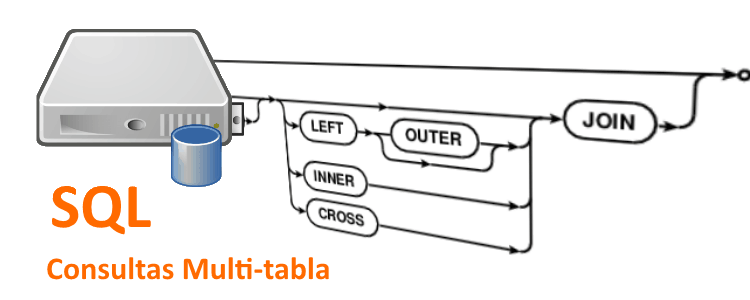
**LENGUAJE DE CONSULTAS A UNA BASE DE DATOS.**

**TALLER CASO PRÁCTICO**

**Objetivo de la jornada:**

* Comprender y ejecutar sentencias para la consulta de datos en base de datos relacionales
* Ejecutar y desarrollar el modelo relacional graficando soluciones para problemas planteados, consultando datos en diversas tablas.
* Practicar conceptos vistos en práctico en modalidad de proyectos.



|  |  |  |  |
| --- | --- | --- | --- |
| **Plan Formativo:** | **Nivel de Dificultad:** | | |
| **Full Stack Java Trainee** | **Básico/medio.** | | |
| **Modulo:** | **Tema:** | | |
| **Lenguaje de consultas a una base de datos** | **Base de datos relaciones.** | | |
| **Nombre del proyecto:** | **Proyecto de cierre módulo Lenguaje de consultas a una base de datos,** | | |
| **Objetivo del proyecto:** | * Construye consultas utilizando sentencias SQL con condiciones de selección para resolver un problema planteado de selección condicional. * Construye consultas utilizando sentencias SQL que requieren la consulta a varias tablas relacionadas. | | |
| **Tipo:** | **Grupal** | | |
| **Descripción del proyecto** | | | |
| 1. **Descripción de la actividad.**   **Dicha actividad tiene como nombre “Proyecto de módulo, Lenguaje construcción de consultas a una base de datos”. El cual busca dar solución a un caso, que se planteara en esta descripción. Esta solución estará basada en lograr la sintaxis y ejecución, con fin de poder obtener información certera de base de datos. Con relación a los requerimientos previamente planteados.**  **Con Respecto al proyecto.**   1. Se debe trabajar bajo el concepto de modelo de datos, entidad-relación. 2. Se debe implementar la mayor cantidad de conceptos vistos en cada una de clases, sin ser el total de ellos de carácter obligatorio. 3. Evitar la copia de código directo, tanto de internet como de sus pares. (La idea es que intente crear su propia sintaxis, de ser necesario busque bibliografía útil. 4. Se valorara participación y trabajo colaborativo. 5. Su revisión se realizara durante clases.  * **Respecto a la duración e integrantes.**  1. Tendrá una duración de máximo 3 clases.   Durante el último plazo, se revisara en unos de los equipos de los participantes.   1. Este proyecto tendrá como mínimo 1 y máximo de 2 2. participantes**.**   Planteamiento.  Una empresa del sector agropecuario, se ha comunicado con el bootcamp de talentos digitales. Con el fin de poder plantear un requerimiento, que es esto momentos, es de suma importancia. Ya que están próximos a una fiscalización de un ente gubernamental, y deben tener toda la información en orden, basada en los últimos 2 años, respecto a la gestión de cada uno de los empleados y el departamento al cual pertenecen. Actualmente cuenta con la información, pero está distribuida en distintos libros, los cuales por el paso del tiempo se han ido deteriorando, y del hecho de no salvarlos, se perderá el total de información.  Los datos requeridos para la conformación de su base de datos son los siguientes:   |  |  | | --- | --- | | Empleado | 1. Código auto\_increment primary key, 2. Rut, nombre 3. ambos apellidos, 4. código departamento | | Depar-tamento | 1. Código auto\_increment primary key. 2. Nombre del departamento. 3. Presupuesto. 4. Gastos. |  1. Desarrolle previamente el modelo entidad relación. 2. Luego de creado dicho modelo, deben ingresar a los menos 10 personas, y un mínimo de 7 departamentos. 3. Cree sentencias para ingresar, eliminar y actualizar cada una de las tablas.   4. Se deben obtener las siguientes consultas : Consultas sobre una tabla  1. Lista el primer apellido de todos los empleados. 2. Lista el primer apellido de los empleados eliminando los apellidos que estén repetidos. 3. Lista todas las columnas de la tabla empleado. 4. Lista el nombre y los apellidos de todos los empleados. 5. Lista el código de los departamentos de los empleados que aparecen en la tabla empleado. 6. Lista el código de los departamentos de los empleados que aparecen en la tabla empleado, eliminando los códigos que aparecen repetidos. 7. Lista el nombre y apellidos de los empleados en una única columna. 