AIF-C01 · CHEAT SHEET · CERTIFYDECK

AWS Certified AI Practitioner

Dominios, pesos oficiales y habilidades clave del temario de AIF-C01, para repasar antes del examen.

Nivel: Beginner Duración: 90 min Nota de corte: 700/1000 Fuente: guía de estudio oficial de Microsoft Learn

1. Fundamentals of AI and ML

20–20%

Explain basic AI concepts and terminologies

- Un modelo de ML se entrena con datos históricos para aprender patrones y después hace inferencias, es decir, predicciones sobre datos nuevos. Un buen ajuste generaliza: el sobreajuste acierta en entrenamiento pero falla con datos nuevos y el subajuste falla en ambos. El sesgo produce resultados desiguales entre grupos y daña la equidad; un LLM es un modelo de deep learning muy grande preentrenado con enormes volúmenes de texto.
- La IA es el término general; el ML es una de sus ramas y aprende patrones de los datos sin instrucciones explícitas, y el deep learning es un tipo de ML con redes neuronales de muchas capas. La IA generativa usa modelos de deep learning para crear contenido nuevo en lugar de solo clasificar. La IA agéntica combina esos modelos con autonomía para planificar y ejecutar acciones mediante herramientas hacia un objetivo.
- La inferencia en tiempo real atiende peticiones interactivas de baja latencia; y la serverless evita gestionar instancias y encaja con tráfico intermitente que tolera arranques en frío. La asíncrona encola peticiones con cargas grandes o procesamiento largo y puede bajar a cero instancias. La inferencia por lotes, como batch transform o la de Amazon Bedrock, procesa grandes volúmenes de datos sin un endpoint persistente.
- Los datos estructurados encajan en tablas con un esquema fijo; los no estructurados, como audio, imágenes o texto libre, no siguen reglas predefinidas. Los datos etiquetados emparejan cada entrada con su respuesta correcta y los exige el aprendizaje supervisado, mientras que los no etiquetados sirven para descubrir patrones y preentrenar LLM. Las series temporales añaden una marca de tiempo a cada valor para predecir valores futuros.
- El aprendizaje supervisado entrena con entradas emparejadas con salidas etiquetadas (clasificación y regresión); el no supervisado descubre patrones en datos sin etiquetar (clustering, detección de anomalías); el semisupervisado mezcla pocos datos etiquetados con muchos sin etiquetar. El aprendizaje por refuerzo aprende por prueba y error para maximizar una recompensa, y el RLHF lo aplica con preferencias humanas para alinear LLM.

Identify practical use cases for AI

- La IA/ML aporta valor al automatizar tareas rutinarias como extraer y clasificar datos de documentos, al ayudar a las personas a decidir mejor y antes destacando patrones en grandes volúmenes de datos, y al escalar servicios como la atención al cliente 24/7. Encaja sobre todo en problemas con patrones complejos o cambiantes, como el fraude o las recomendaciones personalizadas.
- El ML es probabilístico: da predicciones, no resultados exactos garantizados, así que cuando la lógica se conoce y se necesita un resultado concreto y reproducible conviene un sistema basado en reglas. Antes de invertir hay que comparar el ML con una referencia más sencilla y estimar su coste total: preparación y etiquetado de datos, entrenamiento, alojamiento y mantenimiento.
- La regresión predice valores continuos, como el precio de una vivienda, y la clasificación asigna cada elemento a clases predefinidas: binaria con dos clases, como spam o no spam, y multiclase con varias. Sin etiquetas, el clustering descubre grupos de elementos parecidos, la detección de anomalías señala observaciones raras y el modelado de temas organiza documentos en temas no conocidos de antemano.
- La visión artificial analiza imágenes y vídeo, como la inspección de defectos; el NLP extrae sentimiento o entidades de texto y el reconocimiento de voz convierte audio en texto. Las recomendaciones salen de las interacciones de usuarios, el fraude se detecta como anomalía, la previsión usa series temporales, las bases de conocimiento responden con datos propios y la IA agéntica actúa sola hacia un objetivo.
- Amazon Transcribe convierte voz en texto, Amazon Polly convierte texto en voz natural y Amazon Translate traduce texto entre idiomas. Amazon Comprehend extrae entidades, frases clave y sentimiento, y crea clasificadores propios con AutoML; Amazon Lex crea bots de voz y texto que entienden la intención. Amazon SageMaker AI es el servicio gestionado para crear, entrenar y desplegar modelos propios.

