

EVALUACIÓN	Obligatorio	GRUPO	Todos	FECHA	Mar. 2024
MATERIA	Diseño y Desarrollo de Aplicaciones				
CARRERA	Analista en Tecnologías de Información / Analista Programador				
CONDICIONES	<p>- Puntos: 40 – Mínimo 0</p> <p>- Fecha de entrega: 10/06/2024 hasta las 21:00 horas en gestion.ort.edu.uy.</p> <p>LA ENTREGA SE REALIZA EN FORMA ONLINE EN ARCHIVO NO MAYOR A 40MB EN FORMATO ZIP, RAR O PDF.</p> <p>IMPORTANTE:</p> <ul style="list-style-type: none"> - Inscribirse - Formar grupos (Individual o hasta 2 alumnos) - Aquellos de ustedes que presenten alguna dificultad con su inscripción o tengan inconvenientes técnicos, por favor contactarse con el Coordinador o Coordinación adjunta antes de las 20:00hs. del día de la entrega, a través de los mails gervaz@ort.edu.uy, alamon@ort.edu.uy y fernandez_ma@ort.edu.uy, o telefónicamente al 29021505 - int 1156 u 1138 - Subir el trabajo a Gestión antes de la hora indicada, ver hoja al final del documento: "RECORDATORIO" 				



Modelar e implementar un prototipo de sistema de parkings

Nota preliminar

Como se especifica después, el prototipo deberá integrarse con un simulador que lo provee el docente. El simulador funciona con una técnica de compresión de tiempo, lo que permitirá representar eventos que tomarían varias horas en segundos. Para este ejercicio se debe considerar una escala de tiempo 1:60, es decir, 1 segundo en el simulador representa 1 minuto en la realidad.

En adelante, cada referencia a una Unidad Temporal, indicada como <UT>, se debe entender que representa 1 segundo¹. No obstante, es recomendable que el alumno utilice una variable interna con esa equivalencia.

Dominio del negocio

El sistema permite administrar múltiples parkings independientes. Cada parking tiene un nombre, una dirección, un tarifario y un conjunto de cocheras.

El tarifario establece un precio por <UT> para cada tipo de vehículo y puede variar de un parking a otro.

Por cada vehículo que ingresa al parking se crea un registro de la estadía, indicando la fecha y hora de entrada, la fecha y hora de salida y la cochera utilizada. También se llevará registro del valor facturado por esa estadía.

¹ Ejemplo, si la letra refiere 60<UT> eso equivale a 60 unidades temporales, es decir, 60 segundos en la escala del simulador.

Los propietarios y los vehículos ya se encuentran registrados y pueden acceder a cualquiera de los parkings. De los propietarios se sabe su cédula y nombre completo y su lista de vehículos. Los propietarios tienen una cuenta corriente para el pago de las estadías de sus vehículos. Cada vehículo está asociado a un único propietario.

De cada vehículo se cuenta con su número de patente, tipo de vehículo y, opcionalmente, etiquetas.

Las etiquetas son atributos que describen a un vehículo con características especiales y que permiten habilitar su estacionamiento en cocheras igualmente especiales. Ejemplos de etiquetas son: conductor discapacitado, vehículo eléctrico, etc. Cada vehículo puede tener cero o múltiples etiquetas.

Las cocheras tienen un código (irrepetible incluso entre todos los parkings), un estado (libre u ocupada) y opcionalmente también pueden tener etiquetas. El mismo listado de etiquetas se puede asignar a los vehículos y a las cocheras.

Para estacionar en una cochera que tiene etiquetas, los vehículos deberán contar con al menos las mismas etiquetas. De lo contrario estarán en infracción y recibirán una multa (adicional a lo que hubiera facturado si hubieran estacionado correctamente):

- Estacionar incorrectamente en una cochera para discapacitados: la multa es un monto fijo de \$250.
- Estacionar incorrectamente en una cochera para vehículos eléctricos: la multa es equivalente al 50% del monto de la estadía.
- Estacionar incorrectamente en una cochera para empleados internos: la multa es de \$ 1 por cada 10 <UT> de estadía.

Las infracciones son acumulativas.

