Tecnológico de Monterrey
Escuela de Ingeniería y Ciencias

Expo ingenierías

Cloud-Based C- to MIPS Code Generator with Hybrid Infrastructure

Sylvia Fernanda Colomo Fuente, Francisco Flamenco Andrade, Natalia Maya Bolaños, Shaul Zayat Askenazi.

Abstract

CloudStrype is a web application designed to translate C- code into MIPS assembly language. The system features a frontend hosted in a virtual private cloud (VPC) and a backend running locally in a physical server within the classroom. The backend is implemented using FastAPI, and MongoDB is used for secure storage of user credentials and history. The hybrid cloud-local architecture ensures fast interaction, real-time compilation, and private infrastructure control.

Introduction

CloudStrype is a web-based tool designed to translate C- code into MIPS assembly, bridging the gap between high-level programming and low-level architecture understanding. This project aims to support students in learning compiler principles by providing a fast and intuitive interface to test and visualize their code transformations. C- is a simplified academic version of C, while MIPS is a widely used assembly language in education due to its readability and simplicity.

The application is built using a **hybrid infrastructure**. The **frontend**, developed with React, is hosted in a virtual private cloud (VPC). In contrast, the **backend**, written in FastAPI, runs locally on a physical server within the classroom. This setup not only simulates real-world distributed systems but also allows for better control, lower latency for code processing, and the opportunity for students to learn about hybrid deployments. User login, session tracking, and compilation history are stored in a **MongoDB** database, which supports flexible and fast data handling. The backend and database communicate over a secure local network, while cloud-to-local communication is enabled through an institutional gateway configured using assigned physical ports. This architecture gives students hands-on experience with modern web technologies, authentication, database integration, and network routing.

Methods and Materials

Methods and Materials

•**Frontend:** React.js hosted in a Horizon Cloud VPC. Provides user interface, code input, login, and history display.

•**Backend:** FastAPI running on a physical server in the classroom. Processes input and handles REST API requests.

•**Database:** MongoDB used for storing user data and compilation history.

•**Infrastructure:** Custom network gateway setup to allow external communication through classroom-assigned ports.

•**Security:** Basic authentication system with hashed passwords and protected routes.

Results

- Successful translation of C- code to MIPS.
- Stable user management system with session persistence.
- Full integration between cloud-based frontend and local backend.
- Functional internet connectivity using institution-assigned physical ports.

Discussion

Implementing a hybrid architecture allowed us to leverage cloud advantages—such as availability and scalability—while maintaining control over backend processing and network routing. Using FastAPI accelerated development and improved performance, while MongoDB offered flexibility in data storage. Key challenges included ensuring secure API communication and managing real-time data between distant systems.

Figure 1. Cloud Computing.

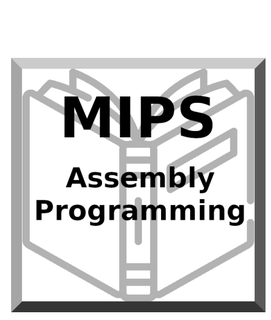
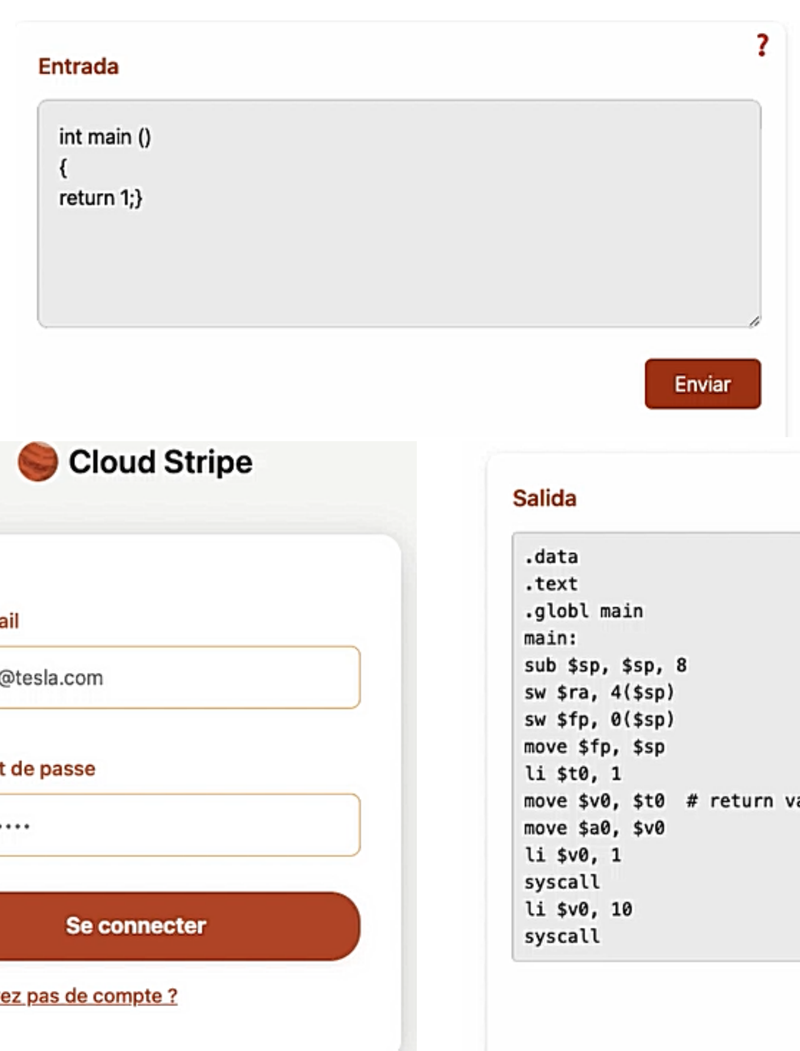


