



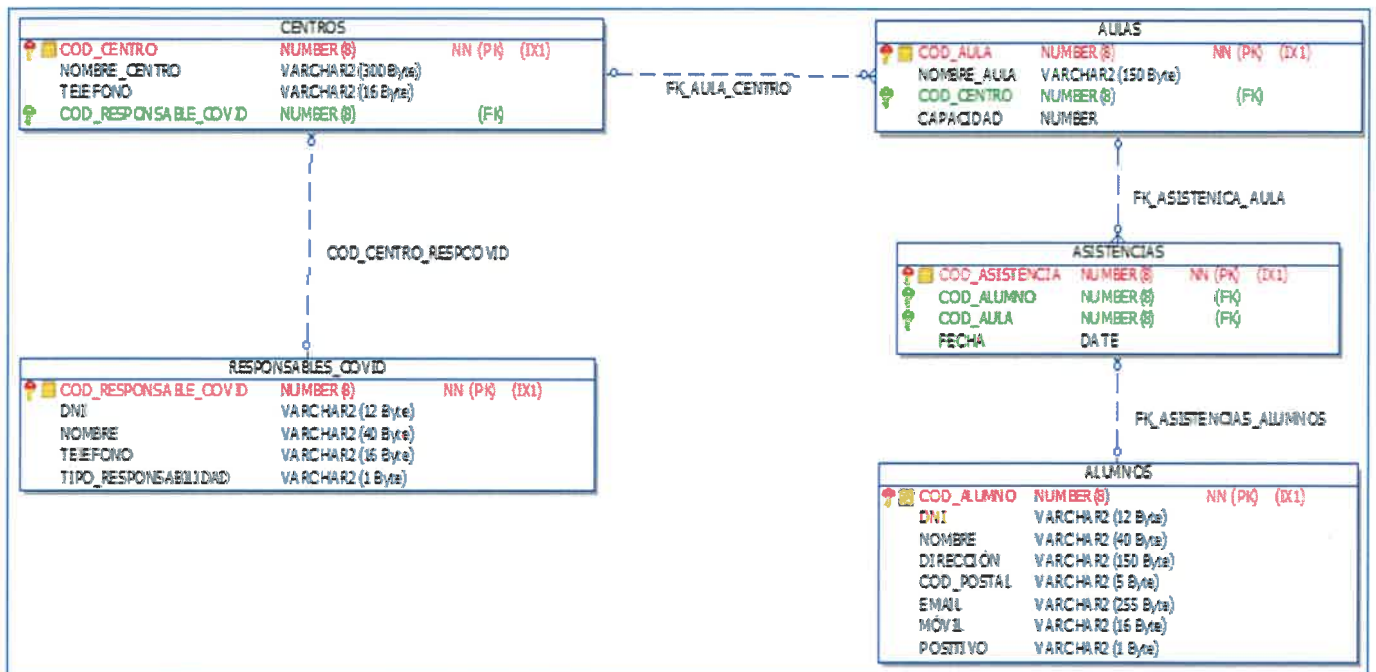
UNIVERSIDAD
DE MÁLAGA

Supuesto Nº 1

La situación actual provocada por la pandemia de COVID-19, ha obligado a la universidad a la creación de un registro de los espacios que han sido utilizados por los estudiantes, a fin de poder conocer, en caso de algún positivo, los compañeros con los que ha compartido aula y decretar su confinamiento.

Para este seguimiento se dispone de una base de datos ORACLE, en la que se ha creado un esquema denominado ESPACIOSCOVID, que permite conocer los espacios visitados por los estudiantes.

El modelo de datos es el siguiente:



Se han colocado códigos QR a la entrada de cada una de las aulas universitarias, para que sean leídos por cada estudiante al entrar en las mismas a través de una aplicación móvil diseñada al efecto.

Cada vez que un estudiante accede a un aula, lee el QR colocado a la entrada, y la app genera una anotación en la tabla ASISTENCIAS que permite el posterior rastreo.

Para las tareas de rastreo existen dos roles autorizados a acceder a esta información:

Rol ResUMA: Responsable COVID de la UMA.

Rol ResCENTRO: Responsable COVID de cada uno de los centros universitarios.

El primero de ellos podrá consultar toda la información almacenada acerca de los accesos realizados, y los segundos, solamente a los accesos realizados en su centro correspondiente.

