

# ENUNCIADOS DE TRABAJOS PRÁCTICOS EVALUABLES

*Ingeniería de Software*

## ÍNDICE

PRÁCTICO 1 -	Dinámica de Manifiesto Ágil (Evaluable) .....	3
PRÁCTICO 2 -	Requerimientos Ágiles – User Stories .....	4
PRÁCTICO 3 -	Requerimientos Ágiles – User Stories Y Estimaciones .....	7
PRÁCTICO 4 -	SCM – Estructura de Repositorio (Evaluable) .....	9
PRÁCTICO 5 -	SCM – Herramientas de SCM (Evaluable) .....	10
PRÁCTICO 6 -	Implementación De User Stories (Evaluable) .....	11
PRÁCTICO 7 -	Dinámicas sobre Scrum (Evaluable) .....	12
PRÁCTICO 8 -	SCRUM – Prácticas de Estimación, Planificación y Seguimiento (Evaluable) .....	13
PRÁCTICO 9 -	REVISIONES TÉCNICAS – Inspección de Código .....	16
PRÁCTICO 10 -	REVISIONES TÉCNICAS – Inspección de Código (Evaluable) .....	18
PRÁCTICO 11 -	TESTING - Métodos de Caja Negra .....	19
PRÁCTICO 12 -	TESTING - Métodos de Caja Negra .....	22
PRÁCTICO 13 -	TESTING – Ejecución de Casos de Prueba (Entregable) .....	23
PRÁCTICO 14 -	TESTING – Métodos de Caja Blanca .....	24
PRÁCTICO 15 -	La Publicidad en Instagram (Evaluable) .....	26
PRÁCTICO 16 -	SCM - Uso del Repositorio (Evaluable) .....	27

## PRÁCTICO 1 - Dinámica de Manifiesto Ágil (Evaluable)

<b>Unidad:</b>	<b>Unidad Nro. 2: Gestión Lean-Ágil de Productos de Software</b>
<b>Consigna:</b>	Elegir 3 principios del Manifiesto Ágil y dar ejemplos de cómo alcanzarlos. Explicar qué debería implementarse para poder alcanzarlos.
<b>Objetivo:</b>	Comprender valores y principios del Manifiesto Ágil expuestos en clase, para aplicarlos a ejemplos concretos de gestión de proyectos de software.
<b>Propósito:</b>	Familiarizarse con los valores y principios del Manifiesto Ágil.
<b>Entradas:</b>	Conceptos teóricos sobre el tema, desarrollados en clase. Manifiesto Ágil. Bibliografía referenciada sobre el tema.
<b>Salida:</b>	Un ejemplo de aplicación por cada uno de los 3 principios elegidos por el grupo.
<b>Instrucciones:</b>	<p>Elegir 3 principios del Manifiesto Ágil proporcionado. Analizar y comprender el motivo de su aplicación en la Gestión de un Proyecto.</p> <p>Dar un ejemplo concreto de aplicación de cada principio en un proyecto de Software. Explicar el contexto, describir las estrategias a utilizar y prácticas a llevar a cabo.</p> <p>Los grupos deberán organizarse para no repetir los principios elegidos.</p> <p>El práctico será evaluado en el aula.</p>

## Manifiesto Ágil

### VALORES

- Individuos e interacciones por sobre procesos y herramientas
- Software funcionando por sobre documentación detallada
- Colaboración por sobre negociación con el cliente
- Responder a cambios por sobre seguir un plan

### PRINCIPIOS

- La prioridad es satisfacer al cliente a través de releases tempranos y frecuentes
- Recibir cambios de requerimientos, aun en etapas finales
- Releases frecuentes (2 semanas a un mes)
- Técnicos y no técnicos trabajando juntos TODO el proyecto
- Hacer proyectos con individuos motivados
- El medio de comunicación por excelencia es cara a cara
- La mejor métrica de progreso es la cantidad de software funcionando
- El ritmo de desarrollo es sostenible en el tiempo
- Atención continua a la excelencia técnica
- Simplicidad - Maximización del trabajo no hecho
- Las mejores arquitecturas, diseños y requerimientos emergen de equipos autoorganizados
- A intervalos regulares, el equipo evalúa su desempeño y ajusta la manera de trabajar

<http://agilemanifesto.org/iso/es/manifesto.html>

## PRÁCTICO 2 - Requerimientos Ágiles – User Stories

<b>Unidad:</b>	Unidad Nro. 2: Gestión Lean-Ágil de Productos de Software
<b>Consigna:</b>	Identificar y escribir las US identificadas en el Dominio presentado para el práctico
<b>Objetivo:</b>	Aplicar los conceptos teóricos desarrollados en clase sobre User Stories.
<b>Propósito:</b>	Familiarizarse con los conceptos de requerimientos ágiles y en particular con US, Épicas y Temas.
<b>Entradas:</b>	Conceptos teóricos sobre el tema, desarrollados en clase. Filminas. Enunciado, consigna y tarjeta de US.
<b>Salida:</b>	US identificadas y tarjetas de US completas. Este práctico no se entrega y por lo tanto no tiene nota. El tema se evalúa en el parcial.
<b>Instrucciones:</b>	Los docentes representarán a expertos del dominio que expresarán sus necesidades vinculadas al desarrollar un software para el manejo de una bomba de insulina. Los estudiantes realizarán preguntas con el objetivo de acordar juntos el alcance del proyecto, y determinar las user stories correspondientes y definir las pruebas de aceptación de usuario para cada una. Cada grupo presentará el resultado obtenido al final de la clase al resto del curso.

### PickMe: Aplicación móvil para servicio de transporte

Una consultora de software de la ciudad de Córdoba prestará sus servicios para desarrollar una aplicación para Smartphones que permita conectar la necesidad de traslado de un pasajero con un conductor permitiendo compartir los gastos del recorrido. A continuación, se transcribe parte de la entrevista realizada al experto del dominio.

A continuación, se transcribe parte de la entrevista realizada al experto en el dominio:

**Product Owner (PO):** La idea es que los conductores anuncien los viajes que van a realizar con sus vehículos para que luego otros pasajeros puedan sumarse. ¿Cómo debería acceder el conductor a la aplicación?

**Experto en el dominio (ED):** Como conductor debe inicialmente registrarse en el sistema completando una solicitud de conducción. La misma debe solicitar nombre, apellido, fecha de nacimiento, contraseña, N° de documento, correo electrónico, teléfono, fotografía y credenciales. Las credenciales obligatorias son el DNI y el carnet de conducir vigentes, las cuales se componen de un archivo y una descripción.

**PO:** ¿El conductor debe proporcionar datos sobre el vehículo que va a utilizar?

**ED:** Exactamente, en la solicitud debe indicar cuáles son los vehículos que va a utilizar y de cada uno de ellos su marca, modelo, versión del modelo, versión del motor, tipo de combustible, capacidad(personas) y las credenciales cédula de identificación del vehículo y renovación de la oblea (en caso de usar GNC).

**PO:** ¿Existe algún procedimiento para autorizar la solicitud?

**ED:** Si, el responsable de autorizar la solicitud de conducción de un interesado (conductor) analiza las credenciales adjuntas y de acuerdo con la fecha de vigencia determina la vigencia de solicitud del conductor.

**PO:** ¿Cómo comenzarían a interactúan los posibles pasajeros con la aplicación? Si estoy registrado como conductor, ¿podría en algún momento utilizar el servicio de pasajero también?

**ED:** Los pasajeros también tienen que registrarse en la aplicación. Puede hacerlo cualquier persona mayor de 18 años ingresando los mismos datos que un conductor, exceptuando sus credenciales y datos de vehículos. No existe un proceso de autorización. Y respondiendo a la segunda pregunta, cualquier conductor puede ser pasajero si así lo desea, pero no al revés.

