

**“UNIVERSIDAD TECMILENIO”
CERTIFICADO EN FUNDAMENTOS DE LA
PROGRAMACIÓN**



FUNDAMENTOS DE LA PROGRAMACIÓN

**- AVANCE DEL PROYECTO: -
VEX ROBOTICS: GESTOR DE PROYECTOS**

Primer Semestre

INTEGRANTES:

Victor Andres Hernandez Castillo - SENIOR

David Durán Rubio - MID

Nicolas Flores Verdad Quintana - MID

Blanca Dennis Soto Rubio - JUNIOR

INTRODUCCIÓN

Lo que buscamos en este proyecto es darle solución a una problemática que nos afecte de manera directa o indirecta, en nuestro caso presentamos una herramienta de apoyo para el taller del TecMilenio “VEX: ROBOTICS”. Dicha herramienta tendrá varias funciones como apoyo a los estudiantes de dicho taller, pues ese es su principal objetivo es ayudar a los alumnos a lograr con mayor facilidad sus objetivos y que cuenten con una mayor organización.

CONTEXTO Y PROBLEMA

Uno de los integrantes de equipo pertenece a VEX, por lo tanto él considera que las principales flaquezas o problemas que tiene el taller, es la falta de organización, que no se conoce el material que se tiene para el trabajo y que las ideas no están bien aterrizadas, ya que muchas veces no sabemos las actividades pendientes que tenemos que realizar.

Separaremos nuestro programa en cuatro secciones, pues consideramos que contamos con cuatro grandes problemas:

1.- Lista de integrantes por sector:

Muchas de las veces, debido a la mala organización, no sabemos a qué sector pertenece cada integrante del equipo. VEX se divide en 3 principales ramas de trabajo que son: *Administración, Mecánica y Programación*; Por ello vamos crear una lista de integrantes, para checar a que paquete pertenece cada integrante y tener de manera más visual los equipos.

MATRÍCULA	NOMBRE	ROL	SEMESTRE
AL07165261	David Duran Rubio	Mecánica	Primer semestre
AL07165262	Víctor Hernandez	Programación	Primer semestre
AL07165263	Blanca Dennis	Administración	Primer semestre

Este es un ejemplo sencillo de lo que buscamos lograr en la lista de integrantes.

2.- Inventario/Materiales:

En VEX el material que se tiene para trabajar es muy limitado o en varias ocasiones no se sabe con cuanto material se cuenta debido a que no hay un registro concreto, por lo tanto en nuestro programa buscamos tener el conteo del material disponible para poder llevar a cabo sus proyectos. Este registro se podría modificar para poder quitar o agregar material según corresponda.

```
Objeto = Tornillos,--- Cantidad = 120  
  
Objeto = Tuercas,--- Cantidad = 120  
  
Objeto = Motores,--- Cantidad = 6
```

Buscamos un resultado que cumpla el mismo propósito, pero con una mejor calidad al de la imagen.

3.- Lista de pendientes:

Consideramos que una manera más sencilla de lograr los objetivos del taller es especificar y listar las cosas por hacer, la mayoría de las veces cuando los integrantes de VEX se presentan para trabajar, no tienen claro que van a hacer, por eso crear una lista con los objetivos haría que el proceso de trabajo sea más veloz.

```
--- Lista de tareas ---  
1. Realizar el boceto del robot - Completada  
2. Preparar el informe del proyecto - Pendiente
```

Esta es la idea de la manera en que buscamos mostrar las tareas pendientes.

4.- Barra de progreso:

Consideramos que una manera algo creativa y que motive a nuestros integrantes de VEX es agregando una barra de progreso al código que muestre cuánto falta de los pendientes a realizar.

```
--- Barra de progreso ---  
[ ██████████ ] 50.0% (1/2 tareas)
```

Esta barra de progreso estaría enlazada a la lista de actividades pendientes, para saber como va el avance de tareas en el equipo.

Las soluciones que mostramos, son para mejorar o eliminar las problemáticas que tiene VEX y buscar un mejor desarrollo como equipo, también buscamos que no solo se quede para esta generación de VEX, sino que perdure para las siguientes generaciones y sea una herramienta de ayuda, la cual puede seguir actualizando y mejorando.