El proyecto se desarrollará con tecnología Oracle ADF 12c, servidor de aplicaciones Oracle WebLogic Server 12c y base de datos Oracle 11g.

(En todo lo no contemplado en el enunciado de este supuesto podrá efectuar las suposiciones que considere conveniente, debiendo siempre hacerlas constar en su propuesta de solución y justificando las mismas.)

Teniendo en cuenta lo anteriormente expuesto, se le pide realice las siguientes cuestiones:

1. Escriba las sentencias Sql necesarias para:

a) (2,5 puntos) Cree la tabla ASISTENCIAS representada en el modelo conceptual proporcionado en el enunciado, teniendo en cuenta las siguientes premisas:

- El campo COD_ASISTENCIA es la clave.
- Se usará el tablespace USERS.
- Un bloque que se ha llenado previamente no se volverá a utilizar para inserciones hasta que ese porcentaje de utilización sea inferior al 40%.
- Se reservará el 20% de cada bloque para evitar encadenamientos.
- La tabla se almacenará en una extensión inicial de 20K, y las extensiones sucesivas serán de 30K, hasta un máximo de 10 extensiones.

b) (2,5 puntos) Cree los objetos de base de datos que permitan que al insertar registros en la tabla ASISTENCIAS se le asigne automáticamente un valor secuencial al campo clave de dicha tabla. Y que, también, se le asigne por defecto la fecha actual al campo FECHA de dicha tabla.

2. (5 puntos) Cree una VISTA materializada denominada VM_ALUMNOS_ASISTENCIAS con la siguiente información:

Datos de Alumnos: Cod_Alumno, Dni, Nombre, Dirección, Cod_Postal, Email y Movil. Fecha de asistencia, Código de Centro y Nombre del Centro al que asistió, Código de responsable Covid y Nombre del Responsable Covid del Centro. Y por último, el número total de asistencias registradas del alumno. Esta última columna deberá llamarse ASISTENCIAS_TOTALES. Y todo ello ordenado por nombre del alumno y fecha de asistencia.

Los datos de esta vista materializada estarán disponibles desde el momento de creación de la misma, y serán refrescados bajo demanda.

3. (5 puntos) Cree una función PL/SQL denominada Cuantos_Vinieron, en un paquete denominado PKG_ASISTENCIAS, que devuelva el número total de alumnos que han estado en una fecha y aula determinada. Estos dos valores se pasarán como parámetros de entrada. Si no hay ninguna asistencia para esa fecha, la función devolverá -1.

4. Teniendo en cuenta la información del proyecto *GestionCovid* proporcionada en el anexo A, se pide:

a) (2,5 puntos) Escriba el código que habría que incluir en *EjemploPopup.jsf*, para mostrar un botón denominado “Mostrar Popup” que, tras pulsarlo, muestre una ventana popup por pantalla, tal y como se muestra en la siguiente imagen:



Nota: No es necesario añadir ninguna funcionalidad al botón Aceptar.

b) (2,5 puntos) Modifique el ejercicio del apartado anterior para que, a continuación del texto: “Ejemplo de ventana popup en ADF.”, muestre la cadena de texto: “El número de asistentes el 09/10/2020, en el aula 1, es:”, seguido del resultado de llamar a la función *Cuantos_Vinieron*, ya implementada anteriormente en el paquete *PKG_ASISTENCIAS*, que espera como argumento la fecha y el aula correspondiente y devuelve el número de asistentes en dicha fecha y aula.

5. La funcionalidad de la página *RastreadorCovid.jsf* consiste en introducir un DNI de un estudiante que ha dado positivo en las pruebas de Covid y, al pulsar un botón denominado *Rastrear contactos*, se genera un fichero en formato *Excel* denominado *ContactosEstudiante.xls* con los estudiantes que han coincidido con él.