**PO:** ¿Cómo debe indicar un conductor que va a iniciar un viaje?

**ED:** El conductor puede programar su viaje hasta 30 minutos antes del inicio y comunicarlos en la aplicación indicando el vehículo que utilizará y el combustible. El viaje es la distancia recorrida por un conductor desde el origen a un destino durante un periodo de tiempo. Una vez comunicado el viaje, estará visible para que los interesados se puedan sumar.

**PO:** ¿Los pasajeros pueden sumarse en cualquier momento del viaje o solo al inicio del mismo?

**ED:** Pueden sumarse en cualquier momento siempre y cuando el vehículo tenga capacidad. El viaje de un pasajero se considera un trayecto, que puede estar conformado por uno o más tramos. Cada vez que hay un cambio en la cantidad de pasajeros (porque sube o baja alguno) se crea un tramo nuevo para cada pasajero. Esto se realiza para facilitar los cálculos de las tarifas.

**PO:** ¿Cómo se calcula la tarifa para el conductor y cada pasajero?

**ED:** El conductor se comportará como un pasajero más cuyo trayecto es igual al del viaje, pero con los datos de inicio y fin reales, conteniendo todos los tramos realizados. A todos los pasajeros se les deberá calcular la tarifa correspondiente al tramo al finalizar el mismo. De cada viaje y cada trayecto se debe mantener su origen, destino, distancia, fecha y hora inicio (programada y real). Mientras que para cada tramo deberá almacenar el origen, destino, distancia e importe. El cálculo de la tarifa está definido como la suma de una tarifa base (3% del valor combustible utilizado) más el cargo del conductor (del 3% al 5% del combustible utilizado) que define en su solicitud de conducción. Esta tarifa se divide entre los pasajeros según el/los tramos efectuados.

**PO:** ¿Cómo pueden abonar las tarifas los pasajeros?

**ED:** Es requisito que tengan cuenta en Mercado Pago, ya que sólo se cobrará a través de este medio.

**PO:** ¿Cómo solicita traslado un pasajero?

**ED:** Para solicitar un traslado hacia algún punto, el pasajero ingresa su dirección de origen, de destino, y una distancia máxima de traslado a pie. Luego, el sistema deberá mostrar en el mapa aquellos conductores que se dirijan en ese momento a la zona deseada por el pasajero, en función de la distancia máxima que desee caminar. Aquellos conductores cuya ruta o destino estén dentro del rango de dicha distancia y tengan capacidad para subir pasajeros, aparecerán en color verde; de lo contrario, en rojo. La solicitud se realiza unidireccionalmente desde el pasajero al conductor. Cuando el usuario lo requiera, seleccionará del mapa el conductor con el cual desea compartir un viaje, y se enviará una solicitud al mismo registrando fecha y hora de solicitud. Mientras un pasajero espera la confirmación de una solicitud, puede realizar otras solicitudes. Si el pasajero recibe una confirmación de su solicitud, las otras solicitudes que haya enviado se cancelan. Mientras un pasajero está viajando no puede realizar nuevas solicitudes.

**PO:** ¿El conductor indica de alguna forma que inicia el viaje programado?

**ED:** Cuando el conductor comienza su viaje debe registrar esta información indicando hora de inicio y punto de partida real, y se les debe informar a los pasajeros que hayan solicitado algún trayecto para ese viaje. En este momento el viaje se considera en curso. Si el conductor está demorado, puede indicarle esta situación al pasajero a través de una opción, la cual enviará la notificación correspondiente.

**PO:** ¿El conductor puede confirmar la solicitud de un pasajero?

**ED:** El conductor debe recibir la solicitud en su pantalla con los datos y la foto del pasajero. Si la acepta, se debe enviar una notificación al pasajero indicando el tiempo de demora estimado y el mapa de su recorrido. Es posible que un conductor le solicite al pasajero que se posicione en otra dirección cercana dónde recogerlo. Cuando el conductor recoge un pasajero deberá indicarle al sistema, registrando en ese momento la hora de inicio y la dirección de origen.

**PO:** ¿Cómo se procede al finalizar un trayecto y el viaje completo?

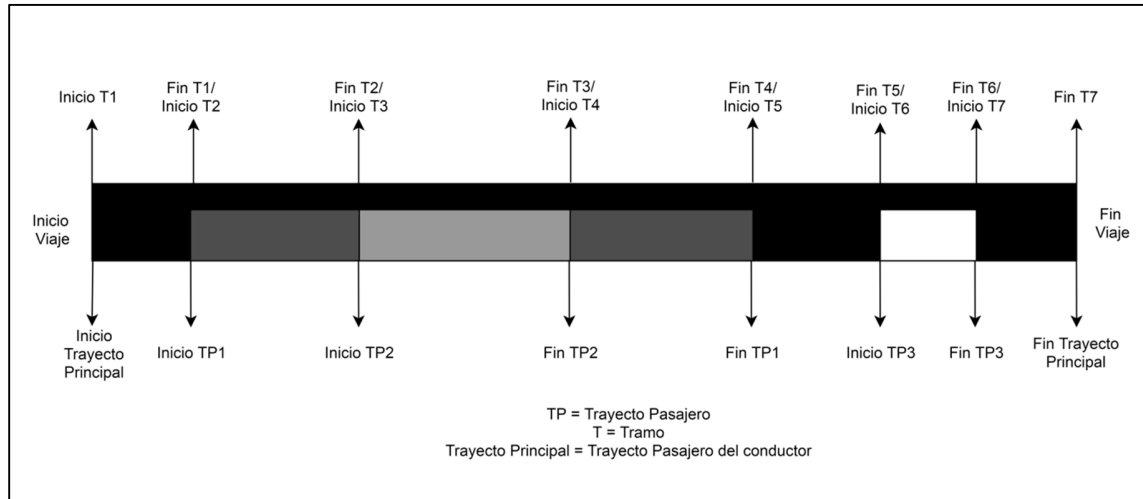
**ED:** Cuando un pasajero llega a su destino, el conductor indica la finalización del trayecto. En este momento se almacena la hora de fin, la dirección de destino y se muestra el tiempo que duró el trayecto, la distancia recorrida, el monto total a abonar por el pasajero, y se procede con el cobro mediante MercadoPago del pasajero. El conductor recibirá una notificación una vez que el ente emisor haya confirmado la acreditación. El viaje finaliza cuando el conductor así lo indica, registrando la hora de fin y el destino. En este punto el mismo se considera finalizado.

**PO:** ¿Le parece importante agregar alguna funcionalidad más a la aplicación que no se haya mencionado aun?

**ED:** Es importante que la aplicación pueda proveer estadísticas e informes acerca de los viajes realizados. El Administrador debería obtener un informe mensual en formato Excel de los viajes realizados con las distancias recorridas en cada uno, el tiempo utilizado,

tarifa final, conductor, vehículo, combustible y cantidad de pasajeros. Además, sería de gran utilidad que otro reporte mensual de pasajeros que indique aquellos que son frecuentes (que utilicen el servicio dos o más veces por semana) y cantidad de altas de pasajeros con formato Excel.

### Ejemplo de viaje con pasajeros



## PRÁCTICO 3 - Requerimientos Ágiles – User Stories Y Estimaciones

<b>Unidad:</b>	Unidad Nro. 2: Gestión Lean-Ágil de Productos de Software
<b>Consigna:</b>	Identificar y escribir las US identificadas en el Dominio presentado para el práctico
<b>Objetivo:</b>	Aplicar los conceptos teóricos desarrollados en clase sobre User Stories.
<b>Propósito:</b>	Familiarizarse con los conceptos de requerimientos ágiles y en particular con US, Épicas y Temas.
<b>Entradas:</b>	Conceptos teóricos sobre el tema, desarrollados en clase. Bibliografía referenciada sobre el tema. Enunciado, consigna y tarjeta de US.
<b>Salida:</b>	US identificadas y tarjetas de US completas. Este práctico no se entrega y por lo tanto no tiene nota. El tema se evalúa en el parcial.
<b>Instrucciones:</b>	Los docentes representarán a expertos del dominio que expresarán sus necesidades vinculadas al desarrollar un software para el seguimiento de taxis. Los estudiantes realizarán preguntas con el objetivo de acordar juntos el alcance del proyecto, determinar las User Stories correspondientes y definir las pruebas de aceptación de usuario para cada una. Cada grupo presentará el resultado obtenido al final de la clase al resto del curso.