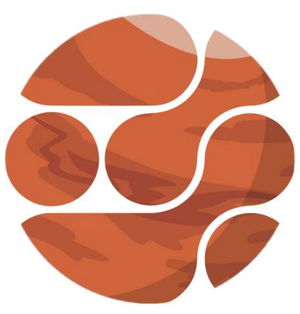
Figure 2. MIPS Assembly.

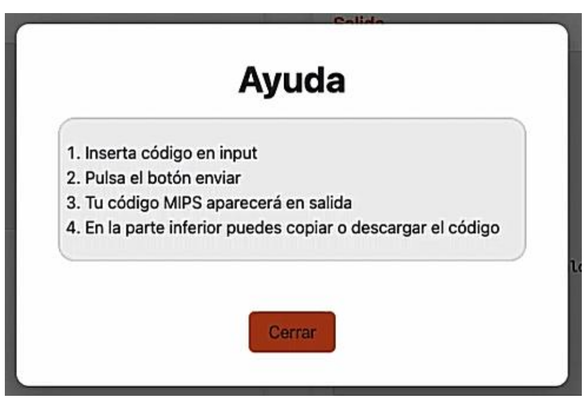
Figure 3. Database.

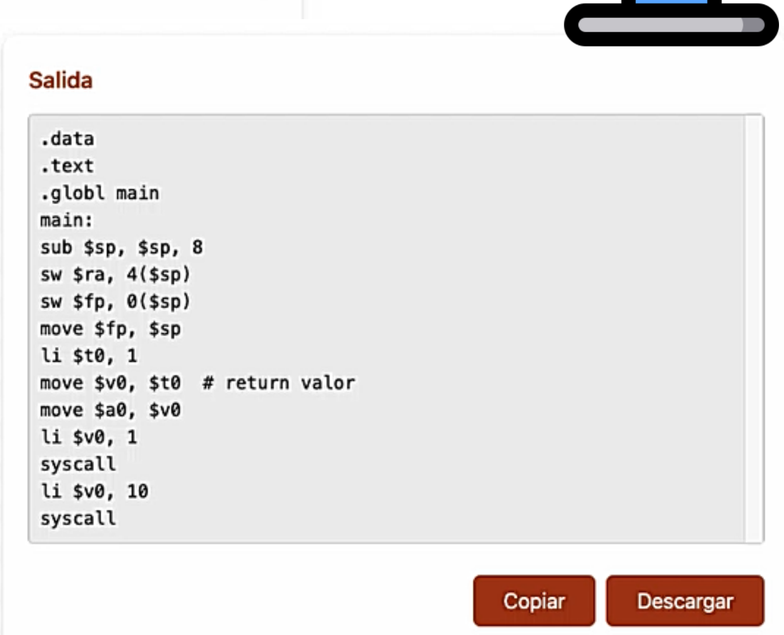
Conclusions

CloudStrype demonstrates the viability of hybrid web applications in educational environments. It provides a fast, intuitive way to teach compiler concepts while also giving students exposure to modern deployment models. Future improvements include extending language support and enhancing security.









References

AWS. Security best practices for your VPC. (2025). https://docs.aws.amazon.com/vpc/latest/userguide/vpc-security-best-practices.html
C Lauden, K. (2004). UNED Construccion de compiladores principios. International Thomson Editores.
FASTAPI. FastAPI Documentation. (2025). https://fastapi.tiangolo.com/
MongoDB. MongoDB Manual. (2025). https://www.mongodb.com/docs/manual
REACT Team. REACT Documentation. (2025). https://reactjs.org