



## Nicolas NIEDDU

28 ans  
Permis B  
Fontaines-sur-Saône,  
69270  
nicolas.nieddu@free.fr  
06.79.05.65.55

### Compétences

Anglais (intermédiaire)  
C  
C++  
Git  
Bash, shell  
Html  
CSS  
SASS, SCSS  
Javascript  
Typescript  
React.js, Redux  
Gestion de relation client

### Connaissances

Docker  
Kubernetes  
Jest  
Unreal engine  
Photoshop

[Mon portfolio](#)

[GitHub](#)

# Développeur front-end

Passionné par les nouvelles technologies, l'informatique et plus largement les sciences, je suis toujours curieux de découvrir de nouveaux domaines afin d'étendre mon panel de compétences. De nature autodidacte, j'ai rejoint l'école **42** afin d'acquérir de solides connaissances dans le développement informatique.

## Formations

### 42 LYON

Architecte du numérique, développeur informatique  
2019- 2022  
Charbonnières-les-Bains, France

### Lycée la Martinière-Diderot

Baccalauréat STI20 (option énergie et environnement)  
2013  
Lyon, France

## Expériences

### EUREXO

Expert terrain, gestion de sinistres  
mars 2018 - juillet 2018  
Région Auvergne-Rhône-Alpes, France

### POLYEXPERT

Expert en assurances  
novembre 2013 - janvier 2018  
Villeurbanne, France

### PIERRES CONSTRUCTION

Ouvrier du bâtiment  
mai 2013 - octobre 2013  
Lyon, France

# Dossier de compétences NIEDDU Nicolas

Téléphone : 06.79.05.65.55

Mail : [nicolas.nieddu@free.fr](mailto:nicolas.nieddu@free.fr)

Liens : [Porfolio](#) [GitHub](#) [LinkedIn](#)

## Compétences

---

Langages	C, C++ HTML, CSS, SCSS / SASS, JavaScript, TypeScript, ReactJS, Redux, Redux toolkit Bash, Shell, Batch
Systèmes	Windows XP, 7, 8, 10, Unix, Linux
Divers	Makefile, CMake Photoshop, Blender Unreal Engine <a href="#">Coding Game</a> <a href="#">Root-me</a>

## Références projets

---

### Cursus 42

---

#### [C]

- Libft : Recoder les fonctions de la libc suivantes :

atoi.c, bzero.c, calloc.c, isalnum.c, isalpha.c, isascii.c, isdigit.c, isprint.c, itoa.c, lstadd\_back.c, lstadd\_front.c, lstclear.c, lstdelone.c, lstiter.c, lstlast.c, lstmap.c, lstnew.c, lstsize.c, memccpy.c, memchr.c, memcmp.c, memcpy.c, memmove.c, memset.c, putchar\_fd.c, putendl\_fd.c, putnbr\_fd.c,\_putstr\_fd.c, split.c, strchr.c, strcmp.c, strdel.c, strdup.c, strjoin.c, strlcat.c, strlcpy.c, strlen.c, strmap.c, strncmp.c, strnstr.c, strchr.c, strtrim.c, substr.c, tolower.c, toupper.c

afin de créer une librairie (.a) qui se compile à partir d'un Makefile.

<https://github.com/nnieddu/libft>

- `get_next_line` : Coder une fonction qui va stocker, dans le paramètre « ligne », une ligne qui a été lue à partir du descripteur de fichier donné et lire la ligne suivante à chaque appel de fonction jusqu'à atteindre EOF, end of file (Initiation aux à l'utilisation de 'file descriptor').

[https://github.com/nnieddu/ft\\_get\\_next\\_line](https://github.com/nnieddu/ft_get_next_line)

- `ft_printf` : Recoder la fonction `printf` avec prenant en charge les conversions 'cspdiuxX' et les flags '0.\*-'. Ce projet utilise notre propre librairie (libft) et notre fonction `get_next_line`.

[https://github.com/nnieddu/ft\\_printf](https://github.com/nnieddu/ft_printf)

- `minishell` : Mise en place d'un interpréteur de commandes comme `bash` en recodant quelques built-in (`echo`, `cd`, `pwd`, `export`, `unset`, `env`, `exit`), en intégrant les redirections `<` `>` `<<` `>>` `|` et en gérant les variables d'environnement.

<https://github.com/nnieddu/minishell>

- `ft_push_swap` : Mise en place d'un algorithme de tri (`quicksort`) avec 2 piles (`stack`) de nombres. Afin de valider ce projet, seuls certains types d'opérations étaient autorisés (voir repo) puis l'objectif était de réaliser le tri avec un nombre limité de déplacements et de temps (sensibilisation à l'optimisation).

<https://en.wikipedia.org/wiki/Quicksort>

[https://github.com/nnieddu/ft\\_push\\_swap](https://github.com/nnieddu/ft_push_swap)

- `cub3d-linux` : Réalisation d'un moteur 3D utilisant le raycasting, pour créer un jeu au style ancien comme `Wolfenstein3D`, en utilisant la `minilibx`. Avec gestion du parsing d'une map à partir d'un fichier texte.

<https://github.com/nnieddu/cub3d-linux>

- `philosophers` : Implémentation du problème du 'dîner des philosophes', travail avec de la programmation multithread (`mutex` et `sémaphore`).

