

# ENUNCIADOS DE TRABAJOS PRÁCTICOS EVALUABLES

Ingeniería de Software

## ÍNDICE

PRÁCTICO 1 -	MÉTRICAS TRADICIONALES - EMPRESA GASTRONÓMICA.....	3
PRÁCTICO 2 -	REQUERIMIENTOS ÁGILES - USER STORIES - Bomba de insulina.....	5
PRÁCTICO 3 -	SCRUM - Instrumentos musicales.....	6
PRÁCTICO 4 -	GESTIÓN DE CONFIGURACIÓN (video).....	8
PRÁCTICO 5 -	GESTIÓN DE CONFIGURACIÓN - Taxi mobile.....	9
PRÁCTICO 6 -	TESTING - MÉTODOS DE CAJA NEGRA - Festival de folklore.....	12
PRÁCTICO 7 -	TESTING - MÉTODOS DE CAJA BLANCA - Festival de folklore.....	14
PRÁCTICO 8 -	TESTING - EJECUCIÓN DE CASOS DE PRUEBA - Festival de folklore.....	16
PRÁCTICO 9 -	LA CAJA DE PRODUCTO.....	16
PRÁCTICO 10 -	PECHA KUCHA.....	17

## PRÁCTICO 1 - MÉTRICAS TRADICIONALES - EMPRESA GASTRONÓMICA

<b>Unidad:</b>	Unidad Nro. 2: Proyectos de Desarrollo de Software basado en procesos definidos
<b>Consigna:</b>	Definir un conjunto de métricas de software a aplicar durante la ejecución de proyectos de desarrollo de software.
<b>Objetivo:</b>	Que el alumno pueda definir y analizar métricas que satisfagan la necesidad de información de un proyecto de software, teniendo en cuenta el enfoque tradicional visto en la clase teórica.
<b>Propósito:</b>	Repasar los conceptos vinculados a la definición de Métricas de Software. Definir y especificar métricas de software.
<b>Entradas:</b>	Conceptos teóricos sobre el tema, desarrollados en clase. Plantilla de definición de métricas.
<b>Salida:</b>	Cada grupo presentará las métricas solicitadas según el enfoque de desarrollo tradicional.
<b>Instrucciones:</b>	<ol style="list-style-type: none"><li>1. Dividirse en grupos</li><li>2. Realizar la identificación de métricas tradicionales, incluyendo al menos una de cada dominio y tipo.</li><li>3. Describir las métricas respetando el template y analizar las métricas provistas.</li><li>4. Seleccionar un representante del grupo para presentar el trabajo realizado.</li></ol>

### Presentación del Caso de Estudio

Una importante compañía de rubro gastronómico de la ciudad de Córdoba ha decidido contratar los servicios de una empresa consultora de sistemas, a fin de contar con un producto de software a medida, para la administración de su negocio.

### Definición del Producto:

#### Objetivo:

Desarrollar un sistema informático que brinde soporte a cada uno de los procesos involucrados en la elaboración y comercialización de productos alimenticios; tales como gestión de compras de insumos, gestión de producción, gestión de stock, gestión comercial, y gestión de recursos humanos.

#### Requerimientos Funcionales (Alcances):

- Administración de productos y promociones ofrecidos.
- Administración de producción.
- Administración de pedidos y su posterior cobro.
- Administración de entrega de pedidos.
- Administración de clientes.
- Administración de insumos.
- Administración de proveedores de insumos.
- Administración de empleados.
- Emisión de estadísticas e informes vinculados a: stock, demanda de productos, insumos, movimientos financieros.

#### Requerimientos No Funcionales:

Nro.	Nombre	Descripción
1	Navegador WEB	El producto debe ser compatible con Internet Explorer 7.0. o superior.
2	Usuarios concurrentes	El producto debe dar soporte hasta 15 usuarios concurrentes sin afectar la performance de la aplicación.

