# Rust — Programmation système et développement performant

*Lien : https://innov-systems.com/formation/rust-programmation-systeme-et-developpement-performant*

| DURÉE | RÉFÉRENCE | CATÉGORIE |
| --- | --- | --- |
| **5 jours (35h)** | **DID25** | **Elm, Unity, Go, ROS, Rust** |

Innov Systems

## OBJECTIFS DE LA FORMATION

À l'issue de cette formation, vous serez capable de :

- Comprendre les fondamentaux du langage Rust
- Maîtriser la gestion de la mémoire sans garbage collector
- Développer des applications système performantes
- Utiliser le modèle ownership/borrowing efficacement
- Concevoir des programmes concurrents sûrs
- Exploiter l’écosystème Rust et Cargo
- Développer des APIs et services backend en Rust
- Optimiser les performances des applications
- Appliquer Rust dans des contextes système et cloud
- Écrire du code sûr et robuste
- Réduire les bugs liés à la mémoire
- Étudier des cas concrets de développement système

## POUR QUI ?

- Développeurs logiciels
- Ingénieurs systèmes
- Backend developers
- Software engineers
- DevOps engineers
- Développeurs systèmes embarqués
- Architectes logiciels
- Ingénieurs performance
- Développeurs C/C++ souhaitant évoluer vers Rust
- SRE (Site Reliability Engineers)
- Toute personne impliquée dans le développement logiciel performant et système

Innov Systems

## Programme détaillé

### 1 / Introduction au langage Rust

- Présentation de Rust et philosophie du langage
- Sécurité mémoire sans garbage collector
- Cas d’usage en systèmes critiques et performants

### 2 / Fondamentaux du langage Rust

- Variables, types et mutabilité
- Fonctions, modules et organisation du code
- Gestion des erreurs (Result, Option)

### 3 / Ownership, borrowing et lifetime

- Modèle de propriété mémoire
- Emprunts (borrowing) et références
- Durée de vie des données (lifetimes)

### 4 / Programmation système avec Rust

- Accès bas niveau au matériel
- Interaction avec le système d’exploitation
- Développement de composants système

### 5 / Concurrence et parallélisme

- Threads et async/await
- Programmation concurrente sûre
- Synchronisation et partage de données

Innov Systems

### 6 / Structures de données et abstractions

- Structs, enums et traits
- Génériques et polymorphisme
- Design patterns en Rust

### 7 / Gestion de la mémoire et performance

- Allocation mémoire contrôlée
- Optimisation des performances
- Comparaison avec C/C++

### 8 / Développement d’applications réseau et backend

- Serveurs web performants
- APIs REST et microservices
- Intégration avec frameworks (Actix, Axum)

### 9 / Écosystème Rust et tooling

- Cargo et gestion de projets
- Crates et dépendances
- Tests, linting et documentation

### 10 / Cas pratiques et projets systèmes

- Développement d’un service backend performant
- Outils système ou CLI en Rust
- Bonnes pratiques de production

Innov Systems

## Approche pédagogique

- Support Ecrit et Projection
- Exposés Interactifs, Podcasts et Vidéos
- Brainstorming et Jeux de Rôle
- Cas Pratiques et Labs inclus pour leur impact opérationnel
- Test de Validation des Acquis des Connaissances

## Prochaines dates programmées

| | |
|---|---|
| 18 au 22 Jan. 2027 | Présentiel - Casablanca |
| 15 au 19 Mars 2027 | Présentiel - Casablanca |
| 10 au 14 Mai 2027 | Présentiel - Casablanca |
| 05 au 09 Juil. 2027 | Présentiel - Casablanca |
| 30 Août au 03 Sep. 2027 | Présentiel - Casablanca |

Autres dates possibles sur demande. Contactez-nous pour organiser une session intra-entreprise.

## Réservation & Renseignements

**Téléphone :** +212 522 247 210
**Email :** contact@innov-systems.com
**Web :** https//www.innov-systems.com


Innov Systems — Tous droits réservés