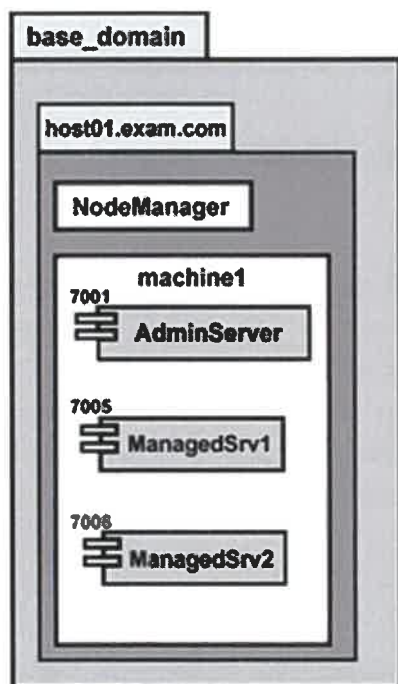
a) (1,5 puntos) Escriba el código correspondiente al componente *ADF Faces* que iría en la página *RastreadorCovid.jsf* para el botón *Rastrear contactos*. Para desarrollarlo, tenga en cuenta que dentro del método *exportaAExcel* de la clase *BackingCovid* se realizan todas las operaciones necesarias para crear un fichero *Excel* y rellenarlo con la consulta de los estudiantes coincidentes.

b) (1,5 puntos) Escriba el código que habría que añadir en la página *RastreadorCovid.jsf* utilizando un componente *ADF Faces* de tipo lista desplegable, que permita seleccionar un solo valor, y que muestre el nombre de todas las aulas.

- c) (2 puntos). Escriba las líneas necesarias a incluir en *BackingCovid.java* para implementar un método que pueda ser llamado desde el botón *btAulas* de *RastreadorCovid.jsf* y que, al pulsarlo, seleccione el primer elemento de la lista desplegable del apartado anterior.

Nota: La información necesaria se proporciona en el anexo A.

6. Disponemos de un entorno de producción con una instalación de Oracle Fusion Middleware Infrastructure, compuesto por un dominio Oracle WebLogic 12c, llamado "base_domain" que tiene la siguiente estructura:

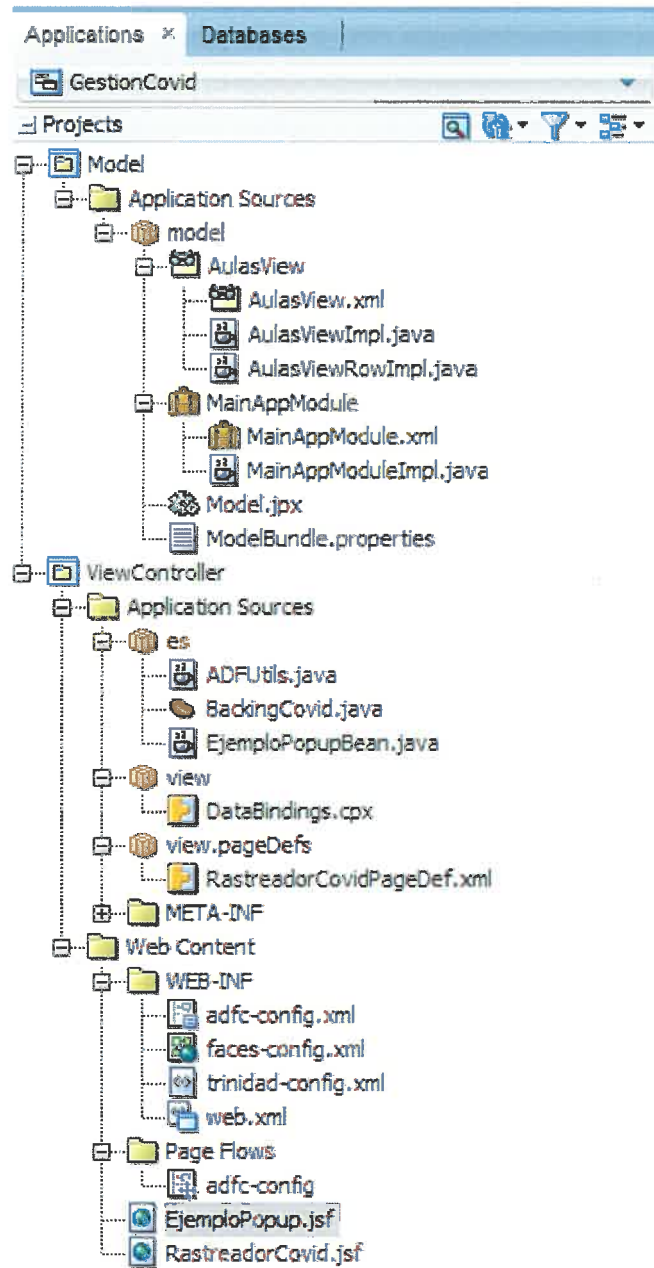


El equipo de desarrollo dispone de un plan de despliegue llamado "CovidProduccionPlan.xml" que contiene la parametrización adecuada al entorno de producción. Sabiendo que el puerto administrativo de WebLogic es 7001, y el usuario administrador del dominio es "weblogic" con contraseña "welcome1":