Describe the AI/ML development lifecycle

- El ciclo de vida de ML pasa por objetivo de negocio, planteamiento del problema, procesamiento de datos (recopilación, preprocesamiento e ingeniería de características), desarrollo del modelo (entrenamiento, ajuste y evaluación), despliegue y supervisión, con bucles de retroalimentación. Los datos se dividen en entrenamiento, validación y prueba, y el mismo preprocesamiento se aplica en inferencia.
- Crear un modelo fundacional desde cero es caro, exige enormes recursos y puede llevar meses, así que lo habitual es partir de un FM preentrenado. SageMaker JumpStart ofrece FM públicos y propietarios de terceros, cada uno con su licencia, que el cliente debe revisar; Bedrock Marketplace amplía el catálogo y Custom Model Import trae a Bedrock modelos personalizados en otro entorno.
- Un modelo puede consumirse como servicio de API administrado, como Amazon Bedrock (sin servidores, pago por tokens y opción de Provisioned Throughput para rendimiento fijo), o autoalojarse. Los endpoints administrados de SageMaker AI dan más control que Bedrock con menos carga que EC2; el autoalojamiento completo da el máximo control, pero exige mantener endpoints y rendimiento.
- Amazon Bedrock da acceso sin servidor a modelos fundacionales; SageMaker AI cubre la creación, el entrenamiento y el despliegue de modelos propios (Feature Store, Pipelines, Model Registry). Kiro es un entorno de desarrollo agéntico guiado por especificaciones para programadores, y Amazon Quick ofrece a usuarios de negocio paneles, investigación y automatizaciones con IA, sin infraestructura.
- MLOps aplica los principios de DevOps al ML: automatiza en pipelines de CI/CD el entrenamiento y el despliegue, versiona código y datos para que todo sea reproducible y auditable, y limita la deuda técnica empezando por un modelo de referencia sencillo. En producción se supervisa la deriva y se reentrena cuando las métricas caen por debajo de los KPI, no con un calendario fijo.

2. Fundamentals of GenAI

24–24%

Explain the basic concepts of generative AI (GenAI)

- Un token es la unidad de texto que procesa el modelo; el chunking divide documentos para que quepan en la ventana de contexto, y los embeddings convierten cada fragmento en un vector que representa su significado. Los LLM basados en transformers usan autoatención, los modelos multimodales entienden texto, imagen o vídeo y los de difusión generan imágenes eliminando ruido.
- La IA generativa crea contenido nuevo: texto, código, imágenes, vídeo y voz. Sus usos habituales son resumir documentos, traducir, sugerir código, redactar textos de marketing, asistentes y agentes conversacionales de atención al cliente, búsqueda de conocimiento, recomendaciones personalizadas y datos sintéticos para probar casos poco frecuentes.
- Un FM se preentrena con aprendizaje autosupervisado sobre grandes volúmenes de datos sin etiquetar, cuidadosamente filtrados para evitar sesgos. Después se elige el modelo según modalidad, contexto, latencia y coste, se personaliza con prompts, RAG o ajuste fino con datos etiquetados, se evalúa, se despliega y se mejora de forma continua con métricas y opiniones de los usuarios.
- Con precios bajo demanda se paga por los tokens de entrada y de salida, así que prompts y respuestas más largos encarecen cada solicitud y además añaden latencia. Para ahorrar se acortan los prompts, se limita la longitud de respuesta, se usa prompt caching con contexto repetido e inferencia por lotes cuando no hace falta respuesta inmediata; Provisioned Throughput se factura por hora.
- La ingeniería de contexto diseña sistemas que reúnen de forma dinámica la información que necesita el LLM: prompt del sistema, consulta, perfil del usuario, memoria, herramientas o servidores MCP y documentos recuperados con RAG. Va más allá de la ingeniería de prompts; un contexto escaso provoca alucinaciones y uno sobrecargado aumenta la latencia y el coste.