REGLAS DE NEGOCIO

Al iniciar el programa mostrará un menú el cual se divide en las cuatro secciones que mencionamos con anterioridad:

1. Lista de integrantes por sector
2. Inventario/Materiales
3. Lista de pendientes
4. Barra de progreso

Estas estarán listadas por números como se mostró anteriormente, para preguntar a qué sección desea ir, después cada sección tendrá un menú propio. *(Todo este proceso se logrará con una variable “opcion” y un input(), el input() pedirá el número de la sección y la variable guardará para después con un if mandarlo a la sección que corresponde).*

Ahora explicaremos a mayor detalle las secciones, comenzando por la “lista de integrantes por sector”:

Aquí también mostrará un menú pero ahora con dos secciones:

1. Mostrar lista
 - a. General
 - b. Administración
 - c. Mecánica
 - d. Programación
2. Agregar integrante

En la primera sección “Mostrar lista” te preguntará si quieres ver una lista general, o de algún sector en específico. *(Aquí esta parte será muy parecida al menú de inicio, pues cada número te llevará a la sección correspondiente)*

La segunda sección te hará añadir a alguien (Con input() pedirá los datos correspondientes), te solicitará su matrícula, nombre, rol que cumple en el proyecto y el semestre al que pertenece. Por último te preguntará si quieres guardar el registro, si la respuesta es “no” o “n” lo elimina y se repite el ciclo (utilizamos while), si tu respuesta es “si” o “s” se guarda el registro y te regresa a la sección de “lista de integrantes por sector”.

En la segunda sección del menú principal “Inventario/Materiales” se mostrará otro menú el cual contará con las siguientes opciones:

1. Mostrar material: Solo mostrará la lista de materiales con su nombre y cantidad, ejemplo: Tornillos de 6/8 → 26/30.
2. Modificar material: Aquí saldrá una lista de todos los materiales con los que se cuentan, estos están numerados, después un input() te preguntará qué material quieres modificar y lo seleccionamos poniendo el número del material, ya seleccionado podemos modificar la cantidad de material con la que se cuenta.

La parte tres del menú es la “lista de pendientes” esta parte del programa será muy similar a la sección anterior, serán dos opciones:

1. Mostrar pendientes:

Aquí mostrará las actividades que hay pendientes por realizar, el estado en el que se encuentra (La tarea puede estar completa, o pendiente).

2. Modificar lista de pendiente:

Aquí será muy similar a la sección anterior, pues mostrará una lista numerada de las actividades por realizar, un input() preguntará qué actividad desea cambiar su estado (los estados son: completo o pendiente).

Por último contamos con la “barra de progreso” esta al ser seleccionada mostrará cómo va el proceso de tus actividades pendientes, por lo tanto la sección tres y cuatro están conectadas, aquí la manera en la que funciona el código es que el número total de tareas pendientes, se dividirá entre el número de actividades de estado completo, el número que nos de será multiplicado por 100, para poder ir printeando la barra de tareas.

REQUERIMIENTOS FUNCIONALES

El código tiene que cumplir con una cantidad de requerimientos funcionales, en especial los siguientes:

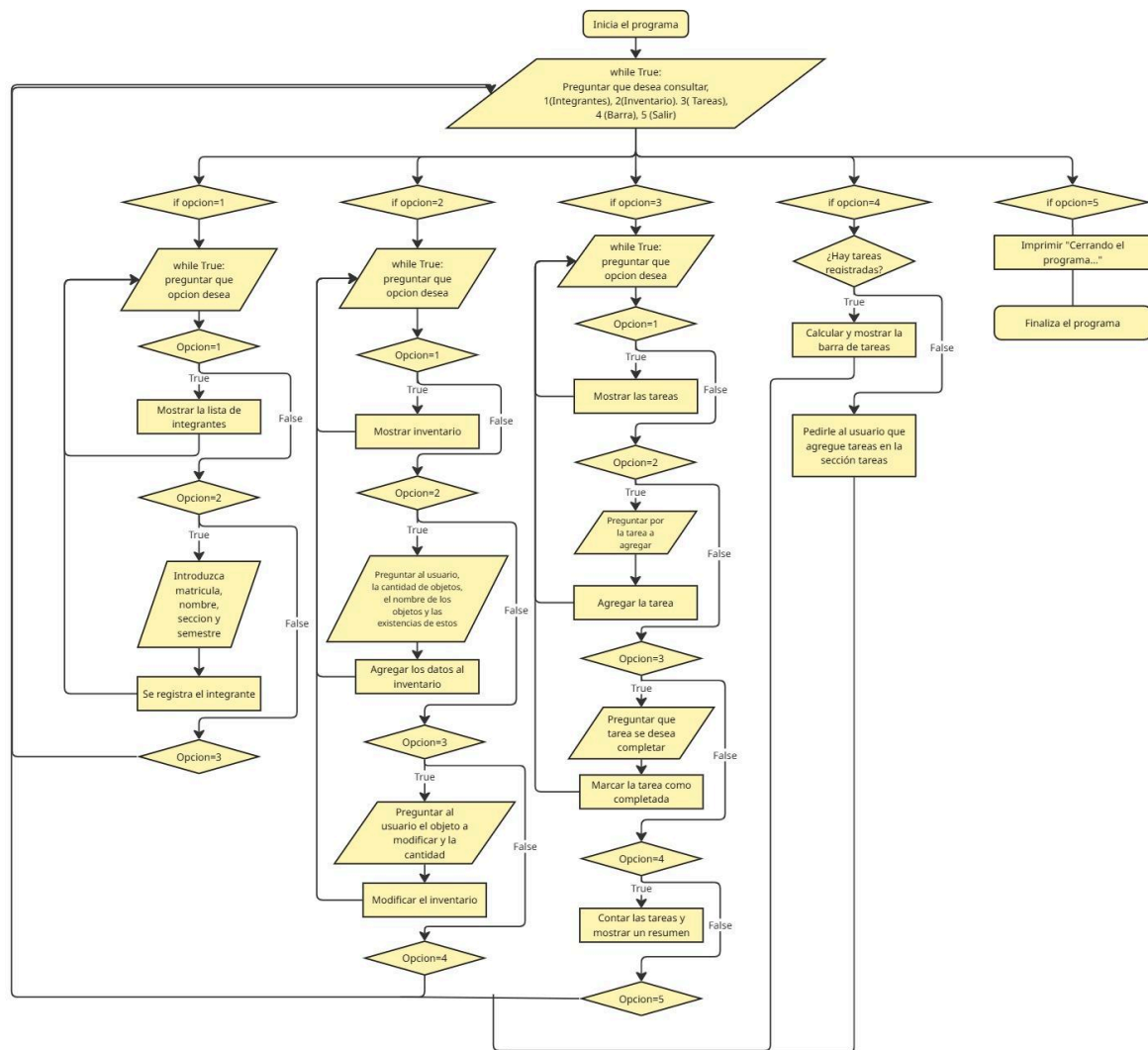
1. `menu_integrantes()`: Te muestra un menú con las opciones que puede hacer el usuario.
 - a. `mostrar_lista_integrantes()`: Aquí muestra los tipos de vista que puedes seleccionar de la lista de integrantes, y después de seleccionar el tipo de vista muestra la lista de los integrantes del modo que seleccionaste.
 - b. `agregar_integrante()`: Te pide los datos necesarios para agregar a un nuevo integrante.
2. `menu_inventario()`: Muestra el menú con las opciones que puede hacer el usuario en la sección de inventario.
 - a. `consultar_inventario()`: Aquí muestra los objetos que hay registrados y la cantidad que hay de ellos, si no hay objetos registrados te dice que no hay objetos.
 - b. `agregar_inventario()`: Primero te pregunta el número de objetos que deseas registrar, después el nombre del objeto y la cantidad de dicho objeto.
 - c. `modificar_inventario()`: Primero te pregunta por el objeto que deseas modificar para buscarlos, después te pregunta por la cantidad que se posee actualmente en el inventario y lo modifica.
3. `menu_tareas()`: En esta función se muestran las opciones que tiene el usuario dentro de esta sección del programa.
 - a. `mostrar_tareas()`: Te muestra las tareas que hay y el estado en que se encuentra (Pendiente/Completada).
 - b. `contar_tareas()`: Cuenta las tareas registradas y muestra un resumen que te informa de: total de tareas, tareas completadas y tareas pendientes.
 - c. `agregar_tarea()`: Te permite registrar una nueva tarea.
 - d. `completar_tarea()`: Aquí te muestra la lista de tareas y te pregunta qué tarea quieres marcar como completada.
4. `barra_progreso`: Cuenta las tareas pendientes y completadas y a base de eso despliega una barra que indica el progreso en las tareas.

