

ENUNCIADOS DE TRABAJOS PRÁCTICOS EVALUABLES

# INGENIERÍA Y CALIDAD DE SOFTWARE

*Índice*

<b>PRÁCTICO 1 - REQUERIMIENTOS ÁGILES – User Stories.....</b>	<b>3</b>
<b>PRÁCTICO 2 - REQUERIMIENTOS ÁGILES – User Stories y Estimaciones .....</b>	<b>4</b>
<b>PRÁCTICO 3 - REQUERIMIENTOS ÁGILES – User Stories, Estimaciones y MVP.....</b>	<b>6</b>
<b>PRÁCTICO 4 - SCM - Herramientas de SCM (Evaluable).....</b>	<b>9</b>
<b>PRÁCTICO 5 - SCM - Uso del Repositorio (Evaluable).....</b>	<b>10</b>
<b>PRÁCTICO 6 - REQUERIMIENTOS ÁGILES - Implementación de User Stories (Evaluable) .....</b>	<b>11</b>
<b>PRÁCTICO 7 - SCRUM - Ejercicio de aplicación (Evaluable) .....</b>	<b>13</b>
Proyecto: Zoológico de plastilina .....	13
<b>PRÁCTICO 8 – SCRUM - Planificación de Release y de Sprint (Evaluable).....</b>	<b>15</b>
<b>PRÁCTICO 9 - TESTING - Métodos de Caja Negra .....</b>	<b>23</b>
<b>PRÁCTICO 10 - TESTING - Métodos de Caja Negra .....</b>	<b>25</b>
<b>PRÁCTICO 11 - TESTING – Ejecución de Casos de Prueba (Evaluable) .....</b>	<b>26</b>
<b>PRÁCTICO 12 - KANBAN – Ejercicio de aplicación (Evaluable) .....</b>	<b>27</b>
<b>PRÁCTICO 13 - SCRUM - Dinámica de Retrospectiva (Evaluable).....</b>	<b>28</b>
<b>PRÁCTICO 14 - DESIGN THINKING – Publicidad de Instagram (Evaluable) .....</b>	<b>29</b>

## PRÁCTICO 1 - REQUERIMIENTOS ÁGILES – User Stories

<b>Unidad:</b>	<b>Unidad Nro. 2: Gestión Lean-Ágil de Productos de Software</b>
<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 Historias de Usuario (User Stories), Épicas y Temas
<b>Entradas:</b>	Conceptos teóricos sobre el tema, desarrollados en clase. Bibliografía referenciada en la modalidad académica. Enunciado, consigna y tarjeta de US.
<b>Salida:</b>	<ul style="list-style-type: none"> <li>Identificación de los roles principales</li> <li>US identificadas con sus tarjetas completas</li> </ul> <p><b>Este práctico no se entrega y por lo tanto no tiene nota. El tema se evalúa en el parcial.</b></p>
<b>Instrucciones:</b>	<p>Los docentes representarán a expertos del dominio que expresarán sus necesidades vinculadas a desarrollar un software para un servicio de transporte.</p> <p>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.</p> <p>Cada grupo presentará el resultado obtenido al final de la clase al resto del curso.</p>

### Aplicación Mis Gastos Familiares

*Objetivo del producto:* facilitar a las personas la gestión de sus gastos cotidianos y poder acceder a información relacionada a estos.

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

Product Owner (PO): ¿Cómo debe visualizarse la planilla de gastos?

Experto en el Dominio (ED): Similar a una tabla en Excel, con columnas donde se pueda indicar el monto, el tipo de gasto, y la fecha en que se realizó el gasto.

PO: ¿La fecha del gasto es la fecha actual? ¿Se toma automáticamente?

ED: Debería mostrar la fecha actual, pero permitir modificarse.

PO: ¿Cuáles son los tipos de gastos permitidos?

ED: Cada persona debería poder registrar sus tipos de gastos, así como indicar si el gasto es propio o de otra persona (por ejemplo, de su esposa).

PO: ¿El nombre o la relación con el usuario debe indicarse?

ED: Debe indicarse el nombre y apellido. También por defecto debe mostrarse el nombre y el apellido del usuario logueado, permitiendo modificarlo, y si alguna vez se registró el nombre y apellido que se comienza a ingresar (no importa si esta vez o una vez anterior), el sistema debería mostrar aquellos nombres y apellidos que comiencen de forma similar, como hace el Excel.

PO: Ah ¿Es decir que el usuario debe registrarse y cuando va a usar la aplicación debe iniciar sesión?, y ¿la sesión caduca en algún momento?

ED: El usuario debe registrarse para permitirle descargar la aplicación, y la primera vez debe iniciar sesión, pero luego se consideran los datos de la sesión registrados, salvo que el usuario decida de loguearse. La sesión no caduca nunca.

PO: ¿Y la planilla de gastos muestra todos los gastos? ¿Cómo se ordenan?

ED: Por defecto se deben mostrar todos los gastos del mes en curso ordenados desde el gasto más actual, pero el sistema debería permitir ver cualquier período que el usuario quiera, y que pueda filtrar por tipo de gasto, por responsable de gasto, por rango de montos. Además debe poder modificar el criterio de ordenamiento. Y para cada filtro que se aplique arriba se debe mostrar el total de gastos según el filtro aplicado.

## PRÁCTICO 2 - REQUERIMIENTOS ÁGILES – User Stories y Estimaciones

<b>Unidad:</b>	<b>Unidad Nro. 2: Gestión Lean-Ágil de Productos de Software</b>
<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, Temas y MVP..
<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>	<ul style="list-style-type: none"><li>● Identificación de los roles principales</li><li>● El MVP explicando el alcance propuesto y justificando la inclusión de las User Stories seleccionadas y el porqué de aquellas excluidas en el mismo</li><li>● US identificadas con sus tarjetas completas</li><li>● User Story canónica</li><li>● User Stories estimadas</li></ul> <p><b>Este práctico no se entrega y por lo tanto no tiene nota. El tema se evalúa en el parcial.</b></p>
<b>Instrucciones:</b>	Los docentes representarán a expertos del dominio que expresarán sus necesidades vinculadas al desarrollar un software para el servicio de Delivery de “lo que sea”. 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.

### TANGO APP – Plataforma de Envíos en colaboración

En el interior del país hay localidades donde los servicios logísticos son deficientes, ya sea por la falta de cobertura o por la poca frecuencia. En un mundo donde las compras por Internet se incrementaron por la pandemia, hay una mayor necesidad de cubrir más lugares y con mayor frecuencia, por eso nace la idea de crear una plataforma colaborativa que sume tanto a las personas particulares, comisionistas, fleteros como empresas de transporte para construir una red logística. Estos transportistas podrán trasladar lo que les permitan sus vehículos: paquetes, correspondencia o cualquier otro tipo de carga (muebles, electrodomésticos, granos, hacienda, maquinarias, etc.).

La plataforma será desarrollada en tecnología mobile (para teléfonos con OS Android y iOS) y debe permitir realizar envíos en colaboración, en donde una misma persona puede enviar alguna carga y, a su vez, puede ser el transporte de otro. A efecto de simplificar los términos y el rol que cada usuario puede asumir en un momento del tiempo, llamaremos "**Dador de carga**" a los usuarios que desean enviar algo y "**Transportista**" a los usuarios que transportan los distintos tipos de carga.

Las personas que necesiten enviar o retirar algo deben publicar un pedido de envío con el tipo de carga que debe ser transportado, la dirección y fecha de retiro, la dirección y fecha de entrega, y, opcionalmente, una observación que detalle necesidades de almacenamiento o transporte (por ejemplo: fragilidad, temperatura mínima, evitar el sol). También, podrán subir una o más fotos para facilitar al transportista presupuestar el servicio de transporte.

Los transportistas se enteran de los pedidos pendiente de cotización a través de 3 modalidades: 1) visualizando en un mapa con la ubicación de retiro de todos los pedidos de envío cerca de su ubicación actual; 2) visualizando un listado de todos los pedidos de envíos; 3) recibiendo notificaciones PUSH (en su teléfono móvil) si configura su zona de cobertura; la zona de cobertura se compone de todas las localidades que visita o puede realizar transporte.

Cuando al transportista le interese realizar un envío deberá cotizarlo, en la aplicación, informando fecha de retiro y entrega (si es diferente a la solicitada), el importe del envío, las formas de pago (tarjeta de crédito/débito, contado al retirar o contado contra entrega) y el vehículo con el cual realizará el transporte.

El usuario dador de carga debe poder visualizar y comparar todas las cotizaciones recibidas y elegir la que más se ajuste a sus necesidades en cuanto a fecha de retiro y entrega, costo y calificación del transportista (de 1 a 5 estrellas que mide la experiencia de otros usuarios en cuanto a la confianza y calidad del servicio prestado). Para contratar la cotización elegida, el usuario deberá elegir la forma de pago y aceptarla desde la aplicación, y se le enviará una notificación PUSH y un email al transportista para comunicarle que su servicio fue contratado. Cuando la forma de pago elegida sea tarjeta de crédito o débito, deberá ingresar

los datos de su tarjeta (número, pin, nombre completo, tipo y número de documento) y se procederá a realizar el pago online a través de una pasarela de pago (por ejemplo: MercadoPago, Stripe, etc.)