8. Lista el nombre y apellidos de los empleados en una única columna, convirtiendo todos los caracteres en mayúscula. 9. Lista el nombre y apellidos de los empleados en una única columna, convirtiendo todos los caracteres en minúscula. 10. Lista el código de los empleados junto al Rut, pero el Rut deberá aparecer en dos columnas, una mostrará únicamente los dígitos del Rut y la otra el guión. 11. Lista el nombre de cada departamento y el valor del presupuesto actual del que dispone. Para calcular este dato tendrá que restar al valor del presupuesto inicial (columna presupuesto) los gastos que se han generado (columna gastos). Tenga en cuenta que en algunos casos pueden existir valores negativos. Utilice un alias apropiado para la nueva columna que está calculando. 12. Lista el nombre de los departamentos y el valor del presupuesto actual ordenado de forma ascendente. 13. Lista el nombre de todos los departamentos ordenados de forma ascendente. 14. Lista el nombre de todos los departamentos ordenados de forma descendente. 15. Lista los apellidos y el nombre de todos los empleados, ordenados de forma alfabética teniendo en cuenta en primer lugar sus apellidos y luego su nombre. 16. Devuelve una lista con el nombre y el presupuesto, de los 3 departamentos que tienen mayor presupuesto. 17. Devuelve una lista con el nombre y el presupuesto, de los 3 departamentos que tienen menor presupuesto. 18. Devuelve una lista con el nombre y el gasto, de los 2 departamentos que tienen mayor gasto. 19. Devuelve una lista con el nombre y el gasto, de los 2 departamentos que tienen menor gasto. 20. Devuelve una lista con 5 filas a partir de la tercera fila de la tabla empleado. La tercera fila se debe incluir en la respuesta. La respuesta debe incluir todas las columnas de la tabla empleado. 21. Devuelve una lista con el nombre de los departamentos y el presupuesto, de aquellos que tienen un presupuesto mayor o igual a 1500000 de pesos. 22. Devuelve una lista con el nombre de los departamentos y el gasto, de aquellos que tienen menos de 500000 pesos de gastos. 23. Devuelve una lista con el nombre de los departamentos y el presupuesto, de aquellos que tienen un presupuesto entre 1000000 y 2000000 de pesos. Sin utilizar el operador BETWEEN. 24. Devuelve una lista con el nombre de los departamentos que **no** tienen un presupuesto entre 1000000 y 2000000 de pesos. Sin utilizar el operador BETWEEN. 25. Devuelve una lista con el nombre de los departamentos que tienen un presupuesto entre 1000000 y 2000000 de pesos. Utilizando el operador BETWEEN. 26. Devuelve una lista con el nombre de los departamentos que **no** tienen un presupuesto entre 1000000 y 2000000 de pesos. Utilizando el operador BETWEEN. 27. Devuelve una lista con el nombre de los departamentos, gastos y presupuesto, de aquellos departamentos donde los gastos sean mayores que el presupuesto del que disponen. 28. Devuelve una lista con el nombre de los departamentos, gastos y presupuesto, de aquellos departamentos donde los gastos sean menores que el presupuesto del que disponen. 29. Devuelve una lista con el nombre de los departamentos, gastos y presupuesto, de aquellos departamentos donde los gastos sean iguales al presupuesto del que disponen. 30. Lista todos los datos de los empleados cuyo segundo apellido sea NULL.  Consultas multitabla (Composición interna) Resuelva todas las consultas utilizando la sintaxis de SQL1   1. Devuelve un listado con los empleados y los datos de los departamentos donde trabaja cada uno. 2. Devuelve un listado con los empleados y los datos de los departamentos donde trabaja cada uno. Ordena el resultado, en primer lugar por el nombre del departamento (en orden alfabético) y en segundo lugar por los apellidos y el nombre de los empleados. 3. Devuelve un listado con el código y el nombre del departamento, solamente de aquellos departamentos que tienen empleados. 4. Devuelve un listado con el código, el nombre del departamento y el valor del presupuesto actual del que dispone, solamente de aquellos departamentos que tienen empleados. El valor del presupuesto actual lo puede calcular restando al valor del presupuesto inicial (columna presupuesto) el valor de los gastos que ha generado (columna gastos). 