

# Evaluación de un plan agregado de producción utilizando programación en Visual Basic.

## Evaluation of an aggregate production plan using Visual Basic programming.

Carlos Venturino De Coss Pérez (1).  
Tecnológico Nacional de México/Instituto Tecnológico de Tuxtla Gutiérrez.  
[carlos.dp@tuxtla.tecnm.mx](mailto:carlos.dp@tuxtla.tecnm.mx).

Edali Ramos Mijangos\* (2). TecNMéxico/I. T. de Tuxtla Gutiérrez, [edali.rm@tuxtla.tecnm.mx](mailto:edali.rm@tuxtla.tecnm.mx).

Roberto Antonio Meza Meneses (3). TecNM/I. T. de Tuxtla Gutiérrez, [roberto.mm@tuxtla.tecnm.mx](mailto:roberto.mm@tuxtla.tecnm.mx).

\*corresponding author.

Artículo recibido en enero 08, 2024; aceptado en febrero 09, 2024.

### Resumen.

*La planeación agregada se refiere a la adjudicación de las diversas fuentes de capacidad a la demanda para algún nivel de planeamiento". Este tema es de gran importancia, ya que se comprende la aplicación a una empresa, lo cual permitirá llegar a un punto de equilibrio entre los niveles de servicio, las restricciones sobre las capacidades que se fijan y los ajustes temporales a la demanda de servicios a medio plazo, para hacer mejor uso de los recursos. Así surge la idea de un software que cumpla con las características básicas para solucionar planes agregados y que tanto el docente como el alumno hagan uso de este como herramienta educativa.*

**Palabras claves:** Métodos, nivelación, plan agregado, programación, persecución.

### Abstract.

*Aggregate planning refers to the allocation of the various sources of capacity to the demand for some level of planning". This issue is of great importance, since the application to a company is understood, which will allow reaching a point of balance between the service levels, the restrictions on the capacities that are set and the temporary adjustments to the demand for services to medium term, to make better use of resources. Thus arises the idea of a software that meets the basic characteristics to solve aggregate plans and that both the teacher and the student make use of it as an educational tool.*

**Keywords:** Aggregate plan, leveling, methods, programming, persecution.

## 1. Introducción.

La implementación de planes agregados ayuda a la toma de decisiones tácticas respecto a los niveles adecuados de producción, inventarios y recursos que deben utilizarse en fabricación (Escobar, 2012). Con ello se intenta optimizar las necesidades y recursos para la producción, teniendo en cuenta siempre la eficiencia, calidad y costos totales del proceso. En definitiva, el plan agregado trata de establecer la capacidad de las operaciones, para responder a la demanda

de los clientes. Actualmente no existen softwares libres de planes agregados para fines educativos, además de que estos softwares están enfocados al sector industrial, por lo que alumnos y docentes difícilmente pueden hacer uso de estos. Analizando las problemáticas causadas por la falta de un software educativo, se plantea la siguiente pregunta: ¿Es factible el diseño funcional de un software que sea útil para docentes y alumnos que desean poner en práctica sus conocimientos y habilidades para la elaboración de planes agregados? Para la elaboración del programa, tomando en cuenta la experiencia obtenida en la asignatura "Algoritmos y Lenguaje de Programación", en la que se desarrollan habilidades de programación en Visual Basic de Excel se decidió utilizar dicho programador. El objetivo del programa es mejorar el proceso enseñanza-aprendizaje práctico del tema "Planes Agregados" que se contempla en la asignatura de Administración de las Operaciones II del plan de estudio de Ingeniería Industrial; a través de la evaluación del diseño que necesita el programa para cumplir con las funciones básicas, se genera y aportan conocimientos de las alternativas de planes agregados para los docentes y estudiantes.

## 2. Métodos.

La metodología utilizada se basó en ocho pasos para lograr obtener un programa funcional. Era necesario la validación de cada uno de estos pasos para proceder al siguiente. A continuación, se describe cada uno de estos.