El valor a facturar para cada estadía se calcula de la siguiente manera:

$$\text{Valor de estadía (\$)} = (\text{PB} * \text{<UT>} * \text{FD}) [+ M]$$

Donde:

- PB = El precio-base asociado al tipo de vehículo.
- <UT> = la cantidad de unidades de tiempo de la estadía.
- FD = El factor de demanda.
- M = Multas.

El factor de demanda es un valor de entre 0.25 y 10 que se establece en cada parking y responde a una estrategia de precios dinámica, como se explica más adelante. En una estadía se toma el factor de demanda que corresponde al momento de ingresar en una cochera.

El factor de demanda inicial es 1 y con “tendencia estable”. Luego, este dependerá fundamentalmente del estado de la tendencia. La tendencia se evalúa cada vez que un vehículo ingresa o egresa del parking y se determina como sigue:

Estado	Condición	Estrategia de precios	Formula
Tendencia estable	En los últimos 10 <UT> <i>el valor absoluto</i> de la diferencia entre ingresos y egresos es menor o igual al 10% de la capacidad del parking.	El factor de demanda se reduce lentamente.	El factor de demanda se le resta 0.01 por cada <UT> en que la tendencia transcurre en estado estable. El factor de demanda nunca puede ser menor a 0.25.
Tendencia positiva	En los últimos 10 <UT> la diferencia entre ingresos y egresos es positiva y mayor al 10% de la capacidad.	El factor de demanda aumenta moderadamente en función de la ocupación.	Si la ocupación del parking es menor al 33%, el factor de demanda aumenta 0.05 por cada <UT> en que la tendencia transcurre en estado positivo.

			Si la ocupación esta entre 33% y 66% el factor de demanda aumenta en 0.1 por cada <UT>. Si la ocupación es mayor a 66% el factor de demanda aumenta en 0.15 por cada <UT>. El factor de demanda nunca puede ser mayor a 10.
Tendencia negativa	En los últimos 10 <UT>, la diferencia entre ingresos y egresos es negativa y menor al $(10\% * -1)$ de la capacidad.	El factor de demanda se reduce significativamente.	Si el factor de demanda actual es > 1 , se establece en 1. Si el factor de demanda es ≤ 1 , se reduce en 0.05 por cada <UT> hasta llegar a 0.25. El factor de demanda nunca puede ser menor a 0.25.

Nota final

El parking se integra con un sistema de sensores que identifican el número de patente de cada vehículo que ingresa y egresa de una cochera e informa esos eventos. El sistema de sensores recoge información de todos los parkings y ofrece una interfaz única para nuestra solución².

Los sensores tienen un mínimo grado de error y eventualmente omite ciertas lecturas. Por ejemplo, puede ser que sensor informe que una patente ingresa a una cochera que figura como ocupada, debido a que “olvidó” informar cuándo egresó el vehículo anterior, por lo que imposibilitara facturar esa estadía. Este tipo de situaciones se denominan estadías anómalas o, simplemente, anomalías, y el sistema deberá registrarlas. De una anomalía se sabe la estadía relacionada, la fecha y hora en que se detectó la anomalía y un código de error.

1) Precarga de información.

No es necesario implementar una interfaz de usuario para el mantenimiento de los siguientes datos. El sistema deberá tener precargada la información, de modo que al iniciarse ya cuente con un conjunto de datos definido.

La información que se debe precargar es:

- **Parkings:** se debe disponer de al menos 2 parkings.
- **Cocheras:** se debe disponer de entre 50 y 100 cocheras por parking.
- **Propietarios:** se debe disponer de al menos 50 propietarios con los siguientes datos: Nombre, Cedula, Cuenta Corriente con saldo (valor aleatorio de entre \$-10 y \$100) y listado de Vehículos.
- **Etiquetas:** se deben contar con las siguientes etiquetas: Discapacitado, Eléctrico, Empleado.
- **Tipos de vehículos:** se deben contar con los siguientes: Motocicleta, Standard, Carga, Pasajeros.
- **Vehículos:** se debe disponer de, al menos, el doble de vehículos que total de cocheras considerando todos los parkings).
- **Tarifas:** inicialmente todos los parkings tendrán el mismo precio para todos los tipos de vehículos, equivalente a \$0,1, salvo para Motocicletas que será de \$0.05.