5. Devuelve el nombre del departamento donde trabaja el empleado que tiene el rut =” ”. 6. Devuelve el nombre del departamento donde trabaja el empleado Pepe Ruiz Santana. 7. Devuelve un listado con los datos de los empleados que trabajan en el departamento de I+D. Ordena el resultado alfabéticamente. 8. Devuelve un listado con los datos de los empleados que trabajan en el departamento de Sistemas, Contabilidad o I+D. Ordena el resultado alfabéticamente. 9. Devuelve una lista con el nombre de los empleados que tienen los departamentos que **no** tienen un presupuesto entre 1000000 y 2000000 de pesos. 10. Devuelve un listado con el nombre de los departamentos donde existe algún empleado cuyo segundo apellido sea NULL. Tenga en cuenta que no debe mostrar nombres de departamentos que estén repetidos.  1.2.5 Consultas multitabla (Composición externa) Resuelva todas las consultas utilizando las cláusulas LEFT JOIN y RIGHT JOIN.   1. Devuelve un listado con **todos los empleados** junto con los datos de los departamentos donde trabajan. Este listado también debe incluir los empleados que no tienen ningún departamento asociado. 2. Devuelve un listado donde sólo aparezcan aquellos empleados que no tienen ningún departamento asociado. 3. Devuelve un listado donde sólo aparezcan aquellos departamentos que no tienen ningún empleado asociado. 4. Devuelve un listado con todos los empleados junto con los datos de los departamentos donde trabajan. El listado debe incluir los empleados que no tienen ningún departamento asociado y los departamentos que no tienen ningún empleado asociado. Ordene el listado alfabéticamente por el nombre del departamento. 5. Devuelve un listado con los empleados que no tienen ningún departamento asociado y los departamentos que no tienen ningún empleado asociado. Ordene el listado alfabéticamente por el nombre del departamento.  1.2.6 Consultas resumen  1. Calcula la suma del presupuesto de todos los departamentos. 2. Calcula la media del presupuesto de todos los departamentos. 3. Calcula el valor mínimo del presupuesto de todos los departamentos. 4. Calcula el nombre del departamento y el presupuesto que tiene asignado, del departamento con menor presupuesto. 5. Calcula el valor máximo del presupuesto de todos los departamentos. 6. Calcula el nombre del departamento y el presupuesto que tiene asignado, del departamento con mayor presupuesto. 7. Calcula el número total de empleados que hay en la tabla empleado. 8. Calcula el número de empleados que **no tienen** NULL en su segundo apellido. 9. Calcula el número de empleados que hay en cada departamento. Tienes que devolver dos columnas, una con el código del departamento y otra con el número de empleados que tiene asignados. 10. Calcula los códigos de los departamentos que tienen más de 2 empleados. 11. Calcula el número de empleados que trabajan en cada uno de los departamentos. El resultado de esta consulta también tiene que incluir aquellos departamentos que no tienen ningún empleado asociado. 12. Calcula el número de empleados que trabajan en los departamentos que tienen un presupuesto mayor a 2000000 de pesos.   Subconsultas con ALL y ANY  1. Devuelve el nombre del departamento con mayor presupuesto y la cantidad que tiene asignada. Sin hacer uso de MAX, ORDER BY ni LIMIT. 2. Devuelve el nombre del departamento con menor presupuesto y la cantidad que tiene asignada. Sin hacer uso de MIN, ORDER BY ni LIMIT. 3. Devuelve los nombres de los departamentos que tienen empleados asociados. (Utilizando ALL o ANY). 4. Devuelve los nombres de los departamentos que no tienen empleados asociados. (Utilizando ALL o ANY).  1.2.7.3 Subconsultas con IN y NOT IN  1. Devuelve los nombres de los departamentos que tienen empleados asociados. (Utilizando IN o NOT IN). 2. Devuelve los nombres de los departamentos que no tienen empleados asociados. (Utilizando IN o NOT IN).  1.2.7.4 Subconsultas con EXISTS y NOT EXISTS  1. Devuelve los nombres de los departamentos que tienen empleados asociados. (Utilizando EXISTS o NOT EXISTS). 2. Devuelve los nombres de los departamentos que tienen empleados asociados. (Utilizando EXISTS o NOT EXISTS). | | | |
