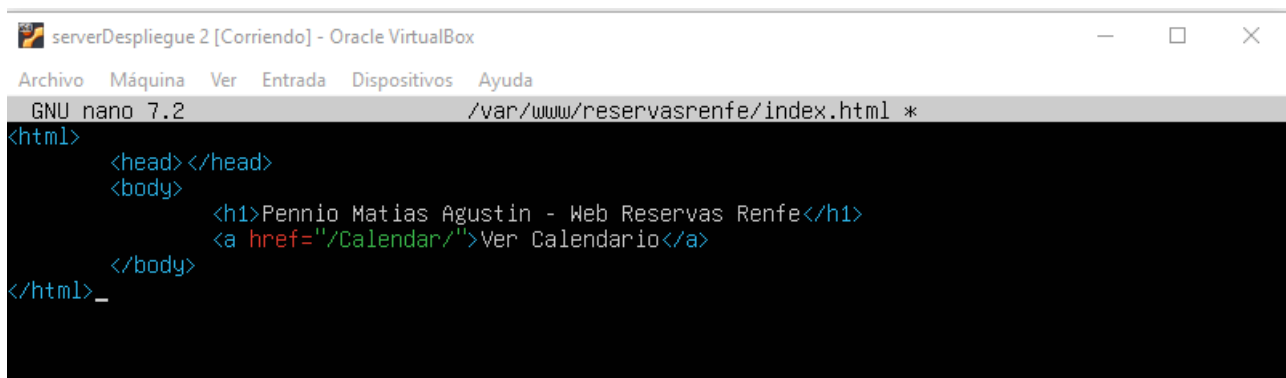


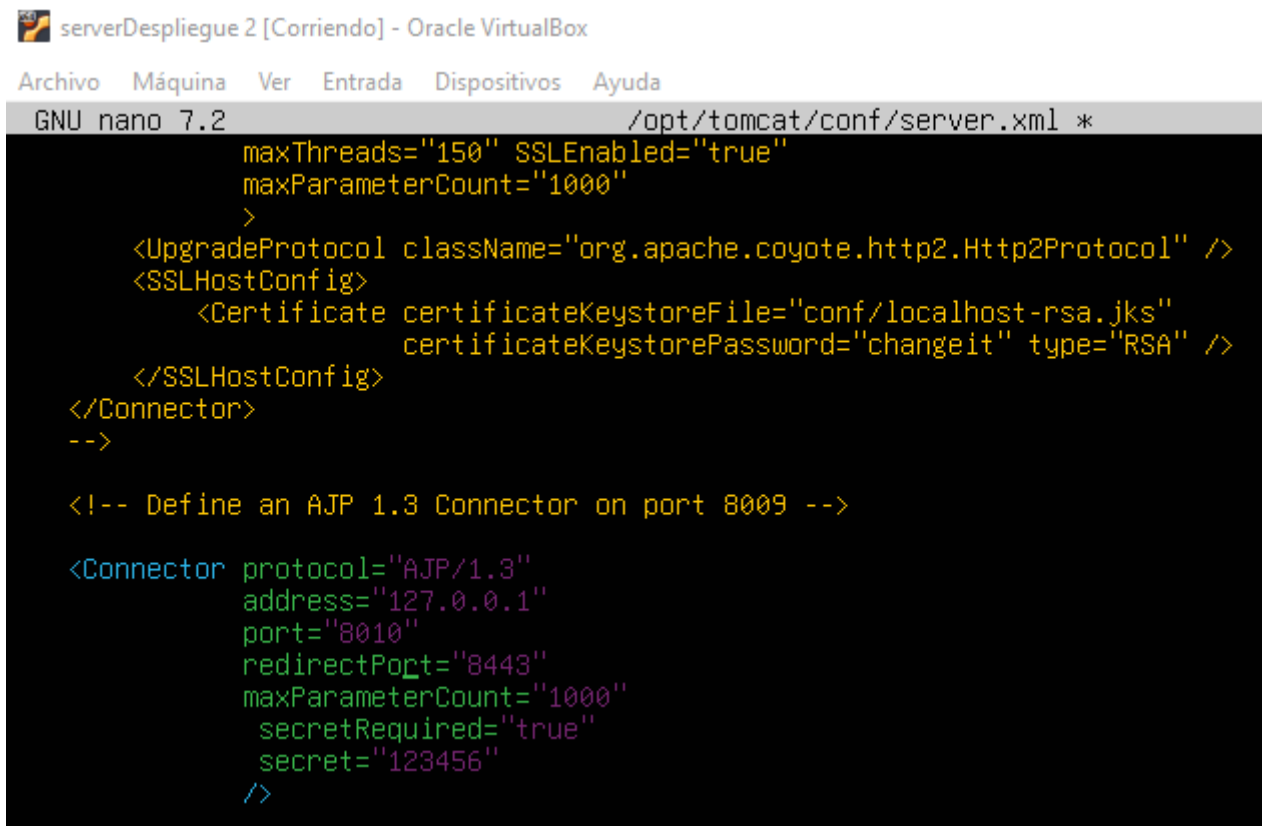
EXAMEN DESPLIEGUE DE APLICACIONES WEB

Alumno: Matias Agustin Pennino



```
serverDespliegue 2 [Corriendo] - Oracle VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
GNU nano 7.2 /var/www/reservasrenfe/index.html *
<html>
  <head></head>
  <body>
    <h1>Pennio Matias Agustin - Web Reservas Renfe</h1>
    <a href="/Calendar/">Ver Calendario</a>
  </body>
</html>_
```

Para la creación de la pagina creamos el directorio **reservasrenfe** y el index mostrado en la imagen



```
serverDespliegue 2 [Corriendo] - Oracle VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
GNU nano 7.2 /opt/tomcat/conf/server.xml *
    maxThreads="150" SSLEnabled="true"
    maxParameterCount="1000"
  >
  <UpgradeProtocol className="org.apache.coyote.http2.Http2Protocol" />
  <SSLHostConfig>
    <Certificate certificateKeystoreFile="conf/localhost-rsa.jks"
      certificateKeystorePassword="changeit" type="RSA" />
  </SSLHostConfig>
</Connector>
-->

<!-- Define an AJP 1.3 Connector on port 8009 -->

<Connector protocol="AJP/1.3"
  address="127.0.0.1"
  port="8010"
  redirectPort="8443"
  maxParameterCount="1000"
  secretRequired="true"
  secret="123456"
/>
```

Antes de continuar, configuraremos como se comunicaran tomcat y apache. El puerto que utilizaran es el 8010 y lo cambiamos en el connector del modulo ajp

```
serverDespliegue 2 [Corriendo] - Oracle VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
