Introduccion

# Motivacion

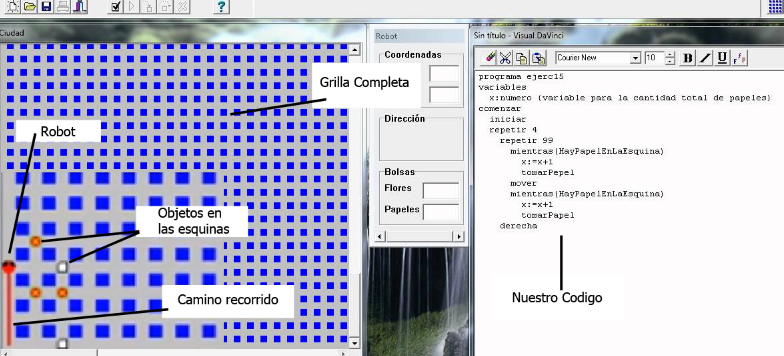
Hace algun tiempo se desarrollo un Software, llamado Visual Da Vinci. El objetivo del mismo es crear una capa entre el lenguaje humano y un lenguaje de programacion, si bien tiene muchas restricciones, se suma el querer que el usuario de VDV, tenga una interfaz menos *arcaica* para aprender a programar, al menos, algoritmos basicos, que incluye estructuras de control, bucles, funciones y atributos, aunque en su maxima simplicidad, pero los incluye al fin.

Esto nos motivo a querer unificar muchas ideas en una sola. En primer lugar, probar, desafiar, mejorar, aprender y volver a desafiar nuestras capacidades de desarrollo, imaginacion, conocimientos y trabajo en grupo. En segundo lugar, tomar un software ya existente y mejorarlo en varios aspectos, escalabilidad, amigabilidad, potencial y modernidad. En tercer y ultimo lugar, la idea de lograr un desarrollo que permita ayudar o mejorar de algun modo, algun aspecto academico o estuantil a cualquier nivel, en nuestro caso, la complejidad de incorporar a las personas con dificultades en la informatica en la programacion.

# Visual Da Vinci, Software de ejemplo

Este programa consta de: Una grilla que representa una ciudad con sus caminos, un *robot* que será controlado por el algoritmo escrito por el usuario. Flores y papeles que se encuentran en las esquinas de la ciudad que el *robot* puede recoger. Ademas, un área de texto donde se escribe el algoritmo que va a controlar al *robot*.

A continuación un ScreenShot del software.



En *Nuestro Codigo*, podemos ver algunos detalles que resaltan (Variables, repetir, ejecución de métodos). El mismo se va a ejecutar de manera secuencial.

Se puede observar la Grilla que representa una ciudad, que será recorrida por el robot. Podrá juntar o no las flores y papeles según lo que codifique el usuario.

Nuevamente si observamos mas el código, el cual es bastante intuitivo, podemos deducir su funcionamiento. Esto lo dejamos a cargo del propio lector, ya que no afecta la comprensión del proyecto.

# 3 Proyecto #NombreDelProyecto

A continuación se describirá brevemente como se podría componer el Software. Sera detallado mas adelante.

## Interfaz Gráfica:



1. Escenario: acá es donde se podrá ver el vehículo desplazándose por la ciudad, junto a otros vehículo, peatones, semáforos, paradas de colectivo y destinos entre otros.
2. Código: Acá se escribirá el código que ejecutaremos.
3. Terminal de comandos. Sería interesante realizar una terminal de comandos para que el usuario valla arrojando comandos y probar. En tal caso no debería estar compilando y ejecutando una y otra vez.
4. Estadísticas: El nombre de esta sección es bastante intuitivo. Las estadísticas serian: cantidad de pasajeros levantados, llevados, dejados en destino, combustible quizás entre otras.

## Ciudad:

La ciudad podría ser en este estilo. Notamos que es un poco mas moderno, mas llamativo, mas natural a la vista de uno.



## Codigo

Falta definir varias ideas de posibles códigos.

### Codigo al Estilo Python, pero en castellano.

### Codigo tipo PilasBloques

## Terminal De Comandos

Esta es una posibilidad. Que el código se valla ejecutando a medida que lo va escribiendo.

## Estadísticas

Estas estadísticas serán según lo necesitado en la lógica funcional.

## Escenario

Seria llamado el mapa donde se interactúa nuestro software.

## Lenguaje de programacion + FrameWork propuestos.

Estuvimos evaluando varias posibilidades:

1)Pilas engine con Python.

2)Go

3)PyGame con Python

4)JMonkeyEngine con Java

5)C# con Unity

Y los que tienen la mayor posibilidad son Pilas-Engine + Python.

Balanza:

Backend- Por falta de un IDE muy desarrollado va a ser un poco mas difícil de lo que podría ser con Java o C#.

Frontend: Este era mucho mas fácil que con otros lenguajes que ya que contamos con Pilas-Engine. Y eso nos otorga muchas ventajas.

### Es producto Argentino, lo que va a llamar la atención. Tanto hacia Pilas-Engine como para ## Nuestro proyecto

### Pilas-Engine es libre y gratuito. No vamos a tener inconvenientes con nuestra publicación

### Es un Framework fácil de usar y de gran potencial, al ser solo 2D(+Z) es una herramienta que no tiene mas de lo que estamos necesitando.