

# Avance de Proyecto - Diseño

Universidad Politécnica de Victoria  
Estructura de Datos

## Integrantes:

Georgina Reta Limas  
Jared de Jesus Olazaran Lopez  
Melissa Jazmin Torres Martinez  
Angel Gabriel Coronado Sánchez

## 1. Especificación del problema de horario de la carrera de ITI

### 1.1. Descripción General del Problema

El problema consiste en asignar un conjunto de clases a franjas horarias, considerando la disponibilidad de profesores y las necesidades curriculares de cada grupo. Este problema se enmarca dentro del *University Time Tabling Problem* (UTTP) aplicado específicamente a la carrera de Ingeniería en Tecnologías de la Información e Innovación Digital de la Universidad Politécnica de Victoria.

### 1.2. Elementos del Problema

#### 1.2.1. Conjuntos Básicos

- **Grupos ( $G$ ):** Conjunto de grupos estudiantiles organizados por cuatrimestre y turno.
  - Ejemplo: ITI 1-1, ITI 2-1, ITI 2-2, ITI 2-3, ITI 4-1, ITI 5-1, ITI 5-2, ITI 5-3, ITI 7-1, ITI 8-1, ITI 8-2
- **Materias ( $M$ ):** Conjunto de asignaturas del plan de estudios.
  - Cada materia tiene asociada una cantidad específica de horas semanales.
- **Profesores ( $P$ ):** Conjunto de docentes disponibles.
  - Cada profesor tiene competencias específicas para impartir ciertas materias.
  - Cada profesor tiene una carga horaria máxima semanal.
- **Franjas Horarias ( $T$ ):** Conjunto de intervalos de tiempo disponibles.
  - Organizadas por día de la semana (Lunes a Viernes).
  - Distribuidas en turnos: Matutino (7:00-13:55) y Vespertino (14:00-20:45).
  - Duración típica: 55 minutos por sesión.

### 1.2.2. Parámetros

- $H_m$ : Horas semanales requeridas para la materia  $m$ .
- $C_p$ : Capacidad horaria del profesor  $p$  (horas disponibles por semana).
- $\text{COMP}_{p,m}$ : Competencia del profesor  $p$  para impartir la materia  $m$  (binario: 1 si puede, 0 si no).
- $\text{disp}_{p,t}$ : Disponibilidad del profesor  $p$  en la franja horaria  $t$  (binario: 1 si está disponible, 0 si no).

### 1.3. Variables de Decisión

- $x_{g,m,p,t}$ : Variable binaria que toma el valor 1 si el grupo  $g$  tiene la materia  $m$  con el profesor  $p$  en la franja horaria  $t$ , y 0 en caso contrario.

### 1.4. Restricciones

Estas restricciones deben cumplirse obligatoriamente para obtener una solución factible:

#### 1.4.1. Cobertura Curricular

- **R1**: Cada materia asignada a un grupo debe cumplir exactamente con el número de horas semanales requeridas.
- **R2**: Todas las materias del plan de estudios de cada grupo deben ser cubiertas.

#### 1.4.2. Restricciones de Profesores

- **R3**: Un profesor no puede estar en dos lugares al mismo tiempo (no puede impartir dos clases simultáneamente).
- **R4**: Un profesor solo puede impartir materias para las cuales está calificado.
- **R5**: La carga horaria total asignada a un profesor no puede exceder su capacidad máxima.
- **R6**: Un profesor solo puede ser asignado en las franjas horarias en las que está disponible.

#### 1.4.3. Restricciones de Grupos

- **R7**: Un grupo no puede tener dos materias al mismo tiempo.
- **R8**: Un grupo no puede tener clases con el mismo profesor en materias diferentes al mismo tiempo.

#### 1.4.4. Restricciones de Coherencia

- **R9**: Una materia de un grupo específico debe ser impartida por un único profesor.
- **R10**: Las sesiones de una misma materia deben distribuirse a lo largo de la semana.

