**Cuestionario Evaluador del Curso**

**DEVOPS**

1. ¿Qué es DevOps?
2. Menciona las fases principales del ciclo de vida de DevOps.
3. ¿Qué herramienta se utilizó para realizar un test de Kahoot sobre los fundamentos de DevOps?
4. ¿Qué es JUnit y para qué se utiliza?
5. ¿Cómo se crea un proyecto con IntelliJ para trabajar con JUnit?
6. ¿Qué anotaciones de JUnit se utilizaron para escribir pruebas unitarias?
7. Explica brevemente qué es TDD (Desarrollo Guiado por Pruebas).
8. ¿Qué significa la anotación **@DisplayName** en JUnit?
9. ¿Para qué sirve la anotación **@Disabled** en JUnit?
10. ¿Qué es **assertThrows** y cómo se utiliza en JUnit?
11. Describe un ejemplo práctico de cómo se implementa TDD para operaciones de débito y crédito.
12. ¿Qué es Mockito y cuál es su propósito?
13. ¿Cómo se configura un proyecto con JUnit y Mockito en IntelliJ?
14. ¿Qué es una dependencia mock y cómo se crea con Mockito?
15. ¿Para qué se utiliza el método **verify** en Mockito?
16. Explica el uso de las anotaciones **@Mock** y **@InjectMocks**.
17. ¿Cómo se prueba un repositorio utilizando Mockito?
18. ¿Qué es Spring Boot y cómo se relaciona con las pruebas unitarias?
19. ¿Qué anotaciones se utilizan para inyectar dependencias en Spring Boot?
20. ¿Cómo se configuran las clases de servicio en un proyecto Spring Boot?
21. ¿Qué es MockMvc y para qué se utiliza en Spring Boot?
22. ¿Cómo se configura Swagger en un proyecto Spring Boot?
23. ¿Qué son los endpoints y cómo se prueban con MockMvc?
24. ¿Qué anotación se utiliza para escribir pruebas de controladores con MockMvc?
25. ¿Qué es la cobertura de código y cómo se mide?
26. ¿Qué es WebTestClient y cómo se utiliza para probar servicios REST?
27. Explica cómo escribirías una prueba de integración para listar elementos usando WebTestClient.
28. ¿Qué diferencia hay entre WebTestClient y TestRestTemplate?
29. ¿Cómo se configura TestRestTemplate para probar servicios REST?
30. Escribe un ejemplo de cómo probarías un endpoint de eliminación con TestRestTemplate.
31. ¿Qué es Jenkins y para qué se utiliza?
32. ¿Cómo se descarga e instala Jenkins?
33. Menciona tres plugins importantes que se pueden usar en Jenkins.
34. ¿Qué es un proyecto de estilo libre (Freestyle) en Jenkins?
35. Explica cómo funcionan los Build Triggers en Jenkins.
36. ¿Cómo se envían parámetros a un Job en Jenkins?
37. ¿Qué herramientas se pueden integrar con Jenkins para proyectos Java y Maven?
38. ¿Qué son las vistas en Jenkins y cómo se crean?
39. Explica qué es una Pipeline en Jenkins.
40. ¿Qué son las directivas en una Pipeline de Jenkins?
41. ¿Cómo se implementa el paralelismo en una Pipeline de Jenkins?
42. Menciona un plugin visual útil en Jenkins y describe su función.
43. ¿Qué son las credenciales en Jenkins y cómo se configuran?
44. ¿Cómo se integra Jenkins con herramientas de control de versiones como Git?
45. Explica cómo configurarías una Pipeline para desplegar una aplicación Spring Boot.
46. ¿Qué es Jenkins y para qué se utiliza?
47. ¿Cómo se descarga e instala Jenkins?
48. Menciona tres plugins importantes que se pueden usar en Jenkins.
49. ¿Qué es un proyecto de estilo libre (Freestyle) en Jenkins?
50. Explica cómo funcionan los Build Triggers en Jenkins.
51. ¿Cómo se envían parámetros a un Job en Jenkins?
52. ¿Qué herramientas se pueden integrar con Jenkins para proyectos Java y Maven?
53. ¿Qué son las vistas en Jenkins y cómo se crean?
54. Explica qué es una Pipeline en Jenkins.
55. ¿Qué son las directivas en una Pipeline de Jenkins?
56. ¿Cómo se implementa el paralelismo en una Pipeline de Jenkins?
57. Menciona un plugin visual útil en Jenkins y describe su función.
58. ¿Qué son las credenciales en Jenkins y cómo se configuran?
59. ¿Cómo se integra Jenkins con herramientas de control de versiones como Git?
60. Explica cómo configurarías una Pipeline para desplegar una aplicación Spring Boot.