

Laboratorio de Ingeniería de Software II	Ingeniería de Sistemas
<b>Nombre:</b> Emmanuel David Martínez Estrada - 2162134 María Paula Riveros Gómez - 2162135 Andrea Juliana Villalba Santos - 2162114	<b>Profesor:</b> Urbano Eliécer Gómez Prada
<b>Caso de Estudio:</b> Sistema de Información para parqueaderos	<b>Objetivos:</b> Proponer el plan de pruebas de un software

Teniendo en cuenta el siguiente enunciado o el requerimiento principal de su proyecto, defina el plan de pruebas y defina como haría la prueba automática.

Si prefiere hacerlo sobre el requerimiento, sería el presentado a continuación:

Sitúese en el primer parqueadero que quiso sistematizar el control de cobro a sus visitantes. El parqueadero presenta las siguientes características:

1. Pide al iniciar la tarifa por hora del parqueadero
2. Solo permite autos (no permite el ingreso de otros vehículos para facilitar una sola tarifa).
3. Cuenta con una sola sede, entrada y salida (para no complicarnos con la infraestructura tecnológica)
4. Para este caso particular, por simplicidad, no maneja persistencia, los datos estarán en memoria RAM).
5. La empresa registra el ingreso del auto asignándole un número de placa, la fecha y la hora de entrada y como valor agregado en los autos registra el mayor color presente en su carrocería.
6. El programa debe calcular el valor a pagar por los vehículos según el tiempo de permanencia (solo se tendrá una tarifa y se asumirá en esta versión que ningún vehículo se quedará de un día para otro)
7. Debe ir presentando en pantalla el listado de autos que van saliendo, con la fecha, hora de entrada, hora salida, total de horas de permanencia, valor a pagar e ir totalizando el informe
8. Se debe tener en cuenta que los autos por cada hora que permanezca de más, después de la primera hora, tiene un descuento de 5% por hora, pero como máximo un 20%, es decir, si lleva 5 o más horas, tiene un descuento del 20%.
9. Y la restricción más absurda de todas, la fecha, la hora de entrada y la hora de salida se ingresan manualmente, de paso sirve para hacer las pruebas
10. Imagine sin desarrollar que ahora hay persistencia, hay sucursales, hay más tipos de vehículos, la tarifa y los descuentos pueden cambiar entre sucursales o entre franjas del día, hay "enemigos" tratando de hackear el sistema, entre otros

**Cada grupo de trabajo debe:**

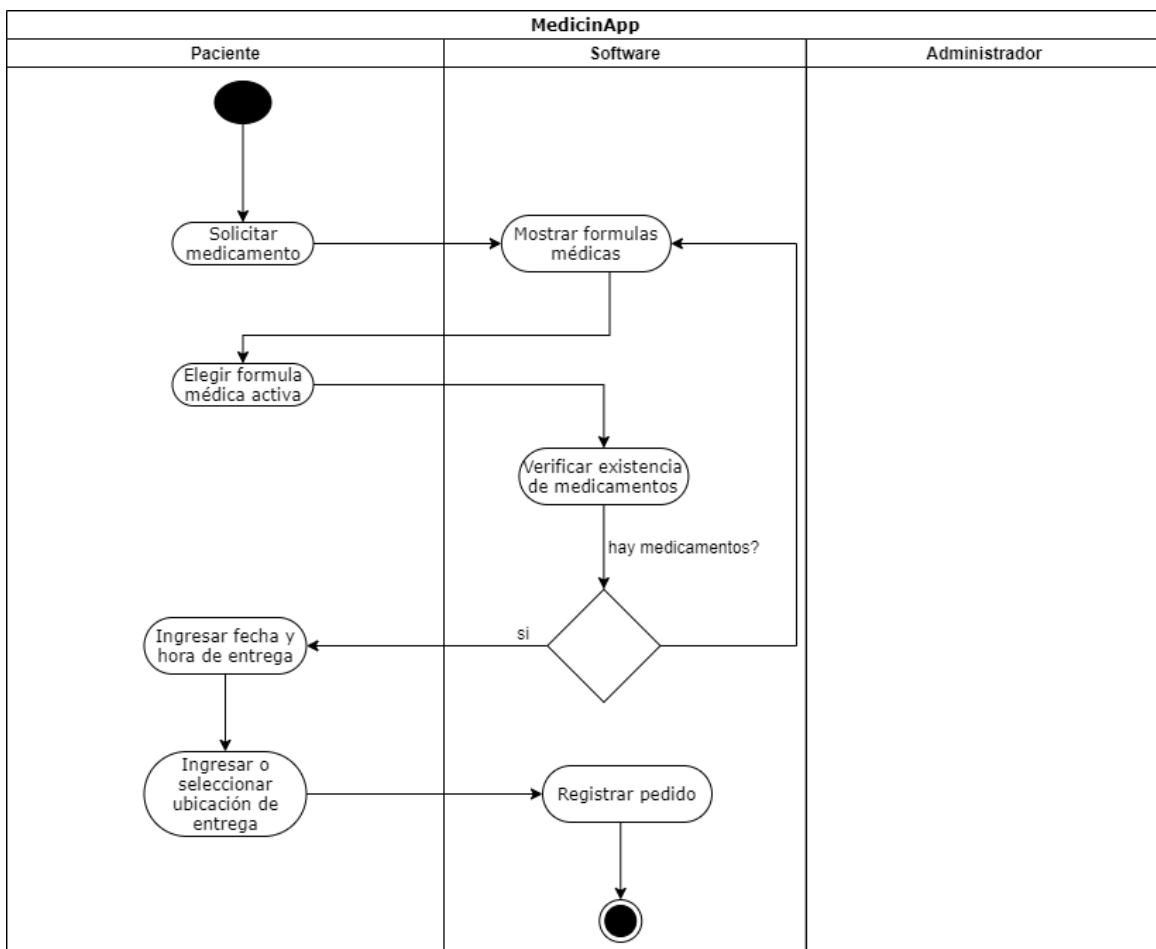
Laboratorio de Ingeniería de Software II	Ingeniería de Sistemas
<b>Nombre:</b> Emmanuel David Martínez Estrada - 2162134 María Paula Riveros Gómez – 2162135 Andrea Juliana Villalba Santos – 2162114	<b>Profesor:</b> Urbano Eliécer Gómez Prada
<b>Caso de Estudio:</b> Sistema de Información para parqueaderos	<b>Objetivos:</b> Proponer el plan de pruebas de un software