### DeliverEat! Servicio de delivery de “lo que sea”

Hace algunos años Marcos se encontró con el problema de que algunos de sus restaurantes favoritos, entre ellos Grabeat que prepara las mejores hamburguesas de Córdoba en Derqui al 88, no tenían servicio de delivery, por lo que para poder hacerse con una de esas delicias necesitaba desplazarse hasta el lugar y esperar hasta que su pedido estuviera preparado, sin importar las condiciones climáticas o la cantidad de cuadras que tuviera que caminar. Totalmente decidido a darle una solución a su problema y viendo una profunda veta de negocios en un servicio aún no explotado en la ciudad, decidió formar un equipo y buscar financiamiento para llevar su idea a una realidad. Hoy en día su servicio es de uso corriente entre los jóvenes y también entre los negocios de comida que se han suscrito para cubrir sus deficiencias de envío a domicilio.

En una entrevista para la prensa tecnológica local, Marcos nos contó un poco acerca de cómo funciona su servicio:

**Marcos:** ¡DeliverEat! es un servicio de delivery de lo que sea, es decir que puedes pedir desde tu casa cualquier cosa que entre en una mochila, nosotros lo buscamos y te lo llevamos a tu domicilio. Esto aún funciona tanto para negocios de comida, como también para farmacias, negocios de ropa, tecnología y mucho más.

**Martina:** ¡Actualmente DeliverEat! es muy conocida, pero yo todavía no la he descargado a mi celular, ¿cómo la podría usar?

**Marcos** (mostrando el paso a paso en su teléfono): El primer paso es descargar la aplicación del Store correspondiente, actualmente soportamos tanto dispositivos Android como iOS, por lo que puedes descargarla desde ambas tiendas, también están disponibles la versión web y desktop ambas desde nuestra página oficial. Luego vas a necesitar una cuenta de usuario: puedes usar Facebook o Google para registrarte o bien mediante e-mail, en este último caso sólo debe tratarse de un e-mail válido indicando la contraseña elegida, que debe tener al menos 8 caracteres (tomamos la seguridad de nuestros usuarios muy en serio).

**Martina:** ¿Perfecto, y ya con una cuenta de usuario puedo llamar a los lugares para que me envíen mi pedido?

**Marcos:** Procesamos los pedidos internamente por lo que no es necesario que vos te pongas en contacto con el restaurante o comercio. Tenemos dos opciones para que puedas hacer tu pedido: elegir de entre nuestros comercios adheridos o realizar un pedido a un lugar no listado. Disponemos de una amplia red de comercios que ya se han sumado a nuestro servicio y esto incorpora una gran ventaja: en el momento en que haces un pedido a un comercio disponible en el listado ellos ya comienzan a prepararlo, en el mismo momento en que el cadete se prepara para retirarlo y llevarlo a tu domicilio, ¡es rapidísimo!

**Martina:** ¡Buenísimo! ¿Y cómo le indico al comercio cuál es el detalle de mi Pedido?

**Marcos:** La aplicación te permite buscar en nuestra lista de comercios adheridos especificando algunos filtros como tipo de comida, tiempo promedio de entrega o si tiene seguimiento del cadete por GPS. Una vez que hayas seleccionado el comercio podrás navegar por su carta de productos en base a su categoría, por ejemplo, para un negocio de comidas podrás ver “Empanadas”, “Pizzas”, “Hamburguesas” mientras que para una librería podrás encontrar algo como “Novelas”, “Ensayos”, “Latinoamericanos”, “Clásicos”, entre otros.

**Martina:** ¿Y de esta forma puedo pedir sólo un producto? ¿Y si luego me arrepiento?

**Marcos:** Vas a disponer de un carrito de productos, en donde puedes ir agregando diferentes productos especificando la cantidad y si es necesario alguna observación. Es importante mencionar que todos los productos deben ser del mismo comercio y que no hay límite de productos en el carrito más que el del pedido final: debe entrar en la mochila de tu cadete. En caso de que te arrepientas aún estás a tiempo porque puedes modificar algún producto del Carrito o bien quitarlo definitivamente. En el momento en que estés lista confirmas el Pedido indicando tu dirección, forma de pago (actualmente sólo disponemos de efectivo y tarjeta VISA pero estamos investigando la integración con otras tarjetas de crédito), cuando quieres recibirlo (por defecto es “Lo antes posible”, pero también puedes programar una fecha/hora de recepción) y, en caso de haber seleccionado pago en efectivo con cuánto vas a pagar. Esto nos permite indicarle al cadete el monto de vuelto que deberá llevar. Al confirmar el Pedido tanto el comercio como el cadete reciben notificaciones en sus teléfonos con toda la información para que tu pedido pueda llegar a su destino tal como lo esperas.

**Martina:** ¿Es posible hacer seguimiento del cadete o comunicarme con él?

**Marcos:** Efectivamente puedes ver el estado de tu Pedidos en tiempo real y seguir a tu cadete mediante un mapa de Google Maps para ver en dónde se encuentra en cada momento y así estimar mejor en qué momento lo recibirás en tu domicilio.

**Martina:** ¿También me habías contado que es posible pedir el envío de algo que no esté disponible en alguno de los comercios adheridos, cómo funciona?

**Marcos:** Efectivamente también puedes realizar un pedido de “lo que sea”: es muy similar al Pedido en base a un Carrito de Productos sólo que en vez de seleccionarlos desde un listado dispones de un campo de texto para indicar qué debe buscar tu Cadete con la opción de agregar una foto descriptiva. Además, debes indicar dónde debe retirarlo seleccionando un punto en un mapa interactivo o completando de forma textual la dirección del comercio. El resto (pago, confirmación y seguimiento) del Pedido son idénticos a los del Pedido a un comercio adherido.

**Martina:** Muchas gracias Marcos por tu tiempo, en la próxima entrega seguramente nos podrás contar un poco más sobre el panel de administración de los cadetes y comercios, hasta la próxima!



## PRÁCTICO 4 - SCM – Estructura de Repositorio (Evaluable)

<b>Unidad:</b>	Unidad Nro. 3: Gestión del Software como producto
<b>Consigna:</b>	Comprender los conceptos de SCM (Administración de Configuración de Software) expuestos en la clase teórica, para aplicarlos en un ejemplo concreto en el contexto de la materia.
<b>Objetivo:</b>	Que el estudiante pueda inferir cómo definir una Estructura de Repositorio e identificar los Ítems de Configuración que deberá asignar en las carpetas definidas en la estructura.
<b>Propósito:</b>	Familiarizarse con las actividades y los conceptos de Gestión de Configuración de Software.
<b>Entradas:</b>	Conceptos teóricos sobre el tema, desarrollados en clase. Plantilla de Ítems de Configuración. Bibliografía referenciada sobre el tema.
<b>Salida:</b>	Estructura del repositorio en formato de árbol de carpetas. Planilla de ítems de configuración con reglas de nombrado, ubicación física y glosario.
<b>Instrucciones:</b>	<ul style="list-style-type: none"> <li>- Para el caso de estudio DeliverEat, pensar en todo el material que se genera en el desarrollo de un proyecto de software en el contexto de una metodología ágil y que necesitará ser mantenido bajo control de versiones.</li> <li>- Definir cómo sería la estructura del repositorio del proyecto como un conjunto de carpetas y su jerarquía.</li> <li>- Identificar cada uno de los ítems de configuración intervinientes</li> <li>- Para cada Ítem de Configuración determinar nombre, regla de nombrado, ubicación física y tipo (Producto, Proyecto o Iteración) y completar la plantilla correspondiente.</li> <li>- Si fuera necesario incluir y detallar el Glosario de etiquetas y nombrado.</li> </ul>