## 1.5. Consideraciones Especiales

### 1.5.1. Turnos

- Turno Matutino: 7:00 - 13:55
- Turno Vespertino: 14:00 - 20:45
- Algunos grupos están designados específicamente para un turno.

### 1.5.2. Periodicidad

- El horario se programa típicamente para un cuatrimestre (aproximadamente 15 semanas).
- El ciclo se repite semanalmente.

## 2. Diseño de Componentes del Sistema

### 2.1. Diseño Conceptual de Componentes

Basado en la especificación del problema UTTP, se identifican 4 componentes principales:

#### 2.1.1. Generador de Horarios

**Descripción Conceptual:** Implementa el algoritmo principal para crear asignaciones, buscando la mejor solución que satisfaga todas las restricciones (R1-R10).

**Responsabilidad:** Resolver el problema de optimización combinatoria asignando cada materia de cada grupo a un profesor calificado y una franja horaria disponible.

**Interfaz Pública:**

- `GenerarHorario()` : `Horario` - Ejecuta el algoritmo de generación y retorna un horario completo.
- `CargarConfiguracion()` : `void` - Carga los datos base: grupos, materias, profesores, franjas.

**Algoritmo:**

1. Itera sobre cada grupo  $g \in G$
2. Para cada materia  $m$  asignada al grupo:
  - Busca profesores  $p$  donde  $\text{COMP}_{p,m} = 1$
  - Busca franjas  $t$  donde  $\text{disp}_{p,t} = 1$
  - Crea asignación  $x_{g,m,p,t} = 1$  respetando restricciones

### 2.1.2. Validador de Restricciones

**Descripción Conceptual:** Verifica que una asignación propuesta o un horario final no incumpla ninguna regla del problema (R1-R10).

**Responsabilidad:** Aplicar las 10 restricciones del documento de especificación para garantizar soluciones factibles.

**Interfaz Pública:**

- ValidarAsignacion(asignacion) : boolean - Retorna true si la asignación es válida.
- ObtenerRestricciones(ID) : string - Consulta la definición de una restricción específica.

**Implementación de Restricciones:**

$$R1: \sum_{t,p} x_{g,m,p,t} = H_m \quad \forall g, m$$

$$R3: \sum_{g,m} x_{g,m,p,t} \leq 1 \quad \forall p, t$$

$$R7: \sum_{m,p} x_{g,m,p,t} \leq 1 \quad \forall g, t$$

### 2.1.3. Gestor de Asignaciones

**Descripción Conceptual:** Maneja el estado temporal de las asignaciones durante el proceso de generación y guarda el resultado final.

**Responsabilidad:** Mantener la estructura de datos que representa las variables de decisión  $x_{g,m,p,t}$  durante la construcción del horario.

**Interfaz Pública:**

- GuardarAsignacion(asignacion) : void - Almacena un resultado temporal.
- ObtenerHorarioFinal() : Horario - Recupera el horario completo.

**Implementación:** Mantiene una matriz o estructura de datos para  $x_{g,m,p,t}$  con métodos de consulta eficientes por grupo, profesor y franja.

### 2.1.4. Gestor de Datos

**Descripción Conceptual:** Actúa como gateway entre la Lógica de Negocio y la Base de Datos, gestionando todas las operaciones CRUD.

**Responsabilidad:** Proporcionar acceso a conjuntos  $G$ ,  $M$ ,  $P$ ,  $T$  y parámetros  $H_m$ ,  $C_p$ ,  $COMP_{p,m}$ ,  $disp_{p,t}$ .

**Interfaz Pública:**

- Guardar(modelo) : void - Persiste una entidad en la base de datos.
- ObtenerPorID(tipo, ID) : Entidad - Recupera una entidad específica.