1. (15 puntos) Elaborar Diagrama de actividades para: el proceso principal si es el proyecto de clase) de calcular el valor a pagar cuando el usuario va a salir.
2. (20 puntos) Presentar los casos de prueba de tal manera que pueda probar escenarios para al menos un auto que se demora desde unos minutos, hasta el que se demoró más de las cuatro horas.
3. (15 puntos si no es el proyecto de clase) Elaborar el programa que permita conocer el valor a pagar para cada vehículo.
4. (Opcional) Automatizar con *TestNG* las pruebas del software, de tal manera que al realizarlas y estar correcto el desarrollo, el cliente cambie, en las políticas de descuento, que el máximo descuento es el 15%.
5. (Opcional) Elaborar un video mostrando la prueba de manera similar al realizado por el profesor Urbano.
6. (Opcional) Realice las siguientes tareas:
  - a. Diagrama de transición de estados
  - b. Vista de usuario
7. ¿Revisó alguna de las siguientes fuentes de información?  
(opcional) Describa en máximo 200 palabras cual presenta mejor la información y qué le llamo la atención?
  - a. <https://www.getzephyr.com/#>
  - b. <https://www.inflectra.com/Trial/Process.aspx?catalogItemId=88>
  - c. <http://www.testlink.org/>

<b>Laboratorio de Ingeniería de Software II</b>	<b>Ingeniería de Sistemas</b>
<b>Nombre:</b> Emmanuel David Martínez Estrada - 2162134 María Paula Riveros Gómez - 2162135 Andrea Juliana Villalba Santos - 2162114	<b>Profesor:</b> Urbano Eliécer Gómez Prada
<b>Caso de Estudio:</b> Sistema de Información para parqueaderos	<b>Objetivos:</b> Proponer el plan de pruebas de un software

## Desarrollo basado en el proyecto de clases – MedicinApp

### 1. Diagrama de actividades



<b>Laboratorio de Ingeniería de Software II</b>	<b>Ingeniería de Sistemas</b>
<b>Nombre:</b> Emmanuel David Martínez Estrada - 2162134 María Paula Riveros Gómez - 2162135 Andrea Juliana Villalba Santos - 2162114	<b>Profesor:</b> Urbano Eliécer Gómez Prada
<b>Caso de Estudio:</b> Sistema de Información para parqueaderos	<b>Objetivos:</b> Proponer el plan de pruebas de un software

## 2. Plan de pruebas

**Escenario:** Ideal

**Caso:** Solicitar medicamentos.

**Condiciones:** Debe tener en la cuenta que el usuario ya debe estar registrado por algún administrador de la aplicación.

