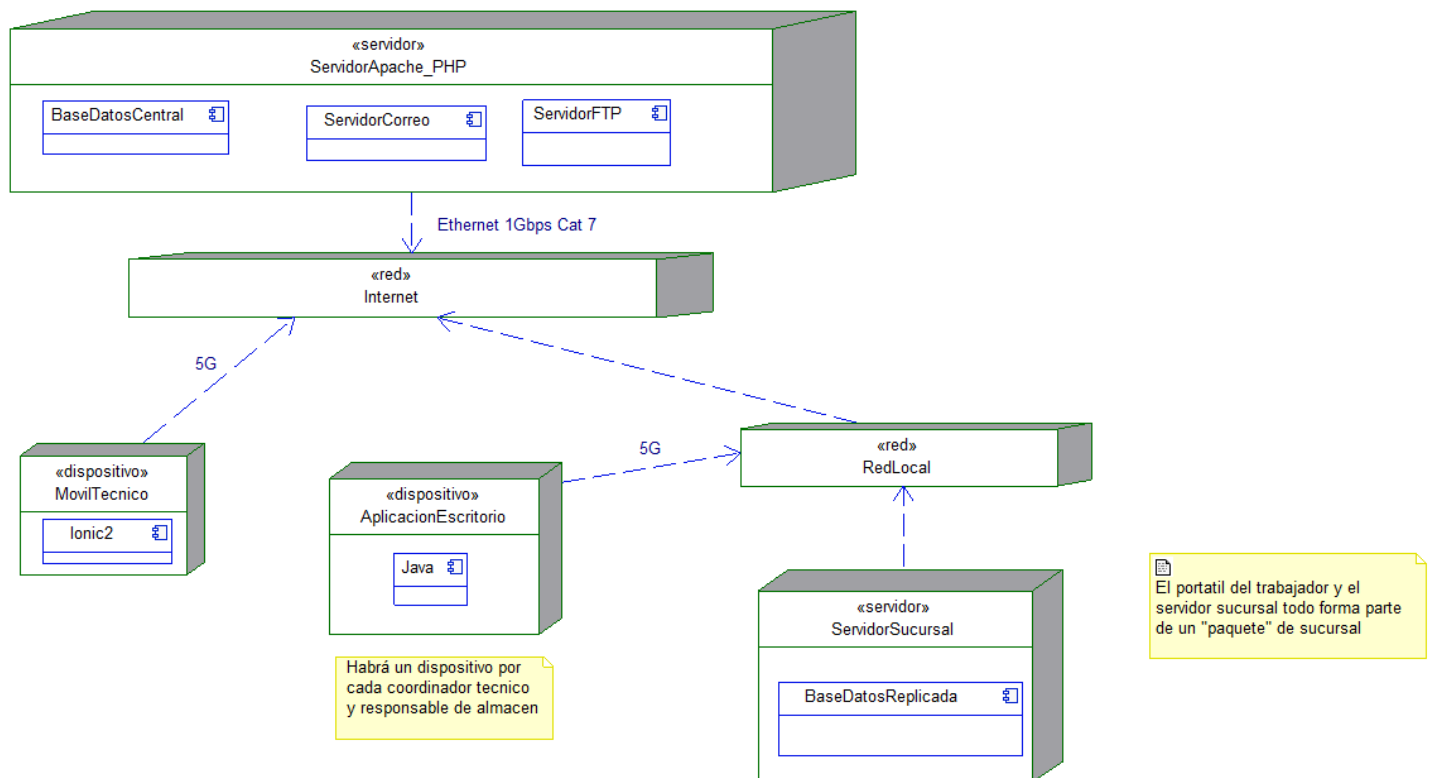


## 10 Modelado de la arquitectura

### 1.1 Diagrama de despliegue

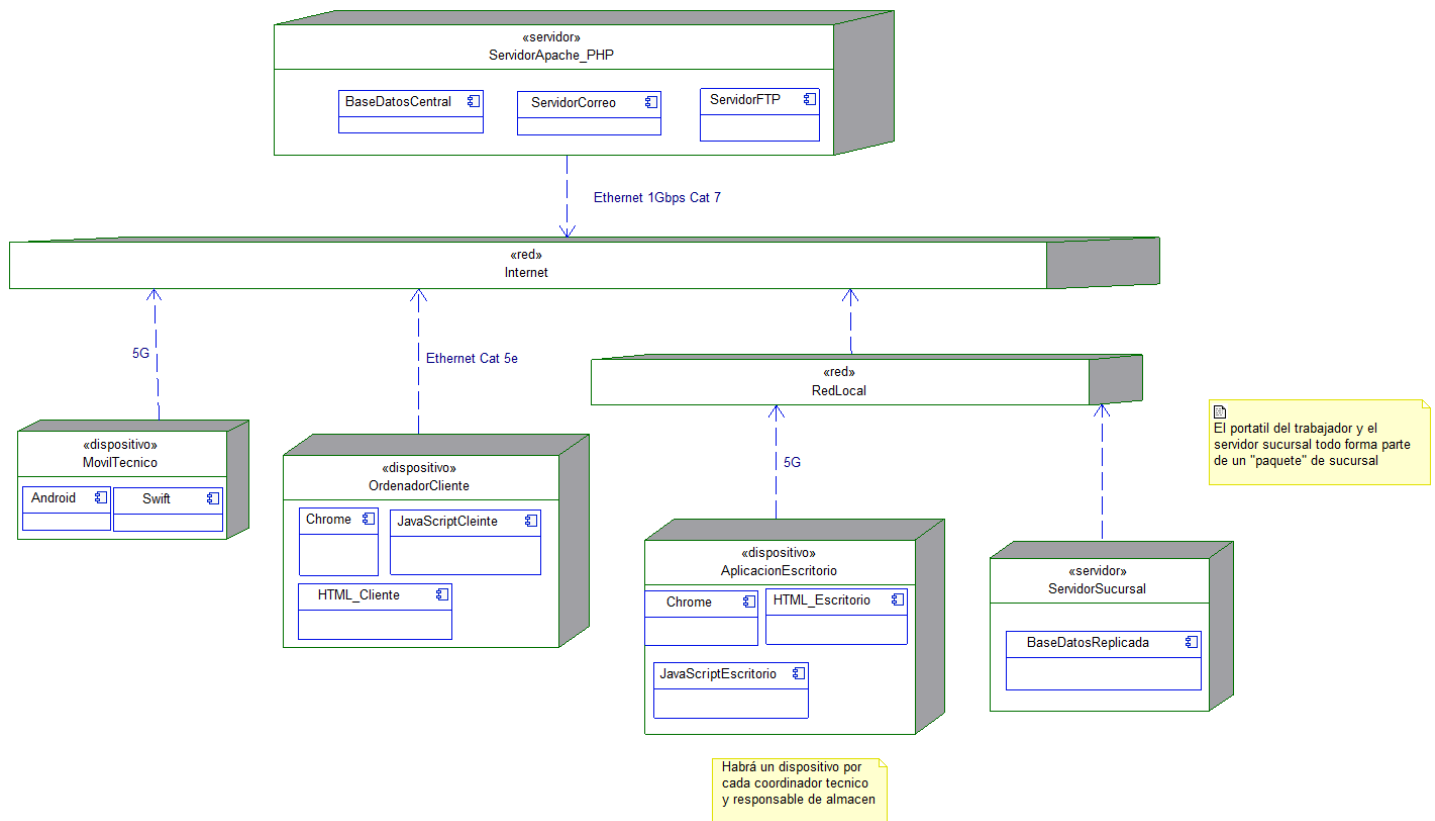
#### 1.1.1 Diagrama de despliegue – SOLUCIÓN 1

En todas las versiones tendremos un servidor Apache\_PHP que funcionará como servidor central del sistema, dicho servidor contiene la base de datos central, un servidor de correo y un servidor FTP (donde se almacenan los informes). Todos los técnicos se conectarán a dicho servidor a través de Internet mediante sus dispositivos móviles, los cuales tendrán en esta versión Ionic2. Todas las sucursales dispondrán de un servidor donde se encuentra la base de datos central replicada y aplicaciones de escritorio (uno por coordinador y responsable) que disponen de Java, todas esas aplicaciones de escritorio y el servidor de la sucursal están conectados mediante una red local.