Otras consideraciones:

- 20% de las cocheras deben poseer etiquetas. Todas las etiquetas deben ser usadas al menos una vez en cada parking.
- 20% de los vehículos deben poseer etiquetas. Todas las etiquetas deben ser usadas al menos una vez.
- Debe haber vehículos de todos los Tipos de Vehículos.

² Esto es posible dado que –como se mencionó– el código de una cochera es universal.

2) Casos de uso:

CU: Ingresar vehículo

Curso normal:

Actor (Sensor)	Sistema
Informa acerca de un evento de ingreso con un número de patente y un código de cochera.	Identifica el vehículo y registra el inicio de una estadía.

Postcondición:

- La cochera pasa a estar “Ocupada”.
- Se registra la fecha y hora del ingreso.

Cursos alternativos:

- Si la cochera informada figura como “Ocupada” por otra estadía, se sigue el flujo normal para el vehículo ingresante, pero antes se procesa la estadía anterior de la siguiente manera:
 - La fecha y hora de salida quedan sin identificar.
 - El precio queda en cero. No se aplican multas.
 - Se registra una anomalía que quedara vinculada a esa estadía.
 - La anomalía llevara el código de error “HOUDINI”.

CU: Egresar vehículo

Curso normal:

Actor (Sensor)	Sistema
Informa acerca de un evento de egreso con un número de patente y un código de cochera.	Identifica el vehículo y procesa la estadía.

Postcondición:

- El monto de la estadía se resta de la cuenta corriente del propietario.
- La cochera pasa a estar “Libre”.

Cursos alternativos:

- Si la cochera se encuentra marcada como “Libre”, es decir, no hay una estadía activa para esa cochera: se registra una estadía con fecha de ingreso y egreso idénticas y una anomalía asociada a ella, y con el código de error “MISTERY”.
- Si la cochera se encuentra marcada como “Ocupada” pero la patente informada no coincide con la patente de la estadía registrada: se registran dos anomalías (y no se aplican multas en ninguna):
 - una anomalía asociada a la estadía que figuraba como ocupante en el sistema con el código “TRANSPORTADOR1”;
 - otra anomalía asociada al vehículo que acaba de identificarse como saliente con el código “TRANSPORTADOR2”. En ambas estadías el precio queda en cero.

CU: Tablero de control

Tablero de control

Estadías: **184**
Facturación: **\$ 971,4**

Parking	# Ocupadas	# Libres	Estado	Factor demanda	# Estadías	Multas	Subtotal

Precios

Cartelera

Fecha/Hora	Propietario	Código de anomalía	Cochera

☒ Monitorear anomalías

Cerrar

Curso normal:

Actor (Administrador)	Sistema
Selecciona la opción “Mostrar tablero de control”	<p>Muestra el tablero de control con la siguiente información:</p> <ul style="list-style-type: none"> - Total de estadías en todos los parkings - Monto total facturado en todos los parkings (\$) <p>Lista de todos los parkings con la siguiente información:</p> <ul style="list-style-type: none"> - Nombre del parking - Cantidad de cocheras ocupadas - Cantidad de cocheras libres - Estado de la tendencia - Factor de demanda - Cantidad de estadías - Subtotal cobrado por multas (\$) - Subtotal del parking (\$)
Opcionalmente marca el checkbox “Monitorear anomalías”	Únicamente mientras el checkbox este seleccionado, si se registra una anomalía, esta se agrega a una lista en pantalla, con la siguiente información: fecha y hora de la anomalía, propietario del vehículo, código de la anomalía, numero de cochera.
Opcionalmente deselecciona el checkbox “Monitorear anomalías”	La lista deja de actualizarse, pero sigue mostrando los registros que tenía hasta el momento.

Cursos alternativos:

- Ninguno

Postcondición:

- Ninguna

CU: Lista de precios

Lista de precios – <nombre del parking>	
Tipo de vehículo	Precio/<UT>
Motocicleta	\$ 0.1
Standard	\$ 1.0
Carga	\$ 3.5
Pasajeros	\$5.0

Nuevo valor:

Curso normal:

Actor (Administrador)	Sistema
Desde el Tablero de control selecciona un parking de la lista e indica la opción “Precios”	
	Muestra el nombre del parking, los tipos de vehículos y el precio por <UT> de cada uno.
Selecciona un tipo de vehículo de la lista	
Ingresa un nuevo valor e indica la opción “Guardar”	
	Actualiza la lista de precios en la lista para el tipo de vehículo seleccionado.

