Eusko Jaurlaritzaren
Informatika Elkartea

Sociedad Informática
del Gobierno Vasco

EXPEDIENTE Nº: EJIE-065-2023

Contratación de servicios de Asistencia Técnica Calidad Osakidetza.

Pliego de condiciones técnicas

Fecha: Mayo 2023

Mediterráneo, 14 • 01010 Vitoria-Gasteiz • Posta-kutxatila / Apartado: 809 • 01080 Vitoria-Gasteiz • Tel. 945 01 73 00* • Fax. 945 01 73 01 • www.ejie.eus

Este documento es propiedad de Eusko Jaurlaritzaren Informatika Elkartea – Sociedad Informática del Gobierno Vasco, S.A. (EJIE). Este documento no puede ser reproducido, en su totalidad o parcialmente, ni mostrado a otros, ni utilizado para otros propósitos que los que han originado su entrega, sin el previo permiso escrito de EJIE. En el caso de ser entregado en virtud de un contrato, su utilización estará limitada a lo expresamente autorizado en dicho contrato. EJIE no podrá ser considerada responsable de eventuales errores u omisiones en la edición del documento.

Versión	Fecha	Resumen de cambios	Elaborado por:	Aprobado por:
1.0	Mayo 2023	Versión Inicial	Javier Pérez Toral	Ander Martinez Juan Jose Villa

Índice

1 Introducción 4

2 Objeto del contrato 5

2.1 Objeto del contrato 5
2.2 Requisitos de los servicios de Asistencia técnica 5

3 Configuración del Servicio 7

3.1 Organización del Equipo de Trabajo 7
3.2 Composición del equipo de trabajo 9
3.3 Formación 9
3.4 Transferencia Tecnológica 9
3.5 Control y Seguimiento 10

4 Metodología aplicable y Entorno Tecnológico 11

4.1 Especificaciones para Desarrollo 11
4.1.1 Arquitectura de Referencia 11
4.1.1.1 Vista lógica 12
4.1.1.2 Directrices de Desarrollo 13
4.1.1.3 Construcción de Aplicaciones por Capas 13
4.2 Directrices tecnológicas de Infraestructuras 17
4.2.1 Entorno tecnológico 17
4.2.2 Especificaciones del Cliente (Maqueta) 20
4.2.3 Dispositivos de entrada/salida 21
4.2.4 Seguridad, acceso, autenticación y autorización 22
4.2.5 Monitorización 23
4.2.6 Backup/Restore 24
4.3 Interoperabilidad 24
4.3.1 Servicios Web 25
4.3.2 14.3.2 Gestor de Eventos 26
4.4 Herramientas del ciclo de vida de las aplicaciones 28

Anexo 1. Asistencia Técnica Calidad Osakidetza 30

1 Introducción

EJIE, Eusko Jaurlaritzaren Informatika Elkartea – Sociedad Informática del Gobierno Vasco, es la Empresa pública de servicios de las tecnologías de la información y las comunicaciones (TIC), cuya razón de existir es contribuir a la consecución de un Sector Público Vasco, moderno y eficiente, en el Marco Legal establecido por el Gobierno, con la seguridad y calidad necesarias y con el debido respeto al medio ambiente.

EJIE tiene como misión ser el órgano gestor tecnológico del Gobierno Vasco que facilita la digitalización de los servicios que el sector público vasco presta a la ciudadanía, garantizando la calidad, seguridad y continuidad de las tecnologías de la información y la comunicación que los soportan, alcanzando resultados satisfactorios en relación a las entidades cliente de los servicios y sus personas usuarias.

EJIE tiene como meta final la consecución de la satisfacción de sus clientes, siendo el instrumento común de prestación de servicios TIC en el Sector Público Vasco, y comprometiéndose en:

- Construir y mantener con eficiencia y calidad la infraestructura de los Sistemas de Información, posibilitando su continuidad y seguridad.
- Garantizar la interoperabilidad entre las distintas administraciones.
- Servir de apoyo a las necesidades de planificación y realización de la función informática del sector público del Gobierno Vasco, asegurando la cobertura de sus demandas con el compromiso y profesionalidad adecuados a las relaciones contractuales que se establezcan.

Se puede obtener información más detallada y extensa en nuestra dirección de Internet http://www.ejie.eus

Por tanto, EJIE es la referencia para la prestación de servicios TIC en el Sector Público Vasco:

- Aportando valor añadido.
- Proporcionando soluciones competitivas.
- Transmitiendo confianza a sus clientes.
- Contando con personas cualificadas y comprometidas.

Con fecha 16 de agosto de 2016 se firmó el Acuerdo entre el Ente Público de Derecho Privado Osakidetza y Eusko Jaurlaritzaren Informatika Elkartea—Sociedad Informática del Gobierno Vasco, S.A. (EJIE) para la integración del servicio de infraestructuras y operaciones (I.O.) de OSAKIDETZA con el análogo servicio (I.O.) en EJIE, en el marco del acuerdo adoptado por el Consejo de Gobierno sobre el proceso de convergencia en materia de tecnología de la información y comunicación –TIC/ICPS el 27 de julio de 2015 y del Acuerdo de 21 de junio de 2016 relativo a dicho proceso de convergencia por el que se aprueba el documento ejecutivo “plan general de actuación en materia de convergencia TIC/ICPS”.

La actividad objeto de contratación, es necesaria para la prestación de los servicios, competencia de EJIE S.A., y está incluida en los servicios que pueden ser objeto de contratación (según Instrucción Sexta de la Instrucciones sobre buenas prácticas en la contratación de servicios de Gobierno Vasco). Su prestación no puede ser asumida con los recursos humanos y técnicos de que dispone EJIE S.A., siendo además inviable su reorganización.

El presente pliego, como el resto de los documentos de la fase de preparación del contrato, incorporan las debidas garantías para dar cumplimiento a la Instrucción octava de las Instrucciones sobre buenas prácticas en la contratación de servicios de Gobierno Vasco.

En este ámbito, el Osakidetza encomienda a EJIE los **servicios de asistencia técnica profesional necesaria** para asegurar el correcto funcionamiento, **calidad** y evolución de los sistemas de información que dan soporte a los procesos de gestión dentro de su ámbito competencial.

2 Objeto del contrato

2.1 Objeto del contrato

La finalidad de la contratación es dotar a EJIE de los servicios de asistencia técnica profesional necesarios para asegurar **la calidad** de los sistemas de información que dan soporte a los procesos de Osakidetza.

La necesidad de contratación se enmarca en el servicio de Asistencia Técnica prestado por EJIE a dichos Departamentos, Organismos Autónomos y Entes Públicos de derecho privado del Gobierno Vasco, gestionados por el área técnica Educativa Socio-Sanitaria de EJIE que tiene como objetivo proporcionarles la asesoría y el apoyo técnico especializado preciso para la definición de planes y proyectos informáticos, así como para el desarrollo, mantenimiento e implantación de los Sistemas de Información. A través de este servicio, EJIE también ayuda a la Dirección de Tecnologías de la Información y la Comunicación para velar por el cumplimiento de los estándares y la adecuada implantación de las aplicaciones.

Para la prestación de los servicios de Asistencia Técnica, EJIE requiere contratar servicios profesionales de soporte técnico especializado en tecnologías de la información.

El **entorno de referencia** para los que se requiere el servicio de asistencia técnica se establece en el **Anexo 1** recogido en este documento.

2.2 Requisitos de los servicios de Asistencia técnica

Para la prestación del servicio de Asistencia Técnica EJIE requiere de la contratación de servicios profesionales de proveedores externos, aunque siempre sujeto a la decisión, control y seguimiento de personal de EJIE. Los servicios requeridos contemplan la realización de las siguientes tareas:

- **Soporte técnico informático especializado "bajo demanda"** en las actividades de aseguramiento de calidad del software. Este modelo bajo demanda aplica al resto de tareas definidas.
- Siendo las pruebas una de las actividades de aseguramiento de calidad del software, el cometido inicial también será la **creación de un modelo de referencia** que marcará la estrategia a seguir para verificar, validar y asegurar que el software de Osakidetza cumple los requerimientos especificados, y las necesidades y expectativas del usuario, es decir, asegurar la calidad del software. Dicho modelo será el Modelo de Pruebas de Osakidetza.
- **Soporte técnico a pruebas de calidad de software,** teniendo en cuenta los siguientes aspectos:
  - Alcance de las pruebas: cual es el objeto de la prueba, es decir, aplicaciones, módulos, funcionalidad, sistemas, subsistemas, interoperabilidad, alta disponibilidad, que serán objeto de prueba, según el caso que se plantee: correctivo, nueva funcionalidad, evolución tecnológica de plataformas/productos comerciales de software, evolución/sustitución de infraestructuras o puesto de trabajo. Cuando sea necesario, la empresa adjudicataria deberá **concertar** el alcance de las pruebas con el desarrollador de los módulos correspondientes o el titular de los derechos de propiedad intelectual del producto.
  - Tipos de Pruebas: la diversidad de software requiere, según el caso, realizar auditorias de código, así como la realización completa o en parte de los siguientes tipos de pruebas:
    - Pruebas funcionales. para probar que el software desarrollado cumple con las funciones específicas para las que ha sido creado.
    - Pruebas no funcionales. referentes a rendimiento, escalabilidad, seguridad, alta disponibilidad y continuidad de negocio, usabilidad, etc.
    - Pruebas de integración: para comprobar la integración del software globalmente, verificando el funcionamiento correcto de las interfaces entre los distintos subsistemas que lo componen y con el resto de los sistemas de información con los que se comunica.
    - Pruebas de aceptación: pruebas para asegurar que un sistema cumple con el funcionamiento esperado y permitan al usuario final determinar su aceptación, desde el punto de vista de funcionalidad y de rendimiento.