[https://fr.wikipedia.org/wiki/D%C3%AEner\\_des\\_philosophes](https://fr.wikipedia.org/wiki/D%C3%AEner_des_philosophes)

<https://github.com/nnieddu/philosophers>

## [C++]

- ft\_irc : Création d'un serveur IRC (conforme aux normes RFC 1459 et 2813).  
[https://github.com/nnieddu/ft\\_irc](https://github.com/nnieddu/ft_irc)
- ft\_containers : Réimplémentation de certains conteneurs de la STL (vector, map, stack) et de leurs mécanisme d'itérateurs + réalisation de tests automatisés.  
[https://github.com/nnieddu/ft\\_containers](https://github.com/nnieddu/ft_containers)

## Web

---

### HTML, CSS, JS, TS, ReactJS, Redux, SEO

- ft\_transcendence : Création d'une 'single page application' en typescript pour un jeu de Pong multijoueur (rendu dans un Canvas) en utilisant NestJS, React, React router, Redux, librairie MUI pour l'UI, une base de données PostgreSQL et Google authenticator pour la 2FA. Conception d'une possibilité de création de comptes grâce à l'API de 42.  
  
En équipe de 3 dans laquelle je m'occupais de la partie Frontend et de quelques ajouts sur le back pour des fonctionnalités du chat.  
  
[https://github.com/nnieddu/ft\\_transcendence](https://github.com/nnieddu/ft_transcendence)

## DevOps

---

- ft\_server : Projet de déploiement de serveur, en utilisant Docker, nginx maria-db (mysql), phpmyadmin et wordpress. Le projet se lance directement avec le Dockerfile qui appelle un script déployant les services au sein du conteneur.  
  
[https://github.com/nnieddu/ft\\_server](https://github.com/nnieddu/ft_server)
- ft\_services : Le projet consiste à mettre en place une infrastructure de différents services, avec ses propres règles (tableau de bord web de Kubernetes, Load Balancer (metallb), WordPress, phpMyAdmin, serveur FTPS, Grafana, Influxdb et base de données Mysql), en utilisant Kubernetes et Docker.  
  
[https://github.com/nnieddu/ft\\_services](https://github.com/nnieddu/ft_services)

## Hors 42

---

### Web

---

HTML, CSS, JS, TS, ReactJS, Redux, SEO

- Mon site CV : utilisant du vanilla JS, HTML et CSS

<https://nnieddu.github.io/cv/index.html>

- Plusieurs exemples de 'CRUD'

<https://nnieddu.github.io/CRUD/>

Simulation d'une base de données avec JsonServer ou le localStorage et exemple avec ElectronJS pour créer un app Desktop pour PC.

<https://github.com/nnieddu/CRUD-example>

<https://github.com/nnieddu/CRUD-localStorage-example>

<https://github.com/nnieddu/CRUD-localStorage-RTK-ElectronJs>

- MoviePicker : Création d'un site à page unique en typescript avec ReactJS et Redux Toolkit pour sélectionner ou supprimer un seul film préféré pour chaque lettre de l'alphabet en utilisant l'API omdb.

<https://nnieddu.github.io/moviePickerV2/>

- ToDoList : Création d'une 'todolist' avec un système de validation de tâches par date de fin et d'une fonction recherche par texte ou date (utilisant une API privée) en vanilla JS, HTML et CSS.

<https://github.com/nnieddu/toDoList-VanillaJs>

- Mon portfolio : En ReactJS, HTML et SCSS en utilisant des fichiers JSON pour le mécanisme de traduction

<https://nnieddu.github.io/>

## Hors 42

---

### Web

---

- Application React qui affiche une liste d'appartements.

L'application permet à l'utilisateur de créer, éditer et supprimer des appartements.

Chaque appartement doit avoir un nom, une description, un prix et une photo.

Les appartements sont triés par ordre alphabétique (lexicographique) et il est possible de les afficher dans une page dédiée.

Cette application a été conçue en React/Typescript, utilise React Context pour gérer l'état de l'application, React Router pour la navigation et Tailwind (+ icônes de fontawesome) pour le style.

La liste des tâches est stockée dans une source de données externe : Firebase (Firestore Database), en utilisant l'API REST fournis par Google.

L'application est déployée sur github page.

<https://github.com/nnieddu/Chez-Nestor-test>

## Software Development

---

### C++

- Modification d'un plugin Unreal Engine : Permet de créer un serveur Web HTTP (mongoose) qui communique avec le moteur UE. Mise à jour pour avoir la prise en charge du HTTPS et permettant d'utiliser les fonctions gyroscopiques des téléphones Apple via une page web et de récupérer les données dans le moteur Unreal (pas de repo GitHub public car le plugin original est payant).
- red-black-tree : Implémentation de la structure de données 'arbre binaire rouge-noir' avec templates en C++.

[https://github.com/nnieddu/red\\_black\\_tree](https://github.com/nnieddu/red_black_tree)

- IA Process Image Recognition : Création d'un programme permettant la reconnaissance d'objets dans un processus sur Windows avec OpenCV via certains modèles d'IA type YOLO. (WIP)

<https://github.com/nnieddu/iaProcessImageRecognition>

### Scripting

---

- Windows : Création d'un script batch pour le déploiement et le maintien de mon portfolio, push de la version build (avec Webpack) sur github.io, commit et push sur le repo des sources qui est 'private' et qui met à jour tous les sous repo (git submodules).
- Unix (ft\_service et ft\_server) : Déploiement de services dans des containers.