Nro.	Nombre	Descripción
3	Motor de Base de Datos	El motor de base de datos debe ser open source y no tener costo de licenciamiento. El motor de base de datos deberá soportar como mínimo la creación de storeprocedures, triggers y funciones, subconsultas y las consultas básicas (select, update, delete, create, drop) y de resumen (sum, avg, may, min).
4	Configuración de Monitores	Debe asumirse una configuración de monitores de 1024 x 768.
5	Sistemas de Ayuda	Debe tener un sistema de ayuda integrado y sensible al contexto.
6	Requerimientos de Hardware para usuarios	El producto no debe demandar instalaciones de ningún tipo en las máquinas de los usuarios. Debe ser suficiente contar con un navegador web.
7	Lenguaje de Programación	El lenguaje de programación debe permitir desarrollos de productos utilizables en Internet.
8	Interfaz para los reportes y estadísticas	Para todos los reportes y estadísticas se ha requerido que los mismos se muestren tanto en formato de cuadros como en formato gráfico.
9	Tiempo de Respuesta	El tiempo de respuesta para transacciones debe ser de hasta 5 segundos y para consultas debe ser de hasta 7 segundos en el 95 % de los casos.

### Consideraciones para el Proyecto

El alcance del proyecto implica la especificación de los requerimientos, el desarrollo del producto, las pruebas, el despliegue del mismo y la capacitación a los empleados de la empresa cliente. Es fundamental para el éxito del proyecto la aceptación del producto por parte de todos los usuarios del sistema. Dado que muchos usuarios finales no están familiarizados con sistemas de información de este tipo, es indispensable que se trate de un sistema fácil de usar, intuitivo y con una ayuda integrada y sensible al contexto.

El plazo expuesto para la finalización del proyecto es de 30 semanas y se requiere que además se contemple un período de "garantía" de 15 días hábiles. En el contrato se encuentran estipuladas multas que la empresa contratada deberá pagar por incumplimiento de plazos. Por tal motivo, el cumplimiento de los plazos establecidos es de fundamental importancia. Una apropiada performance del equipo de proyecto es fundamental para el logro de este objetivo, así como también un correcto seguimiento sobre los defectos encontrados a lo largo del proyecto.

En el contrato se establece que la empresa que desarrollará el sistema no podrá subcontratar actividades del proyecto, lo que es un condicionante con respecto a la disponibilidad necesaria del personal de la empresa y de la evaluación de necesitar contratar y capacitar personal nuevo y los tiempos que esto implique.

### Plantilla de Definición de Métricas a utilizar

<b>Nombre de la Métrica</b> [Que sea claro y significativo. Debería ser elegido para evitar ambigüedad, confusión.]
<b>Razón para usarla (cómo ayuda a crear valor) (¿Por qué?)</b> [Qué motivó la creación de la métrica, cómo se la ha utilizado históricamente]
<b>Nivel de Audiencia y de uso (¿Para quién?)</b> [Indique el uso esperado en los niveles de la organización. Distinguir quién toma la métrica de quien lo mide].
<b>Base de Medición (¿Qué?)</b> [Defina claramente qué es lo que se está midiendo, incluyendo unidades de medición y límites.]
<b>Fórmula de cálculo (¿Cómo?)</b> [Identificar todo lo necesario para asegurar comprensión clara de los datos que se están utilizando y representando. Indicar la fórmula para el cálculo de la métrica]
<b>Dominio</b> [Identificar el dominio de la métrica (Proceso / Proyecto / Producto)]
<b>Frecuencia (¿Cada cuánto?)</b>

**Universidad Tecnológica Nacional - Facultad Regional Córdoba**

**Cátedra de Ingeniería de Software - 2016**

Prof. Ing. Judith Meles / Ing. Daniel Battistelli - Cursos: 4k1-4k2-4k3-4K4

**[Especificar el momento en el que se obtendrá la métrica]**

**Directa/Indirecta**

**[Indicar si se trata de una métrica directa o indirecta]**

**Se pide:**

Defina cinco métricas tradicionales que considere adecuadas para obtener visibilidad durante la ejecución del proyecto de desarrollo de software, teniendo en cuenta las consideraciones del mismo, y utilizando la plantilla propuesta.