Al concretarse la operación, se liberan los datos de contacto tanto del usuario dador como del transportista para que puedan contactarse por teléfono o WhatsApp y terminar de coordinar detalles del envío -si es necesario.

Llegado el día y hora acordados, el transportista, iniciará el viaje desde la aplicación y se le enviará una notificación vía PUSH y email al usuario dador de carga para informar de que está en camino para retirar la carga. No se podrá iniciar el viaje si el transportista no tiene habilitado el GPS o permiso de ubicación en su dispositivo móvil.

El transportista realizará el retiro de la carga en la dirección indicada. También, realizará el cobro del servicio si la forma de pago elegida fue “contado al retirar”.

El dador de carga podrá realizar el seguimiento online de la última ubicación registrada del transportista con su carga, y la fecha y hora de esa posición. La ubicación deberá actualizarse conforme el transportista avanza (por ejemplo: cada 1000 metros) o a intervalos de tiempo regulares (por ejemplo: cada 5 minutos).

El transportista, al llegar a la dirección de destino, realizará la entrega de la carga transportada, cobrará el importe del servicio (si la forma de pago fue “contado contra entrega”) y finalizará el viaje en la aplicación. El dador de carga recibirá una notificación PUSH y un email con la confirmación de la entrega y deberá calificar el servicio del transportista de 1 a 5 estrellas en 4 criterios de evaluación.

El transportista también deberá calificar al usuario dador de carga para mejorar la reputación digital de las personas que integran la comunidad de esta plataforma.

Cuando los pagos se realizan con tarjeta de crédito o débito, un administrador de la plataforma realizará la liquidación de los servicios, descontando la comisión del servicio y pagará a los transportes dentro de las 48 hs. hábiles de finalizar el viaje. La comisión del servicio es un 10% del importe del viaje y sólo se cobra para viajes pagados con tarjetas.

Al ser una aplicación con muchos usuarios conectados, para evitar demoras en el envío de las notificaciones PUSH y/o emails, el proceso que dispara las notificaciones debe funcionar de manera asíncrona (o sea, la funcionalidad principal debe continuar ejecutándose sin necesidad de esperar a que se realicen todos los envíos de notificaciones).

#### Glosario:

- **Notificaciones Push:** son mensajes que se envían de forma directa a dispositivos móviles (Smartphones y/o tablets) con sistema operativo iOS y Android.
- **Zona de cobertura:** se compone de las localidades o ciudades donde opera un transportista.
- **Dador de Carga:** es el usuario de la plataforma que desea enviar una mercancía o carga.
- **Transportista:** es el usuario de la plataforma que realiza el servicio de transporte o traslado de una mercancía o carga.
- **Pasarela de Pago:** es una entidad intermediaria que proporciona servicios web mediante los cuales se procesan de manera segura los pagos por tarjeta de crédito y débito y, luego, acreditan el dinero en la cuenta bancaria del comercio.

## PRÁCTICO 3 - REQUERIMIENTOS ÁGILES – User Stories, Estimaciones y MVP

<b>Unidad:</b>	<b>Unidad Nro. 2: Gestión Lean-Ágil de Productos de Software</b>
<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, Temas y MVP..
<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>	<ul style="list-style-type: none"> <li>• Identificación de los roles principales</li> <li>• El MVP explicando el alcance propuesto y justificando la inclusión de las User Stories seleccionadas y el porqué de aquellas excluidas en el mismo</li> <li>• US identificadas con sus tarjetas completas</li> <li>• User Story canónica</li> <li>• User Stories estimadas</li> </ul> <p><b>Este práctico no se entrega y por lo tanto no tiene nota. El tema se evalúa en el parcial.</b></p>
<b>Instrucciones:</b>	<p>Los docentes representarán a expertos del dominio que expresarán sus necesidades vinculadas a desarrollar un software.</p> <p>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.</p>

### Recircula tus prendas - Cuida el planeta

#### Glosario:

- **Marketplaces:** es un sitio de ventas por internet que conecta compradores y vendedores.
- **Segunda selección:** se denomina así a las prendas que tienen alguna falla, mancha u otro defecto menor.
- **Pasarela de pago:** es un servicio online para el cobro de transacciones con tarjetas de débito o crédito.

#### Descripción del Dominio

Un grupo de emprendedores preocupados por el alto impacto ambiental provocado por la industria textil quiere desarrollar una plataforma de moda circular y sustentable para comprar y revender prendas únicas. Los emprendedores apuestan a que existe un mercado para compradores de ropa usada, pero no saben si hay personas dispuestas a vender su ropa. A continuación, se transcribe parte de la entrevista realizada al experto en el dominio:

**PO (Product Owner):** Nos cuentan ¿cómo espera que funcione su idea?

**ED (Experto del dominio):** Bueno. Nuestra idea es que las personas puedan vender aquellas prendas que ya no usan y que están como nuevas o -incluso- aún con la etiqueta puesta. La plataforma funcionará como un Marketplaces en la cual los clientes podrán comprar prendas, aplicando filtros de búsqueda por distintas categorías (remeras, pantalones, camperas, etc.), marcas y estados (como nueva, nueva con etiqueta, segunda selección o segunda selección con etiqueta). Por otro lado, los vendedores eligen las prendas que desean vender y nos la envían a nosotros para que nos ocupemos de su comercialización. Nuestro proceso arranca desde la selección (donde elegimos la ropa a publicar bajo estrictos controles o la descartamos), fotografiado, publicación para su venta en nuestro ecommerce y, una vez que la prenda se vende, nosotros nos encargamos de la cobranza, de entregarla al cliente y, finalmente, del pago de la ganancia al vendedor. Nuestro modelo de negocio consiste en cobrar un % de comisión al vendedor según el precio de venta, por ejemplo: si el precio de venta es entre 1-\$1.000 cobramos un 50%, si es entre \$1.001 a \$5.000 cobramos un 45%, etc.

**PO:** ¡Buenísimo! Ahora, nos cuentan ¿cómo hago para vender mis prendas? ¿puedo vender cualquier cosa?

**ED:** Ok. Si, podés vender muchas categorías de prendas, que podrás enterarte en nuestra web. El vendedor debe llenar un formulario indicando la cantidad de prendas de cada categoría que nos van a enviar para seleccionar. Ah! pero para poder completarlo, primero debe registrarse en nuestro sitio web con su email y contraseña o con su cuenta de Google, y debe completar sus datos(nombre completo, DNI, y teléfono). Es importante que nos aseguremos de que el teléfono sea real para poder contactarlo. Luego, deberíamos pedirle que indique su domicilio completo (por si tenemos que retirar sus prendas o entregarle su compra) y que nos indiquen un CBU o alias para realizar la transferencia bancaria cuando se venda su prenda. Volviendo a cómo vender, nos tienen que indicar la cantidad de cada categoría de prenda, una descripción de cada prenda a enviar (por ejemplo buzo rojo talla S) y opcionalmente una foto (de hasta 512kb), y, finalmente, la forma de envío: retirar en domicilio o llevarlo a un punto de recolección. Los puntos de recolección son distintas sucursales nuestras o comercios amigos que recolectan las bolsas de prendas. Tendremos muchos puntos en distintas ciudades del país, por ahora sólo 5 en Córdoba. Una vez elegida la forma de envío, deben confirmar el formulario de prendas a vender, y se enviará un mail al vendedor y a nosotros a modo de comprobante con el detalle de prendas. Es importante que sepan que sólo pueden enviar un mínimo de 10 prendas por cada vez que quieran vender.

**PO:** Clarísimo. ¿Cómo seleccionan las prendas? y ¿Cómo continúa el proceso?

**ED:** Primero, el Encargado de Logística se encargará de coordinar el retiro y el arribo de las bolsas con prendas para vender en nuestro Centro de Procesamiento, que es uno sólo aquí en Córdoba. Ahora sí, tenemos tres Analistas de Selección que se ocupan de revisar prenda por prenda para verificar que cumplen nuestras políticas de productos a vender. Ellos completarán la primera parte de una ficha básica del producto indicando categoría, marca, si está seleccionada o no. En caso de estar seleccionada, le debe asignar el estado (como nueva, nueva con etiqueta, segunda selección o segunda selección con etiqueta). Si es segunda selección, deben indicar un motivo para que el comprador se entere cuando mire el producto en el ecommerce, por ejemplo: “Tiene una pequeña mancha”. Independientemente, de si está seleccionada o no, a las prendas se les pega una etiqueta autoadhesiva con un código QR que tiene un código de producto único y permite identificar a la prenda. Las prendas que no queden seleccionadas son separadas para luego informarle el motivo de no selección y preguntarle al vendedor que desea hacer: retirar o donarla.

Una vez seleccionadas las prendas, son procesadas por 4 Analistas de Publicación que identifican cada prenda escaneando su código QR con un lector de QR, visualizan los datos cargados por el Analista de Selección, se ocupan de tomar las fotos (máximo 5 fotos, y cada una con un peso hasta 1Mb) y de completar el resto de la ficha del producto (nombre de la prenda, peso en kilogramos, precio sugerido y atributos dinámicos que describen características de la prenda según la categoría, por ejemplo: tipo de tela, color, talla, etc.) para ser publicados en el ecommerce. El precio sugerido se calcula en base a una lista de precios por categoría, marca, estacionalidad y estado de la prenda, e igualmente podrá editarlo.