### A. Capacitación.

En este paso, se tuvieron tutorías de programación, dedicando dos horas diarias durante una semana para estas reuniones. Se comprendió la teoría de la programación, para proceder con las actividades de capacitación, tales como la grabación de macros, cálculos de promedios, máximos y mínimos, creación de bordes, impresiones e inserciones de componentes.

#### Option Explicit

---

```
Private Sub cmdProcesar_Click()
    Dim i%, Opera(12) As Integer, Demanda(12) As Integer, Meses%, OperaTExt(12) As Integer 'Se coloca las variables
    Dim Dias(12) As Integer, CMaterial#, CInv#, CVenta#
    Dim CSub#, CCap#, CDes#, HTrab%, HExtras%, HDia%, CTNor#, CTEtra#, CHOciosa#
    Dim MaxSub%, II(12) As Integer, TrabCont%, TrabDesp%
    Dim CT(11) As Double, ProdNormalv5(12) As Integer, CostoTotal#, SumaTotal#, j%, ProdTEtra(12) As Integer, ProdNormal%
    Dim Colu%, SumaDemanda%, SumaDias%, Paso#, Faltante(12) As Integer, Faltante1(12) As Integer, Faltante2(12) As Integer
    Dim Fila%, CapProd%, ProdNoEfectuada%, HOCiosas%, Subcontratar(12) As Integer
```

**Figura 1.** Declaración de variables.

### B. Estructuración del software.

Se colocó dentro de tablas los conceptos que integran a un plan agregado y los necesarios para la obtención de datos. De igual forma se agregaron los controles que permiten al usuario ingresar información y elegir el método que desea utilizar. También, se realizó la declaración de variables, asignando un nombre a cada variable a utilizar en el proceso, como se ve en la Fig. 1. Posterior a este paso, el equipo se hizo la siguiente pregunta: ¿La estructuración del software es funcional?; si la respuesta es sí, se continúa al paso 3, si la respuesta es no, se regresa al paso 2 para verificar.

### C. Programación de métodos generales.

En el proceso de planificación agregada se tienen en cuenta diversos factores, que han de combinarse. Entre ellos: la producción normal, las horas extra necesarias, posibles retrasos en entregas, etc. Con todo ello se intentan optimizar las necesidades y recursos para la producción, teniendo en cuenta siempre la eficiencia, calidad y costos totales del proceso. En definitiva, el plan agregado trata de establecer la capacidad de las operaciones, para responder a la demanda de los clientes.

Existen dos tipos de estrategias de planificación agregada disponibles:

- Estrategia de nivelación:

Busca mantener un ritmo de producción y fuerza de trabajo en nivel constante. En esta estrategia, la organización requiere de la previsión de la demanda ya sea para aumentar o disminuir la producción en función de reducciones o incrementos en la demanda de los clientes. La ventaja de esta estrategia es que el nivel de mano de obra es constante (Conexión ESAN. 2016).

- Estrategia de persecución:

Se trata de encajar, en forma dinámica, la demanda con la producción. Para lograrlo se modifica el nivel de la fuerza de trabajo o la tasa de producción. La ventaja de la estrategia de persecución es que logra bajos niveles de inventario (Salazar, 2019).

Para desarrollar el método general de nivelación se comienza estableciendo la fórmula que determina el número de operarios con los que se trabajará en todos los periodos, ya que el método de nivelación busca mantener un ritmo de producción y fuerza de trabajo en nivel constante. En la ecuación (1) se desarrolla la fórmula utilizada para que el programa calcule este valor.