GNU nano 7.2 /etc/apache2/workers.properties *
# Define 1 real worker using ajp13
worker.list=worker1

# Set properties for worker1 (ajp13)
worker.worker1.type=ajp13
worker.worker1.host=localhost
worker.worker1.port=8010_
worker.worker1.secret=123456
```

Tambien le decimos al worker que debe utilizar ese puerto

```
serverDespliegue 2 [Corriendo] - Oracle VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
GNU nano 7.2 /etc/apache2/sites-available/reservasrenfe.conf *
<VirtualHost *:80>
    # The ServerName directive sets the request scheme, hostname and port that
    # the server uses to identify itself. This is used when creating
    # redirection URLs. In the context of virtual hosts, the ServerName
    # specifies what hostname must appear in the request's Host: header to
    # match this virtual host. For the default virtual host (this file) this
    # value is not decisive as it is used as a last resort host regardless.
    # However, you must set it for any further virtual host explicitly.
    #ServerName www.example.com

    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/reservasrenfe
    ServerName reservasrenfe.local
    ServerAlias www.reservasrenfe.local

    JkMount /Calendar worker1
    JkMount /Calendar/* worker1_
    # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
    # error, crit, alert, emerg.
    # It is also possible to configure the loglevel for particular
    # modules, e.g.
    #LogLevel info ssl:warn
```