### Plantilla de Ítems de Configuración

Nombre del Ítem de Configuración	Regla de Nombrado	Ubicación Física	Tipo de Ítem
<b>Clase (código)</b>	[NombreDeLaClase].java	/producto/src/[paquete]	Producto

### GLOSARIO

Sigla	Significado
[NombreDeLaClase]	<b>Nombre de la clase respetando camel case, como: ActaDeInspeccion</b>

## PRÁCTICO 5 - SCM – Herramientas de SCM (Evaluable)

<b>Unidad:</b>	<b>Unidad Nro. 3: Gestión del Software como producto</b>
<b>Consigna:</b>	Comprender los conceptos de administración de configuración de software (SCM) expuestos en la clase teórica para aplicarlos en un ejercicio propuesto por la cátedra.
<b>Objetivo:</b>	Que el estudiante sea capaz de realizar actividades básicas de la gestión de configuración mediante el uso de una herramienta tales como la definición de una estructura de repositorio, ingreso y extracción de ítems de configuración del repositorio y definición de líneas base.
<b>Propósito:</b>	Aplicar los conceptos de gestión de configuración estudiados en una herramienta de software específica
<b>Entradas:</b>	Conceptos teóricos sobre los temas desarrollados en clase. Bibliografía referenciada sobre el tema. Estructura del Repositorio y Plantilla de Ítems de Configuración del Trabajo Práctico Evaluable 4
<b>Salida:</b>	<p>URL y Credenciales de acceso para el repositorio implementado</p> <p>Documento con el criterio para la creación de una línea base</p> <p>Se evaluará lo siguiente:</p> <ul style="list-style-type: none"> <li>• El repositorio debe ser accesible de forma pública</li> <li>• Implementación de la estructura de carpetas propuesta</li> <li>• Los archivos se deben encontrar en la ubicación correspondiente a su definición como ítem de Configuración.</li> </ul>
<b>Instrucciones:</b>	<ul style="list-style-type: none"> <li>• Crear un repositorio de acceso público</li> <li>• Crear cuentas de usuario para cada uno de los integrantes del grupo</li> <li>• Implementar la estructura del repositorio propuesta en el Trabajo Práctico Evaluable 4 para el Proyecto de software.</li> <li>• Realizar Commit (colocar) de cada ítem de configuración disponible al momento.</li> <li>• Definir al menos un momento que considere adecuado para marcar una línea base y luego marcar la línea base definida en el repositorio.</li> </ul>
<b>Observaciones:</b>	El repositorio implementado debe ser de acceso público utilizando Git o Subversion como motor de control de versiones.

## PRÁCTICO 6 - Implementación De User Stories (Evaluable)

<b>Unidad:</b>	Unidad Nro. 3: Gestión del Software como producto
<b>Consigna:</b>	Implementar una User Story determinada usando un lenguaje de programación elegido por el grupo respetando un documento de reglas de estilo.
<b>Objetivo:</b>	Que el estudiante comprenda la implementación de una User Story como una porción transversal de funcionalidad que requiere la colaboración de un equipo multidisciplinario.
<b>Propósito:</b>	Familiarizarse con los conceptos de requerimientos ágiles y en particular con User Stories en conjunto con la aplicación de las actividades de SCM correspondientes.
<b>Entradas:</b>	Conceptos teóricos sobre el tema desarrollados en clase. Definición completa de las User Stories correspondientes al Trabajo Práctico 3 "Requerimientos Ágiles – User Stories y Estimaciones"
<b>Salida:</b>	Implementación de la User Story correspondiente en un programa ejecutable Documento de estilo de código
<b>Instrucciones:</b>	<ul style="list-style-type: none"> <li>- Seleccionar una User Story a implementar de entre las siguientes opciones:                             <ul style="list-style-type: none"> <li>o Realizar Pedido a Comercio adherido (<b>grupos pares</b>)</li> <li>o Realizar un Pedido de "lo que sea" (<b>grupos impares</b>)</li> </ul> </li> <li>- Seleccionar el conjunto de tecnologías para implementar la funcionalidad elegida.</li> <li>- Buscar y seleccionar un documento de buenas prácticas y/o reglas de estilo de código para el lenguaje de programación a utilizar.</li> <li>- Implementar la US siguiendo las reglas de estilo determinadas.</li> </ul>

## PRÁCTICO 7 - Dinámicas sobre Scrum (Evaluable)

<b>Unidad:</b>	<b>Unidad Nro. 2: Gestión Lean-Ágil de Productos de Software</b>
<b>Consigna:</b>	Por medio de las dinámicas propuestas por los docentes reforzar los conceptos del framework SCRUM.
<b>Objetivo:</b>	Profundizar el conocimiento del grupo de estudiantes en los conceptos, prácticas y lineamientos definidos en el framework SCRUM.
<b>Propósito:</b>	Familiarizarse con la forma gestionar utilizando el framework de Scrum, empleando para ello dinámicas colaborativas.
<b>Entradas:</b>	<p>Conceptos teóricos sobre el tema, desarrollados en clase.</p> <p>Materiales necesarios, que deberán ser provistos por los estudiantes:</p> <ul style="list-style-type: none"> <li>• Al menos 30 tarjetas o Post-It</li> <li>• Marcadores</li> <li>• Dos afiches</li> </ul>
<b>Salida:</b>	Cada equipo de estudiantes con conocimientos esclarecidos relacionados a lo que propone SCRUM. La dinámica de los juegos facilitará el debate sobre los resultados obtenidos y sobre la posición que tiene cada uno respecto a estos temas.
<b>Instrucciones:</b>	El docente asumirá el rol de instructor para moderar la ejecución de cada dinámica.

## PRÁCTICO 8 - SCRUM – Prácticas de Estimación, Planificación y Seguimiento (Evaluable)

<b>Unidad:</b>	<b>Unidad Nro. 2: Gestión Lean-Ágil de Productos de Software</b>
<b>Consigna:</b>	A través de un juego y de las instrucciones comunicadas por los docentes, implementar las siguientes prácticas de Scrum: <ul style="list-style-type: none"> <li>• Planificación de Sprint</li> <li>• Ejecución de Sprint</li> <li>• Daily Scrum Meeting</li> <li>• Sprint Review</li> <li>• Sprint Retrospective</li> </ul>
<b>Objetivo:</b>	Aplicar las prácticas de estimación, planificación y monitoreo propuestas por Scrum, aplicadas en un caso práctico.
<b>Propósito:</b>	Familiarizarse con las prácticas de Scrum a través de una dinámica grupal, en forma de juego.
<b>Entradas:</b>	Conceptos teóricos sobre el tema, desarrollados en clase. Bibliografía referenciada sobre el tema. Instrucciones del juego. Materiales necesarios que deben ser llevado por los estudiantes.
<b>Salida:</b>	Mostrar los resultados de la dinámica en cada Sprint Review. Entregar al docente <ul style="list-style-type: none"> <li>• Product Backlog con priorización de las US</li> <li>• Sprint Backlog de cada Sprint (incluyendo estimaciones)</li> <li>• Gráficos de los burndown chart de cada Sprint</li> <li>• Resultados de las Retrospectives</li> <li>• Hoja final de conclusiones con respuesta a las preguntas planteadas.</li> </ul>
<b>Instrucciones:</b>	El docente que dirige la dinámica asumirá el rol del Product Owner y será quién indique la consigna y el alcance de la dinámica a realizar.