$$\text{Operario} = (\text{Demanda total de plan de producción} + \text{Faltante del periodo anterior} - \text{Inventario del periodo anterior}) / (\text{Días laborables de todo el periodo de planeación} * \text{Horas de trabajo en el día} * \text{capacidad de trabajo del Operario en horas}) \dots\dots\dots (1)$$

```

If Opera(i) >= Opera(i - 1) Then
    TrabCont = Opera(i) - Opera(i - 1)
    Cells(Fila + 5, Colu) = TrabCont
    CT(1) = TrabCont * CCap
    Cells(Fila + 6, Colu) = CT(1)
Else
    TrabDesp = Opera(i - 1) - Opera(i)
    Cells(Fila + 7, Colu) = TrabDesp
    CT(2) = TrabDesp * CDes
    Cells(Fila + 8, Colu) = CT(2)
End If
    
```

**Figura 2.** Programación de número de operarios.

El método de persecución permite modificar el número de operarios de cada periodo, cuando sea necesario, a manera de optimizar la producción (Vergara, 2006). En la ecuación (2) se encuentra la fórmula utilizada para este método.

$$\text{Operario} = (\text{Demanda de un periodo} + \text{Faltante del periodo anterior} - \text{Inventario del periodo anterior}) / (\text{Días laborables del periodo} * \text{Horas de trabajo en el día} * \text{capacidad de trabajo del Operario en horas}) \dots\dots\dots (2)$$

Ahora se debe identificar si existirán despidos o contrataciones; esto aplica para ambos métodos. Si el número de operarios necesarios es mayor al número de operarios disponibles entonces se realizará la contratación de los operarios faltantes, en caso contrario se realizará el despido de los operarios excedentes; se calculan los costos correspondientes. En la Fig. 2 se encuentran los códigos utilizados para cada caso.

### 3. Desarrollo.

#### D. Programación de métodos específicos.

Como se mencionó anteriormente, para la planeación agregada se toman en cuenta diferentes factores que han de combinarse o separarse según las necesidades particulares del usuario.

Los factores determinísticos identificados para el programa desarrollado fueron: inventario, faltante, subcontratación, tiempo extra y pedidos pendientes.

Con base en esto, se establecieron los métodos específicos siguientes:

- Persecución con inventario.
- Nivelación sin faltantes recurriendo a tiempo extra.
- Nivelación sin faltantes recurriendo a subcontratación.

A partir de la programación “Método de persecución” se dio inicio al método específico “Persecución con inventario”. Se agregó el cálculo de la producción normal. Para ello basta medir cuánto tiempo se gasta para producir determinado ítem y multiplicar por la disponibilidad en un día entero. Considerando también el cálculo de costo de producción normal, CT(4), obtenido de la multiplicación de la producción normal por el costo por el material.

Posteriormente, se agregó el cálculo del inventario sumando la producción normal al inventario anterior menos la demanda pronosticada para el periodo actual menos el faltante del periodo anterior. Para obtener el costo de inventario, CT(5), se multiplica el resultado del inventario por el costo por mantener inventario. Véase la Fig. 3.

$$\begin{aligned} II(i) &= \text{ProdNormal} + II(i - 1) - \text{Demanda}(i) - \text{Faltante}(i - 1) \\ CT(5) &= II(i) * CInv \\ \text{Cells}(\text{Fila} + 15, \text{Colu}) &= II(i) \\ \text{Cells}(\text{Fila} + 16, \text{Colu}) &= CT(5) \end{aligned}$$

**Figura 3.** Programación de inventarios.

Después de esto se calcula el costo del trabajador en tiempo normal, CT(8). El cálculo es resultado del producto del costo de la producción normal por el número de operarios por las horas trabajadas al día por los días trabajados.

Se realiza la suma de todos los costos para generar el costo total por el periodo de producción. Finalmente, se calcula la suma de todos los periodos, generando el costo final de producción.

A partir de la programación “Método de nivelación” se dio inicio al método específico “Nivelación sin faltantes recurriendo a tiempo extra”. En este método se utilizan dos tipos de producciones (en tiempo normal y en tiempo extra) y tres costos de materiales (en tiempo normal, en tiempo extra y material total). Se obtiene el cálculo de la producción normal y el cálculo de costo de producción normal, CT(10), obtenido de la multiplicación de la producción normal por el costo por el material en tiempo normal.