    - Pruebas de regresión: para evaluar que el sistema continúa operando de manera correcta y de acuerdo con lo especificado después de introducir una nueva funcionalidad o de realizar un cambio al código.
  - Análisis, diseño y planificación de las pruebas: Detalle de los niveles de pruebas y tipos de pruebas a realizar, métricas, herramientas que se van a utilizar, qué se automatiza y por qué, qué y cómo serán tratados aquellos elementos que no se pueden probar; definición del entorno de ejecución de las pruebas detallando los recursos hardware y software necesarios, criterios de inicio, suspensión y finalización de las pruebas, posibles riesgos, equipo de pruebas con sus roles y responsabilidades, estimación de todas las actividades a realizar acotadas en tiempo y recursos asociados a las mismas, relación de entregables, …
  - Ejecución y evaluación de resultados: actividades para la ejecución de las pruebas (preparación del entorno, juegos de datos, relación de métricas, …); detección y registro de defectos o incidencias encontrados, así como la evaluación del software en base a los resultados y posibles recomendaciones de mejoras. Seguimiento y control de estos consensuado con EJIE.
  - Estandarización y automatización de los procesos y tareas fruto de la aplicación de la metodología y estrategia de pruebas, de modo que el trabajo realizado sea fácilmente reutilizable en otros casos o repeticiones de estas.
  - Creación de un catálogo de servicios para para ser consumido por los clientes con entradas y salidas perfectamente definidas.
  - Proporcionar las **herramientas** y su gestión, necesarias para realizar todas las tareas que se especifiquen en la metodología y estrategia de pruebas, así como las requeridas para soportar la propia Calidad de Software, objeto del expediente, siendo preferente el uso de software libre y la valoración de las herramientas que usa EJIE para otros Departamentos. La decisión final de la mismas debe ser consensuada con EJIE.
  - Revisión y mejora continua: Se deberá contemplar una revisión periódica tanto de la metodología como de la estrategia de pruebas para determinar mejoras sobre las mismas, adaptarse y adoptar evoluciones tecnológicas futuras de todos aquellos elementos que las componen, así como las del software de entorno de negocio/funcional de referencia.
- Soporte al personal de EJIE para las actividades de **gestión de configuración, cambios y despliegue de aplicaciones** en los distintos entornos de despliegue de aplicaciones. El personal de EJIE recibe de los equipos de mantenimiento o proyectos de sus clientes las modificaciones o necesidades de redespliegue del software de las distintas aplicaciones de negocio. Para su despliegue se requieren servicios de asistencia técnica para su correcta configuración y parametrización en producción. Asimismo, es habitual hacer modificaciones en dicha configuración y despliegue para la actualización del software.

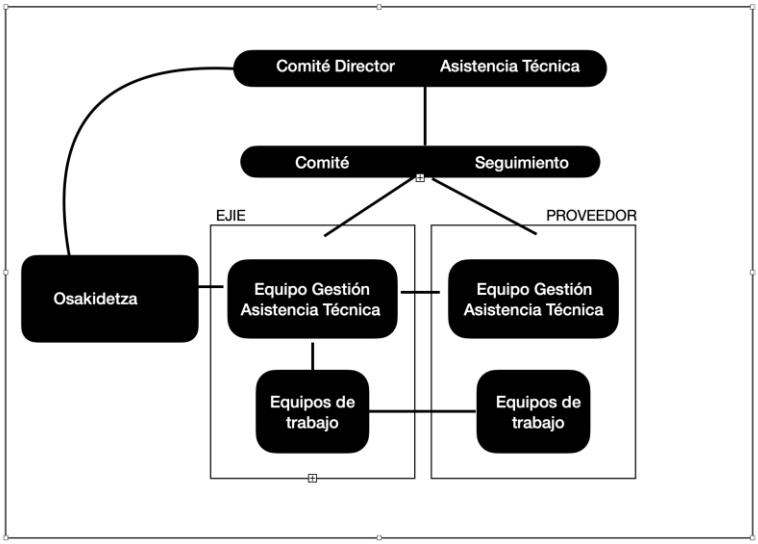
3 Configuración del Servicio

3.1 Organización del Equipo de Trabajo

La empresa licitadora deberá describir en su Documento de Propuesta Técnica:

- La organización (perfiles) del equipo de proyecto asignado a la realización de las actividades resultantes del presente pliego, así como
- Las funciones de estos.

No obstante, se detalla a continuación una posible aproximación de modelo organizativo:



COMITÉ DIRECTOR ASISTENCIA TÉCNICA

Ostenta la máxima autoridad en el proyecto y representa a las organizaciones involucradas en el mismo, EJIE y la empresa adjudicataria. A este comité se incorporarán las personas que en cada momento estime oportuno el propio Comité, principalmente de las áreas representadas en el servicio.

Participantes:

- EJIE:
  - Dirección de Área de EJIE o director técnico
    - Jefe/a de proyecto responsable Osakidetza
- OSAKIDETZA
  - Jefe/a de servicio responsable
- Empresa adjudicataria:
  - Director/a de Proyecto de la empresa adjudicataria en representación de la Dirección
  - Responsable del proyecto/servicio

Periodicidad de las reuniones del Comité:

- Trimestral. Deberá quedar constancia de un acta de reunión y de los informes facilitados por la empresa adjudicataria.

Roles y responsabilidades:

- Liderar y apoyar la consecución de los objetivos fundamentales.
- Resolver los conflictos o problemas que, por cualquier razón, no puedan hacerse desde el comité de Seguimiento.
- Definir o aprobar los indicadores de control cuyo seguimiento se va a realizar.
- Evaluar y aprobar, en su caso, las posibles variaciones de alcance, plazos, coste y requerimientos que pudiesen surgir
- Asegurar la disponibilidad de los medios y recursos necesarios para la ejecución del proyecto.
- Seguimiento exhaustivo de la prestación del servicio de Asistencia Técnica, evaluando y aprobando las mejoras a aplicar.

COMITÉ DE SEGUIMIENTO

Representa la máxima autoridad en relación a la operativa del proyecto, realizando un seguimiento del trabajo (planificaciones y recursos), de las problemáticas surgidas, escalando los aspectos que se consideren que sobrepasan los límites de actuación establecidos.

Participantes:

- EJIE:
  - Jefe/a de proyecto responsable Osakidetza
  - Otro personal de EJIE que, a solicitud del Jefe de Proyecto, fuera requerido incluir en este comité.
- OSAKIDETZA:
  - Jefe de proyecto responsable
- Empresa adjudicataria:
  - Responsable del proyecto/servicio

Periodicidad de las reuniones del Comité:

- Mensual. Deberá quedar constancia de un acta de reunión y de los informes facilitados por la empresa adjudicataria

Roles y responsabilidades:

- Seguimiento permanentemente del servicio.
- Seguimiento del plan de formación establecido para el servicio.
- Gestión y seguimiento de los trabajos encargados
- Agilizar la toma de aquellas decisiones que, por su naturaleza, pudieran exceder de las competencias del Equipo de Trabajo.
- Elevar al comité de dirección las decisiones o incidencias que consideren necesarias.
- Elaborar los Informes de progreso con el fin de informar del desarrollo del servicio

EQUIPOS DE TRABAJO DE LA EMPRESA ADJUDICATARIA

Para la realización de los servicios especificados en el Objeto del contrato, el adjudicatario dispondrá de un equipo de trabajo, que estará formado por perfiles con capacidad, experiencia, y conocimientos necesarios para el correcto desempeño de su trabajo.