## PRÁCTICO 2 - REQUERIMIENTOS ÁGILES - USER STORIES - BOMBA DE INSULINA

<b>Unidad:</b>	Unidad Nro. 3: Gestión Ágil de Proyectos
<b>Consigna :</b>	<p>Aplicar los conceptos teóricos desarrollados en clase sobre user stories. Para ello:</p> <ul style="list-style-type: none"><li>▪ Los docentes representarán a expertos del dominio que expresarán sus necesidades vinculadas al desarrollar un software para el manejo de una bomba de insulina.</li><li>▪ Los alumnos 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.</li></ul> <p>Cada grupo presentará el resultado obtenido al final de la clase al resto del curso. Se entregará una copia al docente la clase siguiente.</p>

### Sistema de bomba de insulina

**Objetivo:** Desarrollar un software para el manejo de una bomba de insulina.

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

**Product Owner (PO):** ¿Qué tamaño tiene la bomba de insulina que vamos a fabricar? ¿Cómo se colocará la bomba de insulina?

**Experto en el Dominio (ED):** Estamos trabajando en una bomba del tamaño de una billetera. Algo como lo que ves en esta fotografía:



La bomba de insulina (**BI**) deberá colocarse en la cintura para que inyecte insulina cuando corresponda.

**PO:** ¿Cómo debería funcionar para administrar la insulina necesaria a un diabético?

**ED:** Se debe poder configurar la insulina basal, es decir la insulina que se debe inyectar constantemente. Para esto se definen períodos de tiempo, cada cuánto se debe inyectar, y la cantidad de insulina que se debe inyectar. Por ej., para los días hábiles de 8 a 12 se debe inyectar 0.625 mlgr por decilitros de insulina cada 1 hora. Esto generalmente lo define el médico de cabecera del paciente que usará la BI. Por otro lado, se debe definir la cantidad de insulina por carbohidratos para el bolo. El bolo es la insulina que se inyecta cuando se está por realizar alguna acción que requiere de una mayor cantidad de insulina, por ej., al momento de comer. El bolo debe ser inyectado únicamente cuando el paciente lo indique.

**PO:** ¿Y qué información deberíamos requerir para definir la cantidad de insulina del bolo?

**ED:** La cantidad de carbohidratos a ingerir y la medida actual de insulina.

**PO:** ¿Alguna otra funcionalidad que deberíamos tener en cuenta? ¿Alarmas? ¿Algún cálculo extra?

**ED:** Debería poder calcularse la vida de la insulina inyectada, para conocer en este momento cuánto falta para inyectar insulina nueva. Además deberían sonar alarmas cuando pase un período de tiempo configurado en que la bomba no está colocada en el paciente, y cuando se pase un momento de tiempo en que el paciente debería haberse inyectado una cantidad de insulina de bolo. Por último sería útil contar con una funcionalidad para cuando el paciente indique se van a ingerir grasas o proteínas, para que la inyección de insulina se realice de forma más racionalizada.

**PO:** Respecto a la batería, ¿Deberíamos informar la cantidad de batería que le queda a la BI verdad?

**ED:** Sí, deberíamos mostrar constantemente una pila con tres niveles de carga, y al llegar al último tercio de carga la pila debería comenzar a titilar y emitir una alerta para que el paciente ponga a cargar la BI.

## PRÁCTICO 3 - SCRUM – INSTRUMENTOS MUSICALES

<b>Unidad:</b>	Unidad Nro. 3: Gestión Ágil de Proyectos
<b>Objetivo:</b>	Aplicar las prácticas de estimación, planificación y seguimiento de Scrum en la ejecución de un caso práctico.
<b>Consigna :</b>	<p>A través de un juego y de las instrucciones comunicadas por los docentes, implementar las siguientes prácticas de Scrum:</p> <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>