Configuramos el virtual host para que utilice el worker creado y le hacemos su a2ensite para dejarlo listo

Luego de configurar apache subiremos desde el gestor de aplicaciones de tomcat el archivo **Calendar.war**, que sera nuestra pagina de calendario

← No seguro | 192.168.56.101:8080/manager/html/upload?org.apache.catalina.filters.CSRF_NONCE=05D1BB3FD64DC09FAC9A8C3586548FB8

Contexto	Nombre	Descripción	Estado	Acciones
/host-manager	Ninguno especificado	Tomcat Host Manager Application	true	0
/manager	Ninguno especificado	Tomcat Manager Application	true	1
/migraterservlet	Ninguno especificado	Hello World	true	0

Desplegar

Desplegar directorio o archivo WAR localizado en servidor

Trayectoria de Contexto (opcional):

Version (for parallel deployment):


URL de archivo de Configuración XML:

URL de WAR o Directorio:

Archivo WAR a desplegar

Seleccione archivo WAR a cargar

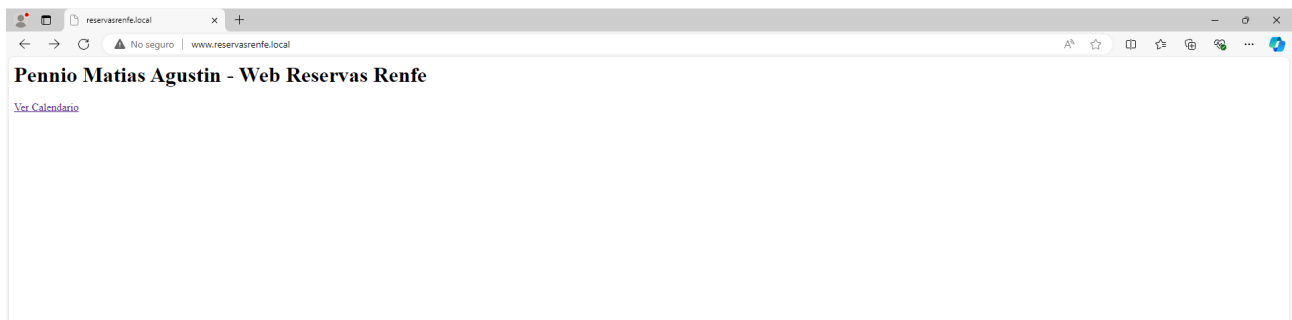
Ya con todo subido reiniciamos apache y tomcat para ver como quedan las paginas