Participantes:

- Con carácter obligatorio para todos los casos:
  - Responsable del servicio
- Con carácter opcional en función de la demanda concreta del ámbito de la aplicación, el equipo estará compuesto por los siguientes perfiles:
  - Jefe/a de proyecto / Consultor de testing
  - Técnicos de testing

NO existen necesidades iniciales de servicio, **es un servicio bajo demanda**, que se solicitaran por parte de EJIE, de acuerdo con las necesidades que se den en cada momento, siempre en el ámbito de las actividades planteadas en este pliego de condiciones técnicas y de acuerdo a las demandas Osakidetza.

Para cada lote, cada licitador deberá proponer un **responsable de servicio.** Este responsable de relación será **una persona** de su organización **encargada** de la gestión del contrato adjudicado al amparo del presente expediente. De modo adicional dicha persona responsable deberá ejercer las facultades de dirección, organización y disciplinarias en lo referente a las personas que presten los servicios contratados, así como de gestionar y coordinar la actividad preventiva, en base a los procedimientos establecidos por EJIE en www.ejie.eus. En caso de incumplimiento de cualquiera de estas obligaciones, EJIE se reserva el derecho de resolver el contrato.

3.2 Composición del equipo de trabajo

El equipo de trabajo propuesto estará formado por personal técnico con categoría profesional y nivel de especialización adecuados a la solvencia requerida, y a los servicios especificados en este pliego de condiciones técnicas.

La totalidad del equipo presentado, deberá estar disponible para la incorporación de forma inmediata a la formalización del pedido/contrato por parte de EJIE.

3.3 Formación

Es objeto del Servicio de Asistencia Técnica de EJIE a su clientela, ayudar a la Dirección de Tecnologías de la Información y la Comunicación para velar por el cumplimiento de los estándares y la adecuada implantación de las aplicaciones. Para poder realizar las tareas de asistencia especificadas en el objeto de este contrato, es responsabilidad del adjudicatario garantizar la formación adecuada del personal del mismo asignado al equipo de trabajo.

En el inicio de la prestación del servicio, se establecerá con la empresa adjudicataria en el Comité de Seguimiento un **plan de formación específico**. Además, en todos los casos, de forma **obligatoria**.

En este sentido, la empresa adjudicataria deberá acreditar con carácter general la **realización de al menos 30 horas de formación anuales por cada persona miembro del equipo de trabajo**, contabilizando dentro de dichas horas aquellas que se realicen en los ámbitos establecidos de forma específica al comienzo de la prestación del servicio en el Comité de Seguimiento.

Deberá proponerse por parte del adjudicatario un **plan** de las acciones formativas. Dicho proceso deberá ser **concretado en el primer mes de la prestación del servicio** y podrá ser sometido a modificaciones a propuesta tanto del adjudicatario del contrato como de la persona responsable del servicio objeto de licitación en EJIE, teniendo en cuenta las necesidades que se plantean en cada momento en Osakidetza.

Cualquier coste que pudiera derivarse de las acciones formativas del personal asignado al servicio demandado será responsabilidad del proveedor y no serán imputables a EJIE.

3.4 Transferencia Tecnológica

Durante la ejecución de los trabajos objeto del contrato el adjudicatario se compromete, en todo momento, a facilitar a las personas de EJIE designadas por la persona responsable del servicio a prestar en EJIE (Jefe de Proyecto), y a tales efectos, la información y documentación que ésta solicite para disponer de un pleno conocimiento de los trabajos desarrollados, así como de los eventuales problemas que puedan plantearse y de las tecnologías, métodos, y herramientas utilizados para resolverlos.

Así mismo el adjudicatario y antes de la finalización del servicio, se compromete a realizar las actividades formativas, que garanticen la continuidad del servicio, al equipo de EJIE u Osakidetza que pudiera dar

soporte a este servicio en cada momento, tras la finalización del servicio al que se refiere el presente pliego, según las directrices marcadas por EJIE.

3.5 Control y Seguimiento

El propósito de las tareas de control y seguimiento es el de proveer una visión objetiva del estado actual de la prestación del servicio y determinar las posibles desviaciones a fin de aplicar las acciones correctoras que sean necesarias. Es en este sentido en el cual llamamos seguimiento a la evaluación rutinaria del estado, en tanto que llamamos control a la toma de los valores que definen el estado.

Los productos mínimos de esta práctica son los informes de seguimiento, un documento o artefacto, donde anotamos los resultados de la evaluación de una iteración de control; de momento sin decir las correcciones a tomar.

Una vez identificadas las desviaciones, es necesario decidir oportunamente las correcciones requeridas, llevándolas a cabo en el momento en que sea necesario. Finalmente es importante que la corrección planteada sea a su vez, objeto de seguimiento, lo que implica que la planificación debe ser actualizada para que refleje las acciones que se han determinado necesarias para corregir la desviación.

A la hora de realizar las reuniones de seguimiento es conveniente calcular previamente las medidas o métricas que aplicadas al servicio objeto de la contratación sirvan de indicador sobre el estado del mismo. Esto con el objeto de obtener una evaluación lo más completa y objetiva posible de la prestación del servicio.

Así, bajo esta perspectiva, la empresa licitadora deberá describir en su Documento de Propuesta Técnica:

- El modelo de control y seguimiento a aplicar durante la prestación del servicio, incluyendo las reuniones de seguimiento a celebrar, su objetivo, su periodicidad, los/las asistentes necesarios/as, los controles que deben ser ejecutados, etc.
- Los informes de seguimiento que deberán obtenerse en cada fase del servicio, y para cada solicitud de trabajo realizada. Se detallará su objetivo y al menos parcialmente, su contenido.
- Los indicadores de calidad del servicio a medir, y su criticidad asociada.
- Los métodos de aplicación de las posibles acciones correctoras

Previamente al inicio de la prestación del servicio se concretará por el responsable del servicio de EJIE cuáles serán las tareas de Control y Seguimiento que deben realizarse, los entregables a generar y la periodicidad de las reuniones a celebrar. Aun así, **deberá al menos existir una reunión de seguimiento con carácter mensual del Comité de Seguimiento,** quedar constancia de la misma con el consecuente informe de seguimiento y acta de reunión asociada.

4 Metodología aplicable y Entorno Tecnológico

La organización del trabajo y la prestación del servicio **de Asistencia Técnica**, tal y como se ha mencionado con anterioridad, el nivel de cumplimiento de los mismos dentro de los servicios a prestar deberá concretarse en las fases iniciales del proyecto. Aun así, las ofertas presentadas deberán tomar como referencia dichos modelos y metodologías, que se explican brevemente a continuación.

El entorno tecnológico de referencia será el establecido en los estándares de Osakidetza. Se deberán aplicar por tanto aquellos que se consideren necesarios para la ejecución del servicio, teniendo en cuenta como referencias las características específicas de las aplicaciones incluidas en el Anexo 1 del presente pliego de condiciones técnicas.

La empresa adjudicataria se compromete a adoptar y soportar en todo momento cualquier modificación o adaptación del entorno tecnológico establecido por Osakidetza e incluido dentro de sus estándares, así como todas aquellas otras actualizaciones específicas impuestas por el responsable del Área Funcional.

4.1 Especificaciones para Desarrollo

Este apartado refleja los principales aspectos del diseño técnico de la arquitectura de referencia para aplicaciones web, definida por Osakidetza.

Una arquitectura de referencia es un conjunto de patrones diseñados y probados para ser usados en un contexto específico, en nuestro caso, en los sistemas de misión crítica de Osakidetza. Es una guía de ayuda para ser usada por los arquitectos y desarrolladores, que permite conceptualizar nuevas aplicaciones rápidamente reutilizando un diseño suficientemente probado.

La arquitectura de referencia refleja las decisiones y aspectos técnicos de diseño basados en la experiencia de Osakidetza en el desarrollo, mantenimiento y evolución de aplicaciones web. El principal propósito de la arquitectura de referencia para aplicaciones web es, por lo tanto, homogeneizar el desarrollo, despliegue y explotación en producción de las aplicaciones.

En este sentido, la implementación de la arquitectura de referencia aportará los siguientes beneficios perseguidos por Osakidetza:

- Mejorar el rendimiento de las aplicaciones.
- Garantizar la escalabilidad y la tolerancia a fallos.
- Facilitar las integraciones entre sistemas.
- Aprovechar las tecnologías estándares y capacidades proporcionadas por los recursos hardware y software existentes en la organización.
- Reducir el coste del mantenimiento (evolutivo y correctivo) mediante la simplificación del diseño de la solución y la implementación del mismo.

4.1.1 Arquitectura de Referencia

Una de las características generales de los proyectos con éxito es que existe un diseño previo bien fundamentado que define la arquitectura general del sistema, las soluciones de infraestructura y plataforma (por ejemplo, alta disponibilidad, recuperación frente a desastres, copia de seguridad y restauración, monitorización), y realiza un mapeo de requerimientos a elementos de la arquitectura.