### 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 2:

- 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**





## PRÁCTICO 4 - GESTIÓN DE CONFIGURACIÓN (VIDEO)

<b>Unidad:</b>	Unidad Nro. 4: Gestión de Configuración del Software
<b>Consigna 1:</b>	Comprender los conceptos de SCM expuestos en la clase teórica, para aplicarlos en un ejemplo de la vida cotidiana.
<b>Objetivo:</b>	Que el alumno pueda inferir cómo definir una Estructura de Repositorio e identificar los Ítems de Configuración.
<b>Propósito:</b>	Familiarizarse con las actividades de SCM y repasar los conceptos de Gestión de Configuración.
<b>Entradas:</b>	Conceptos teóricos sobre el tema, desarrollados en clase. Plantilla de Ítems de Configuración.
<b>Salida:</b>	Se presentará en el pizarrón un ejemplo de Estructura de Repositorio y de un Listado Ítems de Configuración.
<b>Instrucciones:</b>	<ul style="list-style-type: none"> <li>- Pensar en todo el material que le suministra la materia y el generado por el alumno (trabajos prácticos, resoluciones de ejercicios, etc.).</li> <li>- Definir cómo sería la estructura del repositorio de la materia en el disco duro personal de cada alumno.</li> <li>- Identificar cada uno de los ítems de configuración.</li> <li>- Completar la plantilla de ítems de configuración, indicando para cada uno de ellos: nombre, regla de nombrado, ubicación física y tipo de ítem (Cátedra, Clase o Producción Propia).</li> </ul>

<b>Unidad:</b>	Unidad Nro. 4: Gestión de Configuración del Software
<b>Consigna 2:</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 alumno 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 y presentar una demostración del trabajo realizado.
<b>Entradas:</b>	Conceptos teóricos sobre el tema, desarrollados en clase. Propuesta de repositorio, vista en la clase práctica.
<b>Salida:</b>	Construcción y uso de un repositorio Presentación de demo en un cd o dvd.
<b>Instrucciones:</b>	<p>Haciendo uso de una herramienta para generar un video, mostrar (utilizando una herramienta de gestión de configuración) lo siguiente:</p> <ul style="list-style-type: none"> <li>▪ Crear el repositorio definido en el ejercicio visto en la clase práctica.</li> <li>▪ Mostrar cómo se puede bajar al puesto de trabajo local el contenido del repositorio (check-out).</li> <li>▪ Agregar una carpeta al repositorio.</li> <li>▪ Hacer un cambio en el repositorio y actualizarlo (Commit/Check in).</li> <li>▪ Bajar (actualizar) los cambios que posea el repositorio, realizados por otros usuarios (Update).</li> <li>▪ Mostrar cómo se puede definir una línea base (etiquetado).</li> <li>▪ Mostrar cómo se puede visualizar una línea base.</li> </ul>
<b>Observaciones:</b>	El docente no tendrá que instalar nada en su pc, por lo tanto se recomienda a los alumnos que utilicen un formato de video conocido y compatible (mp4, wmp, etc.). El video debe ser autoejecutable y debe mostrar todas las actividades solicitadas en el apartado de instrucciones.

## PRÁCTICO 5 - GESTIÓN DE CONFIGURACIÓN - TAXI MOBILE

<b>Unidad:</b>	Unidad Nro. 4: Gestión de Configuración de Software
<b>Consigna:</b>	Aplicar los conceptos de Gestión de Configuración de Software en el proyecto planteado como caso de estudio.
<b>Objetivo:</b>	Que el alumno, en el rol de Gestor de Configuración, pueda incorporar en la práctica los conceptos desarrollados en el teórico.
<b>Propósito:</b>	Familiarizarse con las actividades de SCM en el contexto de la planificación de proyectos de software. Repasar los conceptos de Gestión de Configuración.
<b>Entradas:</b>	Conceptos teóricos sobre el tema, desarrollados en clase. Caso de estudio. Plantilla de Ítems de Configuración.
<b>Salida:</b>	Cada grupo presentará una el trabajo realizado para ser evaluado.
<b>Instrucciones:</b>	Dividirse en grupos. Realice una propuesta de las consignas indicadas luego del caso de estudio

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

Una empresa de emprendedores requiere el desarrollo de una aplicación de uso gratuito que permita a pasajeros solicitar el servicio de taxis directamente a los taxistas a través de un Smartphone, para taxis de una ciudad.