Cuando la ficha de todas las prendas recibidas de un vendedor es completada, se genera la propuesta de venta y se notifica al vendedor por email que ya la tiene disponible en la plataforma. La propuesta detalla la fecha y cantidad de prendas, el listado de todas las prendas seleccionadas con su correspondiente nombre, categoría, estado, precio sugerido y la ganancia que obtendrá al venderse; y también el listado de todas las prendas que no quedaron seleccionadas. El vendedor debe ingresar a la plataforma y confirmar si desea vender todas las prendas seleccionadas, o no vender algunas o todas y, también, definir qué hacer con las prendas que quedaron no seleccionadas (si lo retira o dona). Cuando confirma, quedan las prendas seleccionadas listas para publicarse en el ecommerce, si hay prendas no seleccionadas y/o que decidió no vender, se le envía un email informando la donación o los pasos a seguir para retirarlas.

**PO:** ...y ahora: ¿cómo es el proceso de venta?

**ED:** Bueno, acá es muy parecido a cualquier ecommerce, el cliente accede a nuestra web, mira el catálogo de productos (con su nombre, precio y ficha de producto), realiza búsquedas por nombre o filtrar por categoría, estado de la prenda, marca y talla. Una vez elegida la prenda, debe registrarse o iniciar sesión para continuar con la compra, debe elegir la forma de entrega. Si

elige entrega a domicilio, se calculará un costo de envío a través de una integración con un Servicio de Entregas Online que terceriza el servicio de logística. Si elige retiro en puntos de entrega, se listarán los domicilios de nuestras sucursales/comercios amigos para que elija uno y no tendrá costo. Finalmente, debe indicar el medio de pago (tarjetas de crédito o débito) para lo cual se utilizará como pasarela de pago MercadoPago. La comercialización online de las prendas y el envío a los clientes lo vamos a realizar con una plataforma de ecommerce llamada Tiendanube hasta que podamos validar el modelo de negocio y realizar un desarrollo personalizado. Al vendedor, se le envía un email notificando de la venta con el detalle y foto de la prenda vendida, el precio de venta y la ganancia obtenida.

**PO:** Entonces, para cerrar el proceso, sólo falta pagarle al vendedor: ¿Cómo se realiza?

**ED:** Claro. Es simple, a principio de mes, el Administrador realiza las liquidaciones y pago a cada vendedor por las ventas del mes anterior que no tuvieron devolución pasado los 14 días de ocurrida la venta. El administrativo consulta todas las prendas vendidas del mes y calcula la ganancia, descontando nuestra comisión, y realiza el pago al vendedor por transferencia bancaria, registrando la misma y enviando un email con el total pagado al vendedor y el detalle de las prendas liquidadas con su ganancia.

Todo el proceso de devolución de las prendas vendidas, del envío de las prendas no seleccionadas (a donar o retirar) y del pago a los vendedores lo realizaremos inicialmente a mano.



## PRÁCTICO 4 - 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.
<b>Salida:</b>	<ul style="list-style-type: none"> <li>• URL de acceso al repositorio implementado</li> <li>• Documento con la definición de los ítems de configuración y su regla de nombrado y la definición del criterio para la creación de una línea base.</li> </ul> <p><b>Se evaluará lo siguiente:</b></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>• Realizar el diseño del repositorio y reglas de nombrado de ítems de configuración</li> <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 para resguardar el trabajo generado durante el cursado de la materia Ingeniería 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 5 - 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, Plantilla de Ítems de Configuración, URL y Credenciales de acceso del repositorio implementado en el Trabajo Práctico 4
<b>Salida:</b>	Se evaluará que: <ul style="list-style-type: none"> <li>• Cada Integrante del Grupo ha 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 presentan una evolución acorde a su contenido a lo largo del cursado con participación de los integrantes del grupo.</li> <li>• Existe al menos una Línea Base creada en el repositorio según el criterio definido en el Trabajo Práctico 4.</li> <li>• Que los ítems de configuración respeten la ubicación propuesta en la estructura del repositorio.</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.

## PRÁCTICO 6 - REQUERIMIENTOS ÁGILES - Implementación de User Stories (Evaluable)

<b>Unidad:</b>	<b>Unidad Nro. 3: Gestión del Software como producto</b>
<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 2 “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>○ <i>Publicar pedido de envío</i> (<b>grupos pares</b>)</li> <li>○ <i>Aceptar cotización</i> (<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>

### User stories:

<b>Publicar pedido de envío</b>	<b>5</b>
<b>Como Dador de Carga quiero publicar un pedido de envío para recibir cotizaciones de transportistas</b>	
<b>Criterios de Aceptación</b> <ul style="list-style-type: none"> <li>● Debe seleccionar el tipo de carga (documentación, paquete, granos, hacienda)</li> <li>● Debe ingresar el domicilio de retiro (calle y número, localidad, provincia y referencia opcional).</li> <li>● Debe seleccionar la fecha de retiro (mayor o igual a hoy)</li> <li>● Debe ingresar el domicilio de entrega (calle y número, localidad, provincia y referencia opcional)</li> <li>● Debe indicar la fecha de entrega (mayor o igual a hoy y mayor o igual a fecha de retiro)</li> <li>● Debe poder adjuntar, de manera opcional, una o más fotos (jpg o png).</li> <li>● Debe enviar una notificación PUSH y email de nuevo pedido a todos los transportes que tengan la localidad de la dirección de retiro en su zona de cobertura</li> </ul>	
<b>Pruebas de Usuario:</b> <ul style="list-style-type: none"> <li>● Probar crear un pedido de envío con fechas correcta sin fotos (pasa)</li> <li>● Probar crear un pedido de envío con fechas correcta con fotos (pasa)</li> <li>● Probar crear un pedido de envío con fechas correcta con observaciones (pasa)</li> <li>● Probar crear un pedido de envío con fecha de retiro y entrega incorrecta (falla)</li> <li>● Probar crear un pedido de envío sin completar datos requeridos (falla)</li> <li>● Probar recepción de email en usuarios transportistas con dirección de retiro o entrega dentro de su zona de cobertura (pasa)</li> <li>● Probar recepción de notificación PUSH en usuarios transportistas con dirección de retiro o entrega dentro de su zona de cobertura (pasa)</li> <li>● Probar recepción de email en usuarios transportistas con dirección de retiro y entrega fuera de su zona de cobertura (falla)</li> </ul>	

## Aceptar Cotización

8

Como Dador de carga quiero aceptar una cotización para contratar el servicio de transporte

### Criterios de Aceptación

- Debe mostrar el nombre del transportista y su calificación.
- Debe mostrar la fecha de retiro y entrega del traslado.
- Debe visualizar el importe del viaje y las formas de pago establecidas.
- Debe elegir una de las formas de pago (tarjeta, contado al retirar o contado contra entrega) definidas por el transportista.
- Si la forma de pago es tarjeta, debe ingresar los datos de la tarjeta (número, pin, nombre completo, tipo y número de documento) y procesar el pago mediante la pasarela de pago elegida.
- Debe informar que el pago se procesó correctamente y el nro. de pago devuelto por la pasarela de pago.
- Debe informar que el pago se rechaza y debe permitir elegir otro medio de pago u ingresar otra tarjeta.
- Debe cambiar el estado del pedido de envío a "Confirmado".
- Debe enviar una notificación PUSH al transportista informando que se confirmó su cotización y la forma de pago elegida.
- Debe enviar un email al transportista informando que se confirmó su cotización y la forma de pago elegida.

### Pruebas de Usuario:

- Probar aceptar una cotización con pago al retirar o contra entrega (pasa)
- Probar aceptar una cotización sin elegir la forma de pago (falla)
- Probar pagar la cotización con tarjeta de crédito vigente y con saldo (pasa)
- Probar pagar la cotización con tarjeta de crédito sin saldo suficiente (falla)
- Probar pagar la cotización con tarjeta de crédito con datos no válido (falla)
- Probar pagar la cotización con tarjeta de crédito sin ingresar los datos de la tarjeta (falla)
- Probar pagar la cotización con tarjeta de débito vigente y con saldo (pasa)
- Probar pagar la cotización con tarjeta de débito sin saldo suficiente (falla)
- Probar pagar la cotización con tarjeta de débito con datos no válido (falla)
- Probar pagar la cotización con tarjeta de crédito sin ingresar los datos de la tarjeta (falla)
- Probar recepción de email de confirmación por parte del transporte (pasa)
- Probar recepción de notificación PUSH de confirmación por parte del transporte (pasa)
- Probar que no se pueda aceptar otra cotización cuando ya está confirmada (falla)

## PRÁCTICO 7 - SCRUM - Ejercicio de aplicación (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 internalizar 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 a través de la ejecución de un caso práctico.
<b>Propósito:</b>	Familiarizarse con la forma gestionar el trabajo de un equipo, utilizando el framework de Scrum, empleando para ello dinámicas colaborativas.
<b>Entradas:</b>	<ul style="list-style-type: none"> <li>● Guía Oficial de Scrum 2020</li> <li>● Conceptos teóricos sobre el tema, desarrollados en clase.</li> <li>● Materiales necesarios, que deberán ser provistos por los estudiantes.</li> <li>● Estimación de las User Stories, detallando las actividades necesarias para cada una</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>	<p>A través de una dinámica de juego y de las instrucciones comunicadas por los docentes, implementar las siguientes prácticas de Scrum:</p> <ul style="list-style-type: none"> <li>● Priorización y planificación de Sprint, definiendo el objetivo del sprint</li> <li>● Ejecución de Sprint</li> <li>● Daily Scrum Meeting</li> <li>● Sprint Review Meeting</li> <li>● Sprint Retrospective</li> </ul> <p>El docente asumirá el rol de instructor para moderar la ejecución de cada dinámica.</p>