### Proyecto: Instrumentos musicales para niños

**Objetivo:** Fabricar prototipos de instrumentos musicales para una importante Juguetería de Córdoba. La empresa desea adquirir instrumentos musicales para niños, para lo cual ha solicitado una serie de prototipos de distintos instrumentos, lo que permitirá la atracción de más niños y potenciales clientes.

Los dueños de la Juguetería desean ponerse en contacto con distintas fábricas de estos instrumentos para comprobar sus propuestas, ya que la idea está muy orientada a la calidad y no se va a aceptar cualquier instrumento.

**Materiales que cada grupo deberá tener para el desarrollo de la clase:**

- 1 caja vacía y pequeña (tamaño deseado, para la guitarra)
- Banditas elásticas (aproximadamente 20), tamaño acorde a la caja
- Una lata vacía y limpia y sin bordes filosos, tamaño deseado (leche Nido, Pringles, etc.)
- 1 globo
- 2 vasitos de plástico
- Porotos, arroz, cualquier semilla pequeña, botones o piedritas
- 2 palitos de madera (palitos de helado)
- Una tira de papel del tamaño de los palitos de madera
- Un paquete de palillos de dientes
- 3 conos de papel higiénico o 2 de papel de cocina
- Un trozo de cartón delgado o cartulina
- 8 sorbetes o pajitas

- Tijeras - Marcadores de colores - Cinta adhesiva - Goma de pegar

**Reglas:**

- Cuando finaliza un sprint se contarán cuántos prototipos de instrumentos musicales están terminados y listos para ser entregados a la Juguetería. Sólo se aceptarán aquellos prototipos que cumplan con el 100% de lo acordado. Si un prototipo no pasa por la revisión final de calidad no se aceptará. Los prototipos sin terminar se pueden continuar en la siguiente iteración para ser completados.
- Cada equipo representará a una fábrica. Cada equipo tendrá un logotipo, que será el número de grupo que le corresponde.

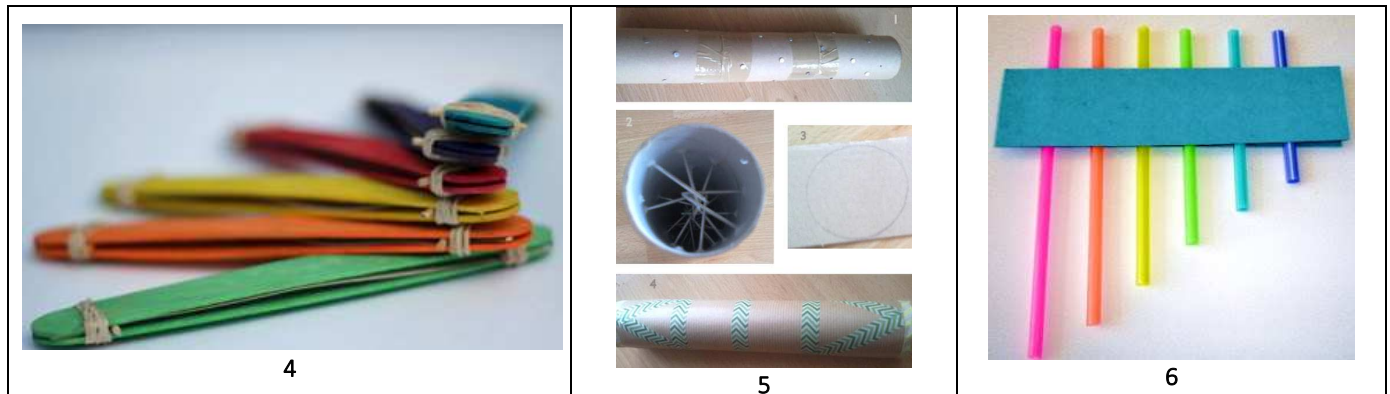
**Theme:** Como Juguetería, deseamos ampliar nuestras ofertas de juguetes para niños y ser reconocidos como la mejor juguetería de Córdoba y con la más amplia variedad.

**Epic:** Contar con diferentes instrumentos musicales para niños.

ID	User Stories	Valor de Negocio
1	Yo como dueño de la juguetería quiero contar con guitarras, a fin de ofrecer el instrumento de cuerda más divertido para los chicos	15
2	Yo como dueño de la juguetería quiero contar con tambores de lata, a fin de ofrecer instrumentos de percusión en mi empresa	10
3	Yo como dueño de la juguetería quiero contar con maracas, a fin de ofrecer instrumentos idiófonos en mi empresa	20
4	Yo como dueño de la juguetería quiero contar con armónicas, a fin de ampliar la variedad de instrumentos musicales ofrecidos	18
5	Yo como dueño de la juguetería quiero contar con palos de lluvia, a fin de ofrecer instrumentos que generan sonidos ambientales	17
6	Yo como dueño de la juguetería quiero contar con zampoñas, a fin de ampliar la variedad de instrumentos musicales ofrecidos	20

Imágenes (a modo ilustrativo) de los prototipos deseados





Detalle de historias de usuario.

ID de la historia	Actividades (orientativas)	Esfuerzo (minutos)	Puntos de historia
1	Hacer un agujero a la caja	2	
	Atravesar las bandas elásticas de una punta a la otra, cuidando que pasen por encima del agujero, y sujetarlas	1	
	Personalizar prototipo	3	
2	En la boca superior de la lata extender el látex del globo para cubrirla	1	
	Sujetar con una bandita elásticas el látex del globo	1	
	Personalizar prototipo	3	
3	Colocar dentro de los vasitos los porotos, arroz, botones y/o piedritas	1	
	Pegar con cinta el otro vaso, juntando boca con boca	1	
	Personalizar prototipo	3	
4	Colocar el papel entre los dos palos de madera	1	
	Fijar banda elástica en cada extremo de los palos	1	
	En cada extremo, introducir un palillo por encima y otro por debajo del papel, situados justo al lado de las bandas elásticas	2	
	Personalizar prototipo	3	
5	Unir con cinta los conos	1	
	Atravesarlos por todos lados con palillos de dientes, formando cruces.	4	
	Cortar las puntas salientes de los palillos	2	
	Pegar o encintar cada palillo para que queden en su lugar	2	
	Depositar arroz dentro del cono	1	
	Tapar ambos extremos del cono con el cartón	2	
	Personalizar prototipo	3	
6	Cortar dos pequeñas piezas rectangulares de cartón	1	
	Colocar y pegar los sorbitos en una de las piezas de cartón	2	
	Pegar la segunda pieza de cartón sobre la primera, de modo tal que las pajitas queden entre los dos cartones	1	
	Recortar la parte inferior de las pajitas, cada una debe ser más corta que la anterior	1	
	Personalizar prototipo	3	

## PRÁCTICO 9 - REVISIONES TÉCNICAS – Inspección de Código

<b>Unidad:</b>	<b>Unidad Nro. 4: Aseguramiento de Calidad de Proceso y de Producto</b>
<b>Consigna:</b>	Ejecute una revisión técnica a la porción de código entregada y genere el reporte correspondiente.
<b>Objetivo:</b>	Que el estudiante pueda ejecutar un proceso formal de inspección de código, identificando hallazgos en una porción de código.
<b>Propósito:</b>	Familiarizarse con proceso formal de inspección de código.
<b>Entradas:</b>	Conceptos teóricos sobre el tema, desarrollados en clase. Estándar de Inspección de Código. Bibliografía referenciada sobre el tema. Paper 11_Best_Practices_For_Peer_Code_Review_Final. Enunciado, consigna y porción de código. Estándar de Reporte de Inspección.  La porción de código bajo inspección se encuentra disponible en Autogestión con el nombre <b>GestorAlquilerPuesto_Java.pdf</b> .
<b>Salida:</b>	Reporte de Hallazgos identificados
<b>Instrucciones:</b>	<ul style="list-style-type: none"> <li>Lea y analice la siguiente porción de código.</li> <li>Realice una inspección del código mediante el método formal de inspección y prepara el reporte de inspección correspondiente respetando el estándar.</li> </ul>

### Breve descripción del Negocio

El Mercado de Abasto de Frutas y Verduras de una ciudad de la región necesita un Sistema de Información que brinde soporte a las actividades que allí se realizan.