Desde el punto de vista de sistemas de información, este requerimiento estratégico debe tener una base de arquitectura y diseño previo que permita cumplirlo durante la evolución del ciclo de vida.

Los principios de la arquitectura de referencia, describen los principios fundamentales que se han considerado para el diseño de dicha arquitectura. Los principios representan los preceptos o reglas básicas que guían el proceso de diseño.

La arquitectura de referencia se describe usando tres vistas diferentes:

- Vista lógica o conceptual, en la que se presenta la arquitectura mediante una visión de alto nivel que permite entender los diferentes componentes que intervienen y las interacciones existentes entre ellos.
- Vista de implementación, en la que se proporcionan detalles adicionales sobre la tecnología y productos para la construcción de los diferentes componentes.
- Vista de despliegue, en la que se describe de forma pormenorizada la información del sistema relativa a la instalación y explotación que se realiza por entornos.

4.1.1.1 Vista lógica

El propósito de la vista lógica es permitir un entendimiento de alto nivel de la arquitectura de referencia y de las diferentes capas y componentes en las que está estructurada. También se define la integración con otros sistemas.

Sobre esta vista lógica se deberían plasmar todos los elementos que deben ser considerados cuando se diseña una nueva aplicación.

Las aplicaciones deben de tomar este diseño como referencia e indicar:

- Las capas de la arquitectura de referencia que se implementarán
- Servicios externos con los que interactúa la aplicación
- Clientes que usarán la aplicación
- Servicios de persistencia de datos

En la siguiente figura se muestra la vista lógica de la arquitectura de referencia para aplicaciones web.

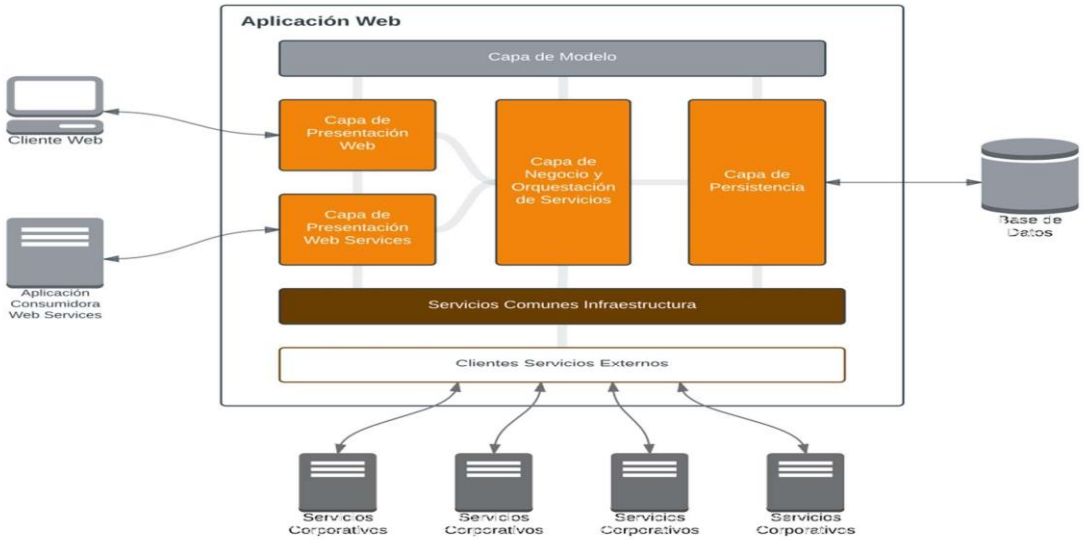


Figura 1. Vista lógica de la arquitectura de referencia

4.1.1.2 Directrices de Desarrollo

Se contemplan como directrices de desarrollo, las normativas, estándares y recomendaciones para la elaboración de un código fuente homogéneo, estandarizado y de calidad, con el objeto de minimizar las tareas de mantenimiento y la deuda tecnológica.

También se incorporan las especificaciones para la obtención de sistemas de información seguros, con un rendimiento óptimo y adaptado a las necesidades de las tecnologías definidas por las Arquitecturas de Software que se tomarán como referencia en los desarrollos.

A continuación, se enumeran algunas de las directrices de desarrollo que cumplen las soluciones desarrolladas para Osakidetza:

4.1.1.3 Construcción de Aplicaciones por Capas

El principal objetivo de la construcción de aplicaciones por capas, es la separación de

la lógica de negocio, de la lógica de diseño. Su principal ventaja, es que el desarrollo se puede llevar a cabo en varios niveles y, en caso de que sea necesario llevar a cabo algún cambio, sólo se necesita intervenir al nivel requerido.

El ejemplo clásico de este tipo de práctica y el que se contempla a nivel de directriz, es el consistente en separar las aplicaciones en tres capas: Presentación, Negocio y Persistencia.

Por lo tanto, las aplicaciones serán desarrolladas en capas; es decir, se diseñará las aplicaciones separando entre los modelos de datos, la lógica de aplicación, y la interfaz de usuario. Si optamos por esta separación de responsabilidades, podremos compartir totalmente el modelo de datos y sólo tendremos que adaptar la interfaz de usuario.

Por cada capa se consideran una serie de directrices de desarrollo basadas en el modelo de tres capas propuesto por la Arquitectura de Referencia de Osakidetza.

Capa de Presentación

Mediante la capa de presentación se separa la interacción del usuario respecto a la lógica de negocio.

El uso de la arquitectura en tres capas en el desarrollo de aplicaciones ha favorecido la aparición de tecnologías que facilitan la implantación de esta capa, además de un conjunto de buenas prácticas que mejoran el proceso de su implementación.

Directrices de Desarrollo para la Capa de Presentación:

Finalidad: Presentar la información al usuario.

La capa de presentación, también llamada "capa de usuario", deberá presentar el sistema al usuario, comunicar la información necesaria y capturar la misma en un proceso que realiza un filtrado para validar los datos respecto al formato. No deberá procesar datos ni tomar decisiones.

Esta capa se comunica únicamente con la capa de negocio. También es conocida como interfaz gráfica y debe tener la característica de ser "amigable" (comprensible y fácil de usar) para el usuario.

Modularidad: Controlar la relación entre las vistas.

En una vista sólo debe hacerse referencia a otras vistas relacionadas con su funcionalidad. En ejecución, se recomienda integrar la vista con un sistema de plantillas, para poder aumentar de esta manera la reusabilidad y el encapsulamiento de funcionalidades.

Internacionalización y localización: Preparar las aplicaciones para diferentes idiomas y convenciones.

Las aplicaciones estarán preparadas para que puedan adaptarse a diferentes idiomas y convenciones (formatos de fecha, moneda, etc.), sin necesidad de realizar cambios de relevancia en el código.

Validaciones: Realizar validaciones sobre los datos introducidos por los usuarios.

La vista será la responsable de la validación inicial de los datos introducidos por el usuario. La validación debe ser obligatoria sólo para el caso de campos y formato. Nunca deberá realizarse una validación de negocio desde esta capa.

Catálogo de controles: Elaborar un catálogo con la lista de controles.

Se especificará, si es posible, la función del control, los posibles validadores/conversores que se le puedan asociar, los eventos que pueda lanzar y los atributos que puedan cambiarse para manipular su aspecto externo.

Los controles deberían permitir el tratamiento de componentes y eventos que posibiliten extraer de la vista toda la lógica de interfaz. La consecuencia principal de este enfoque es que desde la vista nunca se manipularán los controles. El objetivo es que la construcción de la vista sea una tarea totalmente autónoma del resto de componentes de la función de negocio.

Uso de plantillas: Facilitar el mantenimiento haciendo uso de plantillas.

Se debe utilizar un framework de plantillas (templates) en la capa de presentación, ya que de esta manera se facilita el mantenimiento del diseño de los sistemas de información. Existen frameworks que posibilitan aislar el diseño de la aplicación en un único fichero de plantilla (o más, si se necesitan). Esto facilita la tarea de hacer modificaciones en el diseño.

Controles de interfaz: Utilizar un identificador único para los controles de interfaz.

Todos los controles de interfaz deben mantener un identificador único para facilitar su compresión, reutilización y batería de pruebas de este.

Independencia entre capas: Evitar el acoplamiento entre distintas capas.

Cada capa debe tomar la responsabilidad que le corresponde. En este caso, debe evitarse que la capa de presentación realice atribuciones que le correspondan a otras capas.

Capa de Negocio

La capa de negocio expone la lógica necesaria a la capa de presentación para que el usuario, a través de la interfaz, interactúe con las funcionalidades de la aplicación.

Se utilizarán componentes de negocio para abstraer la lógica de negocio de otros problemas generales de las aplicaciones, como la concurrencia, transacciones, persistencia, seguridad, etc…

Directrices de Desarrollo para la Capa de Negocio:

Reglas de negocio: Implementar las reglas de negocio en esta capa.

