

Introduction

This project embarks on an exciting exploration of using formal language theory tools such as automata, transducers, context-free grammars (CFGs), and regular expressions (regex) can be seamlessly integrated with storytelling. Specifically, it

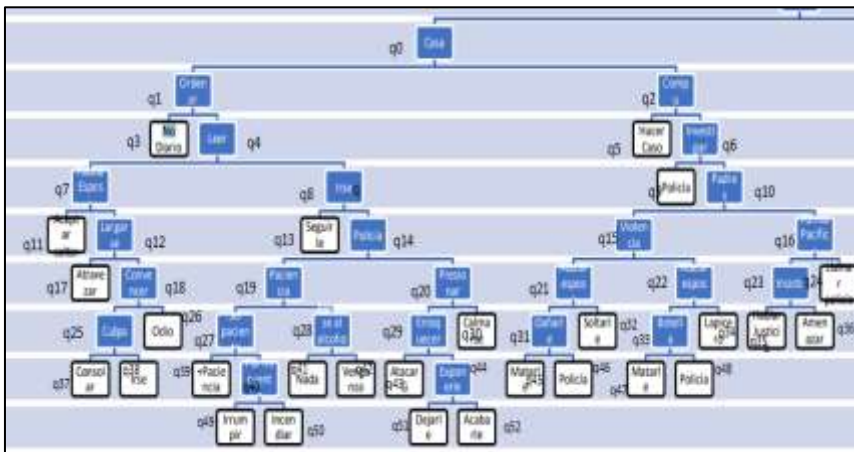
showcases the application of Pyformlang, a Python library, to create immersive narratives that respond to reader decisions, offer multiple endings, and adapt to evolving plotlines.

Goal

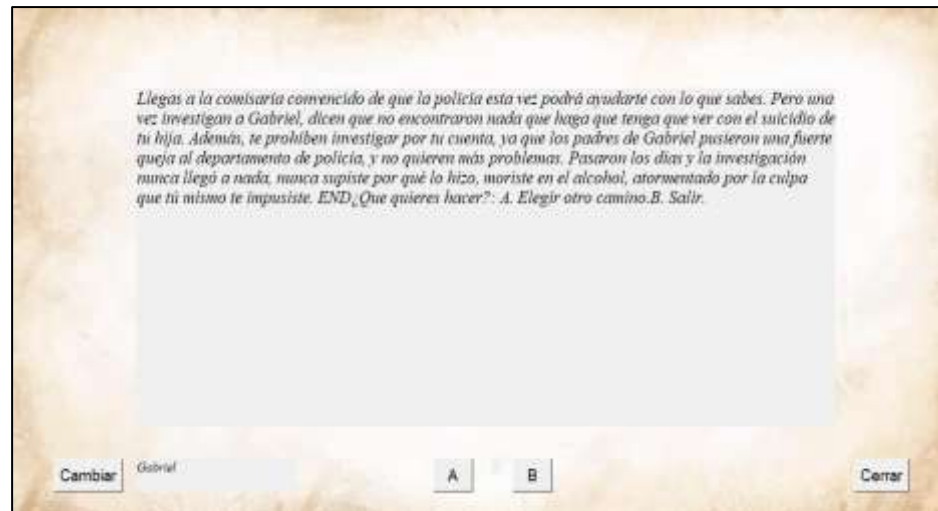
Understand and apply the notions of languages and regular expressions, as well as types of computational models, for pattern recognition, processing, validation, and text extraction using a programming language.¹

Methodology

First, we needed a story to tell, so we used the brainstorming method to come up with a good story. When we had the scene, the next thing we thought about was making it more interactive, so we created a decision tree. After that it was necessary to define the states of the automata, so we marked them on the graph.

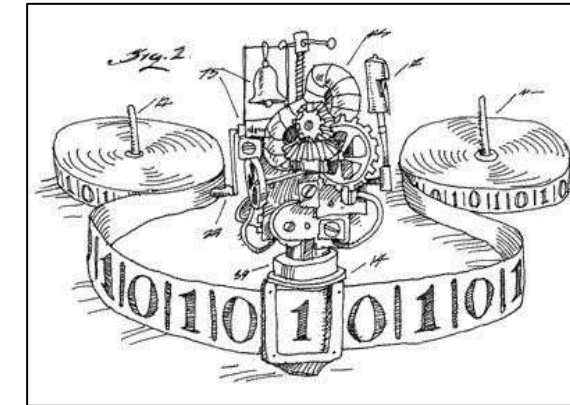


We use all the knowledge learned along the semester about formal language theory to create a structure of computational models that allow us to tell an interactive story. As long as we made it in python, we needed to learn about some libraries that allow us to reach the objective. These libraries are `re` for the regular expressions and `Pyformlang` for the computational models like automates, transducers and context-free grammars implementers.



Final product

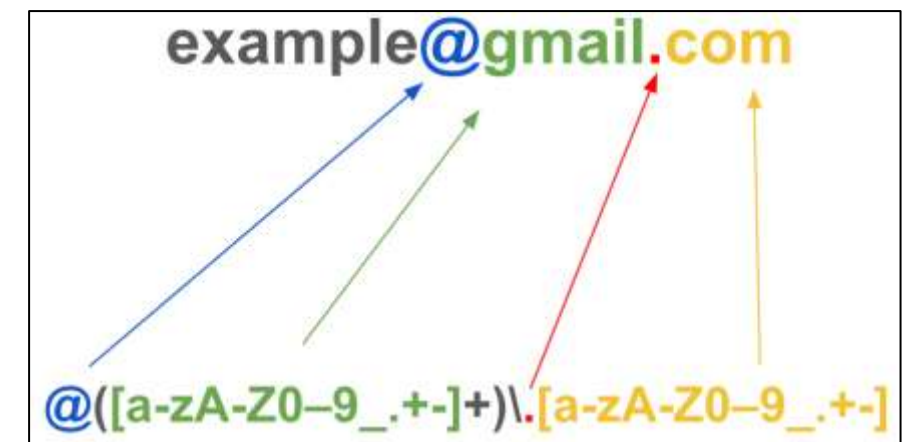
Our final product consists of an interactive story that has a considerable number of endings. In any moment, the user have the opportunity to change the antagonist name, which we will verify with a regular expression. The story will then begin by asking the user to decide how to proceed in the different scenarios until it reaches one of the endings, some longer and others more difficult to achieve. We will also ask the user if he wants to continue playing to reach different endings. Regular expressions were also used to identify when the user reached one of the ends, changing the options to play again or quit. Automaton were used to manage transitions between scenes and the available options for users, while transducers facilitated communication between the system and the user, enabling user actions to influence the direction of the story.



Conclusions

During the development of this project, we witnessed the utility that computational models focused on automation bring when it comes to

interpreting and transforming information. Moreover, we recognized the significant importance of formal language theory in the field of computer science, and in our specific case, in the development of interactive software. Furthermore, the implementation of automata and transducers allowed us to develop a seamless interactive experience for users. These computational models seamlessly integrate into a multi-ending story, greatly simplifying its design. Context-free grammars played a crucial role in validating user-inputted data. Undertaking this project was of paramount importance in comprehending the fundamentals and applications of formal language theory, as well as unleashing our creativity through the use of new tools that replaced certain programming paradigms we had previously considered as the sole alternatives for developing solutions.



References and Acknowledgements

1. Programa de curso: computación y estructuras discretas III -
Facultad de Ingeniería, Diseño y Ciencias Aplicadas ICESI