Understand the capabilities and limitations of GenAI for solving business problems

- La IA generativa responde de forma natural a la conversación humana, lo que permite asistentes que resuelven consultas en el primer contacto, y genera contenido nuevo: textos, prototipos, código, resúmenes o datos sintéticos etiquetados. Los modelos fundacionales son adaptables: un mismo modelo realiza tareas muy distintas según el prompt, frente al ML tradicional, que se entrena para una sola tarea.
- La IA generativa puede alucinar o dar información inexacta con apariencia plausible, y refleja los sesgos de sus datos. Sus modelos son cajas negras difíciles de interpretar y su salida no es determinista: cada token se muestrea de una distribución, así que la misma petición puede variar (una temperatura baja la hace más estable). RAG y las contextual grounding checks de Amazon Bedrock Guardrails ayudan a mitigarlo.
- Al elegir un modelo hay que valorar modalidad, tamaño, precisión, datos de entrenamiento, precio, ventana de contexto, latencia, compatibilidad con la infraestructura y las políticas de uso de datos del proveedor. AWS recomienda fijar la calidad mínima, probar primero modelos pequeños y subir solo si hace falta, comparar candidatos con Amazon Bedrock evaluations y enrutar peticiones simples a modelos más baratos.
- Cada métrica técnica debe vincularse a un resultado de negocio, por ejemplo con el marco OGSM. El ROI se sigue de forma continua en un panel que combina costes (coste por interacción, infraestructura) y valor (horas ahorradas, ingresos, CSAT). Según el caso: tasa de conversión en marketing, valor del ciclo de vida del cliente en retención y tiempo de salida al mercado o de revisión de código en eficiencia.

Describe AWS infrastructure and technologies for building GenAI applications

- Amazon Bedrock da acceso gestionado por API a modelos fundacionales; SageMaker AI sirve para crear, entrenar y desplegar modelos propios, y JumpStart ofrece modelos preentrenados para ajustar y desplegar. Strands Agents es un SDK abierto para programar agentes y Amazon Bedrock AgentCore los ejecuta y opera a escala. Kiro es un IDE agéntico guiado por especificaciones y Amazon Quick, un espacio de trabajo con IA para usuarios de negocio.
- Los servicios gestionados de IA generativa de AWS bajan la barrera de entrada: Amazon Bedrock da acceso por API a modelos de muchos proveedores sin infraestructura, y Amazon Quick acerca agentes y automatización a usuarios sin perfil técnico. Evitan el trabajo pesado indiferenciado de alojar modelos, aceleran la salida al mercado y permiten usar datos propios con Knowledge Bases sin entrenar modelos.
- Con el modelo de responsabilidad compartida, AWS protege la infraestructura y el cliente controla sus datos, permisos de IAM y configuración. Amazon Bedrock cifra los datos, admite claves de AWS KMS y AWS PrivateLink, no comparte los datos con los proveedores de modelos y tiene informes de cumplimiento en AWS Artifact. Amazon Bedrock Guardrails añade seguridad: temas denegados, filtros de contenido y de PII.
- En Amazon Bedrock, la inferencia bajo demanda se paga por tokens de entrada y salida; Provisioned Throughput da capacidad fija facturada por horas, con descuento a más compromiso. Los niveles Priority, Standard y Flex cambian velocidad por precio. La inferencia entre regiones absorbe picos (geográfica para residencia de datos) y la destilación crea modelos más pequeños y baratos.

3. Applications of Foundation Models

28–28%

Describe design considerations for applications that use foundation models (FMs)

- Para elegir un FM se parte de los requisitos: modalidad (texto, imagen, audio), idiomas, ventana de contexto, latencia y coste por token. AWS aconseja probar primero modelos pequeños y subir de tamaño solo si la calidad no basta, y enviar las peticiones sencillas a modelos baratos. El prompt caching reduce latencia y coste de entrada cuando se reutiliza un contexto largo.
- Los parámetros de inferencia cambian la respuesta sin tocar el modelo. Temperature baja da respuestas más deterministas y alta, más variadas; Top K y Top P limitan cuántos candidatos se consideran para el siguiente token. El máximo de tokens, las stop sequences y las penalizaciones controlan la longitud y la repetición de la salida, lo que también influye en el coste.
- RAG recupera información relevante de las fuentes de datos de la empresa y la añade al prompt antes de que el FM responda, sin reentrenar el modelo. Es útil cuando los datos son privados o cambian a menudo, y permite citar las fuentes. Amazon Bedrock Knowledge Bases ofrece RAG gestionado: ingesta, fragmentación, embeddings, búsqueda vectorial y aumento del prompt.
- Los embeddings se guardan en almacenes vectoriales para buscar por similitud. Amazon OpenSearch Service da alto rendimiento y búsqueda de texto completo más vectorial; Amazon Aurora y Amazon RDS for PostgreSQL con pgvector combinan SQL y vectores; Amazon Neptune Analytics une grafos y vectores para GraphRAG; Amazon S3 Vectors abarata grandes volúmenes con consultas poco frecuentes.
- AWS recomienda empezar por el prompt engineering (aprendizaje en contexto), seguir con RAG y pasar al fine-tuning solo si hace falta; entrenar un FM desde cero es lo más caro. Los ejemplos en el prompt se pagan como tokens en cada solicitud, mientras que el fine-tuning cuesta entrenamiento y almacenamiento. La destilación crea un modelo más pequeño y barato de servir.