Para utilizar la aplicación el pasajero deberá registrarse. Al seleccionar la opción "Llamar taxi" se solicita un viaje. Se le informará al pasajero los datos del taxista que primero responda la llamada, mostrando la foto y el nombre, junto con los datos del vehículo (modelo y placa) que irá a recogerlo. El taxi está monitorizado durante todo el recorrido, permitiendo al pasajero conocer dónde está el móvil en cada momento. Para aceptar una solicitud de viaje, el taxista recibirá una notificación sonora y podrá visualizar la solicitud en la pantalla. El sistema permitirá una conexión con una central de taxis, donde se visualicen los taxis en diferentes colores para saber su situación. Desde la central también se podrán derivar viajes a los taxistas, quienes registrarán la solicitud manualmente en el Smartphone.

Las pantallas deberán verse como se muestran en los siguientes prototipos, siendo generado el mapa por la plataforma provista por Google Maps:



Consideraciones para el Proyecto

El alcance de este proyecto comprende desde la definición de requerimientos hasta la puesta en producción del producto. Esta es una empresa que realiza emprendimientos innovadores, buscando llegar antes al mercado, por lo que los tiempos de desarrollo deben ser lo más cortos posible, a fin de ganarle a la competencia en el desarrollo de un producto innovador que acompañe las necesidades y tecnología actual. Esta empresa trabaja con micro-innovaciones, por lo que el objetivo propuesto siempre es que el tamaño del producto no crezca exponencialmente, sino que, por el contrario, con pequeños desarrollos de bajo presupuesto se brinde a los usuarios soluciones simples a los problemas diarios.

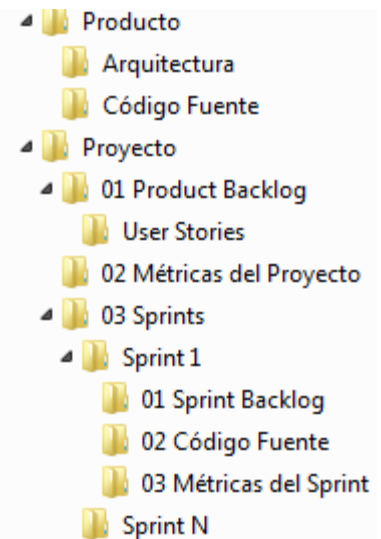
Para construir el software, se utilizará un proceso de desarrollo ágil con un framework SCRUM. Como se desprende de la descripción anterior, el Product Owner mantiene entrevistas con los expertos del dominio para definir los requerimientos, los cuales se registran como tarjetas que representan User Stories. En función del Product Backlog resultante el equipo del proyecto define la arquitectura de la solución.

En las reuniones de planificación (planning meeting) se define la capacidad del equipo y se identifican las actividades a realizar, el responsable y la estimación correspondiente a la tarea, información que se utilizará al inicio de y durante la ejecución de cada sprint.

Cuando se comienza a ejecutar el proyecto, al realizar las daily meeting en un pizarrón se genera el burndown chart para ir conociendo el trabajo ya “quemado”. Y en las sprint Planning meetings se utiliza la información generada en los sprints anteriores para evaluar la capacidad del equipo para el siguiente sprint.

El equipo del proyecto realizará, además, revisiones de pares del código generado. Estas revisiones se ejecutan y los hallazgos identificados por el revisor son guardados para luego realizar un seguimiento y verificar que se corrijan dichos hallazgos.

Se muestra a continuación el repositorio y las reglas de nombrado de los ítems de configuración:



Nombre del Ítem de Configuración	Regla de Nombrado	Ubicación Física	Ubicación Lógica [Producto / Proyecto / Sprint]
User Story	TM_N_US_Nombre_User_Story.doc	TaxiMobile/Proyecto/Sprint N/01SprintBacklog	Producto
Arquitectura	TM_Arquitectura_NombreModelo.png	TaxiMobile/Producto/Arquitectura	Producto
Product Backlog	TM_Product_Backlog.png	TaxiMobile/Proyecto/01ProductBacklog	Producto
Código fuente	TM_NombreClase.java	TaxiMobile/Producto/CódigoFuente	Producto
Burndown Chart	TM_Sprint_N_Métrica_BurndownChart.png	TaxiMobile/Proyecto/SprintN/03Métricas del Sprint	Sprint
Métricas del Proyecto	TM_Métrica_NombreMétrica.xlsx	TaxiMobile/Proyecto/02Métricasdel Proyecto	Proyecto
Sprint Backlog	TM_Sprint_N_Backlog.png	TaxiMobile/Proyecto/SprintN/01Sprint Backlog	Sprint
Revisiones	Revisión_Par_aaaammdd.xlsx	TaxiMobile/Proyecto/SprintN/Revisiones	Sprint

## Referencias:

- N es un número, de sprint o user story según corresponda.
- AAAA año, MM mes, DD día.