Los módulos que requieran una funcionalidad deberán encontrarla en un solo lugar y única versión. No debe haber réplica de funcionalidades en ninguna de las otras capas (Persistencia y/o Presentación)

Validaciones de datos: Garantizar el valor correcto de los datos.

Esta capa debe garantizar que los datos requeridos para procesarla hayan sido debidamente validados, y sólo se puede iniciar el flujo del proceso de negocio si la validación resultó correcta.

Comunicación entre capas: Definir la estrategia de comunicación entre capas.

La comunicación entre capas se debe establecer mediante objetos del modelo. Se crearán entidades sin métodos, sólo tendrán propiedades, campos y atributos, que nos servirán para almacenar información, como puede ser la asociación de las propiedades a las columnas de la base de datos.

Transacciones: Tratar convenientemente las transacciones de los datos.

La capa de modelo de datos proporciona los datos al sistema, pero las transacciones que involucren a dichos datos, se deben manejar desde la capa de negocio.

Excepciones: Encapsular en la capa de negocio las excepciones y trasladarlas de forma correcta a la capa de presentación.

El objetivo es simplificar el manejo de excepciones de la aplicación. Se recomienda tener bien definida una política para el tratamiento de excepciones y disponer de una jerarquía propia de excepciones. Como norma general, las aplicaciones no deberán nunca extender las excepciones del sistema, sino las de su propia jerarquía de clases.

Servicios encapsulados: Encapsular la capa de negocio en servicios.

Se debe encapsular la capa de negocio en servicios, para de esta forma aislar la base de datos respecto de una interacción directa con el usuario. Estos servicios de negocio conforman un "puente" entre el usuario y los datos. Responden a peticiones de usuarios u otros servicios de negocio aplicando procedimientos formales y reglas de negocio a datos relevantes.

Acceso a servicios: Controlar el acceso a los servicios en la capa de negocio.

El control de acceso al servicio de negocio debe hacerse en la capa de negocio, puesto que podemos tener distintas capas de presentación, como, por ejemplo, una capa de presentación web y una capa de presentación Web Services.

El control de acceso puede realizarse tanto a nivel de método, dentro de cada servicio, como a nivel de servicio vertical.

Inyección de Dependencias: Uso del patrón de diseño ‘Inyección de Dependencias’

(DI), como forma de ‘Inversión de control’ (IoC)

La Inversión de Control (Inversion of Control en inglés, IoC) es un método de programación en el que el flujo de ejecución de un programa se invierte respecto a los métodos de programación tradicionales. En la Inversión de Control se especifican respuestas deseadas a sucesos o solicitudes de datos concretas, dejando que algún tipo de entidad o arquitectura externa lleve a cabo las acciones de control que se requieran en el orden necesario y para el conjunto de sucesos que tengan que ocurrir. Es el principio subyacente a la técnica de Inyección de Dependencias, siendo términos frecuentemente confundidos.

La Inyección de Dependencias (Dependency Injection en inglés, DI) es un patrón de diseño orientado a objetos, en el que se suministran objetos a una clase en lugar de ser la propia clase quien cree el objeto. La forma habitual de implementar este patrón es mediante un "Contenedor DI" y objetos planos o simples, por ejemplo, los llamados POJO. El contenedor inyecta a cada objeto, los objetos necesarios según las relaciones plasmadas en un fichero de configuración. Típicamente este contenedor es implementado por un framework externo a la aplicación.

La implementación de un patrón de diseño como la 'Inyección de Dependencias' permite gestionar de una manera eficiente las dependencias entre los diferentes componentes de una aplicación, sin tener que recurrir a un acoplamiento no deseable entre estos.

Capa de Persistencia

La necesidad de vincular los datos guardados en una base de datos relacional, con los objetos de una aplicación orientada a objetos, determina la aparición del concepto de persistencia de objetos. Siguiendo el paradigma de desarrollo en tres capas, la persistencia queda recogida en su propia capa, separada de la lógica de negocio y de la interfaz de usuario.

La persistencia de la información es probablemente la parte más crítica en una aplicación de software y permite ejecutar operaciones en el almacenamiento persistente (base de datos). El modelo de objetos de un sistema difiere en muchos aspectos del modelo relacional. La interfaz que une esos dos modelos se llama asociación objetorelacional (ORM en inglés). La capa de persistencia encapsula el comportamiento necesario para mantener los objetos.

Directrices de Desarrollo para la Capa de Persistencia:

Asociación Objeto-Relacional: Usar un mapeador Objeto Relacional para implementar la capa de persistencia.

Se debe utilizar un mapeador Objeto-Relacional para implementar la capa de persistencia, ya que es una técnica que permite convertir los tipos de datos utilizados en un lenguaje de programación orientado a objetos y los utilizados en una base de datos relacional, lo que posibilita el uso de las características propias de la orientación a objetos.

Uso del patrón DAO: Crear una clase DAO por cada objeto de negocio del sistema

El patrón CRUD, reconocido como el patrón más importante del acceso a datos, indica que cada objeto debe ser creado en base de datos para que sea persistente. Para esto es necesario asegurar que existen operaciones que permiten a la capa inferior (de acceso a datos) leerlo, actualizarlo o borrarlo.

Mediante la implementación de DAO's pueden proporcionarse las operaciones CRUD necesarias para cada aplicación. No es obligatorio que cada DAO implemente todas las operaciones CRUD (puede no ser necesario en la lógica funcional del sistema). Por lo tanto, se recomienda crear un DAO distinto por cada objeto de negocio en el sistema.

Manejo de la caché: Usar la caché en las aplicaciones para reducir tiempos de lectura.

Se recomienda utilizar un sistema de caché en las aplicaciones, para reducir tiempos de lectura, ya que la mayoría de accesos de lectura acceden a una pequeña parte de los datos de la aplicación. Normalmente,

hay un conjunto de datos que son relevantes a todos los usuarios y que, por lo tanto, son accedidos con más frecuencia.

Concurrencia de usuarios: Permitir la concurrencia de usuarios.

Se debe permitir que varios usuarios trabajen en la misma base de datos, protegiendo los datos de ser escritos erróneamente. En la mayoría de los casos, se utilizará el bloqueo optimista, soportado por la mayoría de los motores de persistencia como la opción por defecto.

Referencias circulares entre objetos: Evitar las referencias circulares entre objetos de la capa de persistencia.

Se deben evitar las referencias circulares, facilitando la localización de los objetos. De esta manera, se devuelve el objeto solicitado sin necesidad de realizar un recorrido para localizarlo, lo que supone un acceso más efectivo.

Actualización en cascada: Utilizar la actualización en cascada siempre que sea posible.

Utilizar la posibilidad que ofrecen algunos frameworks para la actualización en cascada de objetos. Una actualización en cascada permite que las modificaciones hechas a un objeto se repliquen en los objetos relacionados. De esta manera se mejora el mantenimiento de los objetos, se asegura que los cambios introducidos se replican de manera eficiente y se mantiene la integridad de los datos.

4.2 Directrices tecnológicas de Infraestructuras

Osakidetza dispone de múltiples aplicaciones desarrolladas en diversas tecnologías (principalmente Java y .Net), y albergadas en diferentes plataformas (principalmente Apache HTTP Server, Microsoft IIS y VMware Horizon como soluciones de presentación, Oracle WebLogic, Apache Tomcat y Microsoft IIS como servidores de aplicación y Oracle RDBMS, Microsoft SQL Server, MongoDB como servidores de BBDD).

El modelo de referencia de las aplicaciones es una solución Web (sin plug-in's), con arquitectura de n-capas preferentemente 3 capas: presentación, negocio y persistencia, compatible con tecnología de virtualización y almacenamiento SAN y NAS. Dicha arquitectura es escalable horizontalmente en cuanto a las capas de negocio y persistencia (reparte la carga equitativamente entre los diversos servidores que ejecutan la lógica de negocio y estos a su vez, mediante un pool de conexiones acceden a la capa de persistencia), y es tolerante a fallos garantizando la continuidad de negocio.

Los servidores que atienden a las capas de negocio y persistencia están ubicados en los dos CPDs de la Dirección General de Osakidetza siendo el reparto de los mismos proporcional en un CPD u otro; la capa de presentación está distribuida en los puestos de trabajo fijos, portátiles, móviles y tabletas en los diversos centros de Osakidetza (Hospitales, Centros de salud, etc..).

De modo que la solución (producto/aplicación/desarrollo) a suministrar deberá ser compatible con el modelo de arquitectura indicado en el párrafo anterior y con la relación de productos y tecnologías que se indican a continuación (las que apliquen).