 hosts: Bloc de notas

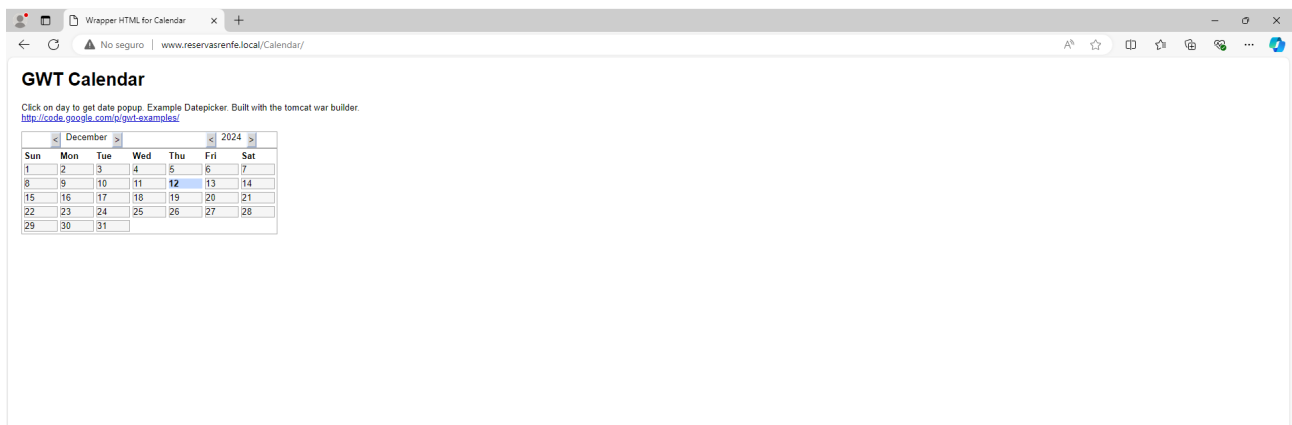
Archivo Edición Formato Ver Ayuda

```
# Copyright (c) 1993-2009 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name.
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#       102.54.94.97       rhino.acme.com          # source server
#       38.25.63.10       x.acme.com              # x client host
#
# localhost name resolution is handled within DNS itself.
#       127.0.0.1         localhost
#       ::1               localhost
#
192.168.56.101           www.reservasrenfe.local
```

En el archivo **hosts** configuramos la direccion del servidor para poder acceder a **reservasrenfe**



Obteniendo este resultado



Y al hacer click en el enlace vemos como nos redirige correctamente a la pagina del calendario

Ahora configuraremos la pagina para que solo se pueda conectar el usuario **examen2425**

- manager-jmx - allows access to the JMX proxy and the status pages
- manager-status - allows access to the status pages only

The users below are wrapped in a comment and are therefore ignored. If you wish to configure one or more of these users for use with the manager web application, do not forget to remove the `<!-- .. -->` that surrounds them. You will also need to set the passwords to something appropriate.

-->

<!--

```
<user username="admin" password="<must-be-changed>" roles="manager-gui"/>
```

```
<user username="robot" password="<must-be-changed>" roles="manager-script"/>
```

-->

<!--

The sample user and role entries below are intended for use with the examples web application. They are wrapped in a comment and thus are ignored when reading this file. If you wish to configure these users for use with the examples web application, do not forget to remove the `<!-- .. -->` that surrounds them. You will also need to set the passwords to something appropriate.

-->

<!--

```
<role rolename="tomcat"/>
```

```
<role rolename="role1"/>
```

```
<user username="tomcat" password="<must-be-changed>" roles="tomcat"/>
```

```
<user username="both" password="<must-be-changed>" roles="tomcat,role1"/>
```

```
<user username="role1" password="<must-be-changed>" roles="role1"/>
```

-->

```
<role rolename="manager-gui"/>
```

```
<role rolename="admin-gui"/>
```

```
<role rolename="usuario"/>
```

```
<role rolename="examen"/>
```

```
<user username="examen2425" password="123456" roles="examen"/>
```

Primero creamos el usuario y su rol

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<web-app>
<display-name>gwt-Calendar Compiled: Sun Jan 27 06:17:26 GMT 2008</display-name>
<description>Google Web Toolkit Project</description>
<security-constraint>
  <display-name>Examen</display-name>
  <web-resource-collection>
    <web-resource-name>Examen</web-resource-name>
    <url-pattern>/*</url-pattern>
  </web-resource-collection>
  <auth-constraint>
    <security-role>examen</security-role>
  </auth-constraint>
</security-constraint>
<security-role>
  <rolename>examen</rolename>
</security-role>
<login-config>
  <auth-method>BASIC</auth-method>
</login-config>
</web-app>
```

Luego en el directorio de la aplicación configuraremos el archivo **web.xml** para crear la restricción de seguridad

(La estructura del archivo es esa pero las etiquetas no son correctas, por esa razón la restricción no funciona)