DP)

Déclaration sur l'honneur

Je soussigné(e) [prénom et nom] [Cliquez ici pour taper du texte.](#),

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 .. [Cliquez ici pour choisir une date](#)

pour faire valoir ce que de droit.

Signature :

Bastien Brunet

DOSSIER PROFESSIONNEL (DP)

Documents illustrant la pratique professionnelle (facultatif)

Intitulé

Cliquez ici pour taper du texte.

DOSSIER PROFESSIONNEL (DP)

ANNEXES

(*Si le RC le prévoit*)