## Proyecto: Zoológico de plastilina

**Objetivo:** Deseamos tener el mejor zoológico de la zona y tenemos 5 años para desarrollarlo.

La construcción es con plastilina. En el zoológico desea incluir:

- Una jaula con una jirafa
- Un estanque con dos peces, una tortuga y una rana
- Un serpentario con dos serpientes
- Una jaula con dos pájaros

Para realizar el zoológico miniatura las herramientas necesarias son:

Las herramientas que deben traer por grupo para la clase indicada son las siguientes:

- Plastilinas de varios colores
- 2 platos descartables
- papeles de colores
- bolígrafos
- Papel de diario
- Revistas
- 1 Cartulina verde
- 1 regla
- 1 afiche blanco
- 1 marcador negro o azul
- 1 taco de Post-it
- Cinta scotch
- 1 tijera

A continuación, se describen los requerimientos solicitados por el cliente:

Theme	Deseamos tener el mejor zoológico de la zona y tenemos 5 años para desarrollarlo.	
Epics	<ul style="list-style-type: none"> <li>• Contar con animales mamíferos</li> <li>• Disponer de un serpentario</li> <li>• Incluir un estanque en el zoológico</li> </ul>	Valor de negocio
User Stories	Yo como Dueño del Zoológico quiero una jaula con una jirafa para que el público pueda ver un mamífero exótico.	10
	Yo como dueño del Zoológico quiero un estanque con dos peces para mostrar al público animales acuáticos	5
	Yo como dueño del Zoológico quiero un estanque con una tortuga para mostrar al público un animal en extinción	6
	Yo como dueño del Zoológico quiero un estanque con una rana para que el público conozca un anfibio	5
	Yo como dueño del Zoológico quiero un serpentario con dos serpientes para presentar al público animales peligrosos	8
	Yo como dueño del Zoológico quiero una jaula con dos pájaros para tener aves	7

## PRÁCTICO 8 – SCRUM - Planificación de Release y de Sprint (Evaluable)

<b>Unidad:</b>	<b>Unidad Nro. 3: Gestión Ágil de Proyectos</b>
<b>Consigna:</b>	Tomando como base la definición de producto realizada para Taxi Mobile: Sistema web mobile para seguimiento de taxis y teniendo en cuenta el MVP definido, realizará con su equipo SCRUM la primera reunión de planificación de Sprint (Sprint Planning).
<b>Objetivo:</b>	Que el estudiante sea capaz de simular una de las ceremonias de SCRUM, Sprint Planning, cuyo propósito es la definición del Sprint Backlog. Que comprenda la importancia de la planificación en el contexto de la gestión ágil de proyectos.
<b>Propósito:</b>	Aplicar los conceptos de Gestión Ágil de Proyectos Vivenciar el ambiente de Scrum simulando la ceremonia de planificación de un sprint.
<b>Entradas:</b>	Conceptos teóricos de SCRUM. Bibliografía referenciada sobre el tema. Definición de Hecho (DoD) para el equipo. Caso práctico de Taxi Mobile: Sistema web mobile para seguimiento de taxis previamente desarrollado. Ejemplo de Minuta de Sprint Planning Ejemplo de Sprint Backlog
<b>Salida:</b>	Se evaluará que: <ul style="list-style-type: none"> <li>● Presente el plan de release para la liberación de la primera versión del producto (MVP).</li> <li>● Describa todas las consideraciones de contexto</li> <li>● Presente la minuta de planificación del Sprint</li> <li>● El Sprint Backlog desagregando las user stories en tareas estimadas en horas ideales</li> <li>● Descripción de todas las consideraciones de contexto que considere necesarias para la comprensión de los entregables mencionados en los ítems anteriores.</li> </ul>
<b>Instrucciones:</b>	<ul style="list-style-type: none"> <li>● En grupos trabajarán la consigna.</li> <li>● Tomarán el MVP definido para el producto de <b>Taxi Mobile: Sistema web mobile para seguimiento de taxis</b>, ya estimado</li> <li>● Definirán las condiciones de contexto necesarias para la planificación del release.</li> <li>● Definirán el Plan de Release indicando cuantos sprints serán necesarios y que user stories entregarán en cada uso y por consiguiente la duración del Plan de Release para la entrega de la versión del producto.</li> <li>● Definirá la minuta para el Sprint 1 y el Sprint Backlog</li> </ul>
<b>Observaciones:</b>	Debe referenciar la Bibliografía consultada.

### Ejemplo de Template de Minuta para Sprint Planning

**Sprint Nro.** <<x>>

**Duración del Sprint en días:** <<XX>>

**Objetivo del Sprint:** <<describa el objetivo del incremento de producto que se obtendrá al finalizar el sprint>>

#### Equipo Scrum:

<<Nombre del Miembro del Equipo>>

<<Nombre del Miembro del Equipo>>

<<Nombre del Miembro del Equipo>>

<<Nombre del Miembro del Equipo>>

<<Nombre del Miembro del Equipo>>

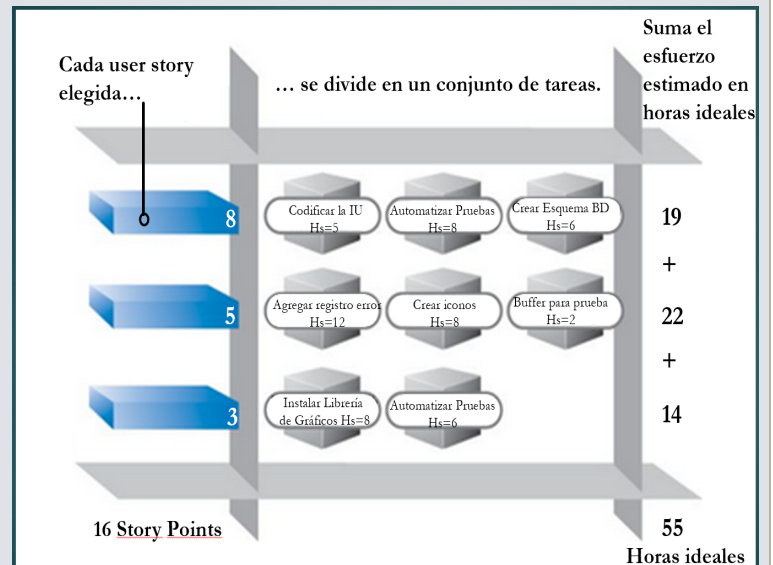
**Capacidad del Equipo en Horas Ideales:** << XXX>>

#### Definición de Hecho para el Equipo

##### Definición de Hecho (DONE)

- ☐ Diseño revisado
- ☐ Código Completo
- ☐ Código refactorizado
- ☐ Código con formato estándar
- ☐ Código Comentado
- ☐ Código en el repositorio
- ☐ Código Inspeccionado
- ☐ Documentación de Usuario actualizada
- ☐ Probado
  - ☐ Prueba de unidad hecha
  - ☐ Prueba de integración hecha
  - ☐ Prueba de Regresión hecha
  - ☐ Plataforma probada
  - ☐ Lenguaje probado
- ☐ Cero defectos conocidos
- ☐ Prueba de Aceptación realizada

#### Ejemplo de Sprint Backlog








## Taxi Mobile: Sistema web mobile para seguimiento de taxis

**Objetivo:** Desarrollar un producto de software para Smartphones, donde los pasajeros de taxis puedan solicitar el taxi más cercano y saber su ubicación y demora en todo momento.

Se describen a continuación los principales roles de usuario:

Rol de Usuario	Descripción
 <b>Pasajero</b>	<p>Persona que va a utilizar la aplicación con mucha frecuencia, para solicitar que le envíen un taxi al lugar donde está ubicado. Para él la simplicidad es importante.</p> <p>Está familiarizado con el uso de smartphones y puede instalar la aplicación.</p> <p>Su expectativa es que el taxi llegue a buscarlo lo más rápido posible.</p>
 <b>Taxista</b>	<p>Está familiarizado con el uso de smartphones y puede instalar la aplicación.</p> <p>Debe utilizar la aplicación mientras está trabajando, por eso necesita recibir solicitudes de viaje con notificaciones sonoras para enterarse de que tiene una solicitud de viaje. También debe poder utilizar la aplicación con manos libres.</p> <p>Necesita visualizar la posición del pasajero para poder llegar y necesita minimizar la cantidad de interacciones con el celular para informar que se dirige a buscar un pasajero o que ya lo encontró.</p>
 <b>Administrador de Central de Taxis</b>	<p>Debe poder trabajar con aplicaciones web y está familiarizado con el uso de Google maps.</p> <p>Necesita poder visualizar los taxis con su estado y poder visualizar la localización de los pasajeros para poder asistir a los Taxistas. La información debe estar disponible en tiempo real.</p>

### MVP (Producto Mínimo Viable)

#### Objetivo:

- Permitir que los clientes (pasajeros) puedan trasladarse al lugar que desean solicitando un taxi cercano.
- Permitir a los taxistas ver la ubicación del pasajero que solicita un viaje para llevarlo.
- Permitir a los taxistas manejar el estado del taxi.

