

Présentation de l'Équipe de Développement

Compilateur IFCC (IF-C-Compiler)

Hexanome 4211 : DALAOUI Riad, AUBUT Antoine, CHIKHI Djalil,
HANADER Rayan, TAIDER Sami, CHAOUKI Youssef, and
LOUVET Samuel

INSA Lyon - Département Informatique - 4IF-PLD-COMP

Février à Avril 2025

Table des matières

1	Composition de l'Équipe	2
2	Répartition des Fonctionnalités	2
3	Organisation du Travail	3
3.1	Méthodologie	3
3.2	Outils	3

1 Composition de l'Équipe

Sept élèves-ingénieurs ont contribué à ce projet de compilateur, combinant leurs expertises complémentaires. Une répartition stratégique des responsabilités a assuré une gestion optimale à chaque étape du développement.

Membre	Rôle	Responsabilités
Sami TAIDER	Chef de Projet	<ul style="list-style-type: none">— Gestion du repository Git— Revue des pull requests— Organisation des sprints (Trello)— Coordination de l'équipe
Rayan HANADER	Responsable Documentation et Soutenance	<ul style="list-style-type: none">— Rédaction des documentations— Coordination de la soutenance— Scripts, démo et supports
Djalil CHIKHI, Antoine AUBUT, Samuel LOUVET	Architectes Logiciel	<ul style="list-style-type: none">— Vue d'ensemble de l'architecture— Choix des structures de données— Supervision technique
Youssef CHAOUKI, Riad DALAOUI	Responsables Qualité	<ul style="list-style-type: none">— Mise en place des tests— Vérification de la couverture— Intégration continue
Tous les membres	Ingénieurs-Développeurs	<ul style="list-style-type: none">— Implémentation des fonctionnalités— Correction de bugs— Revue de code pair-à-pair

2 Répartition des Fonctionnalités

Malgré les responsabilités de chacun, tous les membres ont participé à l'écriture du code source du compilateur.

Fonctionnalité	Implémenté par
Types de base (<code>int</code> , <code>char</code>) et déclarations/affectations variables/constantes	Sami, Rayan, Youssef
Opérations arithmétiques/logiques	Antoine, Djalil, Samuel
IR et structures en bloc	Antoine, Djalil, Youssef, Rayan
Structures de contrôle (<code>if/while</code>)	Antoine
Fonctions	Riad, Samuel
Génération de code x86	Tous
Système de tests	Youssef, Antoine, Riad

3 Organisation du Travail

3.1 Méthodologie

- Méthode agile avec sprints de 2 semaines
- Revue de code systématique avant merge
- Documentation technique continue

3.2 Outils

Git : Gestion de version avec branche `feature`/par-développeur

Trello : Tableaux Kanban pour suivi des tâches