Choose effective prompt engineering techniques

- Un prompt combina instrucción (qué hacer), contexto (información que orienta), datos de entrada, ejemplos e indicadores de salida (formato o longitud). Por API, el modelo no recuerda solicitudes anteriores salvo que se incluyan en el prompt. En generación de imágenes, el negative prompt indica qué excluir, porque estos modelos no entienden palabras de negación como "no" o "sin".
- Zero-shot no da ejemplos y depende del preentrenamiento; single-shot da un ejemplo y few-shot varios, que el modelo aprende en contexto sin cambiar sus pesos. Conviene que los ejemplos se parezcan a los datos reales. Chain-of-thought pide razonar paso a paso en tareas complejas, y las plantillas de prompt fijan una estructura reutilizable con marcadores para las partes variables.
- La ingeniería de prompts mejora la relevancia y precisión de las respuestas sin reentrenar el modelo y da más control sobre lo que los usuarios pueden pedir. Buenas prácticas: instrucciones claras y concisas, formato de salida explícito, una respuesta por defecto si el modelo no sabe, iterar probando con un conjunto de prueba reservado y protecciones breves en la plantilla, porque las sobrecargadas encarecen y restan precisión.
- La inyección de prompts intenta anular las instrucciones del desarrollador (secuestro del objetivo), el jailbreak busca saltarse la seguridad nativa del modelo y la filtración intenta exponer el prompt del sistema, lo que facilita otros ataques. El envenenamiento manipula los datos de entrenamiento o de una knowledge base. Se mitigan validando la entrada, controlando quién edita los datos y con Amazon Bedrock Guardrails.
- Amazon Bedrock Prompt Management guarda prompts reutilizables con su modelo y configuración de inferencia. Las variables entre dobles llaves adaptan un prompt a cada caso y las variantes permiten comparar alternativas. Al guardar se actualiza el borrador; para producción se crea una versión, una instantánea fija que la aplicación referencia y que permite cambiar de configuración o volver a una anterior.

Describe the training and fine-tuning process for FMs

- Un FM se preentrena con enormes volúmenes de datos sin etiquetar mediante aprendizaje autosupervisado. Después puede adaptarse: el preentrenamiento continuo usa texto del dominio sin etiquetar, el ajuste fino usa ejemplos etiquetados de entrada y salida para una tarea, y la destilación entrena un modelo alumno más pequeño y barato con las respuestas de un modelo profesor más grande.
- El aprendizaje por transferencia reutiliza un modelo preentrenado y lo ajusta para una tarea relacionada con menos datos. El ajuste fino basado en instrucciones usa pares etiquetados de prompt y respuesta; la adaptación al dominio y el preentrenamiento continuo usan texto del dominio sin etiquetar para aprender jerga y mantener el conocimiento al día sin empezar desde cero.
- Los datos de ajuste fino deben estar limpios, ser representativos y tener un tamaño adecuado: se eliminan duplicados, contenido tóxico e información personal, y las etiquetas conviene que procedan de varios expertos para reducir el sesgo. RLHF entrena un modelo de recompensa con las preferencias humanas, y el ajuste fino por refuerzo usa funciones de recompensa cuando etiquetar sale caro.