El mercado está organizado en sectores. Cada sector contiene puestos, los cuales son alquilados a empresas y quinteros (genéricamente clientes) para que allí realicen sus ventas. Existen distintos tipos de puestos (con techo, sin techo, con cámara refrigerante, etc.) y distintas dimensiones para cada puesto (10m2, 15m2, etc.), para poder ajustarse mejor a las necesidades de cada cliente.

El precio del alquiler depende del sector en el que se encuentre el puesto, el tipo de puesto y sus dimensiones, y está predefinido.

Cuando un cliente desea alquilar uno o más puestos, se verifica la disponibilidad del tipo de puesto que requiere. Si existe disponibilidad y el cliente está de acuerdo con el precio, se realiza un contrato de alquiler por cada puesto que se alquile. En el contrato se especifica la fecha de inicio y fin del alquiler, el monto mensual del alquiler y tiene además un número que identifica el contrato que es único y el nombre del responsable por parte del Mercado que intervino en la firma del contrato y el responsable de la registración del mismo. Además, cada puesto cuenta con un medidor para el consumo de energía eléctrica. Mensualmente se registran las lecturas de cada medidor, ya que el consumo de cada puesto es facturado al cliente que está alquilando ese puesto.

En el momento de efectuar el alquiler, se registra en el contrato la última lectura del medidor del puesto que se está alquilando. Los aspectos vinculados a la facturación quedan excluidos del alcance del sistema, como así también la gestión de cobro de los alquileres.



Reporte de Inspección				
Fecha de Inicio:				
Revisores:	Nombre	Rol		
Alcance de la revisión:				
Nombre	Versión	Autor		
Hallazgos				
Nº	Fecha	Descripción	Línea	Severidad
Decisión Final:		Aceptar/Rechazar/Aceptar Provisoriamente		

## PRÁCTICO 10 - REVISIONES TÉCNICAS – Inspección de Código (Evaluable)

<b>Unidad:</b>	<b>Unidad Nro. 4: Aseguramiento de Calidad de Proceso y de Producto</b>
<b>Consigna:</b>	Ejecute una revisión técnica a la porción de código entregada y genere el reporte correspondiente.
<b>Objetivo:</b>	Qué el estudiante pueda ejecutar un proceso formal de inspección de código, identificando hallazgos en una porción de código.
<b>Propósito:</b>	Familiarizarse con proceso formal de inspección de código.
<b>Entradas:</b>	Conceptos teóricos sobre el tema desarrollados en clase. Estándar de Inspección de Código. Bibliografía referenciada sobre el tema. <i>Paper 11_Best_Practices_For_Peer_Code_Review_Final.</i> Enunciado, consigna y porción de código. Estándar de Reporte de Inspección. La porción de código a analizar y el documento de reglas de estilo serán provistos de forma anticipada por otro Grupo que haya implementado la misma User Story.
<b>Salida:</b>	Reporte de Hallazgos identificados
<b>Instrucciones:</b>	<ul style="list-style-type: none"> <li>Entregar al siguiente grupo par o impar, según corresponda, una porción de su código a elección con un mínimo de 100 líneas y el documento de reglas de estilo o buenas prácticas</li> <li>Lea y analice la porción de código recibida.</li> <li>Realice una inspección del código mediante el método formal de inspección y prepara el reporte de inspección correspondiente respetando el estándar indicado.</li> </ul>

Reporte de Inspección				
Fecha de Inicio:				
Revisores:	Nombre	Rol		
Alcance de la revisión:				
Nombre	Versión	Autor		
Hallazgos				
Nº	Fecha	Descripción	Línea	Severidad
Decisión Final:		Aceptar/Rechazar/Aceptar Provisoriamente		

## PRÁCTICO 11 - TESTING - Métodos de Caja Negra

<b>Unidad:</b>	<b>Unidad Nro. 4: Aseguramiento de Calidad de Proceso y de Producto</b>																		
<b>Consigna:</b>	Realice el diseño de las pruebas de caja negra según los métodos de Clases de Equivalencia y Análisis de Valor límite																		
<b>Objetivo:</b>	Que el estudiante pueda diseñar pruebas mediante los métodos de Clases de Equivalencia y Análisis de Valor límite, para historias de usuario dadas.																		
<b>Propósito:</b>	Familiarizarse con los métodos de Clases de Equivalencia y Análisis de Valor límite para el diseño de pruebas de caja negra. Descubrir escenarios de prueba que no se derivan a través de los métodos.																		
<b>Entradas:</b>	Conceptos teóricos sobre el tema, desarrollados en clase. Bibliografía referenciada sobre el tema. Enunciado, consigna y User Story.																		
<b>Salida:</b>	Se realizará una puesta en común del conjunto de clases de equivalencia y el resultado de la aplicación del método de análisis del valor límite. Este práctico no se entrega y por lo tanto no tiene nota. El tema se evalúa en el parcial.																		
<b>Instrucciones:</b>	<p>Según el método de partición de equivalencia, defina las clases existentes utilizando el siguiente cuadro para la Historia de Usuario dada.</p> <table border="1"> <thead> <tr> <th>Condición externa</th><th colspan="2">Clases de equivalencia válidas</th><th colspan="2">Clases de equivalencia inválidas</th></tr> </thead> <tbody> <tr> <td></td><td></td><td></td><td></td><td></td></tr> <tr> <td></td><td></td><td></td><td></td><td></td></tr> </tbody> </table> <p>Describa la cantidad mínima de casos de prueba que requiera para asegurar la prueba de todas las clases identificadas y para asegurar que se realicen todas las pruebas de aceptación requeridas por el PO, aplicando los métodos de <i>Partición de Equivalencias</i> y de <i>Análisis de Valores Límite</i> (para los precios de las entradas).                      Utilice el template Template_Casos_De_Prueba.</p>				Condición externa	Clases de equivalencia válidas		Clases de equivalencia inválidas											
Condición externa	Clases de equivalencia válidas		Clases de equivalencia inválidas																

## Publicidad Multipunto

Una empresa que opera en la ciudad de Córdoba se dedica a la comercialización de publicidad Multi-Punto. Es decir, posee en diferentes ubicaciones del país puntos de emisión, que son pantallas con conexión a internet que reproducen tandas publicitarias. La empresa cuenta con un software construido con tecnología web que permita la gestión de la programación y reproducción de las publicidades en los diferentes puntos de emisión y la gestión de los contratos de servicios de publicación. Los puntos de emisión deben tener un nombre, una descripción y un precio por segundo de publicidad. Los clientes que deseen publicar en los puntos de emisión de la empresa deben contratar un servicio de publicación, el cual es registrado por el administrador, donde se acuerda la cantidad de veces que será reproducida una publicidad durante un periodo definido de tiempo, para un punto de emisión determinado. El servicio contratado puede ser de tipo periódico o por única vez. Si el cliente acepta el precio del servicio el administrador le asignará un usuario con el cual operar el sistema, en caso de que aún no tenga uno. El cliente que posea un usuario asignado podrá subir sus publicidades y generar reportes. Las publicidades deben tener un nombre y una descripción y los formatos de video permitidos son .flv (Flash Video), .AVI o .MP4.

El responsable de programación es el encargado de crear las tandas publicitarias, para lo cual agrupará las publicidades subidas por los clientes según algún criterio (rubro, público dirigido, marca) teniendo en cuenta el punto de emisión donde se reproducirá. También tendrá en cuenta si las publicidades son periódicas o de reproducción única, para agruparlas en tandas publicitarias de igual tipo, lo cual generará una tanda publicitaria de esa característica. En cada tanda publicitaria se debe indicar el nombre de la tanda y orden de reproducción para cada publicidad, de esta manera las publicidades serán reproducidas secuencialmente según el orden indicado.