4.2.1 Entorno tecnológico

En conjunto, el entorno tecnológico de las soluciones actualmente en producción es:

- Plataformas de desarrollo:
  - .NET sobre Windows 2016*
  - Java sobre Red Hat Enterprise Linux 8 Aplicaciones móviles:
  - Node.js, Express, AngularJS, MongoDB sobre Red Hat Enterprise Linux 8
  - Ionic, Cordova sobre Android, iOS

  (*)En la actualidad 2019 en construcción

- Plataforma de virtualización:
  - VMWare vSphere 7.0 U2

- Almacenamiento:
  - SAN: EMC VMax
  - NAS: EMC Isilon

- Sistemas Operativos:
  - Servidores:
  - Windows 2016 (en la actualidad en construcción 2019)
  - Red Hat Enterprise Linux 8
  - Puesto de trabajo fijo: Windows 10, Enterprise x64
  - Puesto de trabajo móvil: Android, iOS

- Servidores Web:
  - Apache Webserver 2.4.X (en la actualidad 2.4.6)
  - MS IIS 10 (S.O. Windows Server 2016)
  - SAP Web Dispatcher 7.73 (actual 7.20 obsoleta)
  - Nginx 1.18.0

- Servidores de Aplicación:
  - Apache Tomcat 9.0.32 + JDK 8
  - Oracle Weblogic Server 12c + JDK 8
  - MS IIS 10 + .NET Framework 4.6.2
  - Application Server de SAP Netweaver 7.5
  - Node.js 12.17.0

- Sistemas de Gestión de Bases de Datos (*)
  - Oracle RAC 19c (para réplicas se usa Active Data Guard)
  - Oracle 19c Single instance
  - Microsoft SQL Server 2016 (En la actualidad 2019 en construcción)
  - MongoDB 4.4.1

(*) Continuidad de negocio y Alta disponibilidad: Oracle RAC, SQL Server Always ON solo para aplicaciones críticas

- Gestión de contenidos:
  - MS SharePoint Online (O365)
  - Documentum 20.4

- Infraestructura SOA:
  - Oracle Service Bus 12c (12.2.1.4) + JDK 8.0

- Plataforma de firma (PKI):
  - Izenpe

- Autenticación:
  - Directorio Activo de MS (AD)

- Modo de ejecución/escritorio:
  - local: puesto fijo o móvil
  - Cliente de Horizon 5.3
  -
- Dispositivos Inalámbricos:
  - Seguridad:

- Utilización del estándar IEEE 802.11 b/g/n
- Capacidad de asociarse a un SSID de la red WIFI correspondiente
- Soporte SSID oculto
- Soporte WPA2 Enterprise
- Cifrado de datos AES
- Autenticación EAP basada en certificado de máquina
- Soportar certificados:
  - Los certificados emitidos son de 1024 bits
  - El certificado de la CA raíz es de 4096 Soporten certificados con hash SHA-2
  - El tipo de certificado: x.509
  - Protocolo: RSA Señal:
- Ganancia de la antena: Mínimo 3 dBi
- Sensibilidad <= -70 dBm a 11Mbps

Consideraciones en relación a los de productos y tecnologías indicados anteriormente:

- En cuanto al navegador, se debe evitar el uso de Applets (Java), componentes ActiveX (.NET), Flash u otras tecnologías que impliquen la descarga y ejecución de software embebido en el navegador.
- Al respecto de Office 2010, no está permitido el uso, dependencia o interrelación con Access.
- Al respecto de los dispositivos inalámbricos se han especificado los requisitos mínimos que deben cumplir.

4.2.2 Especificaciones del Cliente (Maqueta)

Actualmente, la versión de maqueta que se instala en los equipos de Osakidetza es la 2.1, aunque convive con versiones anteriores instaladas.

- Hardware (mínimo existente):
  - PC arquitectura Intel o equivalente
  - Procesador con un mínimo de Tecnología de 45 nm, Caché L2 de 3 Mb y FSB de 1.066 MHz.
  - Memoria RAM 2 Gb (4 GB en W10)
  - Disco duro 250 Gb (500 Gb en W10)
  - Monitor 19 pulgadas con resolución 1440x900 mayoritaria o 1280x1024. (21-22" 1920x1080 en W10)

- Restricciones de escritura en disco C: para usuarios standard, excepto en su perfil de usuario (C:|users\) o En cuanto al navegador, se debe evitar el uso de Applets (Java), componentes ActiveX (.NET), Flash u otras tecnologías que impliquen la descarga y ejecución de software embebido en el navegador.
  - Al respecto de Office 2016, no está permitido el uso, dependencia o interrelación con Access. o Mismas restricciones (navegador y Office) que W7.

istema Operativo que instala esta imagen

X	Actual - Windows 10, Enterprise x64 1709
	Actualizando - Windows 10, Professional x64 1909
Idiomas Disponibles:	Castellano (es-ES); Euskera (eu-ES)

Hotfixes:	
Descripción	Clasificación
Compatible desde 1703 hasta el update y actualizaciones	Updates Críticas y de seguridad

Parámetros de Configuración de Software

Fabricante	Producto	Versión	Idioma	Plataforma
Microsoft	Silverlight	5.5.1	Castellano	x64
Microsoft	Visual C++ 2008	9.0.30729.4148	Castellano	X64,x86
Microsoft	Visual C++ 2010	10.0.30319	Castellano	x64;x86
Microsoft	Visual C++ 2012	11.0.60610	Castellano	x64;x86
Microsoft	Visual C++ 2013	12.0.30501	Castellano	x64;x86
Microsoft	Visual C++ 2017	14.11.25325	Castellano	x64;x86
Oracle	Java	8.1510.12	Castellano	X64;x86
Adobe	Acrobat Reader	1.502.020.039	Castellano	x64
Adobe	Shockware	12.2.4.194	Castellano	x64
7zip Org	7zip	16.04	Castellano	x64
FNMT	DNIe	14.0	Castellano	x64
FNMT	Certificados Raiz			
FNMT	TC-FNMT	6.1.3	Castellano	x64

Xuxen	Corrector Xuxen	5.2	Euskera	X86
LibreOffice	LibreOffice	5.4.5.1	Multiligual	X64
Firefox	Firefox	58.0.2	Castellano	X64
Microsoft	RSTA Cliente	1709	Castellano	X64
Microsoft	Cliente One Drive	17.3.6816.0313	Castellano	x64
IZENPE	Middleware_Izenpe	4.0.2.436	Castellano	X64
IZENPE	Idazki-desktop-win	3.3.1	Castellano	X64
Vmware	Horizon Client	4.7	Multilingual	x64
Parámetros de Configuración de Software Personalización de Osakidetza				
Oracle	Cliente Oracle	12.1.0.2	Castellano	x64
VMWare	Cliente Horizon	5.0.0		
Microsoft	Office Standard	2016	Castellano	x86
Microsoft	Office ProPlus	2016	Castellano	x86
Microsoft	Access Runtime (Con versión Standard 2016)	2016	Castellano	X86
Business Objects	Crystal Reports	10.5.0.0.sp16	Ingles	X86
Business Objects	Crystal Reports runtime engine	13.0.22	Ingles	X86
NDDigital	nddDigital DCS Server (Pago por impresion)	2.0.0.6	Ingles	X64
NDDigital	nddDigital nddPrint Agent	5.20.5	Ingles	X64
BitDefender	Bitdefender Endpoint Security Tools	6.6.11	Castellano	X86
Microsoft (integrado con el SO)	NET Framework	3.5 (incluye .NET 2.0 y 3.0)		X64
Microsoft (integrado con el SO)	Servicios avanzados de .NET Framework	4.7(ASP.NET 4.7)		X64
Unify	Team Developer 5.2 Deployment	5.2.0000		x86

4.2.3 Dispositivos de entrada/salida

La solución deberá ser capaz de interactuar con diversos dispositivos de entrada y salida de diversos fabricantes; siendo dichos dispositivos: pantallas táctiles, impresoras térmicas, monitores, por tanto, la interface con los mismos será en base a estándares del mercado.

Así mismo la solución deberá adaptarse a la resolución de las diversas pantallas.

En relación con dispositivos móviles se deben tener en cuenta ciertas particularidades en relación a las características de movilidad, tamaño, comunicación inalámbrica e interacción con las personas:

- Son aparatos pequeños.
- La mayoría de estos aparatos se pueden transportar en el bolsillo del propietario o en un pequeño bolso.
- Tienen capacidad de procesamiento.
- Tienen conexión permanente o intermitente a una red.
- Tienen memoria (RAM, tarjetas MicroSD, flash, etc.).
- Normalmente se asocian al uso individual de una persona, tanto en posesión como en operación, de modo que puede adaptarlos a su gusto.
- Tienen una alta capacidad de interacción mediante la pantalla o el teclado.