Posteriormente, se procedió a calcular el faltante 1. Si el resultado es mayor a cero, se imprime el resultado; también se agrega el cálculo de los operarios en tiempo extra. En la Fig. 4 se observa la programación de esta actividad.

Si los operarios en tiempo extra son mayores a los operarios en tiempo normal, se imprimen los mismos operarios en tiempo normal; si son menores, se imprimen los operarios en tiempo extra.

Se agrega el cálculo del costo por trabajador en tiempo extra, CT(3), la producción en tiempo extra y se imprimen estos cálculos.

Para calcular si aún existe faltante, posterior al denominado faltante 1, se procede a calcular el faltante final y su costo, CT(7).

Después, se agregó el cálculo del inventario. Para obtener el costo de inventario se multiplica el resultado del inventario por el costo por mantener inventario. Para finalizar este método, se calculan todos los costos de producción en tiempo extra.

A partir de la programación “Método de nivelación” se dio inicio al método específico “Nivelación sin faltantes recurriendo a subcontratación”. Se obtiene el cálculo de la producción normal y el cálculo de costo de producción normal, CT(10), utilizando los mismos códigos del método de nivelación sin faltantes recurriendo a tiempo extra.

En este caso, al no utilizar tiempo extra, se recurre a la subcontratación y se calcula el costo, CT(6), en donde si el faltante 1 es mayor a la máxima subcontratación, se imprime la misma máxima subcontratación; si es menor, se imprime el faltante 1.

Se imprimen los costos por subcontratar en caso de realizar esta acción. De lo contrario, se imprime cero en subcontrataciones y en costo por subcontratar, CT(6).

```
Faltante1(i) = Demanda(i) + Faltante(i - 1) - ProdNormalv5(i) - II(i - 1)
If Faltante1(i) > 0 Then
  Cells(Fila + 10, Colu) = Faltante1(i)
  If Faltante1(i) > 0 Then
    OperaTExt(i) = Faltante1(i) / (HDia / HTrab)
```

**Figura 4.** Programación de operarios en tiempo extra.

```
Option Explicit
Private Sub cmdProcesar_Click()
  Dim i%, Opera(100) As Integer, Demanda(100) As Integer, Periodos%, OperaTExt(100) As Integer 'Se coloca las variables
  Dim Dias(100) As Integer, CMaterial#, CInv#, CVenta#
  Dim CSub#, CCap#, CDes#, HTrab#, HExtras%, HDia#, CTNor#, CTEExtra#, CHOciosa#
  Dim MaxSub%, II(100) As Integer, TrabCont%, TrabDesp%
  Dim CT(11) As Double, ProdNormalv5(100) As Integer, CostoTotal#, SumaTotal#, j%, ProdTEExtra(100) As Integer, ProdNormal%
  Dim Colu#, SumaDemanda#, SumaDias#, Paso#, Faltante(100) As Integer, Faltante1(100) As Integer, Faltante2(100) As Integer
  Dim Fila#, CapProd#, ProdNoEfectuada#, HOciosa%, Subcontratar(100) As Integer
```

**Figura 5.** Modificación de variables.

E. Depuración del código del programa.

Para confirmar que el software funciona correctamente, se hizo una revisión de todos los códigos realizados, considerando posibles modificaciones, mejoras o eliminaciones,

Entre las correcciones que se realizaron se encuentran cambios de variables, intercambio o agregación de filas en la matriz de resultados, y, por lo tanto, modificaciones en los códigos que referencian la matriz.

La variable “Meses” se modificó por “Periodos” debido a que el usuario no necesariamente usará meses como datos de entrada; pueden ser días, semanas o incluso años. Debido al cambio anterior, se modificaron todas las variables declaradas como arreglos, cambiando el número 12, que representaba los meses de un año, por 100, eligiéndolo como un número grande para cubrir los periodos que desee el usuario; este cambio se observa en la Fig. 5.