Una vez creadas las tandas publicitarias, el responsable de programación periódicamente se encargará de generar las programaciones de las tandas para cada punto de emisión. En cada programación se deberá indicar la fecha de inicio, fecha de fin, el conjunto de tandas publicitarias, el orden en que éstas se reproducirán y un nombre descriptivo de la programación. Nuevamente es el encargado de programación que decide la manera de agrupar las tandas publicitarias, para conformar una programación teniendo en cuenta además el tipo de tanda y la duración de la misma. Cabe aclarar que las diferentes programaciones de un punto de emisión no pueden superponerse, es decir un punto de emisión en un momento de tiempo sólo tiene una programación vigente. Aun así, es posible generar varias programaciones para un mismo punto de emisión respetando esta restricción.

#### Ejemplo de una Programación

Punto de emisión: NuevoCentroShopping\_piso1\_bonafide

#### **Programación Mujeres Otoño (19/10/17 - 20/11/17)**

##### **01. Tanda Ropa Femenina**

- 01. Corto Ossira "Muy Cool"
- 02. Wanama Nueva Temporada

##### **02. Tanda Batistella**

- 01. Botas Cuero 100% argentino
- 02. Zapatos Fiesta
- 03. Publicidad carteras y cintos 50 % descuento.

##### **03. Tanda Perfumeria**

- 01. Makeup Dior tonalidades otoño
- 02. Julieraque descuentos

Cada punto de emisión tiene instalado un software que resuelve la sincronización con la aplicación central para descargar actualizaciones de las programaciones y además poder reproducirlas según el orden establecido tomando las métricas de cuantas veces se reprodujo cada una.

Una vez instalado el punto de emisión en el local donde se ubicará, el administrador le da de alta en el sistema como habilitado para poder asignarle programaciones, indicando el nombre y las coordenadas de su ubicación geográfica, la cual también puede seleccionar de un mapa interactivo. A partir de esto, si el punto de emisión posee conexión a internet deberá conectarse al servidor y descargar las programaciones que debe reproducir. Una vez completa la descarga, el punto de emisión comenzará a reproducir secuencialmente las publicidades.

Cada hora, al mismo tiempo que en pantalla del punto de emisión se reproducen las publicidades, el software del dispositivo debe sincronizarse con el servidor central para descargar actualizaciones en el caso de que existan. El momento de sincronización puede durar varios minutos y en el caso de que el punto de emisión no pueda conectarse al servidor, se continúa reproduciendo la programación actual y se verificará cada 5 minutos el estado de la conexión para retomar la sincronización.

<p><b>Crear Tanda Publicitaria</b></p> <p>Como Responsable de Programación quiero crear una Tanda Publicitaria para asignarla a una programación que luego será emitida en un punto de emisión.</p> <p><b>Nota 1:</b> Se debe seleccionar el Punto de Emisión al cual corresponde.</p> <p><b>Nota 2:</b> Se deben seleccionar las publicidades a incluir, seleccionando también el orden de reproducción para cada publicidad.</p> <p><b>Nota 3:</b> Se debe diferenciar entre tanda publicitaria de reproducción única y periódica, y todas las publicidades incluidas deben ser del mismo tipo.</p> <p><b>Nota 4:</b> El sistema calcula y muestra la duración de la tanda en función de las publicidades seleccionadas.</p> <p><b>Nota 5:</b> El nombre de la tanda las identifica unívocamente, por lo cual es un dato obligatorio.</p> <p><b>Nota 6:</b> Sólo se pueden crear tandas con publicidades de formatos de video válidos.</p>	<p>3</p>
<ul style="list-style-type: none"> <li>▪ Probar crear una Tanda Publicitaria especificando un Punto de Emisión con Publicidades ordenadas (pasa)</li> <li>▪ Probar crear una Tanda Publicitaria sin especificar un Punto de Emisión (falla)</li> <li>▪ Probar crear una Tanda Publicitaria sin Publicidades (falla)</li> <li>▪ Probar crear una Tanda Publicitaria sin indicar el orden de reproducción de las Publicidades (falla)</li> <li>▪ Probar crear una Tanda Publicitaria de reproducción periódica y asignarle publicidad de reproducción única (falla)</li> <li>▪ Probar crear una Tanda Publicitaria con publicidades con un error en el formato de video, no pudiendo establecer la duración (falla)</li> </ul>	

## PRÁCTICO 12 - TESTING - Métodos de Caja Negra

Unidad:	Unidad Nro. 4: Aseguramiento de Calidad de Proceso y de Producto																			
Consigna:	Realice el diseño de las pruebas de caja negra según los métodos de Clases de Equivalencia y Análisis de Valor límite																			
Objetivo:	Que el estudiante pueda diseñar pruebas mediante los métodos de Clases de Equivalencia y Análisis de Valor límite, para historias de usuario dadas.																			
Propósito:	Familiarizarse con los métodos de Clases de Equivalencia y Análisis de Valor límite para el diseño de pruebas de caja negra. Descubrir escenarios de prueba que no se derivan a través de los métodos.																			
Entradas:	Conceptos teóricos sobre el tema, desarrollados en clase. Bibliografía referenciada sobre el tema. Enunciado, consigna y User Story. Planilla de Excel con Plantilla de Casos de Prueba y Reporte de Defectos Delivereat_Template_Caso_De_Prueba_Defectos.xlsx																			
Salida:	Se realizará una puesta en común del conjunto de clases de equivalencia y el resultado de la aplicación del método de análisis del valor límite. Este práctico no se entrega y por lo tanto no tiene nota. El tema se evalúa en el parcial.																			
Instrucciones:	<p>Según el método de partición de equivalencia, defina las clases existentes utilizando el siguiente cuadro para la User Story de las siguientes que <b>no haya implementado</b> en el Trabajo Práctico 3 “Requerimientos Ágiles – User Stories y Estimaciones”, sobre el servicio DeliverEat!:</p> <ul style="list-style-type: none"><li>○ Realizar Pedido a Comercio adherido (grupos impares)</li><li>○ Realizar un Pedido de "lo que sea" (grupos pares)</li></ul> <table><thead><tr><th>Condición externa</th><th colspan="2">Clases de equivalencia válidas</th><th colspan="2">Clases de equivalencia inválidas</th></tr></thead><tbody><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr></tbody></table> <p>Describa la cantidad mínima de casos de prueba que requiera para asegurar la prueba de todas las clases identificadas y para asegurar que se realicen todas las pruebas de aceptación requeridas por el PO, aplicando los métodos de <i>Partición de Equivalencias</i> y de <i>Análisis de Valores Límite</i> (para las condiciones externas en las que sea posible).</p> <p>Utilice el template <i>Template Casos De Prueba</i>.</p>					Condición externa	Clases de equivalencia válidas		Clases de equivalencia inválidas											
Condición externa	Clases de equivalencia válidas		Clases de equivalencia inválidas																	

## PRÁCTICO 13 - TESTING – Ejecución de Casos de Prueba (Entregable)

Unidad:	Unidad Nro. 4: Aseguramiento de Calidad de Proceso y de Producto
Consigna:	Ejecute los casos de pruebas de caja negra diseñados
Objetivo:	Que el estudiante pueda aplicar los distintos aspectos del testing, desde la planificación hasta ejecución de las pruebas.
Propósito:	Comprender como se realiza la ejecución de los casos de prueba diseñados
Entradas:	Conceptos teóricos sobre el tema desarrollados en clase. Bibliografía referenciada sobre el tema. Planilla de Excel con User Story, Plantilla de Casos de Prueba y Reporte de Defectos <i>DeliverEat_Template_Caso_De_Prueba_Defectos.xlsx</i>
Salida:	Planilla de Excel con Clases de Equivalencias identificadas y Casos de Prueba diseñados en el Trabajo <b>Práctico 12 “Testing – Métodos de Caja Negra”</b> Ejecución y Reporte de Defectos
Instrucciones:	<ul style="list-style-type: none"> <li>• Obtener y ejecutar la User Story a probar en base a la implementación de otro grupo del curso</li> <li>• Ejecutar los Casos de Prueba diseñados y registrar los Defectos encontrados completando la Plantilla correspondiente</li> </ul>