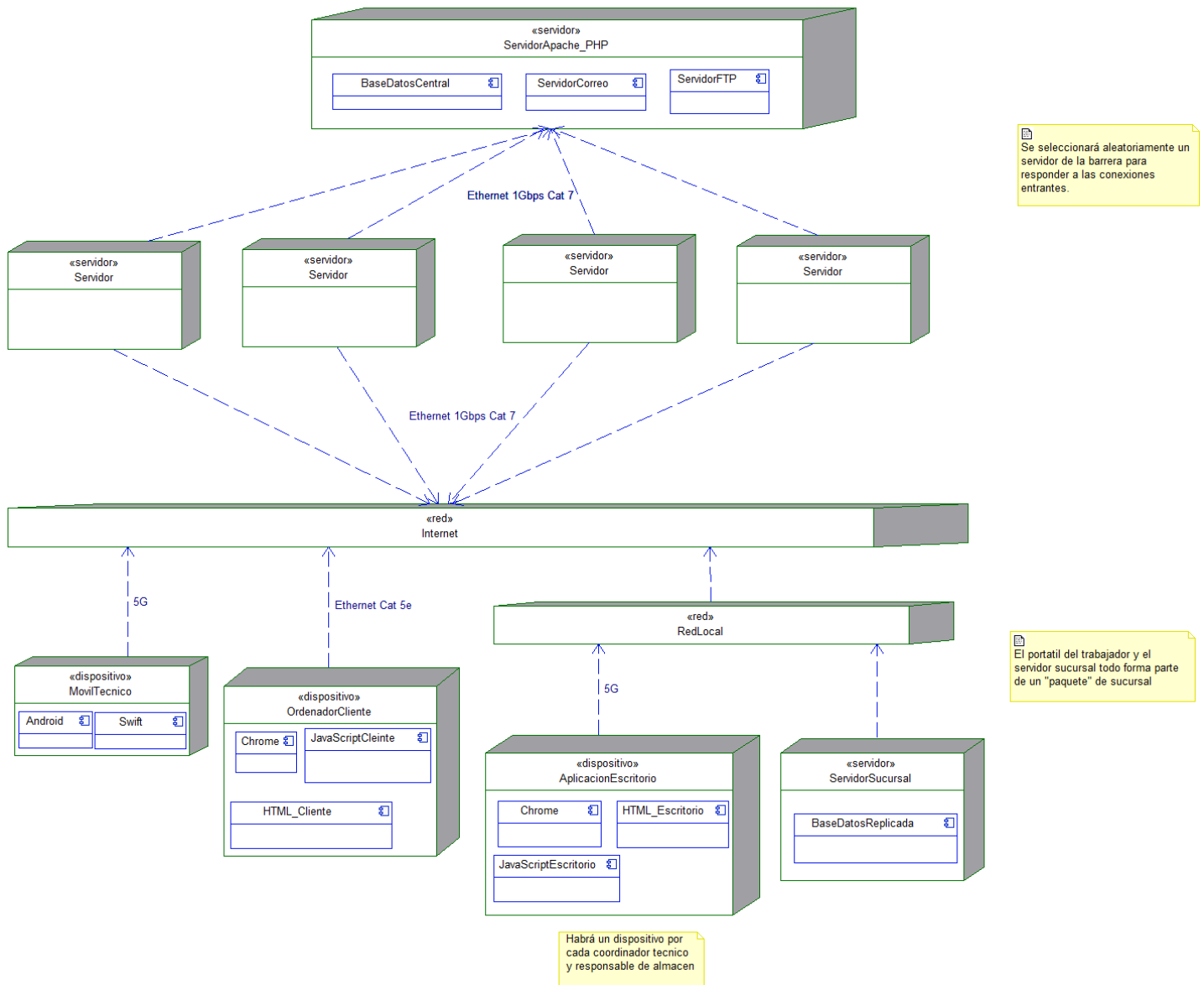
#### 1.1.2 Diagrama de despliegue – SOLUCION 2

Esta solución parte de la solución anterior. A diferencia de la solución anterior los móviles de los técnicos, en vez de Ionic2, disponen tanto de Android como de Swift. En esta versión se añade la funcionalidad de los ordenadores de los clientes, los cuales incorporarán JavaScript, HTML y Chrome. En lo respectivo a la sucursal la aplicación de escritorio ya no cuenta con Java, sino que al igual que en el caso de los clientes, dispone de JavaScript, HTML y Chrome.