Al llegar al método de nivelación sin faltantes recurriendo a tiempo extra, el equipo se percató de que habría que modificar la matriz de resultados a causa de los costos de material. En el inicio del desarrollo del programa, se colocó únicamente el costo total del material; en el método de nivelación sin faltantes recurriendo a tiempo extra, se menciona

que en este método se utilizan dos tipos de producciones (en tiempo normal y en tiempo extra) y tres costos de materiales (en tiempo normal, en tiempo extra y material total), por lo que se agregaron estos a la matriz de resultados.

F. Diseño visual.

En este punto, posterior a las correcciones, se considera que el software ya es funcional y realiza los cálculos esperados de la forma y en el momento adecuado. Sin embargo, aún no se tiene una presencia visual ideal, lo que podría ocasionar que el usuario no entienda la forma en que se debe de usar el programa para que este realice los cálculos para cada caso en particular.

Por lo anterior se procedió a, principalmente, darles orden visual a los elementos presentados en el programa. En la Fig. 6 se presenta la apariencia general final del programa.

Concepto	1	2	3	4	5	6
Pronóstico	1800	1500	1100	900	1100	1600
Días	22	19	21	21	22	20
Número de Trabajadores	40	50	32	27	31	60
Contrato	0	30			4	19
\$ Contrato	\$0	\$2,000			\$800	\$3,800
\$ Despido			18	5		
\$ Despido			\$4,500	\$1,250		
Producción normal	1408	1520	1076	908	1092	1600
\$ Material en tiempo normal						
Faltante después de producción en tiempo normal						
Trabajador Horas Extras						
Producción en tiempo extra						
\$ Trabajador en tiempo extra						
\$ Material en tiempo extra						
\$ Material total	\$140,800	\$152,000	\$107,600	\$90,800	\$109,200	\$160,000
Faltante después de producción en tiempo extra						
Subcontratación						
\$ Subcontratación						
Faltante (Pedidos por cubrir)						
\$ Faltante						
Inventario	8	28	4	12	4	4
\$ Inventario	\$12	\$42	\$6	\$18	\$6	\$6
\$ Trabajador en tiempo normal	\$28,160	\$30,400	\$21,504	\$18,144	\$21,824	\$32,000
Producción no efectuada						
Hora ociosa						
\$ Hora ociosa						
TOTAL	\$168,972	\$184,442	\$133,610	\$110,212	\$131,830	\$195,806
<b>SUMA TOTAL</b>	<b>\$924,872</b>					

Figura 6. Apariencia general del programa (Interface).

Planeación Agregada

MÉTODO

NÚMERO DE PERIODOS

Insertar los datos. Si no se utiliza cierto dato, insertar 0.

\$ Material <input type="text"/> por unidad	Horas por día <input type="text"/>
\$ Inventario <input type="text"/> por unidad	\$ Tiempo normal <input type="text"/> por hora
\$ Ventas perdidas <input type="text"/> por unidad	\$ Tiempo extra <input type="text"/> por hora
\$ Subcontratación <input type="text"/> por unidad	Operarios disponibles <input type="text"/>
\$ Capacitación <input type="text"/> por operario	Máximo a subcontratar <input type="text"/> unidades
\$ Despido <input type="text"/> por operario	Inventario inicial <input type="text"/> unidades
Horas de trabajo <input type="text"/> por unidad	Pedidos pendientes <input type="text"/> unidades
Horas extras <input type="text"/> por operario al mes	\$ Hora ociosa <input type="text"/>

Dar clic en Guardar Datos, de lo contrario no se podrá procesar la planeación agregada.

Figura 7. Formulario de inicio (Interface).