- a) (2,5 puntos) Describa cómo realizaría el despliegue de la aplicación "GestionCovid" (GestionCovid.ear) en el servidor manejado ManagedSrv1 mediante el uso de la herramienta de línea de comandos basada en Java que proporciona WebLogic.
- b) (2,5 puntos) Para que la aplicación "GestionCovid" disponga de alta disponibilidad, ¿qué requisitos debe cumplir y qué ficheros de nuestra aplicación tendría que modificar para cumplir con las especificaciones de alta disponibilidad para ADF?

ANEXO A:

Imagen de la estructura del proyecto GestiónCovid ADF:



Se extrae la siguiente información de los ficheros más relevantes del proyecto:

Plantilla del fichero *MainAppModuleImpl.java*

```
package model;

import oracle.jbo.server.ApplicationModuleImpl;
import java.sql.CallableStatement;
import java.sql.SQLException;
import java.sql.Types;
import oracle.jbo.server.DBTransaction;

// -----
// --- File generated by Oracle ADF Business Components Design Time.
// --- Thu Dec 05 21:41:49 CET 2019
// --- Custom code may be added to this class.
// --- Warning: Do not modify method signatures of generated methods.
// -----
public class MainAppModuleImpl extends ApplicationModuleImpl {
    /**
     * This is the default constructor (do not remove).
     */
    public MainAppModuleImpl() {
    }

    @SuppressWarnings("oracle.jdeveloper.java.insufficient-catch-block")
    public int callDb_ObtenerCuantosVinieron(String funcion, String fecha,
oracle.jbo.domain.Number aula)
    {
        int numAsist = 0;
        CallableStatement st = null;
        try {
            DBTransaction transaccion=this.getDBTransaction();
            st =transaccion.createCallableStatement("begin ? := " + funcion + ";
end;", 0);

            st.registerOutParameter(1, Types.NUMERIC);
            st setObject(2, fecha);
            st setObject(3, aula);
            st.executeUpdate();

            numAsist = st.getBigDecimal(1).intValue();
        } catch (SQLException e) {
            e.printStackTrace();
        } finally {
            if (st != null)
                try {
```



```

        st.close();
    } catch (SQLException e) { }
}
return numAsist;
}
/**
 * Container's getter for AulasView.
 * @return AulasView
 */
public AulasViewImpl getAulasView() {
    return (AulasViewImpl) findViewObject("AulasView");
}
}

```

Plantilla del fichero *ADFUtils.java*

```

package es;

import javax.faces.context.FacesContext;
import javax.el.ELContext;
import javax.el.ExpressionFactory;
import javax.el.ValueExpression;
import javax.faces.application.Application;

public class ADFUtils {

    public static Object resolveExpression(String expression) {
        FacesContext facesContext = FacesContext.getCurrentInstance();
        Application app = facesContext.getApplication();
        ExpressionFactory elFactory = app.getExpressionFactory();
        ELContext elContext = facesContext.getELContext();
        ValueExpression valueExp = elFactory.createValueExpression(elContext,
expression, Object.class);
        return valueExp.getValue(elContext);
    }
    /*Se obvia el resto de métodos de la clase*/
}

```

Plantilla del fichero *EjemploPopup.jsf*

```
<?xml version='1.0' encoding='UTF-8'?>
<!DOCTYPE html>
<f:view xmlns:f="http://java.sun.com/jsf/core"
xmlns:af="http://xmlns.oracle.com/adf/faces/rich">
  <af:document title="Ejemplo de un Popup en ADF" id="d1">
    <af:form id="f1">

      </af:form>
    </af:document>
  </f:view>
```