El equipo trabaja generando una versión intermedia diaria a partir de un merge de lo realizado por cada uno de los 5 miembros del equipo. Al final de cada sprint se genera una línea base de todo el código fuente, y en las fechas acordadas se generan los release. Luego del primer release el equipo se dividirá para trabajar dos personas en la app de los taxistas, mientras que los otros tres trabajarán en la app de los pasajeros. La app de los taxistas tendrá un segundo release a los 2 meses, mientras que el release de la app de pasajeros será a los 3 meses. En ese mismo momento también se actualizará nuevamente la versión de los taxistas.

La forma de nombrar cada versión respeta el siguiente formato:

<b>Versión de cada miembro de equipo de proyecto</b>	V.X.Y.Y.nombreMiembroEquipoAAMMDD
<b>Versión de cada día</b>	V.X.Y.Y.AAMMDD
<b>Línea base del sprint</b>	V_.X.0LB_SprintX
<b>Release</b>	R_X.Y

**Referencias:**

- Siendo AA el año, MM el día, DD el día en que se genera la versión.
- Siendo X el número del sprint, y en el caso de versiones anteriores al fin del sprint, se indica con el número X-1.
- Y el número de versión que puede ser Y.Y.Y...

**Se pide:**

1. Defina cuál de los siguientes productos de trabajo identificaría como ítems de configuración y justifique su respuesta:

Producto de Trabajo	Es IC	Justificación
1. User stories		
2. Foto de priorización del backlog		
3. Mail de resolución de un impedimento enviado por el Scrum Master		
4. Mail de aprobación de un release enviado por el product owner		
5. Mail que envió el scrum master solicitando aprobación del reléase		
6. Foto del burn down chart cada día del sprint		
7. Foto del burn down chart del último día de un sprint		

2. Realice una lista de al menos 3 hallazgos de los siguientes ítems de configuración, en función de las consideraciones del proyecto indicadas anteriormente, basándose en los siguientes criterios de severidad:
  - **Cosmético:** hallazgos de sintaxis que no afecten al significado del ítem de configuración y su identificación unívoca.
  - **Menor:** hallazgos de sintaxis que afectan la integridad del ítem de configuración.
  - **Mayor:** hallazgos que no son de sintaxis y afectan la integridad del producto.

Ítem de configuración	
1. TM_001_US_Ver_mapa_de_taxi.docx	
2. TM_Arquitectura_Modelo de Despliegue.png	
3. TM_Product_Backlog.png	
4. TM_Taxi .java(clase)	
5. TM_Sprint_3_Métrica_BurndownChart	Enero 2015.png
6. TM_Métrica_Velocidad.xlsx	
7. TM_Sprint_003_Backlog.png	
8. TM_pasajero.java (clase)	
9. Taxista.java (clase)	

Ítems de configuración	Severidad	Hallazgo

3. Realice el árbol de versiones en función de las consideraciones del proyecto.



## PRÁCTICO 6 - TESTING - MÉTODOS DE CAJA NEGRA - FESTIVAL DE FOLKLORE

<b>Unidad:</b>	Unidad Nro. 5: Aseguramiento de Calidad de Proceso y de Producto																		
<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 alumno 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. Filminas. Enunciado, consigna y User Story.																		
<b>Salida:</b>	Se presentará en el pizarrón el conjunto de clases de equivalencia y el resultado de la aplicación del método de análisis del valor límite.																		
<b>Instrucciones:</b>	<p>Según el método de partición de equivalencia, defina las clases existentes utilizando el siguiente cuadro para la Historia de Usuario "Definir precios".</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 <u>Template_Casos_De_Prueba</u>.</p>				Condición externa	Clases de equivalencia válidas		Clases de equivalencia inválidas											
Condición externa	Clases de equivalencia válidas		Clases de equivalencia inválidas																