### Solicitar medicamentos.

Para solicitar medicamentos, el paciente debe seleccionar una formula médica activa del listado que le muestra la aplicación, luego deberá ingresar una fecha y hora, en este caso pondrá el 19 de junio de 2020, el formato debe ser DD/MM/AAAA, y la hora será a las 08:00 AM, formato estándar. Una vez ingresado estos datos deberá ingresar su ubicación, la cual será en la Calle 13 # 96-20 en el barrio San Francisco. Antes de finalizar con la solicitud la aplicación debe mostrarle una lista con los medicamentos, la fecha y hora de entrega, y la ubicación de entrega. Finalmente, deberá confirmar para completar la solicitud la cual quedará registrada en la aplicación.

#	Paciente	Software	Datos	Revisar
1		Listar las fórmulas médicas.		
2	Seleccionar formula médica activa.			
3		Verificar existencia del medicamento.		En caso de que no existan medicamentos, mostrar un mensaje de inexistencia de medicamentos.
4	Ingresar fecha y hora.		19/06/2020 08:00 AM	<b>Observaciones:</b> Queden inactivas fechas y horas invalidas (fuera del horario establecidos de activad laboral de la app, domiciliarios y administrador). -Se le notifica al usuario el tiempo estimado de entrega de los medicamentos. - Que no deje ingresar una fecha no valida. - Que no deje ingresar una hora no valida.
5	Ingresar ubicación.		Calle 13 # 96-20 San Francisco	Que no deje ingresar una dirección no valida.
6		Mostrar pedido completo		<b>Observaciones:</b> Se debe mostrar: medicamentos, fecha y hora de entrega, lugar de entrega.
7	Confirmar solicitud			<b>Observaciones:</b> Una vez confirmado, no hay vuelta atrás.
8		Registrar pedido		
				Informe del pedido
#	Errores - Descripción		Corregido	Fecha de Corrección
1				
2				

<b>Laboratorio de Ingeniería de Software II</b>	<b>Ingeniería de Sistemas</b>
<b>Nombre:</b> Emmanuel David Martínez Estrada - 2162134 María Paula Riveros Gómez – 2162135 Andrea Juliana Villalba Santos – 2162114	<b>Profesor:</b> Urbano Eliécer Gómez Prada
<b>Caso de Estudio:</b> Sistema de Información para parqueaderos	<b>Objetivos:</b> Proponer el plan de pruebas de un software

### 3. Automatización

#### ▼ Librerías

```
[4] import pandas as pd
    from pandas._testing import assert_frame_equal
```

#### ▼ Automatización: solicitar medicamentos

Definimos la función a la cual le queremos realizar la prueba.

```
[5] def solicitar_medicamentos(medicamentos, fecha, hora, ubicacion):

    solitud_completada = pd.DataFrame({'Medicamentos': medicamentos,
                                      'Fecha y hora': [fecha + ' ' + hora],
                                      'Ubicación': ubicacion})

    return solitud_completada
```

Creamos la prueba de unidad automatica

```
[6] salida_verdadera = pd.DataFrame({'Medicamentos': ['Ibuprofeno, Paracetamol, Loratadina'],
                                     'Fecha y hora': ['10/03/2018 08:00 AM'],
                                     'Ubicación': ['Calle 13 # 96-20 San Francisco']})
salida_obtenida = solicitar_medicamentos('Ibuprofeno, Paracetamol, Loratadina',
                                         '10/03/2018', '08:00 AM',
                                         'Calle 13 # 96-20 San Francisco')

pd.testing.assert_frame_equal(salida_verdadera, salida_obtenida)
print('La salida esperada es igual a la salida obtenida: ',
      salida_verdadera.equals(salida_obtenida))

pd.testing.assert_series_equal(salida_verdadera['Medicamentos'], salida_obtenida['Medicamentos'])
print('La salida esperada para la columna medicamentos es igual a la salida obtenida para la columna medicamentos:',
      salida_verdadera['Medicamentos'].equals(salida_obtenida['Medicamentos']))

pd.testing.assert_index_equal(salida_verdadera.index, salida_obtenida.index)
print('La cantidad de registros de la salida esperada es igual a la cantidad de registros de la salida obtenida: ',
      salida_verdadera.index.equals(salida_obtenida.index))
```

```
↳ La salida esperada es igual a la salida obtenida: True
La salida esperada para la columna medicamentos es igual a la salida obtenida para la columna medicamentos: True
La cantidad de registros de la salida esperada es igual a la cantidad de registros de la salida obtenida: True
```