| **Contribuciones** | | | |
| 1. **Trabajo colaborativo.** 2. **Práctica de contenido visto durante el módulo.** 3. **Desarrollo de lógica de consultas a base de datos.** 4. **Resolución de problemas** 5. **Inferencia, abstracción de soluciones a partir de un problema.** 6. **Empleo de sintaxis básica, para la creación de Queries en Sql.** 7. **Implementación correcta de proyectos.** 8. **Proactividad y búsqueda de soluciones.** 9. **Medición de avances del módulo.** | | | |
| **Requerimientos de los alumnos** | | | |
| **Conocimientos previos**  ¿Qué necesitarán saber para lograr la realización del proyecto? | **Actitudes para el trabajo**  ¿Qué comportamientos necesitan manifestar y desarrollar durante el proyecto? | **Valores**  ¿Qué valores deberán ponerse en juego para cumplir las metas? | |
| * Base de datos relaciones * Sentencias para la definición de tablas * Consulta de datos multitabla. * Ejecución de sentencias para la creación e inserción de datos. | * Proactividad. * Manejo de la frustración. * Consenso. * Trabajo colaborativo * Optimización del trabajo en base al tiempo | * Compañerismo. * Disciplina en base al trabajo. | |
| **Objetivo General de Aprendizaje**  Redactar el objetivo tomando en cuenta el nivel taxonómico del conocimiento que se desea lograr. (ver tabla de verbos) | El alumno al finalizar el proyecto será capaz de:   * Generar Queries necesarias, para la creación de modelos lógicos tanto físicos de base de datos relaciones. * CRUD: podrá Crear, leer, actualizar y borrar, desde una base de datos. * Hacer consultas, simples y avanzadas en una o muchas tablas. * Obtener información en relevante de tablas de base de datos. | | |
| **Objetivos particulares**  Solo redactar sí lo considera necesario. |  | | |
| **Duración del proyecto**  (clases, días, semanas) | Tendrá una duración de máximo 3 clases. Durante el último plazo, se revisara en unos de los equipos de los participantes. | | |
| **Tips o listado de Preguntas Guía** | | |
| ¿Qué significa la sigla DBA? ¿Qué es una Base de Datos?  ¿Qué es una tabla maestro?  ¿Qué son las Bases de Datos estáticas?  ¿Qué significa la sigla LOPD? ¿Qué son Bases de Datos dinámicas?  ¿Qué es modelo entidad-relación?  ¿Cuál es modelo más óptimo para la creación de tablas? | | |
| **Productos a obtener durante la realización del proyecto** | | |
| **1.-Avances diarios, revisión de trabajo, en base a la totalidad del proyecto.**  **2.- Entrega de proyecto final, revisión de cumplimiento de objetivos.** | | |
| **Especificaciones de desempeño** | | |
|  | | |
| **Cronograma de actividades** | | |
| **1.-Presentación de objetivos.**  **2.- Conformación de grupos.**  **3.-Revision de avance clase a clase, resolución de problemas y preguntas.**  **4.-Revision Final de proyecto.** | | |
| **Sugerencias bibliográficas para la investigación** | | |
| <https://www.ecured.cu/Programaci%C3%B3n_Orientada_a_Objetos> | | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **RUBRICA MODULO 2 LENGUAJE DE CONSULTAS A UNA BASE DE DATOS.** | | | | | | | |
|  |  |  |  |  |  |  |  |
| **Base de datos relacionales** | | | | | | | |
| **Habilidad** | **Descripción** | **Nivel 1** | **Nivel 2** | **Nivel 3** | **Nivel 4** | **Nivel 5** | **Expectativa** |
| Base de datos relaciones | Desarrolla el código para la creación de  Queries | No, en ningún caso o incorrecto | No, en general | Sí, pero no en todos los casos | Si, en general (80%) | Sí, siempre y perfectamente (100.00%) | 3 |
|  |  |  |  |  |  |
| Verifica que el código se ejecuta correctamente y en caso de presentar errores realiza la depuración y/o corrección. | No, en ningún caso o incorrecto | No, en general | Sí, pero no en todos los casos | Si, en general (80%) | Sí, siempre y perfectamente (100.00%) | 3 |
|  |  |  |  |  |  |
| Inserta comentarios en el código desarrollado para facilitar la comprensión y documentación. | No, en ningún caso o incorrecto | No, en general | Sí, pero no en todos los casos | Si, en general (80%) | Si, siempre y perfectamente (100.00%) | 3 |