Plantilla del fichero *EjemploPopupBean.java*

```
package es;

import javax.faces.bean.ManagedBean;
import javax.faces.bean.RequestScoped;

@ManagedBean(name="ejemploPopupBean")
@RequestScoped
public class EjemploPopupBean {
  public EjemploPopupBean() {
    super();
  }
}
```

Extracto del fichero *BackingCovid.java*

```
package es;
import java.io.ByteArrayOutputStream;
import java.io.OutputStream;

import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.ResultSetMetaData;
import java.sql.Statement;

import org.apache.poi.hssf.usermodel.HSSFCell;
import org.apache.poi.hssf.usermodel.HSSFRow;
import org.apache.poi.hssf.usermodel.HSSFSheet;
import org.apache.poi.hssf.usermodel.HSSFWorkbook;

import javax.faces.context.FacesContext;
import oracle.adf.model.binding.DCIteratorBinding;
import oracle.adf.model.binding.DCBindingContainer;
import oracle.binding.BindingContainer;

public class BackingCovid {

    public static FacesContext getFacesContext() {
        return FacesContext.getCurrentInstance();
    }

    public static BindingContainer getBindingContainer() {
        return (BindingContainer)ADFUtils.resolveExpression("#{bindings}");
    }

    public static DCBindingContainer getDCBindingContainer() {
        return (DCBindingContainer)getBindingContainer();
    }

    public static DCIteratorBinding getIteratorBinding(String name) {
        return ((DCBindingContainer)getBindingContainer()).findIteratorBinding(name);
    }

    public void exportaAExcel(FacesContext facesContext, OutputStream
outputStream){
        /*a partir de aquí se obvia el resto del código de la clase*/
    }
}
```

Dentro de la página *RastreadorCovid.jsf* se encuentra el siguiente componente:

```
<af:button id="btAulas" text="Inicializar aulas" type="default"
    actionListener="#{BackingCovid.inicializarAulas}">
</af:button>
```

Del pagedef *RastreadorCovidPageDef.xml* asociado a la página *RastreadorCovid.jsf* se extrae la siguiente información:

```
<?xml version="1.0" encoding="UTF-8" ?>
<pageDefinition xmlns="http://xmlns.oracle.com/adfm/uimodel"
version="12.2.1.16.48" id="RastreadorCovidPageDef"
    Package="view.pageDefs">
    <parameters/>
    <executables>

        <iterator id="AulasViewIterator" Binds="AulasView"
DataControl="MainAppModuleDataControl" RangeSize="10"/>
    </executables>

    <bindings>

        <list ListOperMode="navigation" IterBinding="AulasViewIterator"
id="Aulas">
            <AttrNames>
                <Item Value="Nombre"/>
            </AttrNames>
        </list>
    </bindings>
</pageDefinition>
```



UNIVERSIDAD
DE MÁLAGA

Supuesto Nº 2

La Universidad de Málaga dispone de una cabina de discos con capacidad de ofrecer almacenamiento tanto mediante iSCSI como sobre la infraestructura de red Fibre Channel (FC). La red FC consiste en dos fabric separados a los que se conectan la cabina de discos y los distintos servidores que necesitan disponer de almacenamiento en la SAN.

Se dispone igualmente de una docena de servidores blade, con sistema IPMI para gestión y monitorización del hardware. Cada servidor tiene, aparte de una interfaz Ethernet de Management, una doble tarjeta 10 Gigabit Ethernet, así como una tarjeta HBA de dos puertos FC. No disponen de disco local, debiéndose cubrir las necesidades de disco desde la cabina en la SAN.

Sobre ellos se quieren proporcionar distintos tipos de recursos sobre una plataforma de virtualización basada en software abierto:

- despliegue de servicios comunes como syslog, servidores web, bases de datos, etc.
- despliegue de servidores virtualizados para un proyecto de análisis robotizado de muestras.
- despliegue de una solución de virtualización de escritorios para personal de administración y servicios.
- despliegue de una solución de virtualización de escritorios para alumnos.
- frontal de acceso público a algunos de los recursos anteriores.

Para la conectividad Ethernet, existe una red conmutada separada en distintas VLANs, desplegadas con el protocolo 802.1q, con uno o mas enrutadores que permiten la interconectividad entre ellas, tanto a nivel de direccionamiento IP en versión 4, como en IPv6.