DISEÑO DE ENTRADAS Y SALIDAS

ENTRADAS	SALIDAS
Número de la sección que desea visitar.	Te lleva al menú seleccionado y muestra el menú de dicha sección.
En la lista de integrantes, el usuario decide cuál es la vista en la que quiere ver la lista (General, Administración, Mecánica, Programación)	Muestra la tabla de integrantes como se la pediste.

Al registrar un nuevo integrante, te pide los datos necesarios (Matrícula, nombre, rol, semestre). Estos datos los almacena para después mostrarlos	Muestra en la tabla de integrantes los datos que fueron anteriormente solicitados.
En el inventario te pide que si quieres ver la lista o modificarla.	Según sea la selección mostrará la lista de material o mostrará una opción para modificar la lista.
Al modificar la lista te pide el tipo de material y la cantidad con la que se cuenta.	Esta información que se solicitó, se mostrará después en la tabla de materiales.
En la lista de pendientes, tu añades la tarea a realizar.	La tarea que vas a realizar se muestra en la tabla de lista de pendientes.
Al modificar la tabla de actividades pendientes, el usuario puede modificar el estado de la tarea.	Ese estado se guarda y se muestra en la lista de actividades pendientes.
En la barra de progreso los valores de entrada son los mismos que los de la lista de pendientes, pues calcula el porcentaje según el número de tareas pendientes y completadas.	Muestra el resultado del cálculo de porcentaje de tareas pendientes, tanto como porcentaje como barra de progreso de manera visual.

DIAGRAMA DE FLUJO



PLAN DE TRABAJO

Aquí mostraremos las tareas que iremos realizando durante el periodo de trabajo del proyecto.

Miércoles 27/08/2025:

Este día el docente de la clase “Fundamentos de la Programación” nos dio una clase de como hacer uso del GitHub, pues será nuestra principal herramienta para trabajar en este proyecto. También lo que hicimos fue delimitar la idea para llevar a cabo.

Jueves 28/08/2025:

Empezamos con el trabajo del proyecto.

Victor (SENIOR) se encargó de la creación del repositorio de GitHub y agregar las carpetas de cada archivo correspondiente.

Nicolas (MID) se encargó de la creación del esqueleto.

David (MID) se encargó de la redacción del documento.

Dennis (JUNIOR) se encargó de checar y cambiar las correcciones del esqueleto

Viernes 29/08/2025:

Este día está enfocado para mandar el primer avance del proyecto, que es un documento que cuente con los siguientes requisitos:

- Portada
- Contexto y problema
- Reglas de negocio
- Requerimientos funcionales
- Diseño de entradas y salidas
- Diagrama de flujo
- Plan de trabajo
- Asignación de roles

De eso se encargó David (MID)

Victor (SENIOR) se encargó del diagrama de flujo.

Dennis (JUNIOR) se encarga de terminar el esqueleto del código.

Semana del 01/09/2025 al 05/09/2025:

Veremos los temas del 9 al 12, gracias a estos temas vamos a ver qué información nos puede ayudar a poder mejorar la aplicación e implementarla.

Los integrantes de MID (David y Nicolas) se encargaran de seguir con el desarrollo del código, los JUNIOR (Dennis) se encargaran del testeo del código, mientras el SENIOR (Victor) verifica que el trabajo que se está haciendo se esté realizando de manera adecuada y está en los repositorios correspondientes.