## PRÁCTICO 14 - TESTING – Métodos de Caja Blanca

<b>Unidad:</b>	<b>Unidad Nro. 4: Aseguramiento de Calidad de Proceso y de Producto</b>
<b>Consigna:</b>	Analice el pseudocódigo según los métodos de caja blanca
<b>Objetivo:</b>	Que el estudiante pueda realizar actividades que cubran los distintos aspectos que conforman el proceso de Testing, desde la planificación hasta ejecución de las pruebas, pasando por el diseño de los casos de prueba utilizando distintos métodos de prueba de caja.
<b>Propósito:</b>	Familiarizarse con los conceptos y actividades principales de los métodos de Testing de caja blanca.
<b>Entradas:</b>	Conceptos teóricos de Testing, desarrollados en clase. Bibliografía sobre el tema, citada. Enunciado, consigna, templates y porción de pseudocódigo.
<b>Salida:</b>	Casos de prueba de caja blanca diseñados. Este práctico no se entrega y por lo tanto no tiene nota. El tema se evalúa en el parcial.
<b>Instrucciones:</b>	Describa la cantidad mínima de casos de prueba que se requieran para el pseudocódigo presentado, aplicando el método de Caja Blanca. Utilice la tabla que indica los distintos métodos de cobertura. Para cada fila, en la primera columna deberá completar la mínima cantidad de casos de prueba que deben ejecutarse y en la columna "Dato" las características de las entradas de cada una de las pruebas.

### PickMe

PickMe es una organización que se dedica a prestar servicios de transporte en la ciudad de Córdoba. Su objetivo es conectar la necesidad de traslado de un pasajero con un conductor, permitiendo compartir los gastos del recorrido. Para brindar el servicio, la empresa cuenta con una aplicación móvil a través de la cual los conductores anuncian los viajes que van a iniciar para que luego los pasajeros pueden sumarse. En un mismo viaje podrían subir varios pasajeros y bajar a destiempo, es decir en puntos distintos del recorrido.

El conductor puede recibir una solicitud en su pantalla con los datos y la foto del pasajero que desea realizar un viaje. Si la acepta, se debe enviar una notificación al pasajero indicando el tiempo de demora estimado y el mapa de su recorrido. Es posible que un conductor le solicite al pasajero que se posicione en otra dirección cercana dónde recogerlo. Cuando el conductor recoge un pasajero deberá indicarle al sistema, registrando en ese momento la hora de inicio y la dirección de origen.

A continuación, se presenta una porción de pseudocódigo que resuelve la US "Confirmar solicitud de traslado":

```

If (Solicitud tiene nombre = Yes AND Solicitud tiene ubicación AND Solicitud tiene foto = Yes)
    If (Ubicación pasajero <= 500 metros de ruta planificada por conductor)
        [Enviar notificación: Tiempo de demora y Mapa de recorrido]
    Else
        [Solicitar nueva ubicación al pasajero]
        If (Pasajero acepta modificar ubicación)
            [Enviar notificación: Tiempo de demora y Mapa de recorrido]
        Else
            If (conductor modifica recorrido planificado por el pasajero)
                [Enviar notificación: Tiempo de demora y Mapa de recorrido]
            Else
                [Rechazar Solicitud]
    Else
        [Rechazar Solicitud]
  
```

### Precondiciones:

- Pasajero Juan con ubicación San Martín 100(Córdoba) posee foto de identificación.
- Conductor Armando tiene ruta planificada por San Martín al 200(Córdoba)



	Cantidad de Casos de Prueba	Dato
Cobertura de sentencias		
Cobertura de decisión		
Cobertura de condición		
Cobertura de condición /decisión		
Cobertura múltiple		

## PRÁCTICO 15 - La Publicidad en Instagram (Evaluable)

Unidad:	Unidad Nro. 2: Gestión Lean-Ágil de Productos de Software
Consigna:	Simular la definición de un producto de software. Para ello se utilizará una dinámica en la que se diseñará la publicidad en Instagram de un producto para vender en el mercado.
Objetivo:	Aplicar los conceptos de gestión ágil en el desarrollo de productos de software y el concepto de MVP.
Propósito:	Familiarizarse con la gestión ágil de proyectos y el desarrollo ágil de software utilizando SCRUM.
Entradas:	Conceptos teóricos sobre el tema desarrollados en clase. Bibliografía referenciada sobre el tema. Instrucciones de la dinámica. Materiales necesarios: <ul style="list-style-type: none"> <li>• Afiche</li> <li>• Marcadores de colores</li> <li>• Cinta de papel</li> <li>• Revistas</li> <li>• Tijeras</li> <li>• Pegamento</li> </ul>
Salida:	El diseño de la publicidad de un producto que cumple con el MVP, es decir que es la versión mínima de mi producto que me permite validarlo con el Mercado y obtener feedback del cliente.
Instrucciones:	<ol style="list-style-type: none"> <li>1. Cada grupo debe definir un producto y armar un prototipo de una publicidad para Instagram.</li> <li>2. Un representante del grupo debe presentar la propuesta al grupo inmediato siguiente.</li> <li>3. Luego de escuchar la propuesta, el grupo visitado debe brindar feedback.</li> <li>4. En función del feedback recibido sobre el prototipo, armar en el afiche la publicidad en Instagram del producto mínimo viable (MVP).</li> <li>5. Presentar su publicidad a todo el curso.</li> </ol>



## PRÁCTICO 16 - SCM - Uso del Repositorio (Evaluable)

<b>Unidad:</b>	<b>Unidad Nro. 3: Gestión del Software como producto</b>
<b>Consigna:</b>	Comprender los conceptos de administración de configuración de software (SCM) expuestos en la clase teórica para aplicarlos en un ejercicio propuesto por la cátedra.
<b>Objetivo:</b>	Que el estudiante sea capaz de realizar actividades básicas de la gestión de configuración de software, utilizando una herramienta automatizada. Tales actividades son: la definición de una estructura de repositorio, ingreso y extracción de ítems de configuración del repositorio y definición de líneas base.
<b>Propósito:</b>	Aplicar los conceptos de gestión de configuración de software estudiados, utilizando una herramienta específica
<b>Entradas:</b>	Conceptos teóricos sobre los temas desarrollados en clase. Bibliografía referenciada sobre el tema. Estructura del Repositorio y Plantilla de Ítems de Configuración del Trabajo Práctico 4 URL y Credenciales de acceso del repositorio implementado en el Trabajo Práctico 5
<b>Salida:</b>	Se evaluará que: <ul style="list-style-type: none"> <li>• Cada Integrante del Grupo haya realizado operaciones de <i>Commit</i> (colocar) y <i>Update</i> (obtener) sobre el proyecto de desarrollo de software para el cursado de la materia.</li> <li>• Los ítems de configuración disponibles presenten una evolución acorde a su contenido a lo largo del cursado con participación de los integrantes del grupo.</li> <li>• Exista al menos una Línea Base creada en el repositorio según el criterio definido en el Trabajo Práctico 5.</li> </ul>
<b>Instrucciones:</b>	<ul style="list-style-type: none"> <li>• Realizar <i>Commit</i> (colocar) de cada ítem de configuración disponible al momento con las cuentas de usuario de cada integrante del grupo.</li> <li>• Generar una Línea Base según criterio definido.</li> </ul>
<b>Observaciones:</b>	Las operaciones de gestión de configuración deben realizarse a lo largo del cursado de la materia, a medida que evoluciona el proyecto de software.