Todos los servidores físicos se han conectado a los correspondientes conmutadores Ethernet (cada tarjeta a un conmutador distinto) y FC (un puerto a cada fabric).

Los servidores DNS institucionales que pueden usar todas los servidores y máquinas virtuales tienen configuradas las IPs 150.244.99.12 y 150.244.155.11. Se dispone también de un servidor proxy (redundado a nivel de DNS) proxy.uma.es que atiende en el puerto 3128 para posibles usos de todas las máquinas virtuales.

Los robots destinados a análisis están contruidos con hardware y software abiertos y disponen de conexión Ethernet, pero la implementación de seguridad es muy débil. No deben disponer de conectividad excepto contra los servidores de la aplicación que los controla.

CUESTIÓN 1 (5 puntos)

Para todos los servidores físicos y virtualizados, se requiere asignar direccionamiento tanto en IPv4 como IPv6. En cuanto a IPv4 se dispone de un pequeño rango de direccionamiento público (150.214.240.0/26) para posibles accesos desde el Internet a algunos de los servicios desplegados y del rango 192.168.128.0/17 en direccionamiento privado (RFC 1918) para todo lo demás, en el. En cuanto a IPv6, se dispone del rango 2001:720:c20:c000::/50.

a) (3 puntos) Planifique el proyecto desde el punto de vista del direccionamiento IP y la distribución de VLANs, indicando las distintas necesidades que se podrían cubrir y justificando la opción elegida en el caso en que se pueda optar por distintas opciones. Tenga en cuenta factores como la seguridad y el crecimiento futuro a la hora de diseñar el plan de direccionamiento.

b) (2 puntos) Detalle las distintas opciones de las que dispone un enrutador para influir en las configuraciones IPv6 de los equipos que se conecten a las redes que gestione.

CUESTIÓN 2 (5 puntos)

Describa los distintos pasos a seguir tanto a nivel de la cabina de discos, fabric y a nivel de los distintos servidores para lograr el uso de almacenamiento que se solicita.

- a) (1.5 puntos) Se necesita que los servidores físicos sean capaces de arrancar desde los discos proporcionados por el sistema de almacenamiento en SAN. En la cabina de disco existe una LUN que contiene una imagen de un disco de arranque con un OS Centos 7, sobre el que existe un entorno de virtualización que permite desplegar VMs mediante KVM y mediante contenedores LXC, que se debe usar para este requerimiento.
- b) (2 puntos) Igualmente, a cada servidor se debe proveer de un único espacio de almacenamiento común a todos ellos ofertado sobre FC desde la cabina de discos, Dicho espacio debe ser visible a nivel de Centos como un sistema LVM, sobre el que sea posible desplegar los discos virtuales de las distintas máquinas virtuales (VMs) que se proporcionen.
- c) (1.5 puntos) Además, uno de los servidores debe poder montar un espacio de almacenamiento ofertado por la cabina de discos via SCSI, en el que se almacenará el sistema centralizado de logs.

CUESTIÓN 3 (5 puntos)

Para el despliegue de máquinas virtuales en cada uno de los servidores físicos, como contenedores LXC, se necesita definir la conectividad de red que van a utilizar las distintas máquinas virtuales (VM). Cada servidor físico se ha configurado con su interfaz eth0 en la VLAN correspondiente a la de acceso a los servidores que alojan las VM, y su interfaz eth1 con un enlace contra el conmutador ethernet en modo trunk, por el que reciben varias VLANs definidas para las distintas VM. Asuma que se han definido tres de estas VLANs con los IDs 200, 210 y 220.

- a) (1.25 puntos) Indique la configuración de la interfaz de red de la puerta del conmutador al que conecta cada interfaz eth1 de los servidores, que debe reflejar el hecho de que no deje pasar tramas Ethernet pertenecientes a otra VLANs distintas a las indicadas. Asuma que se trata de un conmutador del fabricante Cisco o, en su caso, hágalo para un conmutador imaginario, indicando los detalles a contemplar.
- b) (1.25 puntos) Indique las configuraciones de ambas interfaces eth0 y eth1 en uno de los servidores físicos. En el caso de la segunda, basta con que lo haga para una de las VLANs que recibe. En ambos casos, esta configuración debe asignar una IPv4 manualmente a la interfaz, y configurar IPv6 para que pueda conseguir direccionamiento de modo automático a partir de los anuncios RA del router correspondiente.
- c) (1.25 puntos) A la hora de desplegar VMs mediante LXC sobre los equipos, se desea que algunas de ellas tengan visibilidad en la red Ethernet externa adonde están definidas las distintas VLANs mientras que otras deben permanecer invisibles y lograr la conectividad a través de los servidores físicos que los alojan. Indique cómo se podría lograr mediante el uso de bridges de Linux. No es necesario indicar los comandos de configuración.
- d) (1.25 puntos) Se ha desplegado una máquina virtual con KVM en la que se instala un Windows 10 Profesional. Utilizando la línea de comandos, indique las instrucciones para configurar con una