Cursos alternativos:

- Si el nuevo valor ingresado es negativo se muestra el siguiente mensaje “Valor inválido. El precio debe ser igual o mayor a cero.”
- Si el nuevo ingresado es igual o mayor a $2x$, donde x es el promedio de la tarifa de todos los parkings para el tipo de vehículo seleccionado, la acción no se completa y se muestra el siguiente mensaje: “Valor demasiado alto. El sistema no permite dispersión de precios por encima del 100%. Ingrese un valor menor a $<2x>$.”

Postcondición:

- El nuevo valor comienza a regir para los vehículos que inicien estadías en el parking indicado.

CU: Cartelera electrónica

Se podrán desplegar múltiples instancias de carteleras electrónicas que simulan ser carteleras físicas ubicadas en las inmediaciones de un parking.

Cartelera electrónica – <nombre del parking>

Disponibilidad: 34

Cocheras	Disponibilidad
Discapacitado	12
Eléctrico	8
Empleado	2

Tipo de vehículo	Precio/<UT>
Motocicleta	\$ 0.1
Standard	\$ 1.0
Carga	\$ 3.5
Pasajeros	\$5.0

Cerrar

Curso normal:

Actor (Administrador)	Sistema
Desde el Tablero de control selecciona un parking de la lista e indica la opción “Cartelera”	
	Muestra la cartelera electrónica con los siguientes datos: <ul style="list-style-type: none"> - Nombre del parking -Cantidad de cocheras libres (disponibilidad) -La lista de etiquetas y la cantidad de cocheras libres para cada una. -La lista de tipos de vehículos y su precio base.

Cursos alternativos:

- Ninguno

Postcondición:

- Ninguna

4) Interfaz gráfica

El objetivo de este prototipo es emular una situación en la cual cada usuario interactúa con el sistema desde una computadora diferente y donde los parkings exponen carteleros electrónicos en los diferentes accesos a las instalaciones.

Para emular esta situación, se deberá implementar una ventana general para testing de la aplicación, desde la cual se podrán lanzar múltiples instancias del tablero de control.

La información de todas las ventanas debe obligatoriamente actualizarse de manera automática, sin necesidad de que el usuario indique que desea actualizar la información.

El alumno también puede incorporar *opcionalmente* otra interfaz gráfica para testear el ingreso y egreso de vehículos a cocheras y así poder testear situaciones particulares.

Componente provisto

Las implementaciones deberán importar e integrar el componente `SimuladorTransito`. Este componente **no puede ser modificado** por el alumno. La integración del `SimuladorTransito` con la solución del alumno es parte del ejercicio y será evaluado.

Requerimientos de diseño para esta versión:

- 1) Maximizar la modularidad y claridad del código. Para esto utilice el indicador que dice que ningún método debería tener más código que el que se puede visualizar en una pantalla.
- 2) Evitar la duplicación de código y métodos o porciones de código que realizan la misma tarea.
- 3) División física de las clases en paquetes.
- 4) División lógica.
- 5) Uso del patrón de diseño “Fachada”.
- 6) Aplicar el principio Experto
- 7) Utilizar una arquitectura MVC.
- 8) Utilizar polimorfismo donde corresponda.
- 9) Utilizar manejo de excepciones.

Notas

Las posibles omisiones, ambigüedades o contradicciones que surjan del estudio de los requerimientos detallados en este documento serán analizadas y corregidas en clase durante el curso.

Se pide entregar

- Implementación del sistema en Java con interfaz de usuario gráfica cumpliendo con todos los requerimientos funcionales y de diseño solicitados.
- 2 diagramas de Clases:
 - Un diagrama de clases conceptual modelando el dominio del problema.
 - Un diagrama de clases de diseño³ (o más de uno si lo considera apropiado) que incluya a todas las entidades que participan en la solución (modelo, vistas y controladores)
- Autoevaluación de errores: si la solución entregada incluye defectos, errores, limitaciones o deficiencias conocidas, el alumno puede incorporar a la entrega un documento enumerándolos y explicando su impacto con relación a los criterios de evaluación.