Se agregaron botones de inicio y formularios. Para crear el formulario se usó la herramienta UserForm. Esto permite agregar controles a una ventana e ir modificando sus propiedades; con estas propiedades, se coloca un nombre a cada control, permitiendo alguna futura referencia. El formulario de la Fig. 7 es el primero que ve el usuario.

#### G. Validación preliminar.

La validación de software es un proceso que demuestra a partir de pruebas o documentos que el programa cumple con las funciones para las cuales fue diseñado, de acuerdo con las especificaciones de los requisitos del usuario, y con la garantía de seguridad y confiabilidad de la información.

Se decidió realizar una validación preliminar, la cual tiene como objetivo verificar que el programa realice todas las funciones básicas relacionadas con la elaboración de planes agregados (y sus distintos métodos) de manera correcta.

La manera más eficiente para llevar a cabo esta validación fue comparando con una plantilla de Excel con más de cincuenta ejercicios resueltos proporcionada por el asesor de este proyecto en la materia Administración de Operaciones, mismos que son utilizados para el proceso de enseñanza – aprendizaje en el curso mencionado

#### H. Operación del programa.

Para interpretar y desenvolver el programa, la manera en la que la didáctica del diseño se desarrolla, alcanza una serie de pasos secuenciales metódicamente analizados.

Principalmente, al abrir el programa se debe dar clic al botón “Iniciar planeación agregada”, en donde posteriormente se va a abrir una ventana donde es necesario incluir el método a seguir, así como los datos del ejercicio; en dado caso no sea necesario utilizar algún dato en particular, se debe insertar el valor 0 en la casilla correspondiente.

Posteriormente, se debe dar clic al botón “Guardar datos”, para que de esta manera sea redirigido a la matriz de resultados. En ese momento, el programa muestra el método anteriormente seleccionado y el número de periodos, permaneciendo en espera de la preparación de captura de los últimos datos para poder desarrollar el plan agregado en su totalidad. Debe dar clic al botón “Preparar captura de datos”, en donde el programa crea la matriz a la dimensión correcta según los periodos previamente elegidos y, resalta en color verde, donde se deben agregar los valores del pronóstico y los días por cada periodo. Finalmente, para completar el proceso en el programa, se debe dar clic al botón “Procesar”, el cual se encarga de que el programa desarrolle e interprete los resultados correspondientes del plan agregado.

En dado caso, el usuario quiera corregir algún valor de los conceptos pronóstico o días, el usuario puede cambiar sus datos directamente en las casillas correspondientes de pronóstico y días; o, si el usuario quiere corregir estos valores y de igual manera eliminar y limpiar todos los datos, debe dar clic nuevamente al botón “Preparar captura de datos”, dando opción a agregar los nuevos datos deseados.

Si, por otra parte, el usuario se decide por utilizar otro método, en la misma sección debe dar clic al botón “¿Desea utilizar otro método?”, y posteriormente elegir el método deseado. Después de esto, únicamente será necesario dar clic al botón “Procesar”, para obtener los resultados del ejercicio

### **Conclusiones.**

De esta forma, se logró desarrollar al cien por ciento un software didáctico que resuelve las funciones básicas para evaluar alternativas de planes agregados con ayuda del programador Visual Basic en Excel. Este presenta una entrada amigable, lo cual permitirá que su manejo sea sencillo y comprensible para los usuarios, buscando fomentar la mejora del proceso enseñanza-aprendizaje del tema.

El programa desarrollado para la resolución de planes agregados es una herramienta útil y funcional, ya que cumple con las expectativas de dar solución a problemas planteados en la materia Administración de las Operaciones II del tema “Planes Agregados”.

El programa es susceptible a mejoras referentes a su aplicación, programación o diseño bajo otras estrategias para hacer planes de producción o bien formular el modelo óptimo. Estas aportaciones podrán ser realizadas por alumnos y docentes de la institución; en caso de ser alumnos, deben haber cursado la materia de “Algoritmos y Lenguaje de Programación” o afín, o contar con una base de conocimientos sólida en programación con Visual Basic de Excel.