Describe methods to evaluate FM performance

- Amazon Bedrock Evaluations ofrece evaluaciones programáticas con conjuntos de datos integrados o propios, evaluaciones con trabajadores humanos para criterios subjetivos, evaluaciones con un LLM como juez y evaluaciones de sistemas RAG. Los benchmarks públicos complementan, pero no sustituyen, a un conjunto de referencia (golden dataset) con prompts y respuestas esperadas de tu propio caso de uso.
- ROUGE mide la coincidencia de n-gramas entre un resumen generado y uno de referencia, y BLEU compara traducciones automáticas con traducciones humanas. BERTScore compara embeddings y capta paráfrasis que ROUGE penaliza. Cuando no hay respuestas de referencia, un LLM como juez puntúa criterios como la utilidad o el tono y explica cada puntuación.
- Para saber si un FM cumple los objetivos de negocio hay que medir una línea base antes del despliegue y vincular cada métrica técnica con una meta de negocio medible, por ejemplo con el marco OGSM. La productividad y la adopción de funciones reflejan el valor y la interacción de los usuarios, y el ROI se sigue de forma continua porque cambian el coste y el uso.
- Una aplicación con FM se evalúa por capas, no solo por su respuesta final. En RAG, Amazon Bedrock Evaluations mide la recuperación (relevancia y cobertura del contexto) y la generación (fidelidad, corrección, citas). En agentes, Amazon Bedrock AgentCore Evaluations puntúa sesiones, trazas y llamadas a herramientas, y en los flujos por lotes priman el rendimiento y el coste por tarea.
- Las métricas de infraestructura indican si una aplicación de IA está en marcha, no si aporta valor. Para alinearla con el negocio se miden la tasa de finalización de tareas y de escalado, la satisfacción del usuario (NPS, CSAT, comentarios) y el coste por interacción o por tarea. Estas métricas se comparan con una línea base, se vigilan con alertas de anomalías y se revisan al cambiar los objetivos.

4. Guidelines for Responsible AI

14–14%

Explain the development of AI systems that are responsible

- AWS define la IA responsable con varias dimensiones: equidad (impacto en distintos grupos), explicabilidad, privacidad y seguridad, seguridad frente a salidas dañinas y mal uso, controlabilidad, veracidad y robustez (salidas correctas incluso ante entradas inesperadas o adversarias), gobernanza y transparencia (que las personas decidan informadas sobre su uso del sistema).
- Amazon Bedrock Guardrails evalúa las entradas y las respuestas con salvaguardas configurables: content filters, denied topics, word filters, sensitive information filters (bloquear o enmascarar PII), contextual grounding checks y Automated Reasoning checks contra alucinaciones. Se aplica al invocar el modelo o con la API ApplyGuardrail sin invocar ningún modelo fundacional.
- Para elegir un modelo de forma sostenible, AWS recomienda valorar primero servicios de IA gestionados y modelos preentrenados, empezar con algoritmos sencillos como referencia y no decidir solo por precisión. En IA generativa ayudan RAG en lugar de reentrenar modelos grandes, el ajuste fino eficiente en parámetros (LoRA), la destilación a modelos más pequeños y la cuantización.
- Los riesgos legales de la IA generativa incluyen reclamaciones de propiedad intelectual por salidas que reproducen contenido protegido, salidas sesgadas, alucinaciones y pérdida de confianza del cliente. La AWS Responsible AI Policy recuerda que las salidas son probabilísticas y que el cliente debe evaluarlas y aplicar supervisión humana en decisiones con consecuencias; además, el cliente debe revisar las licencias de los modelos que usa.
- Un buen conjunto de datos representa a los usuarios y contextos reales del caso de uso, con suficientes ejemplos de cada grupo demográfico para evaluar la equidad; las carencias se corrigen con más datos o con muestreo estratificado. También debe cubrir las variaciones relevantes y documentarse en una ficha (datasheet) con usos previstos, fuentes, método de recogida y sesgos conocidos.