dirección IP el equipo, para facilitarle el acceso web a Internet y para registrar o instalar la clave de producto de Windows 10 (p.ej. NRH6B-ZKK1X-CXVVJ-9G5XM-9Q84J) forzando su activación mediante un servidor KMS ya existente en la misma red.

CUESTIÓN 4 (5 puntos)

Se ha decidido migrar el MTA del servicio de correo desde una configuración monolítica en un servidor usando software libre (postfix y Dovecot), a una solución con múltiples etapas, distribuida y de alta disponibilidad. Dicho MTA, entrega los mensajes bien al servidor de buzones, bien al servidor de listas o bien a estafetas secundarias, según indique la información de encaminamiento.

a) (3 puntos) Diseñe las distintas etapas que podrían tenerse en cuenta, desde el punto de vista de la mayor separación posible de tareas, indicando brevemente las funciones a realizar por cada una y los flujos de los mensajes. Considere que la información sobre las direcciones, cuentas de usuario (y, por tanto, su autenticación) y sus buzones asociados se almacena en un servidor LDAP mediante los atributos adecuados. No es necesario incluir configuraciones. Considere apoyar su descripción con un gráfico.

b) (2 puntos) Una de las principales preocupaciones que se quieren abordar en el servicio de correo es la mayor protección posible frente a correos maliciosos y no deseados. Indique varios mecanismos que se puedan tener en cuenta para ello.

CUESTIÓN 5 (5 puntos)

La Universidad ha adquirido un conjunto de robots de pipeteo que permiten el procesado automatizado de muestras para extracción RT-PCR. Se ha realizado un convenio con diversas universidades europeas para llevar a cabo un estudio sobre la fiabilidad de la toma de muestras por los propios sujetos. Para ello, se dispondrá en las universidades participantes de tubos porta-muestras pre-identificados con un código QR único, que se enviarán al laboratorio de los robots para su procesamiento.

El estudio es anónimo a todos los efectos, salvo que se deberá informar de los resultados a las personas que participan, con la máxima rapidez y privacidad. Por ello, es necesario asociar las muestras a las personas por medio de una App para móvil, preservando su privacidad. Las personas podrán asociar más de un dispositivo para capturar muestras y recibir notificaciones sobre los resultados, pero todos ellos presentarán la información relativa a la misma persona.

Cualquier universidad europea puede adherirse al convenio. La gestión del registro de dispositivos y personas se realizará con una aplicación web Django accesible desde Internet. Los miembros de las universidades participantes podrán autenticarse en el sistema usando las facilidades de la inter-federación eduGAIN (inter-federación de la federación de identidad de las redes nacionales de educación e investigación de alcance mundial basada en protocolo SAML2) y registrar sus dispositivos usando la App móvil que se les proporciona para las funcionalidades del proyecto.

Defina un procedimiento de control de acceso a la aplicación web para registro de personas de orígenes diversos con el máximo nivel de anonimato, utilizando los servicios de eduGAIN. Puede mencionar distintos mecanismos para transferir la información de autenticación recibida de eduGAIN a la aplicación web Django.

CUESTIÓN 6 (5 puntos)

Teniendo en cuenta la información del apartado 5:

- a) (3 puntos) Describa y explique el modelo de datos para Django de la aplicación Web.

- b) (2 puntos) Explique brevemente las funciones básicas de la App móvil y el soporte necesario en la aplicación Django, para gestionar las muestras, el registro de dispositivos móviles asociados a personas, la carga de resultados (a realizar por un operador manualmente, muestra a muestra o por lotes) y las notificaciones a las personas participantes.