### **Descripción del dominio:**

Anualmente la Dirección de Cultura de la Municipalidad de una localidad de la provincia, organiza un festival de folklore. Este festival tiene una duración de generalmente cinco noches, aunque esto puede variar de año en año. En cada una de las noches actúan distintos grupos folklóricos con reconocimiento regional, provincial y nacional. El festival se prepara con mucha anticipación y se realiza la diagramación para determinar qué grupos actúan en cada noche y el orden en el que los mismos realizarán sus presentaciones, teniendo en cuenta que los horarios de presentación de los grupos no pueden superponerse y que no pueden quedar espacios sin ninguna presentación entre medio de dos grupos. Considerar que no puede incluirse la participación de un grupo más de una vez para un mismo festival, en una misma noche.

En cada noche se define la hora de inicio de la misma, pero no se determina la hora de fin, ya que esta puede variar según si las presentaciones se extienden más de lo previsto.

El Festival se realiza en un único estadio, que está dividido en sectores (A, B, C, etc.), que se identifican con colores diferentes, y cada sector se compone de filas (1, 2, 3, etc.), cada fila, a su vez, está conformada por butacas, las cuales están numeradas.

Existen distintos tipos de entradas para el público (mayores, menores, jubilados, etc.). El precio de las entradas depende del tipo de entrada y del sector donde se encuentre la butaca, además puede variar de una noche a otra, dependiendo de los grupos musicales que actúan. Por ejemplo, una entrada para mayores en el sector A, que está cerca del escenario, será más costosa que una para mayores en el sector E que está más alejado del mismo y a su vez puede variar de noche en noche el precio de la entrada en la misma ubicación. Las butacas se venden para una noche en particular así es que una misma butaca puede estar disponible, por ejemplo, para la noche 1 y 3, y ocupada para la noche 2, 4 y 5.

#### **Definir precios**

Como Responsable de Festival quiero fijar precios de un festival para poder vender entradas a los precios fijados.

2

Nota: tener en cuenta que el precio depende de la noche del festival, el sector y el tipo de entrada. Todos los datos son requeridos.

#### **Pruebas de Usuario**

- ☐ Probar registrar precios para una noche, sector y tipo de entrada que ya los tiene (falla)
- ☐ Probar registrar precios para una noche, sector y tipo de entrada (pasa)
- ☐ Probar registrar precios para un festival vigente (pasa)
- ☐ Probar registrar precios cuando no hay festival vigente (falla)
- ☐ Probar registrar precios sin ingresar el precio de la entrada (falla)

## PRÁCTICO 7 - TESTING - MÉTODOS DE CAJA BLANCA - FESTIVAL DE FOLKLORE

<b>Unidad:</b>	Unidad Nro. 5: Aseguramiento de Calidad de Proceso y de Producto
<b>Consigna:</b>	Analice el pseudocódigo según los métodos de caja blanca
<b>Objetivo:</b>	Que el alumno 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. Filminas. Bibliografía sobre el tema, citada. Enunciado, consigna, templates y porción de pseudocódigo.
<b>Salida:</b>	Casos de prueba de caja blanca diseñados.

A continuación, se presenta una porción de pseudocódigo que resuelve la US de “Definir precios”:

```
If (Noche de festival vigente = Yes AND Sector habilitado = Yes)
    If (Precio registrado para tipo Entrada = NO)
        If (Precio valor numérico AND precio >= 0,00)
            [Registrar precio de Entrada]
        Else [Mostrar el mensaje “El precio debe ser mayor a cero”]
```

En base a los métodos de caja blanca, complete la siguiente tabla que se encuentra a continuación. Para cada fila, en la primera columna deberá completar la cantidad de casos de prueba que deben ejecutarse y en la columna “Dato” las características de las entradas de cada una de las pruebas, como lo indican los ejemplos que se encuentran bajo dicha columna.

Precondiciones:

- Los sectores Violeta, Rojo, Verde, y Amarillo fueron registrados y están habilitados
- El sector Azul se deshabilitó
- Del viernes 22/11 al domingo 24/11 se desarrollará el Festival
- Los tipos de entradas registrados son: Menores, Estudiantes, Jubilados, General.
- Todos los precios de entradas para jubilados ya han sido registrados.