Recognize the importance of transparent and explainable models

- Los modelos sencillos, como los lineales o los árboles de decisión, son interpretables: se ve cómo influye cada entrada en la salida. Los modelos complejos, como las redes neuronales profundas o los modelos fundacionales, no lo son por sí mismos y necesitan herramientas a posteriori como SHAP o LIME, puntuaciones de confianza o citas de fuentes. La explicabilidad permite entender las salidas; la transparencia, decidir con información.
- Amazon SageMaker Model Cards documenta en un único lugar el uso previsto, el riesgo, el entrenamiento y la evaluación de los modelos propios, con versiones inmutables y exportación a PDF. Las AWS AI Service Cards describen usos previstos, limitaciones y decisiones de diseño de los servicios de AWS, pero no sustituyen las pruebas propias, que pueden hacerse con Amazon Bedrock Evaluations.
- Los modelos interpretables, como los árboles de decisión, se entienden con facilidad pero pueden perder precisión, así que hay que comparar su rendimiento con el de los complejos. Más transparencia puede chocar con la privacidad o la seguridad: AWS recomienda cumplir el mínimo de cada criterio, documentar las decisiones y medir la explicabilidad con estudios de usuarios.
- El diseño centrado en las personas permite a los usuarios entender, impugnar y comentar las salidas de la IA, adapta las explicaciones al nivel de experiencia de cada grupo y da a los revisores humanos factores, puntuaciones de confianza y casos similares. En los agentes de Amazon Bedrock, la traza muestra los pasos y el razonamiento que llevan a cada respuesta.

5. Security, Compliance, and Governance for AI Solutions

14–14%

Explain methods to secure AI systems

- AWS protege la infraestructura y el cliente controla sus datos, los permisos de IAM de mínimo privilegio y el cifrado con AWS KMS. Amazon Macie descubre PII en Amazon S3, AWS PrivateLink conecta la VPC con Amazon Bedrock sin pasar por Internet y Guardrails filtra contenido. AgentCore Identity guarda credenciales de los agentes y Policy in AgentCore autoriza sus llamadas a herramientas en el gateway.
- Documentar el origen de datos y modelos permite auditar y reproducir resultados. SageMaker ML Lineage Tracking registra qué datos, trabajos y canalizaciones produjeron cada modelo; AWS Glue Data Catalog indexa la ubicación y el esquema de los datos; y SageMaker Model Cards guarda usos previstos, riesgo y evaluación en versiones inmutables. Bedrock Knowledge Bases añade citas a las respuestas.
- La ingeniería de datos segura empieza por identificar y clasificar los datos confidenciales y minimizarlos con enmascaramiento o tokenización. AWS Glue Data Quality mide la calidad con reglas, AWS Lake Formation limita el acceso por columna, fila y celda, AWS Clean Rooms aplica privacidad diferencial y S3 Object Lock protege la integridad impidiendo sobrescribir o borrar datos.
- La inyección de prompts es un riesgo de la aplicación que corresponde al cliente: valida la entrada y usa el filtro de ataques de prompt de Guardrails, que también filtra toxicidad y enmascara PII en las respuestas. GuardDuty detecta amenazas, Inspector busca vulnerabilidades y el registro de invocaciones de Bedrock, desactivado por defecto, guarda prompts y respuestas para auditoría.
- RAG reduce las alucinaciones al fundamentar las respuestas en datos recuperados en cada consulta. La comprobación de fundamentación contextual de Guardrails puntúa la fundamentación y la relevancia de cada respuesta y bloquea las que quedan bajo el umbral. Las comprobaciones de razonamiento automatizado validan las respuestas con lógica formal frente a reglas y devuelven hallazgos sin bloquear.

Recognize governance and compliance regulations for AI systems

- AWS Artifact ofrece los informes de conformidad de AWS (ISO, SOC, PCI) para los auditores. AWS Config evalúa la configuración de los recursos con reglas y guarda su historial; AWS CloudTrail registra quién hizo cada acción y cuándo; Amazon Inspector analiza vulnerabilidades de forma continua; y AWS Trusted Advisor recomienda mejoras de coste, rendimiento y seguridad.
- La gobernanza de datos define cuánto tiempo se conservan, dónde se procesan y cómo se vigilan. S3 Lifecycle mueve y elimina objetos según reglas, CloudWatch Logs guarda los registros indefinidamente salvo que se fije una retención y puede enmascarar datos confidenciales, los perfiles geográficos de inferencia mantienen el procesamiento dentro de una zona y CloudWatch observa las cargas de IA generativa.
- La Generative AI Security Scoping Matrix clasifica los casos de uso en cinco ámbitos, de la aplicación de consumo al modelo entrenado desde cero, y orienta los controles de cada uno. La política de IA debe fijar evaluaciones periódicas frente a datos de referencia, también para modelos ajustados. Las AWS AI Service Cards aportan transparencia, y cada cliente debe formar a su propio personal.