### Créditos.

Los autores agradecen al Tecnológico Nacional de México campus Instituto Tecnológico de Tuxtla Gutiérrez por el apoyo del proyecto a través de la convocatoria de Investigación del 2021 al 2023.

### Referencias bibliográficas.

**Conexión ESAN. (09 de marzo de 2016).** ¿De qué hablamos cuando nos referimos al planeamiento agregado? Obtenido de [www.esan.edu](https://www.esan.edu): <https://www.esan.edu.pe/apuntes-empresariales/2016/03/de-que-hablamos-cuando-nos-referimos-al-planeamiento-agregado/>.

**Escobar, E., Velázquez, S., Aldape, A., & García, J. (03 de marzo de 2012).** Modelo para el Desarrollo de Planes de Producción Agregada para la Industria Refresquera. Obtenido de Instituto Tecnológico de Cd. Juárez: [https://www.academia.edu/8862379/Modelo\\_para\\_el\\_Desarrollo\\_de\\_Planes\\_de\\_Producci%C3%B3n\\_Agregada\\_para\\_la\\_Industria\\_Refresquera](https://www.academia.edu/8862379/Modelo_para_el_Desarrollo_de_Planes_de_Producci%C3%B3n_Agregada_para_la_Industria_Refresquera).

**Salazar, B. (16 de junio de 2019).** Planeación agregada. Obtenido de Ingeniería Industrial: <https://www.ingenieriaindustrialonline.com/produccion/planeacion-agregada/>.

**Vergara, R. (29 de agosto de 2006).** Desarrollo de los Procesos de Planeación y Programación de la Producción en el Sector Manufacturero de la PYME en Bogotá. Obtenido de Desarrollo Sostenible y Tecnología: <https://repository.unilibre.edu.co/bitstream/handle/10901/24886/PROYECTO%20DE%20GRADO.pdf?sequence=1>.

### Información de los autores.

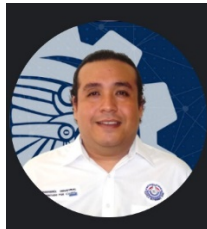


**Carlos Venturino De Coss Pérez**, Profesión Ingeniero Industrial en 2003 del Instituto Tecnológico de Tuxtla Gutiérrez y es graduado como Maestro en Ciencias en Ingeniería Industrial en 2005 por el Instituto Tecnológico de Querétaro. Actualmente Docente desde 2008 en el programa educativo de Ingeniería Industrial en el Instituto Tecnológico de Tuxtla Gutiérrez y adscrito al Departamento de Ing. Industrial. Cuenta con Perfil PRODEP.





**Edali Ramos Mijangos**, Profesión Ingeniero Industrial en junio 2004 del Instituto Tecnológico de Tuxtla Gutiérrez y es graduado como Maestra en Administración pública en 2023 por el Instituto de Estudios Superiores de Chiapas. Actualmente Docente desde 2008 en el programa educativo de Ingeniería Industrial en el Instituto Tecnológico de Tuxtla Gutiérrez y adscrito al Departamento de Ing. Industrial. Actualmente estudiando el Doctorado en Administración de Negocios en el Colegio de Formación Educativa TENAM, S.C.



**Roberto Antonio Meza Meneses**, Profesión Ingeniero Industrial en 2004 del Instituto Tecnológico de Tuxtla Gutiérrez. Graduado como Maestro en Ciencias en Ingeniería Industrial en 2008 por el Instituto Tecnológico de Orizaba. Graduado como Doctor en Administración en 2013 por la Universidad del Sur. Actualmente docente desde 2009 en el programa educativo de Ingeniería Industrial del Instituto Tecnológico de Tuxtla Gutiérrez y adscrito al Departamento de Ingeniería Industrial.