|  |  |  |  |  |  |
| Implementa Conceptos de forma correcta, y maneja programa de administración de base de datos. | No, en ningún caso o incorrecto | No, en general | Si, pero no en todos los casos | Si, en general (80%) | Si, siempre y perfectamente (100.00%) | 3 |
|  |  |  |  |  |  |
| Presenta el código fuente en orden y sintaxis correcta para la creación de base de datos. | No, en ningún caso o incorrecto | No, en general | Si, pero no en todos los casos | Si, en general (80%) | Si, siempre y perfectamente (100.00%) | 3 |
|  |  |  |  |  |  |
| El código se ejecuta con errores, y lograr hacer consultas a la query ejecutada | No, en ningún caso o incorrecto | No, en general | Si, pero no en todos los casos | Si, en general (80%) | Si, siempre y perfectamente (100.00%) | 3 |
|  |  |  |  |  |  |
| Presenta errores de sintaxis y/o de ejecución de Query, en el código desarrollado para implementar su base de datos. | No, en ningún caso o incorrecto | No, en general | Si, pero no en todos los casos | Si, en general (80%) | Si, siempre y perfectamente (100.00%) | 3 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Base de datos relacionales** | | | | | | | |
| **Habilidad** | **Descripción** | **Nivel 1** | **Nivel 2** | **Nivel 3** | **Nivel 4** | **Nivel 5** | **Expectativa** |
| Sentencias para la manipulación de datos | Desarrolla el Queries para la creación de  consultas | No, en ningún caso o incorrecto | No, en general | Si, pero no en todos los casos | Si, en general (80%) | Si, siempre y perfectamente (100.00%) | 3 |
|  |  |  |  |  |  |
| Verifica que el código se ejecuta correctamente y en caso de presentar errores realiza la depuración y/o corrección. | No, en ningún caso o incorrecto | No, en general | Si, pero no en todos los casos | Si, en general (80%) | Si, siempre y perfectamente (100.00%) | 3 |
|  |  |  |  |  |  |
| Inserta comentarios en el código desarrollado para facilitar la comprensión y documentación. | No, en ningún caso o incorrecto | No, en general | Si, pero no en todos los casos | Si, en general (80%) | Si, siempre y perfectamente (100.00%) | 3 |
|  |  |  |  |  |  |
| Implementa Conceptos de forma correcta, y lograr obtener información requerida | No, en ningún caso o incorrecto | No, en general | Si, pero no en todos los casos | Si, en general (80%) | Si, siempre y perfectamente (100.00%) | 3 |
|  |  |  |  |  |  |
| Presenta el código fuente en orden y sintaxis correcta para la creación de consultas | No, en ningún caso o incorrecto | No, en general | Si, pero no en todos los casos | Si, en general (80%) | Si, siempre y perfectamente (100.00%) | 3 |
|  |  |  |  |  |  |
| El código se ejecuta con errores, y lograr hacer consultas a la query ejecutada | No, en ningún caso o incorrecto | No, en general | Si, pero no en todos los casos | Si, en general (80%) | Si, siempre y perfectamente (100.00%) | 3 |
|  |  |  |  |  |  |
| Presenta errores de sintaxis y/o de ejecución de Query, en el código desarrollado para Obtener información | No, en ningún caso o incorrecto | No, en general | Si, pero no en todos los casos | Si, en general (80%) | Si, siempre y perfectamente (100.00%) | 3 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Base de datos relacionales** | | | | | | |
| **Habilidad** | **Descripción** | **Nivel 1** | **Nivel 2** | **Nivel 3** | **Nivel 4** | **Nivel 5** | |
| **Consultando información relacionada en varias tablas** | Desarrolla el código para la creación de  Queries | No, en ningún caso o incorrecto | No, en general | Si, pero no en todos los casos | Si, en general (80%) | Si, siempre y perfectamente (100.00%) | |
|  |  |  |  |  | |
| Verifica que el código se ejecuta correctamente y en caso de presentar errores realiza la depuración y/o corrección. | No, en ningún caso o incorrecto | No, en general | Si, pero no en todos los casos | Si, en general (80%) | Si, siempre y perfectamente (100.00%) | |
|  |  |  |  |  | |
| Inserta comentarios en el código desarrollado para facilitar la comprensión y documentación. | No, en ningún caso o incorrecto | No, en general | Si, pero no en todos los casos | Si, en general (80%) | Si, siempre y perfectamente (100.00%) | |