Semana del 08/09/2025 al 12/09/2025:

En estas semanas veremos los temas del 13 al 16, con la información de estos temas también los integrantes de MID (David y Nicolas) mejoraran el código con los temas vistos durante las clases, los JUNIOR (Dennis) se encargaran del testeo del código y recalcar los errores que hay en el código, mientras el SENIOR (Victor) hará las correcciones que sean necesarias y que ya nos han marcado los JUNIOR.

EVIDENCIA DE PRUEBAS

```
=====
          GESTOR DE PROYECTOS VEX ROBOTICS
=====
1. Lista de integrantes
2. Inventario
3. Lista de pendientes
4. Barra de progreso
5. Salir
=====
Selecciona una opción (1-5): 1
```



```
--- MENU: LISTA DE INTEGRANTES ---
1. Mostrar lista
2. Agregar integrante
3. Volver al menú principal
Selecciona una opción (1-3): 1
```

```
--- LISTA DE INTEGRANTES ---
1. General
2. Administración
3. Mecánica
4. Programación
Selecciona una opción (1-4): 1
MATRÍCULA  NOMBRE                                ROL          SEMESTRE
-----
AL07165261  David Duran Rubio          Mecánica     Primer semestre
AL07165262  Victor Hernandez          Programación Primer semestre
AL07165263  Blanca Dennis              Administración Primer semestre
```

```
--- LISTA DE INTEGRANTES ---
1. General
2. Administración
3. Mecánica
4. Programación
Selecciona una opción (1-4): 2
MATRÍCULA  NOMBRE                                ROL          SEMESTRE
-----
AL07165263  Blanca Dennis              Administración Primer semestre
```

```
--- LISTA DE INTEGRANTES ---
1. General
2. Administración
3. Mecánica
4. Programación
Selecciona una opción (1-4): 3
MATRÍCULA  NOMBRE                                ROL          SEMESTRE
-----
AL07165261  David Duran Rubio          Mecánica     Primer semestre
```

```
--- LISTA DE INTEGRANTES ---
1. General
2. Administración
3. Mecánica
4. Programación
Selecciona una opción (1-4): 4
MATRÍCULA  NOMBRE                                ROL          SEMESTRE
-----
AL07165262  Victor Hernandez          Programación Primer semestre
```

```
--- AGREGAR NUEVO INTEGRANTE ---  
Matrícula: AL07165263  
Nombre: Nicolas Flores Verdad  
Roles disponibles:  
1. Administración  
2. Mecánica  
3. Programación  
Selecciona el rol (1-3): 3  
Semestre: Primer semestre  
¿Deseas guardar este registro? (s/n): s  
Integrante agregado correctamente.
```

```
---INVENTARIO---  
1.-Consultar inventario  
2.-Agregar objetos al inventario  
3.-Modificar las existencias de un objeto en el inventario  
4.-Salir  
¿Que desea?
```

```
---INVENTARIO---  
1.-Consultar inventario  
2.-Agregar objetos al inventario  
3.-Modificar las existencias de un objeto en el inventario  
4.-Salir  
¿Que desea?2  
¿Cuantos objetos desea registrar?3  
¿Objeto que desea registrar?Tornillos  
¿Numero de existencias?120  
¿Objeto que desea registrar?Tuercas  
¿Numero de existencias?120  
¿Objeto que desea registrar?Motor  
¿Numero de existencias?6
```

```
---INVENTARIO---  
1.-Consultar inventario  
2.-Agregar objetos al inventario  
3.-Modificar las existencias de un objeto en el inventario  
4.-Salir  
¿Que desea?1  
Objeto = Tornillos,--- Cantidad = 120  
  
Objeto = Tuercas,--- Cantidad = 120  
  
Objeto = Motores,--- Cantidad = 6
```


```
--- Menu de pendientes ---  
1. Mostrar tareas  
2. Agregar tarea  
3. Marcar tarea como completada  
4. Ver resumen  
5. Salir  
Elige una opcion: 2  
Escribe la nueva tarea: Realizar los bocetos  
Tarea agregada exitosamente
```

```
--- Lista de tareas ---  
1. Realizar los bocetos - Pendiente  
2. Ordenar el taller - Pendiente
```

```
--- Lista de tareas ---  
1. Realizar los bocetos - Pendiente  
2. Ordenar el taller - Pendiente  
Numero de la tarea que quieres marcar como completada: 1  
Tarea marcada como completada
```

```
--- Resumen ---  
Total de tareas: 2  
Tareas completadas: 1  
Tareas pendientes: 1
```

```
=====
                        GESTOR DE PROYECTOS VEX ROBOTICS
=====
1. Lista de integrantes
2. Inventario
3. Lista de pendientes
4. Barra de progreso
5. Salir
=====
Selecciona una opción (1-5): 4

--- Barra de progreso ---
[  ] 50.0% (1/2 tareas)
```