En lo que se refiere a dispositivos sobre los que tienen que ejecutar soluciones de movilidad hay dos ámbitos que son las soluciones orientadas a profesionales de Osakidetza y soluciones orientadas al ciudadano.

Las aplicaciones orientadas al ciudadano se ejecutarán en dispositivos de consumo ya sean móviles, tabletas o cualquier otro dispositivo al que vaya destinada la solución. En este caso el desarrollo deberá ejecutarse en dispositivos con los diferentes sistemas operativos que haya en el mercado.

En las aplicaciones orientadas a los profesionales de Osakidetza las aplicaciones deberán de ejecutarse en el dispositivo elegido y homologado para soportar la solución. Habrá que tener en cuenta los siguientes aspectos:

- Ergonomía, tamaño y usabilidad del dispositivo, se refiere a que cumpla las expectativas del usuario en ámbito en el que se va a desarrollar el proyecto.
- Especificaciones de limpieza y resistencia a los golpes.
- Características de comunicaciones como pueden ser conectividad Wifi y 4G.

4.2.4 Seguridad, acceso, autenticación y autorización

El tratamiento de los datos cumplirá lo establecido por la legislación vigente en materia de seguridad y protección de datos.

En el ámbito intranet se permitirá el acceso mediante los transportes RMI, JMS, HTTP y HTTPS.

El proveedor de servicios de certificación de Osakidetza es Izenpe.

Osakidetza ha dotado a todos sus profesionales de certificados corporativos reconocidos, en soporte tarjeta criptográfica, con capacidad de firma de documentos y mails, cifrado y autenticación en el AD.

Además, para los nuevos SI que necesitan firma electrónica ágil, Osakidetza ha implantado una solución horizontal de firma, basada en certificados alojados en HSMs, llamada “firma centralizada”. Son certificados con las mismas características y garantías que los alojados en tarjeta criptográfica. Estos HSM son de última generación y cumplen con las más exigentes normas internacionales (FIPS 140-2 Level 2 and Level 3, Common Criteria at EAL 4+ y RoHS compliant). Permiten la firma a través de CSP propio (applets) y la firma directa basada en una Pasarela de firma propietaria.

Izenpe es también el proveedor de la infraestructura de autenticación, validación y firma basada en certificados electrónicos. Osakidetza dispone de 2 Zain en HA (equipos TrustedX de Safelayer) en propiedad y su backup, en modo servicio, en los Zain de Izenpe. La integración con estos equipos se puede realizar mediante su librería Smartwrapper. Provee diferentes servicios web (validación, evolución de firmas, firma en servidor, etc.) alojados en los OSB de Osakidetza.

Así mismo, Osakidetza dispone de su propia infraestructura de Custodia de documentos firmados, para asegurar la validación de cualquier firma realizada en sus sistemas de información con certificados electrónicos. Está constituida por 2 equipos Siaval en HA, de la empresa SIA, y HSM Ncipher propios.

Para la firma electrónica de documentos in-situ por parte de sus pacientes, Osakidetza dispone de una solución denominada “firma biométrica”, que asegura la validez de todas las firmas implicadas. Se apoya en la solución SmartAccess de Telefónica.

Osakidetza está inmerso en la adecuación de toda su infraestructura y sistemas de información al reglamento europeo eIDAS. Tanto en cuestión de certificados como de niveles de aseguramiento (autenticación).

Los sistemas de autenticación válidos son:

- Autenticación basada en Directorio Activo (Microsoft Active Directory) corporativo, que permite identificar un empleado de Osakidetza mediante usuario/contraseña o la tarjeta profesional.
- Token de Kerberos, que habrá sido emitido por el Directorio Activo corporativo.
- Autenticación basada en infraestructura de certificación digital (PKI). Esta autenticación permite la identificación de personas (certificados personales) o de sistemas de información (certificados de servidor)
- Certificados X.509 de servidor, que permitirán identificar sistemas (máquinas) y establecer confianza entre ellas.
- Certificados X.509 de cliente, que permitirán identificar individuos (personas)
- Para los servicios web, el estándar WS-Security permitirá aplicar políticas de autenticación.
- El bus de servicios sólo se utilizará para la autenticación basada en certificados X.509 de servidor.

El sistema de autorización se implementará por medio de la definición de roles de usuario en base a los cuales se gestionarán los permisos y acciones sobre los distintos procesos por los que estará compuesta la solución. De modo que cada usuario en función del rol que tenga asignado podrá acceder a una serie de funcionalidades, así como a los datos relativos a su organización de servicio, centro de trabajo, servicio/área de atención.

Se crearán grupos de autorización del Directorio Activo. Por cada perfil que tenga la aplicación (medico, enfermera, administrativo, administrador de la aplicación,…) se creará un grupo de autorización en el DA. Se mapeará el perfil al grupo.

4.2.5 Monitorización

Osakidetza dispone de una plataforma de monitorización a través de la cual ya tiene establecida una línea de monitorización base de su infraestructura y SI.

En este sentido la solución deberá incluir un apartado en el que se describan los elementos a monitorizar para asegurar un correcto funcionamiento del sistema, servicio o producto suministrado de forma integral dentro de la mencionada plataforma de monitorización.

La monitorización se podrá llevar a cabo a través de alguna combinación de los siguientes productos o métodos que Osakidetza integra dentro de su plataforma de monitorización:

- SCOM mediante Management Pack para sistemas Microsoft.
- Nagios XI para sistemas Linux.
- Enterprise Manager Cloud Control para SGBD Oracle / EM 13 y servidor de aplicaciones Weblogic para rendimiento.
- Instalación de agente Dynatrace (1.219.131) para aplicaciones web/Horizon

Las herramientas de monitorización se integran con Evanios/ITOM (gestión de eventos) para las plataformas Batera.

Con el objetivo de monitorizar la solución que vaya a ser suministrada, deberán detallarse los objetos concretos a monitorizar para garantizar su correcto funcionamiento más allá de los aspectos generales de la infraestructura de sistemas que los alberguen; dicha relación de objetos podrá incluir, aunque no exclusivamente, algunos como:

  - Procesos en ejecución
  - Puertos de comunicaciones a la escucha

- Eventos concretos en registros del sistema
- Unidades de disco o File Systems

Para cada uno de ellos deberán incluirse los umbrales de consumo que se consideren anormales, especificándose 2 valores a partir de los cuales establecer el correspondiente nivel de alerta preventiva o alerta crítica según proceda.

Adicionalmente podrán suministrarse scripts o procesos para llevar a cabo:

- Pruebas sintéticas de los sistemas cuyo resultado determine el correcto funcionamiento del sistema o el correspondiente nivel de alerta preventiva
- Operativas sobre el sistema que puedan ayudar a determinar o especificar la condición concreta de los errores o alertas producidos.
- Operativas sobre el sistema para intentar remediar automáticamente la
- condición de la alerta o error, como reinicio de procesos o servicios, etc.

Todos estos procesos deberán entregarse con su correspondiente documentación, y su operativa deberá resultar sencilla de modo que pueda ser realizada por un operador.

4.2.6 Backup/Restore

Osakidetza dispone de una infraestructura para la realización de copias de seguridad y restauración a través de red basada en CommVault.

De modo que la solución deberá especificar los procedimientos necesarios a llevar a cabo para la recuperación del servicio ante la pérdida del mismo, corrupción de sus datos o alguno de sus componentes. En este sentido se plantearán escenarios de recuperación independientes para cada uno de los posibles componentes impactados y el orden de ejecución en caso de que deban ejecutarse secuencialmente para la recuperación de varios componentes o del sistema completo en su conjunto.

A partir de los procedimientos anteriormente descritos y en cada uno de ellos, se inferirá los elementos concretos sobre los que se requiere hacer Backup, en qué orden, con qué frecuencia y política de persistencia deben llevarse a cabo para poder satisfacer los distintos escenarios de recuperación del servicio contemplados. Así mismo deberán indicarse las restricciones que pudieran existir para la realización del Backup: posible impacto en el servicio, restricciones horarias, etc.

Los distintos procedimientos de recuperación suministrados con la solución deberán ser validados por Osakidetza para garantizar su compatibilidad con la infraestructura de copia de seguridad y restauración existente.

4.3 Interoperabilidad

Los requisitos de integración de diversos sistemas de información de Osakidetza se realizarán de acuerdo a una arquitectura orientada a servicios (SOA) sobre la plataforma Oracle SOA Suite (Oracle BPEL Process Manager, Oracle Service Bus (OSB), Oracle Business Rules, ...).

Dicha arquitectura proporciona una forma bien definida de exposición e invocación de Servicios Web, lo cual facilita la interacción entre diferentes sistemas propios o de terceros.