#### Criterio:

Las user stories incluidas permiten validar la idea de negocio en el mercado, focalizando en la funcionalidad que pone en contacto a Pasajeros con Taxistas. Si bien los taxistas deben poder iniciar sesión para vincularlos con los datos de su vehículo, el análogo para el pasajero no es requisito para poder realizar un pedido. La funcionalidad relacionada con la gestión de taxis por parte de la central no aporta el valor significativo para la validación del mercado por lo que se dejará para futuras iteraciones.

#### Frases Verbales de las historias de usuario incluidas en el MVP:

##### Taxista

- Loguear taxista
- Ocupar taxi
- Liberar taxi
- Ver ubicación del pasajero

##### Pasajero

- Buscar taxis cercanos
- Pedir taxi
- Notificar a taxista solicitud de taxi

### Loguear pasajero

2

Como pasajero quiero loguearme para poder visualizar los taxis más cercanos.

Criterios de Aceptación:

- Datos para el login: nombre, apellido, teléfono celular (opcional).
- Pueden tomarse los datos de Facebook o desde el mismo celular.

#### Pruebas de Usuario

- ☐ Probar obtener datos desde Facebook con conexión a internet vía wifi. (pasa)
- ☐ Probar obtener datos desde Facebook con conexión a internet vía 3g. (pasa)
- ☐ Probar obtener datos desde Facebook sin conexión a internet. (falla)
- ☐ Probar solicitar los datos al pasajero. (pasa)

Justificación de estimación:

- **Complejidad:** user story simple ya que requiere un ingreso de datos mínimos y la interfaz con Facebook, para lo cual se utilizará una solución existente.
- **Esfuerzo:** Se requiere esfuerzo mínimo vinculado a utilizar la solución existente para la interfaz con Facebook.
- **Incertidumbre:** Mínimo, vinculado a la duda técnica respecto a cómo utilizar las APIs correspondientes para resolver la interfaz con Facebook.

### Loguear taxista

2

Como taxista quiero loguearme para poder visualizar los pedidos de taxis.

Criterios de Aceptación:

- Datos para el login: nombre, apellido, teléfono celular, dominio, número de móvil, central.

#### Pruebas de Usuario

- ☐ Probar ingresar los datos completos cuando el taxista se encuentra asociado a una central. (pasa)
- ☐ Probar ingresar los datos cuando el taxista no se encuentra asociado a una central. (falla)

Justificación de estimación:

- **Complejidad:** User story simple ya que requiere un ingreso de datos mínimos y la interfaz con Facebook, para lo cual se utilizará una solución existente.
- **Esfuerzo:** Se requiere esfuerzo mínimo vinculado a utilizar la solución existente para la interfaz con Facebook.
- **Incertidumbre:** Mínimo, vinculado a la duda técnica respecto a cómo utilizar las APIs correspondientes para resolver la interfaz con Facebook. Esta duda será resuelta en la user story de logueo de pasajero.

<b>Registrar central de taxis [user story canónica]</b>	<b>1</b>
Como administrador de central quiero dar de alta la central para poder tomar viajes con Taxi-mobile.	
Criterio de Aceptación: <ul style="list-style-type: none"> <li>El nombre de la central no debe repetirse.</li> </ul>	
<b>Pruebas de Usuario</b> <ul style="list-style-type: none"> <li><input type="checkbox"/> Probar registrar una central inexistente. (pasa)</li> <li><input type="checkbox"/> Probar registrar una central existente. (falla)</li> </ul>	

*Justificación de estimación:*

- **Complejidad:** User story simple ya que requiere un ingreso de datos mínimos y validaciones simples.
- **Esfuerzo:** Se requiere esfuerzo mínimo vinculado a validaciones simples de unicidad.
- **Incertidumbre:** Nula, no hay duda respecto a los datos y validaciones requeridos para registrar una central.

Se la selecciona como la **user story canónica** por ser de poca complejidad e incertidumbre nula, por lo que es factible de ser comparable con otras user stories.

<b>Pedir taxi</b>	<b>5</b>
Como pasajero quiero poder pedir un taxi seleccionando el más conveniente de un mapa para asegurarme de que el taxi está cerca.	
Criterios de Aceptación: <ul style="list-style-type: none"> <li>El celular debe contar con el sistema de geoposicionamiento online.</li> <li>El pasajero se debe visualizar gráficamente en un mapa.</li> </ul>	
<b>Pruebas de Usuario</b> <ul style="list-style-type: none"> <li><input type="checkbox"/> Probar seleccionar el taxi de entre un conjunto de taxis visualizados (pasa)</li> <li><input type="checkbox"/> Probar seleccionar un taxi cuando no existen taxis disponibles (realizando una selección en cualquier parte de la pantalla). (falla)</li> <li><input type="checkbox"/> Probar visualizar los taxis cuando no está activo el sistema de posicionamiento del celular(falla)</li> </ul>	

*Justificación de estimación:*

- **Complejidad:** User story compleja ya que requiere la vinculación con el sistema de geoposicionamiento online y la generación de un mapa.
- **Esfuerzo:** Se requiere esfuerzo alto para resolver la generación del mapa y la ubicación del taxi a partir de la respuesta del sistema de geoposicionamiento. Se requiere esfuerzo de investigación y pruebas.
- **Incertidumbre:** Media. Existen soluciones probadas para generar el mapa y establecer una interfaz con el sistema de geoposicionamiento, pero deben probarse y definirse la solución más adecuada.

<b>Ocupar taxi</b>	<b>2</b>
Como taxista quiero marcar que el taxi se encuentra ocupado para no recibir pedidos de servicio que no podrá atender.	
Criterios de Aceptación: <ul style="list-style-type: none"> <li>Para ocupar el taxi, este debe estar libre o con un pedido pendiente</li> </ul>	
<b>Pruebas de Usuario</b> <ul style="list-style-type: none"> <li><input type="checkbox"/> Probar ocupar el taxi cuando existe un viaje pedido pendiente. (pasa)</li> <li><input type="checkbox"/> Probar ocupar el taxi cuando está libre. (pasa)</li> <li><input type="checkbox"/> Probar ocupar el taxi cuando ya está ocupado. (falla)</li> </ul>	

*Justificación de estimación:*

- **Complejidad:** User story simple ya que requiere un único dato y cambio de estado de una única entidad (el taxi).
- **Esfuerzo:** Poco esfuerzo para el desarrollo y medio para realización de pruebas de taxis en distintos estados.
- **Incertidumbre:** Nula, no hay duda respecto a cómo realizar el cambio de estado.

<b>Liberar taxi</b>	<b>2</b>
Como taxista quiero liberar el taxi cuando estaba ocupado para que esté disponible para un próximo pedido de viaje.	
Criterios de Aceptación: <ul style="list-style-type: none"> <li>● Para liberar el taxi, este debe estar en curso</li> </ul>	
<b>Pruebas de Usuario</b> <ul style="list-style-type: none"> <li><input type="checkbox"/> Probar liberar el taxi cuando está con un viaje en curso. (pasa)</li> <li><input type="checkbox"/> Probar liberar el taxi cuando está libre. (falla)</li> </ul>	

*Justificación de estimación:*

- **Complejidad:** User story simple ya que requiere un único dato y cambio de estado de una única entidad (el taxi).
- **Esfuerzo:** Poco esfuerzo para el desarrollo y medio para realización de pruebas de taxis en distintos estados.
- **Incertidumbre:** Nula, no hay duda respecto a cómo realizar el cambio de estado.

<b>Marcar taxi como fuera de servicio</b>	<b>2</b>
Como taxista quiero marcar que el taxi se encuentra fuera de servicio para no recibir pedidos de servicio que no podrá atender.	
Criterios de Aceptación: <ul style="list-style-type: none"> <li>● El taxi queda en estado fuera de servicio.</li> <li>● El taxi no puede pasar a fuera de servicio si está ocupado o con pedido de viaje pendiente.</li> </ul>	
<b>Pruebas de Usuario</b> <ul style="list-style-type: none"> <li><input type="checkbox"/> Probar marcar el taxi como fuera de servicio cuando el taxi está libre. (pasa)</li> <li><input type="checkbox"/> Probar marcar el taxi como fuera de servicio cuando existe un viaje pedido pendiente. (falla)</li> <li><input type="checkbox"/> Probar marcar el taxi como fuera de servicio cuando ya está ocupado. (falla)</li> </ul>	

*Justificación de estimación:*

- **Complejidad:** user story simple ya que requiere un único dato y cambio de estado de una única entidad (el taxi).
- **Esfuerzo:** Poco esfuerzo para el desarrollo y medio para realización de pruebas de taxis en distintos estados.
- **Incertidumbre:** Nula, no hay duda respecto a cómo realizar el cambio de estado.