|  |  |  |  |  | |
| Implementa Conceptos de forma correcta, y maneja programa de administración de base de datos. | No, en ningún caso o incorrecto | No, en general | Si, pero no en todos los casos | Si, en general (80%) | Si, siempre y perfectamente (100.00%) | |
|  |  |  |  |  | |
| Presenta el código fuente en orden y sintaxis correcta para la creación de base de datos. | No, en ningún caso o incorrecto | No, en general | Si, pero no en todos los casos | Si, en general (80%) | Si, siempre y perfectamente (100.00%) | |
|  |  |  |  |  | |
| El código se ejecuta con errores, y lograr hacer consultas a la query ejecutada | No, en ningún caso o incorrecto | No, en general | Si, pero no en todos los casos | Si, en general (80%) | Si, siempre y perfectamente (100.00%) | |
|  |  |  |  |  | |
| **Sentencias para la definición de tablas** | Desarrolla el código para la creación de  Queries | No, en ningún caso o incorrecto | No, en general | Si, pero no en todos los casos | Si, en general (80%) | Si, siempre y perfectamente (100.00%) | |
|  |  |  |  |  | |
|  | Verifica que el código se ejecuta correctamente y en caso de presentar errores realiza la depuración y/o corrección | No, en ningún caso o incorrecto | No, en general | Si, pero no en todos los casos | Si, en general (80%) | Si, siempre y perfectamente (100.00%) | |
|  |  |  |  |  | |
|  | Inserta comentarios en el código desarrollado para facilitar la comprensión y documentación. | No, en ningún caso o incorrecto | No, en general | Si, pero no en todos los casos | Si, en general (80%) | Si, siempre y perfectamente (100.00%) | |
|  |  |  |  |  | |
|  | Implementa Conceptos de forma correcta, y maneja programa de administración de base de datos. | No, en ningún caso o incorrecto | No, en general | Si, pero no en todos los casos | Si, en general (80%) | Si, siempre y perfectamente (100.00%) | |
|  |  |  |  |  | |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Habilidades de Autogestión** | | | | | | |  |
| **Habilidad** | **Descripción** | **Nivel 1** | **Nivel 2** | **Nivel 3** | **Nivel 4** | **Nivel 5** | **Expectativa Final** |
| **Planificación, organización y manejo del tiempo** | Planifica y organiza el trabajo de manera priorizada, incremental e iterativa, aprovechando su tiempo y utilizando herramientas que la ayuden a lograrlo. Demuestra tener un prototyping mindset. | No planifica y/o no gestiona su tiempo | Logra organizarse de manera general, sin un plan específico, haciendo una estimación muy general del tiempo. | Escribe como historias de usuario y las prioriza en función del valor de negocio/usuario y de las dependencias entre sí. | Escribe como historias de usuario y las prioriza en función del valor de negocio/usuario y de las dependencias entre sí. | Escribe como historias de usuario y las prioriza en función del valor de negocio/usuario y de las dependencias entre sí. | 4 |
|  |  |  |  |
| Conoce las ceremonias y artefactos ágiles (sprint planning, dailys, retrospectiva, tablero al menos, etc) pero los omite y/o no las utiliza de manera adecuada. | Escribe definiciones de terminado para cada historia de usuario del "product backlog". | Escribe definiciones de terminado para cada historia de usuario del "product backlog". | Escribe definiciones de terminado para cada historia de usuario del "product backlog". |
|  |  |  |  |
|  | Cumple, casi siempre, con las ceremonias y mantiene su tablero actualizado. | Cumple siempre con las ceremonias y con mantener su tablero actualizado. | Cumple siempre con las ceremonias y con mantener su tablero actualizado. |
|  |  |  |  |
|  | No pierde tiempo en tareas con menor relevancia para el produto/usuario. | No pierde tiempo en tareas con menor relevancia para el produto/usuario, logrando gestionar su tiempo. | No pierde tiempo en tareas con menor relevancia para el produto/usuario, logrando gestionar su tiempo. |
|  |  |  |  |
|  |  | Hace estimaciones en "puntos" usando como referencia otras tareas ya conocidas ("planning  poker"). | Hace estimaciones con precisión y, por lo tanto, planificar "releases" o entregas parciales con comodidad. |
|  |  |
|  |  |
|  |