CONCLUSIONES Y APRENDIZAJES DEL EQUIPO

Conclusión y aprendizaje de Victor Andres Hernandez Castillo:

A través de este proyecto tuve la oportunidad de aplicar lo aprendido en mi clase de fundamentos de la programación en un caso práctico. Gracias a esto tuve la oportunidad no solo de reforzar mis conocimientos, sino que también, pude forjar una mejor lógica en cuanto a la forma en la que aplicaba cada recurso en busca de encontrar la solución a un problema dado. En adición, aprendí a utilizar herramientas digitales para el trabajo en equipo, en este caso Github, lo cual me ayudará en un futuro a llevar a cabo las actividades necesarias para en un entorno laboral. Por otra parte, el trabajar en equipo me ayuda a ampliar mi panorama de cómo resolver un problema, pues aun a pesar de intentar resolver un mismo problema, cada uno lo resolvía con métodos y códigos diferentes lo cual me permitió ampliar las opciones de cómo resolver problemas a base de diferentes herramientas a las que estoy acostumbrado.

Conclusión y aprendizaje de Nicolas Flores Verdad Quintana:

A lo largo del desarrollo de este proyecto pude aplicar los fundamentos de la programación en un caso práctico y real, lo cual me ayudó a comprender mejor cómo organizar y estructurar

un programa desde cero. Aprendí la importancia de dividir un problema grande en partes más pequeñas y manejables, como lo hicimos con las secciones de integrantes, inventario, pendientes y barra de progreso. También reforcé la importancia del trabajo en equipo, ya que cada integrante tuvo un rol definido y fue necesario coordinar esfuerzos para avanzar. Además, comprendí el valor de herramientas como GitHub para llevar un control del proyecto y colaborar de forma ordenada.

Conclusión y aprendizaje de Blanca Dennis Soto Rubio:

En proyectos colaborativos de programación como este, he aprendido que la clave está en la integración de ideas: no solo codificas solo, sino que construyes sobre el trabajo de otros, lo que acelera el desarrollo y resuelve problemas reales, como organizar un taller de VEX ROBOTICS. Me ha enseñado la flexibilidad al adaptar código ajeno y cómo estas experiencias fomentan habilidades blandas como el trabajo en equipo, mientras aplicas conceptos técnicos en contextos educativos y prácticos. Al final, estos proyectos me motivan a ver la programación no como un hobby aislado, sino como una herramienta colaborativa que impacta positivamente en grupos reales.

Conclusión y aprendizaje de David Durán Rubio:

Considero que lo que más aprendí en este proyecto fue como buscar resolver diversos problemas de distintas maneras, pues desde mi punto de vista vi que nosotros como equipo buscamos solucionar la problemática principal de VEX, pero cada quien tenía una idea diferente para llevar a cabo el inicio del programa, pues como podemos ver en el programa se puede notar quién hizo cada parte del código, ya que cada uno de nosotros tenemos nuestra manera de buscar soluciones y cómo aplicar los conocimientos aprendidos en clase. También, y para seguir haciendo mención de lo anterior, los conocimientos que me llevo de como hacer uso de Python considero que son suficientes para poder tener un desempeño aceptable dentro de otro proyecto o en el ámbito laboral.