	<b>Cantidad de Casos de Prueba</b>	<b>Dato</b>
<b>Cobertura de sentencias</b>		

	<b>Cantidad de Casos de Prueba</b>	<b>Dato</b>
<b>Cobertura de decisión</b>		<i>Precio \$50,00 para la noche del 22/11, sector Rojo, tipo de entrada "Estudiante"</i>
<b>Cobertura de condición</b>		
<b>Cobertura de condición /decisión</b>		
<b>Cobertura múltiple</b>		<i>Precio para la noche del 25/11 y el sector Azul</i>

	<b>Cantidad de Casos de Prueba</b>	<b>Dato</b>

## PRÁCTICO 8 - TESTING - EJECUCIÓN DE CASOS DE PRUEBA - FESTIVAL DE FOLKLORE

<b>Unidad:</b>	Unidad Nro. 5: Aseguramiento de Calidad de Proceso y de Producto
<b>Consigna:</b>	Ejecute los casos de pruebas de caja negra diseñados
<b>Objetivo:</b>	Que el alumno 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. Filminas. Enunciado, consigna, template, resultados del TP6. Funcionalidad ejecutable "Registrar precio de entrada"
<b>Salida:</b>	Se presentará en el pizarrón el conjunto de clases de equivalencia y el resultado de la aplicación del método de análisis del valor límite.
<b>Instrucciones:</b>	Utilice los casos de prueba definidos para la Historia de Usuario "Definir precios". Corra los casos de prueba identificados, con el ejecutable "Registrar Precio de Entrada" Utilice el template Template_Casos_De_Prueba para registrar la ejecución. Describa los bugs encontrados.

## PRÁCTICO 9 - LA CAJA DE PRODUCTO

<b>Unidad:</b>	Unidad Nro. 3: Gestión Ágil de Proyectos
<b>Consigna :</b>	Aplicar los conceptos teóricos desarrollados en clase sobre user stories. Consigna fundamental: <b>"un producto = una Visión = Una Caja con la Visión/Producto"</b>

### Materiales para trabajar:

- 2 afiches blancos
- Fibras y marcadores de colores
- Recortes de revistas
- Papeles de colores
- Goma de pegar
- Tijeras

### Instrucciones:

1. Cada grupo deberá elegir un producto y crear, en 10 minutos, una caja que contenga la visión del mismo, con los siguientes lineamientos:
  - a. Frente de la Caja: Nombre, Foto - Ícono o Dibujo, Slogan, 3 o 4 argumentos de venta
  - b. Contra frente: Vista detallada de la funcionalidad - Pre requisitos
2. La crítica - Feedback (5 minutos): Un representante de cada grupo toma su "caja" y visita otro grupo, expone sus argumentos y escucha y anota lo que dicen los compañeros.
3. El grupo determina el MVP (Minimum Viable Product: es la versión mínima de mi producto que me permite salir a probar al Mercado y obtener feedback)

## PRÁCTICO 10 - PECHA KUCHA

<b>Unidad:</b>	Unidad Nro. 3: Gestión Ágil de Proyectos
<b>Consigna:</b>	Realice una presentación sobre un tema de la materia para presentar a todo el curso
<b>Objetivo:</b>	Que el alumno pueda conocer una forma de presentación rápida haciendo foco en conceptos de la materia, y que pueda escuchar propuestas de la misma forma de sus compañeros.
<b>Propósito:</b>	Comprender como se realiza una presentación de las características propuesta por Pecha Kucha
<b>Entradas:</b>	Conceptos teóricos sobre el tema, desarrollados en clase. Filminas. Papers. Bibliografía Consigna.
<b>Salida:</b>	Cada grupo debe entregar la prestación propuesta y presentarla en el curso.
<b>Instrucciones:</b>	Por grupo realice una presentación de 20 filminas. Cada grupo debe organizarse para realizar la presentación, tomándose no más de 20 segundos por cada diapositiva. La presentación debe cerrar una idea clara en el tiempo asignado. Entre una impresión de la presentación indicando el número de grupo, los integrantes y el curso.