Sobre la misma arquitectura SOA, Osakidetza implementa una solución de integración orientada a Eventos; es un diseño a medida que gestiona un conjunto de sistemas que publican eventos y un conjunto de aplicaciones que se suscriben a determinados eventos. Dicha solución se denomina Gestor de Eventos-Event Manager y es responsable de recibir los eventos publicados, ejecutar las validaciones adecuadas, enrrutar y almacenar los eventos para su envío a los subscriptores que estén asociados a cada uno de los elementos recibidos.

Relación de estándares de comunicación soportados:

- Protocolos a nivel de mensaje:
  - SOAP 1.1 y SOAP 1.2
  - WSDL 1.1 y WSDL 1.2 Binding
  - SOAP con Attachments
  - JSON – REST (para aplicaciones móviles)
- Protocolos de seguridad a nivel de mensaje:
  - WS-Security 1.0/1.1
  - WS-SecurityPolicy
  - WS-Policy
  - WSPolicyAttachment
  - WS-Security: Username Token Profile 1.0/1.1
  - WS-Security: X.509 Token Profile 1.0/1.1
  - WSSecurity: SAML Token Profile 1.0/1.1
  - WS-Security: KerberosToken Profile 1.1
  - WS-Reliable Messaging 1.0 o WS-Addressing
  - WS-I Basic Profile 1.1
  - WSSecurity: JWT Token (aplicaciones móviles en internet)
- Protocolos a nivel de transporte:
  - HTTP 1.0, HTTP 1.1
  - TLS, SSL
  - Interoperabilidad con registros UDDI v3-compliant
  - Sistemas middleware basados en JMS/MQ
  - HTTPS

4.3.1 Servicios Web

Los Servicios Web se implementarán de acuerdo con las especificaciones WSDL v1.1, SOAP v1.1, v1.2, JSON-REST (en el caso de aplicaciones móviles), UDDI v2.XX y XML v1.0, con el objetivo de incorporar las recomendaciones de la WS-I definidas en la especificación Basic Profile v1.0, v2.0 y de esta manera asegurar la interoperabilidad entre los sistemas.

El estándar de codificación que se utiliza en los mensajes XML es UTF-8.

La seguridad aplicada a los servicios web cubrirá los siguientes aspectos:

- Autenticación: Verificar que el cliente (usuario o aplicación) es quien dice ser. La identidad de un usuario se realizará en base a la información presentada por el usuario (usuario/contraseña, certificado, token SAML)

- Autorización: Otorgar acceso a los servicios en base a la identidad del cliente o a los roles asignados.
- Confidencialidad, privacidad: Mantener la información secreta mediante el uso de algoritmos de encriptación estándar de elementos XML.
- Integridad, no repudio: Asegurar que un mensaje permanece inalterado durante la transmisión mediante la firma digital. La firma también valida la identidad del remitente y proporciona una marca de tiempo para garantizar que una transacción no puede ser repudiada más tarde ni por el remitente ni por el destinatario.

Osakidetza usa Oracle Web Service Manager (OWSM) para gestionar y aplicar políticas a los servicios web publicados en la plataforma SOA.

La política estándar que Osakidetza ha definido para los servicios proporciona:

- Autenticación mediante certificado x509
- Protección del mensaje mediante firma (sin encriptado)

Existen dos versiones de la política en OWSM, una para servicios y otra para clientes. Para garantizar la interoperabilidad, cada política tiene su versión compatible con tecnología .NET y Java.

- oracle_wss10_x509_token_with_message_sign_service_policy
- oracle_wss10_x509_token_with_message_sign_service_policy_net
- oracle_wss10_x509_token_with_message_sign_client_policy
- oracle_wss10_x509_token_with_message_sign_client_policy_net
  oracle_http_jwt_token_server_policy

Los clientes y aplicaciones que acceden a través de internet entran a la DMZ a través del firewall de aplicaciones (WAF). El WAF aplica reglas contra ataques y define patrones de seguridad.

- Es responsabilidad de cada aplicación publicada en la DMZ controlar el acceso y autorizar a los usuarios.
- El OSB de internet publica los servicios a los que pueden acceder las aplicaciones de internet.
- El OSB de internet audita todas las llamadas a los Servicios Web mediante una política propietaria de Osakidetza gestionada por OWSM.
- En el OSB de internet se protegen todos los servicios con la política oracle_wss10_x509_token_with_message_sign_service_policy. Esta política autentica a las aplicaciones mediante certificado x509 y firma el mensaje de petición y respuesta.
- El OSB de internet delega la ejecución a servicios publicados en la Intranet.
- En la intranet, se despliegan instancias independientes de servicios web para dar servicio a las peticiones que llegan desde el OSB de Internet.
- Entre el OSB de intranet y el OSB de intranet, se realiza una federación de servicios con las condiciones de seguridad habituales.

Nota: OSB -> Oracle Service Bus 12c

4.3.2 14.3.2 Gestor de Eventos

A continuación, se describen los requisitos técnicos que tienen que cumplir las aplicaciones para publicar y/o recibir eventos.

Estándares de comunicación

La mensajería del Servicio Osakidetza se implementará de acuerdo con las especificaciones del estándar HL7 versión 2.XX o con cualquier otro formato propio de Osakidetza; de esta manera se asegura la interoperabilidad entre los sistemas de información.

Relación de tecnologías soportadas para la publicación y subscripción a eventos. Tecnologías que soporta el gestor de eventos para la publicación y subscripción a eventos, así como si la modalidad soporta transaccionabilidad y las opciones de seguridad disponibles:

Modalidad	Tecnología	Transaccional	Orden	Seguridad
Publicación	Mensajería JMS	Sí	Sí, si el publicador establece el parámetro UnitOfOrder	Autenticación (user/pass)
Publicación	Servicio web	No	No	Ninguna, Autenticación (user/pass) y WS-Security
Subscripción	Mensajería JMS	Sí	Sí. Event Manager garantiza la entrega en el mismo orden que ha recibido los mensajes	Autenticación (user/pass)
Subscripción	Servicio OSB	Sí	Sí. Event Manager garantiza la entrega en el mismo orden que ha recibido los mensajes	Ninguna
Subscripción	Servicio web	Sí	Sí. Event Manager garantiza la entrega en el mismo orden que ha recibido los mensajes	Ninguna, Autenticación (user/pass) y WS-Security

Por cada modalidad y tecnología se deben cumplir unos requisitos técnicos que serán indicados al adjudicatario para la implementación de esta forma de integración.

Mensajería. Definición de Evento

Un evento es un documento XML definido mediante un XSD, donde:

- id: Es el identificador del tipo de evento. Se genera durante el proceso de alta del evento en el sistema de administración de Event Manager. Durante el procesamiento de un evento se verifica que el id sea válido.
- correlation: Es un campo libre en el que el publicador del evento indica un número correlativo relativo a su sistema.
- source: Es el identificador del publicador. Se genera durante el proceso de alta de un publicador en el sistema de administración de Event Manager. Durante el procesamiento de un evento se verifica que el source sea válido.
- timestamp: Lo establece el publicador del evento en el momento del envío.
- metadata: Puede contener un xml que ayude a describir el contenido del evento. Event Manager puede utilizar esta información para tomar decisiones de enrutado.
- payload: Es el contenido del evento. Puede ser cualquier cadena de texto o XML.

El resultado devuelto cuando se publica un evento en Event Manager es un XML definido por un XSD, donde:

- uuid: Es un identificador único que se asigna a cada evento procesado por Event Manager.
- processed: true o false, si el evento se ha procesado correctamente o con errores.
- errorCode: Si se ha producido un error, aqui se informa el código del error.
- errorDescription: Si se ha producido un error contiene la descripción de éste.

Se realiza gestión de errores: codificación de errores y descripción de los mismos.

4.4 Herramientas del ciclo de vida de las aplicaciones

Como soporte e instrumento necesario en la ejecución de todas las fases del proyecto, existe un conjunto de Herramientas que abarcan todo el ciclo de vida de las aplicaciones, y que facilitan la realización de distintas tareas y normalizan la obtención de entregables.

Estas herramientas homologadas son las que se utilizan en el entorno de trabajo de Osakidetza, no pudiendo utilizarse en el mismo otras herramientas similares o equivalentes.

Para los trabajos a realizar en las dependencias de la empresa proveedora, su uso es recomendado frente a otros productos o herramientas del mercado, para dar cobertura a los cometidos para los que están destinadas. No obstante, en los casos en los que el resultado de uso de las herramientas sea un entregable con un formato específico y normado, su uso será obligatorio frente a otras herramientas de mercado, o bien en cualquier caso deberá proporcionarse un formato compatible.

Anexo 1. Asistencia Técnica Calidad Osakidetza



Anexo%201%20AT%
20CALIDAD%20OSAI