<b>Notificar a taxista y a central pedido de taxi</b>	<b>3</b>
Como pasajero quiero enviar una notificación al momento en que solicito un viaje para que el taxista me busque y la central esté enterada del pedido.	
Criterios de Aceptación: <ul style="list-style-type: none"> <li>● La notificación debe llegar en forma simultánea al taxista y a la central.</li> </ul>	
<b>Pruebas de Usuario</b> <ul style="list-style-type: none"> <li><input type="checkbox"/> Probar notificar al taxista y la central cuando ambos tienen conexión de internet. (pasa)</li> <li><input type="checkbox"/> Probar notificar al taxista y la central cuando ninguno tiene conexión de internet. (falla)</li> </ul>	

*Justificación de estimación:*

- **Complejidad:** user story de complejidad media ya que requiere resolver la forma en que se enviará un mensaje push.
- **Esfuerzo:** Esfuerzo de investigación y desarrollo para resolver el envío de mensaje push, y de preparación de ambiente y pruebas.
- **Incertidumbre:** Media, vinculada a la duda técnica respecto a cómo enviar el mensaje push, a partir de soluciones existentes.

### Buscar taxis cercanos

3

Como pasajero quiero ver cuáles son los taxis más cercanos a mi ubicación para pedir el taxi que más me convenga.

Criterios de Aceptación (CA):

- Se muestra la ubicación del taxi y el tiempo estimado para llegar al lugar en el que se encuentra el pasajero.
- Se mostrarán hasta 5 taxis cercanos.

### Pruebas de Usuario

- ☐ Probar visualizar taxis cuando existen al menos 5 taxis libres. (pasa)
- ☐ Probar visualizar taxis cuando no existe ningún taxi libre (falla)

*Justificación de estimación:*

- **Complejidad:** user story de complejidad media ya que requiere resolver la forma de obtener datos de ubicación y tiempo a partir del sistema de geoposicionamiento.
- **Esfuerzo:** Esfuerzo medio de investigación y desarrollo para resolver la obtención de datos de geoposicionamiento y tiempos, así como para la preparación del ambiente simulado para realizar pruebas. Se requiere gran número de combinaciones de pruebas.
- **Incertidumbre:** Media, vinculada a resolver la forma de obtener los datos de geoposicionamiento, y la simulación para realizar las pruebas.

### Ver ubicación del pasajero

5

Como taxista quiero ver la ubicación del pasajero que ha solicitado un viaje para poder ir a buscarlo

Criterios de Aceptación (CA):

- El celular debe contar con el sistema de posicionamiento online.
- El pasajero se debe visualizar gráficamente en un mapa.

### Pruebas de Usuario

- ☐ Probar visualizar ubicación del pasajero para un pedido de viaje pendiente. (pasa)
- ☐ Probar visualizar ubicación del pasajero para un pedido de viaje cuando no se ha solicitado ninguno. (falla)
- ☐ Probar visualizar ubicación del pasajero para un pedido de viaje cuando no está activo el sistema de ubicación del celular. (falla)

*Justificación de estimación:*

- **Complejidad:** User story de complejidad alta ya que requiere resolver la forma de obtener datos de ubicación a partir del sistema de geoposicionamiento y la visualización en un mapa.
- **Esfuerzo:** Esfuerzo medio de investigación y desarrollo para resolver la obtención de datos de geoposicionamiento, así como para la preparación del ambiente simulado para realizar pruebas. Se requiere gran número de combinaciones de pruebas.
- **Incertidumbre:** Media, vinculada a resolver la forma de obtener los datos de geoposicionamiento, la simulación para realizar las pruebas, así como la generación del mapa.

### Ver mapa de taxis

5

Como administrador de la central quiero ver la ubicación de todos los taxis de la central y si tienen viajes en curso para saber la disponibilidad actual

Criterios de Aceptación:

- Se muestran taxis en todos los estados.
- Los taxis que se mostrarán deben tener habilitado el sistema de ubicación.

### Pruebas de Usuario

- ☐ Probar visualizar un conjunto de taxis libres, fuera de servicio y ocupados (pasa)
- ☐ Probar visualizar el mapa cuando ningún taxi de la central está conectado al sistema de ubicación (falla)

*Justificación de estimación:*

- **Complejidad:** user story de complejidad alta ya que requiere resolver la forma de obtener datos de ubicación a partir del sistema de geoposicionamiento y la generación del mapa con todos los taxis.
- **Esfuerzo:** Esfuerzo medio de investigación y desarrollo para resolver la obtención de datos de geoposicionamiento, así como para la preparación del ambiente simulado para realizar pruebas. Se requiere gran número de combinaciones de pruebas.
- **Incertidumbre:** Media, vinculada a resolver la forma de obtener los datos de geoposicionamiento, la simulación para realizar las pruebas y la generación del mapa.

## PRÁCTICO 9 - 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.			
<b>Instrucciones:</b>	<b>Este práctico no se entrega y por lo tanto no tiene nota. El tema se evalúa en el parcial.</b>			
	Según el método de partición de equivalencia, defina las clases existentes utilizando el siguiente cuadro para la Historia de Usuario dada.			
	Condición externa	Clases de equivalencia válidas		Clases de equivalencia inválidas
	<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>.</p> <p>Utilice el template <code>Template_Casos_De_Prueba</code>.</p>			

### Taxi Mobile: Sistema web mobile para seguimiento de taxis

**Objetivo:** Desarrollar un sistema para celulares para que los pasajeros de taxis puedan solicitar el taxi más cercano y saber su ubicación y demora en todo momento.

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

**Product Owner (PO):** ¿La idea es que el pasajero pueda llamar a una central sabiendo dónde está el taxi que va a pedir? ¿Cómo pide un taxi el pasajero?

**Experto en el Dominio (ED):** El pasajero debe ingresar a la aplicación, la cual activará el sistema de posicionamiento. El sistema detecta dónde está el pasajero y le muestra los 5 taxis más próximos a su ubicación, visualizados en un mapa, e informa ubicación, distancia y tiempo estimado. El pasajero selecciona el taxi deseado y esta acción envía una notificación a la central de taxis y al taxista a su celular. De esta forma ambos identifican el pedido de un móvil, y el taxista puede saber a dónde está el pasajero

**PO:** ¿Y cómo sabe el taxista quién es el pasajero?

**ED:** Cuando se instala la aplicación en el celular se pedirán los datos mínimos para registrarlo como usuario. Estos datos se pueden obtener de Facebook y el número de celular del mismo aparato donde se instala la aplicación.

**PO:** ¿Y los taxistas cómo trabajan con esta aplicación?

**ED:** Como taxista también es necesario instalar una aplicación en su celular. El taxista debe estar registrado en una central que haya contratado el servicio de taxi-mobile. Cuando instale la aplicación se solicitarán sus datos identificatorios como taxista y

de su móvil (nro. de chapa), y la selección de la central a la que pertenece. La aplicación también utiliza el sistema de posicionamiento que tiene el teléfono, para poder informar en qué lugar se encuentra el taxista.

**PO:** ¿Si estoy entendiendo bien, tanto el pasajero como el taxista deben contar con un Smart Phone con sistema de posicionamiento para que la aplicación funcione?

**ED:** Así es, sino la aplicación no puede ubicar al taxi ni al pasajero.

**PO:** ¿Y una vez que el pasajero sube al taxi, ¿cómo se indica que ya está en viaje?

**ED:** El taxista debe asentar esto en la aplicación con la opción correspondiente. De esta forma la central está al tanto de la situación. Igualmente, cuando el pasajero descienda el taxista debe indicarlo, para que el taxi quede libre en la aplicación, y se muestre a un próximo pasajero.

**PO:** ¿La aplicación sólo muestra taxis libres?

**ED:** A los pasajeros sí, en la central se pueden ver con distintos colores los taxis ocupados, los libres, los solicitados y los que están fuera de servicio. Pasando el mouse sobre el ícono del auto deben poder visualizarse los datos del viaje si el taxi está ocupado (hora de inicio y costo) y los datos del pasajero (nombre, apellido y número de teléfono celular). Si el taxi está solicitado, sólo se visualizarán los datos del pasajero.

**PO:** ¿Y la aplicación para la central también es mobile?

**ED:** No, debería ser una aplicación web. Y podrán utilizarse filtros por Barrio, por Estado y por chapa del taxi para visualizarlos en el mapa.

**PO:** ¿La idea es que la aplicación del pasajero y del taxista sean con touch screen?

**ED:** Los elementos centrales de la aplicación son el mapa y el taxi. Siempre a través de la selección del taxi se indicarán las acciones que quieren realizarse, como llamarlo, indicar que está ocupado, que está libre, tanto por parte del pasajero como del taxista.

