|  |
| --- |
| Ceballos Vitale Pablo Guillermo, Gómez Tomás |
| Flick Color |
| Documento del Proyecto N°1 de la Materia Lógica para las Ciencias de la Computación |

|  |
| --- |
| 14/05/2018 |

Contenido

[ASPECTOS GENERALES 2](#_Toc514227266)

[COMO JUGAR 2](#_Toc514227267)

[FORMA DE LA INTERFAZ 2](#_Toc514227268)

[JUGABILIDAD 3](#_Toc514227269)

[RESOLUCIÓN GENERAL DEL PROYECTO 3](#_Toc514227270)

[ESTRATEGIAS DE RESOLUCIÓN GENERALES 4](#_Toc514227271)

[LADO DEL CLIENTE: LENGUAJE PROLOG 4](#_Toc514227272)

[MODIFICAR EL TABLERO DE ACUERDO AL INPUT DEL USUARIO. 4](#_Toc514227273)

[IMPLEMENTACIÓN DE LA AYUDA BÁSICA Y EXTENDIDA 4](#_Toc514227274)

[IMPLEMENTACIÓN DEL CAMBIO DE GRILLA 4](#_Toc514227275)

[LADO DEL SERVIDOR: LENGUAJES JAVASCRIPT, HTML Y CSS 4](#_Toc514227276)

[DISEÑO Y FORMA DE LA INTERFAZ 4](#_Toc514227277)

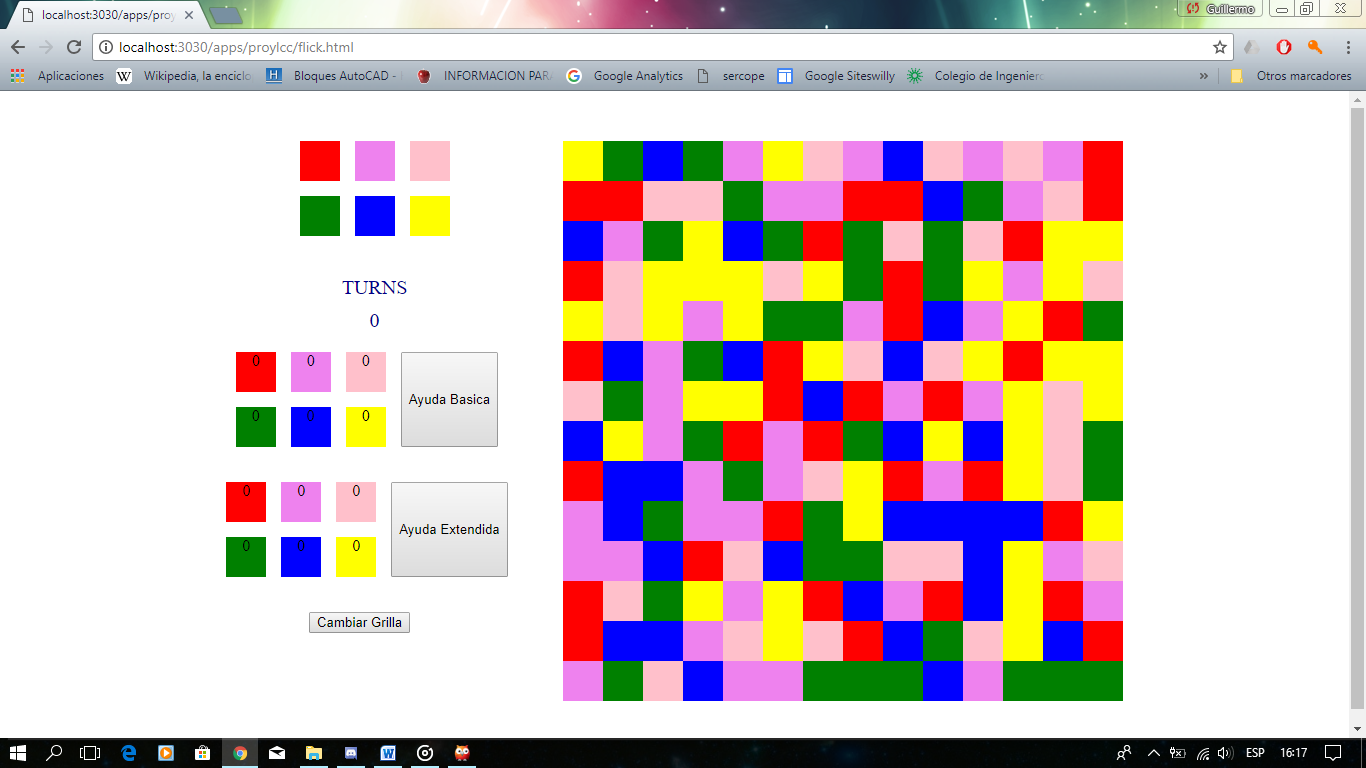
[COMUNICACIÓN CON PROLOG 4](#_Toc514227278)

[CONCLUSIONES Y OBSERVACIONES GENERALES 5](#_Toc514227279)

# ASPECTOS GENERALES

## COMO JUGAR

### FORMA DE LA INTERFAZ



2

5

4

3

1

Ilustración : Interfaz del Juego

|  |  |
| --- | --- |
| 1 | Panel de botones para modificar el tablero de juego |
| 2 | Tablero de juego |
| 3 | Paneles de Ayuda |
| 4 | Botón de Cambio de Grilla |
| 5 | Número de Turnos |