## 2.2. Modelos de Datos

### 2.2.1. Grupos ( $G$ )

Atributos:

- id: string (ej. "ITI 1-1")
- nombre: string
- cuatrimestre: int
- turno: enum {MATUTINO, VESPERTINO}
- materiasAsignadas: Lista<Materia>

### 2.2.2. Materias ( $M$ )

Atributos:

- id: string
- nombre: string
- $H_m$ : int (horas semanales requeridas)

### 2.2.3. Profesores ( $P$ )

Atributos:

- id: string
- nombre: string
- $C_p$ : int (capacidad horaria semanal)
- $\text{COMP}_{p,m}$ : Map<Materia, boolean> (competencias)
- $\text{disp}_{p,t}$ : Map<FranjaHoraria, boolean> (disponibilidad)

### 2.2.4. Franjas Horarias ( $T$ )

Atributos:

- id: string
- dia: enum {LUNES, MARTES, MIERCOLES, JUEVES, VIERNES}
- horaInicio: Time
- horaFin: Time
- turno: enum {MATUTINO, VESPERTINO}

**Nota:** Turno Matutino: 7:00-13:55, Vespertino: 14:00-20:45. Duración típica: 55 minutos.

### 2.2.5. Asignación (Variable de Decisión)

Representa:  $x_{g,m,p,t}$

Atributos:

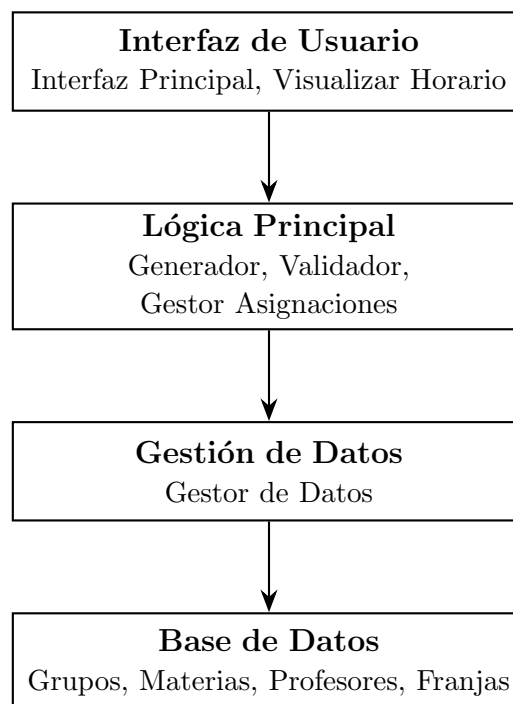
- grupo: Grupo ( $g$ )
- materia: Materia ( $m$ )
- profesor: Profesor ( $p$ )
- franja: FranjaHoraria ( $t$ )
- valor: boolean (1 si asignado, 0 si no)

### 2.2.6. Horario (Solución)

Atributos:

- periodo: string
- asignaciones: Lista<Asignacion> (todas donde  $x = 1$ )
- esFactible: boolean (cumple R1-R10)

## 2.3. Arquitectura del Sistema



## 2.4. Mapeo de Restricciones a Componentes

El Validador de Restricciones implementa las 10 restricciones:

**Cobertura Curricular:**

- R1: Cada materia cumple exactamente  $H_m$  horas

- R2: Todas las materias del plan cubiertas

**Restricciones de Profesores:**

- R3: No dos clases simultáneas
- R4: Solo materias donde  $\text{COMP}_{p,m} = 1$
- R5: Carga total  $\leq C_p$
- R6: Solo asignar donde  $\text{disp}_{p,t} = 1$

**Restricciones de Grupos:**

- R7: No dos materias simultáneas
- R8: No mismo profesor en materias diferentes al mismo tiempo

**Restricciones de Coherencia:**

- R9: Un único profesor por grupo-materia
- R10: Distribución semanal de sesiones

## 2.5. Consideraciones de Implementación

**Algoritmo sugerido:**

1. Cargar datos:  $G$ ,  $M$ ,  $P$ ,  $T$  y parámetros
2. Para cada grupo  $g \in G$ :
  - Para cada materia  $m$  asignada a  $g$ :
  - Asignar  $H_m$  sesiones cumpliendo R1-R10
3. Validar horario completo
4. Guardar solución factible

**Periodicidad:** El horario se programa para un cuatrimestre (aproximadamente 15 semanas) con ciclo semanal repetitivo.