#### Ver mapa de taxis

5

Como Administrador de la Central quiero ver la ubicación de todos los taxis de la central y si tienen viajes en curso para saber la disponibilidad actual

#### Criterios de Aceptación:

- Se muestran taxis en todos los estados (Verde: Libre, Amarillo: Solicitado, Rojo: Ocupado, Negro: Fuera de Servicio).
- Los taxis que se mostrarán deben tener habilitado el sistema de ubicación.
- Pasando el mouse sobre el ícono del auto deben poder visualizarse los datos del viaje si el taxi está ocupado (hora de inicio y costo) y los datos del pasajero (nombre, apellido y número de teléfono celular).
- Si el taxi está solicitado, sólo se visualizarán los datos del pasajero.
- Podrán utilizarse filtros por Barrio, por Estado y por chapa del taxi

#### Pruebas de Usuario

- Probar visualizar un conjunto de taxis libres (pasa)
- Probar visualizar un conjunto de taxis solicitados (pasa)
- Probar visualizar un conjunto de taxis fuera de servicio (pasa)
- Probar visualizar un conjunto de taxis ocupados (pasa)
- Probar visualizar un conjunto de taxis libres, solicitados, fuera de servicio y ocupados (pasa)
- Probar visualizar un taxi conectado al sistema por número de chapa (pasa)
- Probar visualizar en el mapa un taxi no conectado al sistema de ubicación (falla)
- Probar visualizar un conjunto de taxis en un Barrio inexistente (falla)



## PRÁCTICO 10 - 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. Planilla de Excel con Plantilla de Casos de Prueba y Reporte de Defectos Delivereat_Template_Caso_De_Prueba_Defectos.xlsx																		
<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 User Story de las siguientes que <b>no haya implementado</b> en el Trabajo Práctico 6 “Requerimientos Ágiles – Implementación de User Stories”, sobre TANGO APP:</p> <ul style="list-style-type: none"> <li>○ <i>Publicar pedido de envío</i> (<b>grupos impares</b>)</li> <li>○ <i>Aceptar cotización</i> (<b>grupos pares</b>)</li> </ul> <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 las condiciones externas en las que sea posible).          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 11 - TESTING – Ejecución de Casos de Prueba (Evaluable)

<b>Unidad:</b>	<b>Unidad Nro. 4: Aseguramiento de Calidad de Proceso y de Producto</b>
<b>Consigna:</b>	Ejecute los casos de pruebas de caja negra diseñados
<b>Objetivo:</b>	Que el estudiante pueda aplicar los distintos aspectos del testing, desde la planificación hasta ejecución de las pruebas.
<b>Propósito:</b>	Comprender como se realiza la ejecución de los casos de prueba diseñados
<b>Entradas:</b>	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>
<b>Salida:</b>	<ul style="list-style-type: none"> <li>Planilla de Excel con Clases de Equivalencias identificadas y Casos de Prueba diseñados en el Trabajo <b>Práctico 11 “Testing – Métodos de Caja Negra”</b></li> <li>Ejecución y Reporte de Defectos</li> </ul>
<b>Instrucciones:</b>	<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 12 - KANBAN – Ejercicio de aplicación (Evaluable)

<b>Unidad:</b>	<b>Unidad Nro. 4: Aseguramiento de Calidad de Proceso y de Producto</b>
<b>Consigna:</b>	A través de un juego y de las instrucciones comunicadas por los docentes, vivenciar los principios kanban
<b>Objetivo:</b>	<p>Experimentar cómo un sistema Kanban emerge de un proceso existente, como ocurre en el mundo real. Vivenciar un sistema Kanban completo, en lugar de centrarse sólo en el tablero Kanban y las mecánicas relacionadas.</p> <p>Comprender que los tableros dependen del contexto: para cualquier proceso dado hay muchos diseños de tableros diferentes que son adecuados y útiles, pero el tablero perfecto no necesariamente existe.</p> <p>Comprender los efectos de limitar su trabajo en progreso.</p> <p>Experimentar la autoorganización y la adaptación.</p>
<b>Propósito:</b>	Familiarizarse la filosofía lean, internalizando los principios propuestos por Kanban, utilizando una dinámica grupal, en forma de juego.
<b>Entradas:</b>	<ul style="list-style-type: none"> <li>● Conceptos teóricos sobre el tema, desarrollados en clase. Bibliografía referenciada sobre el tema.</li> <li>● Instrucciones del juego.</li> <li>● Materiales necesarios que deben ser llevado por los estudiantes.</li> </ul>
<b>Salida:</b>	Mostrar los resultados de la dinámica en cada iteración.
<b>Instrucciones:</b>	El docente que dirige la dinámica será quién indique la consigna y el alcance de la dinámica a realizar.
<b>Lista de materiales</b>	<ul style="list-style-type: none"> <li>● Post-Its en tres colores: amarillo (piña), rosa (jamón) y verde (rúcula)</li> <li>● Papel de impresora para cortar fondos de pizza (A4 / Carta u otros tamaños)</li> <li>● Marcadores rojos como salsa de tomate.</li> <li>● Pegamento o cinta transparente (para hacer que el Post-Its se pegue mejor)</li> <li>● Cinta de enmascarar (también conocida como cinta de pintor)</li> <li>● Tijeras (una pequeña + una grande por equipo)</li> <li>● Cronómetro</li> <li>● Tarjetas de pedido - un juego por equipo</li> <li>● Placa de horno - una por equipo</li> <li>● Hojas de reglas</li> <li>● • Hojas de información general de Kanban</li> </ul>

### Los principios de Kanban

- **Visualizar el flujo de trabajo:** Normalmente está escondido de alguna manera, se trata de hacerlo explícito y visible por todos.
- **Limitar el WIP:** Limitamos el trabajo el curso. Se limita para evitar cuellos de botella, y para garantizar que solo entran elementos nuevos cuando hay capacidad de completarlos (no se produce por encima del cuello de botella).
- **Gestionar y medir el flujo:** Medir para ver si se mejora o no, y decidir en consecuencia. Podría ser número de tareas completas en el Sprint, o número de Story Points completados, el Lead Time, el tiempo de ciclo, etc...
- **Implementar ciclos de feedback:** En el mundo ágil se corresponden con las retrospectivas. La idea de Kanban va más allá y define que se debe recibir el feedback en todos los niveles (del equipo, del cliente, del PO, stakeholders...)
- **Explicitar políticas y procedimientos:** Por ejemplo los "Definition of Done", "Definition of Ready", valores y principios
- **Mejorar continuamente mediante la colaboración:** Usando modelos y experimentando científicamente. Muchas veces es lo que falla de las retrospectivas, se decide algo y después no se sigue ni se mide la consecuencia de aplicarlo. Kanban propone algo más científico, definir una hipótesis, implementar un experimento y medir las consecuencias. Si funciona se sigue y si no comienzas de nuevo el ciclo

## PRÁCTICO 13 - SCRUM - Dinámica de Retrospectiva (Evaluable)

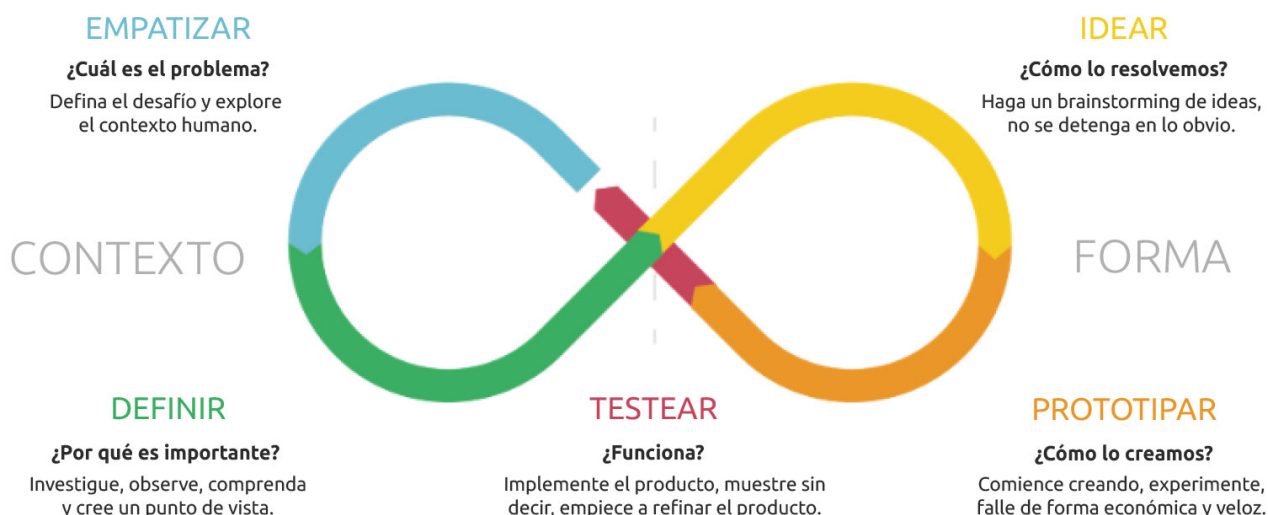
<b>Unidad:</b>	<b>Unidad Nro. 2: Gestión Lean-Ágil de Productos de Software</b>
<b>Consigna:</b>	En función de los conceptos desarrollados en la materia sobre las ceremonias de Scrum y en particular sobre Retrospectivas, aplicar la técnica al trabajo grupal realizado por los estudiantes durante el cursado de la materia.
<b>Objetivo:</b>	Utilizar la técnica de retrospectiva propuesta por SCRUM para realizar una autoevaluación del desempeño de los miembros del grupo de trabajo en el contexto del cursado de esta materia.
<b>Propósito:</b>	Familiarizarse con la gestión ágil de proyectos, conceptos y distintas técnicas de retrospectivas para obtener mejores resultados en los equipos de trabajo SCRUM.
<b>Entradas:</b>	<ul style="list-style-type: none"> <li>● Conceptos teóricos sobre el tema desarrollados en clase e investigación de diferentes técnicas de retrospectivas de Scrum.</li> <li>● Bibliografía referenciada sobre el tema.</li> <li>● Instrucciones de la dinámica.</li> <li>● Materiales de trabajo (post-its, afiches, fibrones)</li> </ul>
<b>Salida:</b>	<ul style="list-style-type: none"> <li>● Presentación con el resultado de la ejecución de la ceremonia retrospectiva y de la técnica empleada.</li> <li>● Conclusiones obtenidas al finalizar la retrospectiva realizada por cada grupo y las conclusiones generales del grupo clase.</li> </ul>
<b>Instrucciones:</b>	<ul style="list-style-type: none"> <li>● Se trabajará con el grupo de clase.</li> <li>● Investigar y elegir una técnica de retrospectiva.</li> <li>● Realizar la retrospectiva respetando la estructura propuesta: <b>apertura (“icebreaker”), retrospectiva (aplicando técnica elegida), definición de ítems de acción y responsables, conclusiones y cierre.</b></li> <li>● Presentar la técnica empleada y las conclusiones de cada grupo.</li> <li>● Debatir las conclusiones generales de la ejecución de la ceremonia de retrospectiva con el grupo clase</li> </ul>