### 

### JUGABILIDAD

El objetivo del juego es pintar el tablero de un único color en la menor cantidad de turnos posible. Para esto, se disponen de 6 botones con los colores disponibles al jugador, como se podrá apreciar en la Ilustración 1 donde debe hacerse clic sobre cualquiera de los 6 colores para modificar el tablero de juego con el color deseado, cuando se presiona uno de los botones, la esquina superior izquierda del tablero cambiará al color deseado, y con eso cambiará a todas las casillas adyacentes a ella que sean del mismo color, hasta que se encuentren casillas adyacentes de colores distintos

Si el jugador todavía no ha empezado a jugar, puede optar por cambiar el tablero de juego actual, haciendo uso del Botón de Cambio de Grilla, esto hará que se cargue otro tablero del repertorio de tableros disponibles del juego

Si el jugador desea saber que jugada le conviene hacer para pintar la mayor cantidad de casillas, el jugador puede optar por pedir ayuda usando alguno de los dos botones de ayuda disponibles

* “Ayuda Básica” brindará información sobre la cantidad de casillas que serán pintadas para cada color en particular, luego de un turno
* “Ayuda Extendida” Brindará información sobre la mejor jugada posible que podría hacerse en dos turnos

## 

## RESOLUCIÓN GENERAL DEL PROYECTO

<Resolución a alto nivel del Proyecto>

Siendo un proyecto comprendido por distintas partes de diferente envergadura, asignar prioridades al proyecto fue crucial para resolverlo satisfactoriamente, siendo que resolver los problemas requeridos desde el lado del Lenguaje Prolog por el proyecto era necesario para asegurar el correcto funcionamiento del juego, decidimos que lo primero a resolver era encontrar la forma de que el código Prolog modifique el estado interno del tablero de juego correctamente, siendo que si se podía asegurar el correcto funcionamiento de un algoritmo que recorra el tablero y modifique el mismo, los demás requerimientos podían ser satisfechos fácilmente con mínimas modificaciones sobre el mismo algoritmo.

Luego, determinamos como segunda prioridad la implementación de los componentes JavaScript necesarias para albergar la lógica de las ayudas y del cambio de tablero de juego, fue necesario recurrir a fuentes de información adicionales, ya que nuestro conocimiento sobre el Lenguaje era limitado

Mientras se llevaba a cabo la implementación de las componentes gráficas, se desarrolló la parte lógica de las ayudas y del cambio de tablero de juego desde el lenguaje Prolog, los predicados usados para resolver estos problemas usan el algoritmo para recorrer el tablero del predicado que modifica el tablero en cierta manera que será descrita en detalle en sus secciones correspondientes.

# ESTRATEGIAS DE RESOLUCIÓN GENERALES

## LADO DEL CLIENTE: LENGUAJE PROLOG

### MODIFICAR EL TABLERO DE ACUERDO AL INPUT DEL USUARIO.

<Algoritmo de recorrido y explicación de la resolución adoptada>

### IMPLEMENTACIÓN DE LA AYUDA BÁSICA Y EXTENDIDA

<Explicar cómo se usó el algoritmo hallado anteriormente para implementar las ayudas>

### IMPLEMENTACIÓN DEL CAMBIO DE GRILLA

<Idem ayudas>

## LADO DEL SERVIDOR: LENGUAJES JAVASCRIPT, HTML Y CSS

### DISEÑO Y FORMA DE LA INTERFAZ

Aunque una interfaz base fue provista por la cátedra, fue necesario extenderla para agregar las demás funcionalidades requeridas por el proyecto, dando lugar a la primer problemática a resolver, como diseñar la Interfaz Gráfica.

Decidimos que la mejor forma de organizarla era mantener todas las componentes interactivas sobre la barra lateral ya provista por la cátedra, para esto, definimos el resto de componentes que necesitaríamos en el Código HTML dentro de la sección correspondiente a esta barra lateral para que luego podamos acceder a ellas desde el Código JavaScript sin tener que definir las variables múltiples veces para proveerles de la Lógica necesaria para operar correctamente

La parte del Código CSS del proyecto solo fue modificada para proveer a las componentes agregadas a la interfaz de un aspecto gráfico adecuado

### COMUNICACIÓN CON PROLOG

El código JavaScript hace uso de una librería provista por la cátedra denominada “Pengines” para comunicarse con la lógica Prolog para hacer consultas a ciertos predicados implementados en dicho lenguaje, los mismos son:

* Flick(+Grid, +Color,-RGrid)
  + El cerebro lógico detrás del juego, este predicado se encarga de, dada la Grilla y un color seleccionado desde el lado del servidor, modificar la grilla de acuerdo a los requerimientos especificados en el proyecto y devolver una nueva grilla, que será levantada por el servidor para modificar el tablero gráfico del juego
* ayudaBasica(+Grid, +Color,-Res)
  + Se encarga de, dada la grilla, calcular para cada color posible cuantas celdas serán pintadas luego de un turno, la respuesta a cada color es levantada por el servidor para modificar los labels correspondientes
* ayudaExtendida(+Grid, +Color, -Res)
  + Se encarga de, dada la grilla, calcular para cada color cuantas celdas serán pintadas luego de dos turnos, calculando la máxima cantidad de celdas pintadas posibles para todas las combinaciones posibles de colores elegidos como segundo color

# CONCLUSIONES Y OBSERVACIONES GENERALES