3 No es necesario especificar atributos y métodos en los diagramas. Ponga el foco solo en las clases y sus relaciones.

Distribución del puntaje

Concepto	Pts.
Funcionalidad: Implementación de la funcionalidad solicitada en Java	(-40 a 0)
Diagrama conceptual de domino.	1
Diagrama de diseño	3
División lógica	4
Experto	6
Fachada y arquitectura	3
MVC	6
Observador y manejo de eventos	6
Manejo de excepciones	3
Polimorfismo	8
Total:	40

Atención: que la solución sea solo funcionalmente correcta no implica que el obligatorio este correcto pues lo que se evalúa es la correcta aplicación de los conceptos, principios y patrones de diseño indicados. Si la solución realiza una implementación parcial a nivel funcional de los casos de uso solicitados se restarán los puntos que correspondan en cada caso de uso.

En caso de que la solución presentada aplique otros patrones o principios no vistos hasta el momento en clase o utilice elementos del lenguaje no vistos en clase no formarán parte de la evaluación, y deberá asegurarse de que su aplicación no viole los puntos anteriores.

Defensa

La defensa del trabajo intenta:


- Evaluar el conocimiento general de los integrantes del grupo sobre la solución propuesta. Todos los integrantes deben conocer toda la solución.
- Evaluar el aporte individual al trabajo por parte de cada uno de los integrantes del equipo.
- El mecanismo de defensa se determinará al momento de la entrega, pudiendo ser el mismo escrito o en el laboratorio. En ambos casos se verificará el aporte al trabajo de ambos integrantes.

RECORDATORIO: IMPORTANTE PARA LA ENTREGA

- **Obligatorios**

La entrega de los obligatorios será en formato digital online, a excepción de algunas materias que se entregarán en Bedelía y en ese caso recibirá información específica en el dictado de la misma.

Los principales aspectos por destacar sobre la **entrega online de obligatorios** son:

1. Ingresá al sistema de Gestión.
2. En el menú, seleccioná el ítem "Evaluaciones" y la instancia de evaluación correspondiente, que figura bajo el título "Inscripto".
3. Para iniciar la entrega hacé clic en el ícono: 
4. Ingresá el número de estudiante de cada uno de los integrantes y hacé clic en "Agregar". El sistema confirmará que los integrantes estén inscriptos al obligatorio y, de ser así, mostrará el nombre y la fotografía de cada uno de ellos. Una vez agregados todos los integrantes, hacé clic en "Crear equipo".

Cualquier integrante podrá:

- **Modificar la integración del equipo.**
- **Subir el archivo de la entrega.**

5. Seleccioná el archivo que deseás entregar. Verificá el nombre del archivo que aparecerá en la pantalla y hacé clic en "Subir" para iniciar la entrega. Cada equipo (hasta 2 estudiantes) debe entregar **un único archivo en formato zip o rar** (los documentos de texto deben ser pdf, y deben ir dentro del zip o rar). El archivo a subir debe tener **un tamaño máximo de 40mb**

Cuando el archivo quede subido, se mostrará el nombre generado por el sistema (1), el tamaño y la fecha en que fue subido.

6. El sistema enviará un e-mail a todos los integrantes del equipo informando los detalles del archivo entregado y confirmando que la entrega fue realizada correctamente.
7. Podés cerrar la pestaña de entrega y continuar utilizando Gestión o salir del sistema.
8. La **hora tope para subir el archivo será las 21:00** del día fijado para la entrega.
9. La entrega se podrá realizar desde cualquier lugar (ej. hogar del estudiante, laboratorios de la Universidad, etc).
10. Aquellos de ustedes que presenten alguna dificultad con su inscripción o tengan inconvenientes técnicos, por favor contactarse con el Coordinador o Coordinación adjunta **antes de las 20:00hs.** del día de la entrega, a través de los mails gervaz@ort.edu.uy, alamon@ort.edu.uy y fernandez_ma@ort.edu.uy, o telefónicamente al 29021505 - int 1156 u 1138.

Si tuvieras una situación particular de fuerza mayor, debes contactarte con suficiente antelación al plazo de entrega, al Coordinador de Cursos (gervaz@ort.edu.uy) o Secretario Docente (paulos@ort.edu.uy).