## PRÁCTICO 14 - DESIGN THINKING – Publicidad de Instagram (Evaluable)

<b>Unidad:</b>	<b>Unidad Nro. 2: Gestión Lean-Ágil de Productos de Software</b>
<b>Consigna:</b>	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.
<b>Objetivo:</b>	Aplicar algunos conceptos de gestión ágil y design thinking en el desarrollo de productos de software y el concepto de MVP.
<b>Propósito:</b>	Familiarizarse con la gestión ágil de proyectos y algunos conceptos de Design Thinking.
<b>Entradas:</b>	<ul style="list-style-type: none"> <li>• Conceptos teóricos sobre el tema desarrollados en clase.</li> <li>• Bibliografía referenciada sobre el tema.</li> <li>• Instrucciones de la dinámica.</li> <li>• Materiales de trabajo (a coordinar)</li> </ul>
<b>Salida:</b>	<ul style="list-style-type: none"> <li>• Presentación con el resultado de la ejecución de las Etapas del proceso propuesto.</li> <li>• 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.</li> </ul>

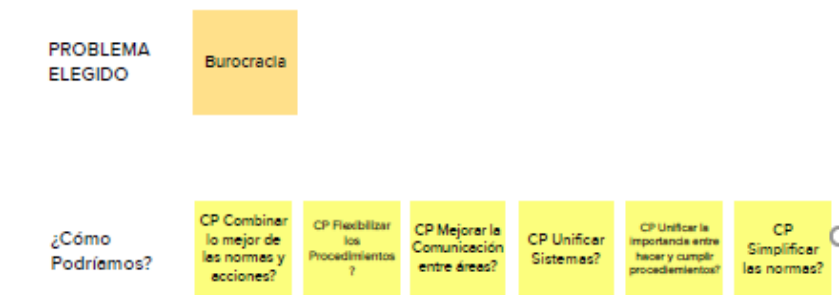
# DESIGN THINKING

## UN FRAMEWORK PARA INNOVAR



### Instrucciones

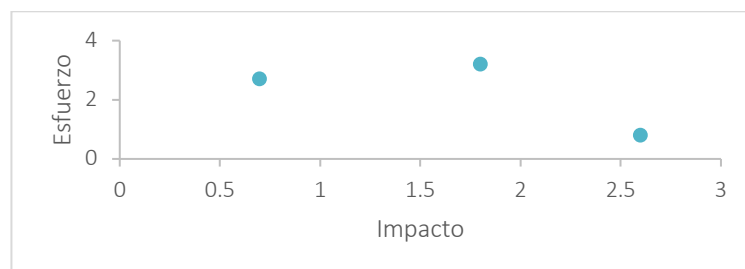
1. Se trabajará sobre un problema que pueda resolverse con un producto de software. Puede ser elegido por cada grupo o planteado por el docente. Por ejemplo, podría plantearse como problema la burocracia en el contexto de la Universidad, el sistema de transporte en Córdoba, los pedidos del Delivery llegan siempre fríos, etc.
2. A partir del problema el grupo planteará los desafíos que el mismo supone, a partir de la consigna “Como Podríamos...” Eso nos permite convertir a los problemas en Desafíos positivos y accionables. Por ejemplo:



3. Y para cada desafío planteamos posibles soluciones

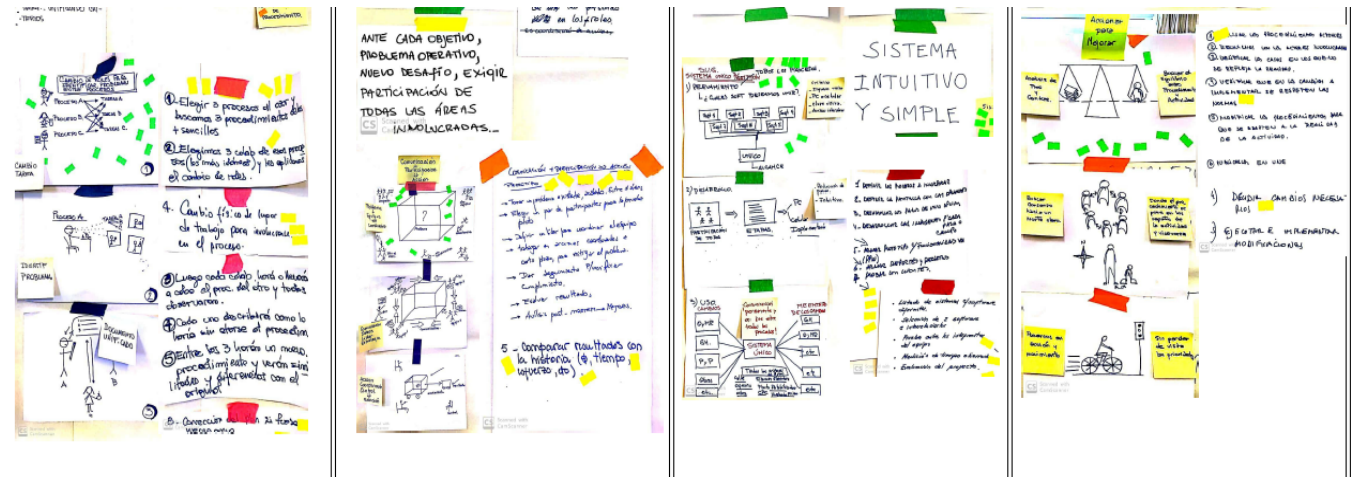


4. Se eligen la solución sobre la cual se trabajará, ponderándolas en una matriz de Esfuerzo – Impacto (se elige la de la combinación de mayor impacto y menos esfuerzo)



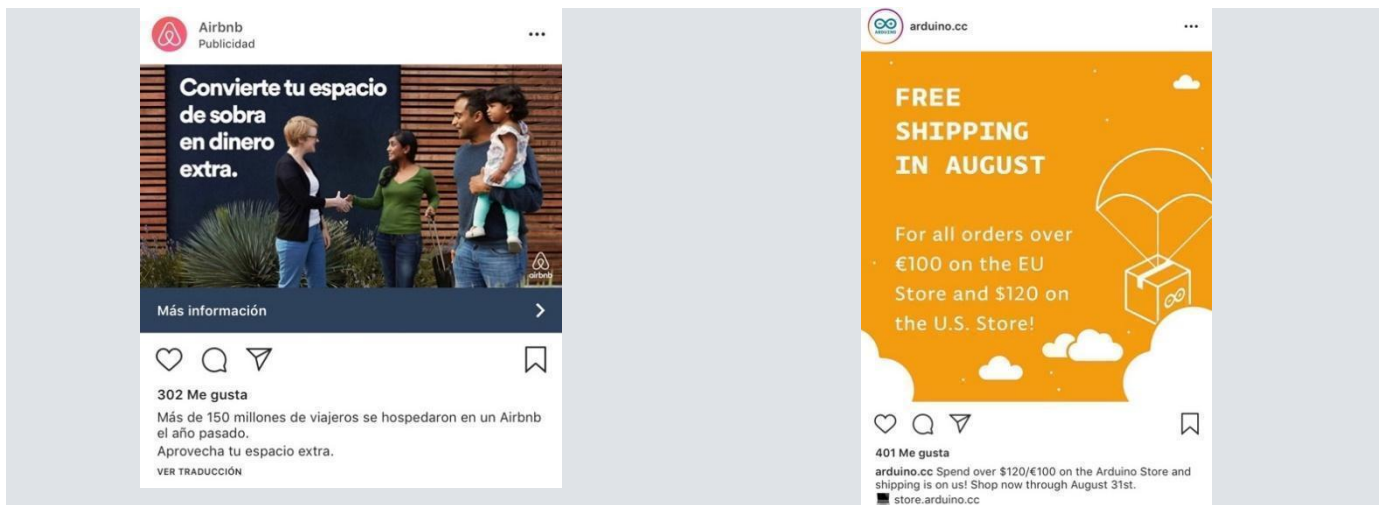
5. Cada alumno del grupo grafica como trabajar sobre la idea elegida en 3 etapas, para modelar el concepto

Ejemplos gráficos:



6. El equipo vota (cada integrante tiene 2 votos) por la representación que más le gustó. La representación más votada es elegida la representación ganadora.
7. El equipo define los requerimientos que debería incluir el Mínimo Producto Viable. Luego arma un prototipo de una publicidad para Instagram de la solución elegida, para el Mínimo Producto Viable.

Ejemplos:



8. El grupo presenta su publicidad a todo el curso.