### 1.1.3 Diagrama de despliegue – SOLUCIÓN 3

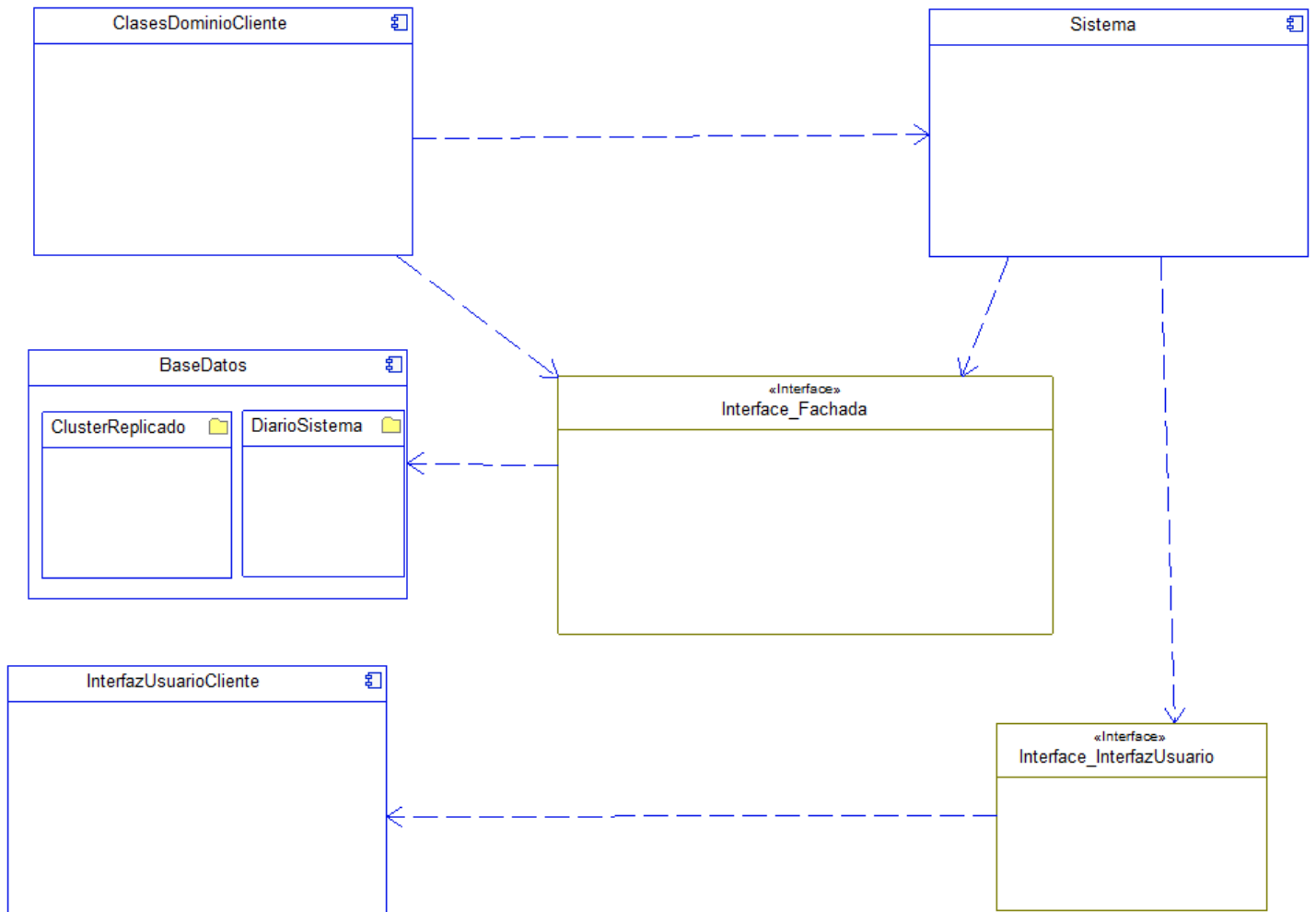
La única diferencia de la solución 2 es que disponemos de una serie de servidores barreras que funcionan de intermediarios entre el servidor Apache central e Internet.



## 1.2 Diagrama de componentes

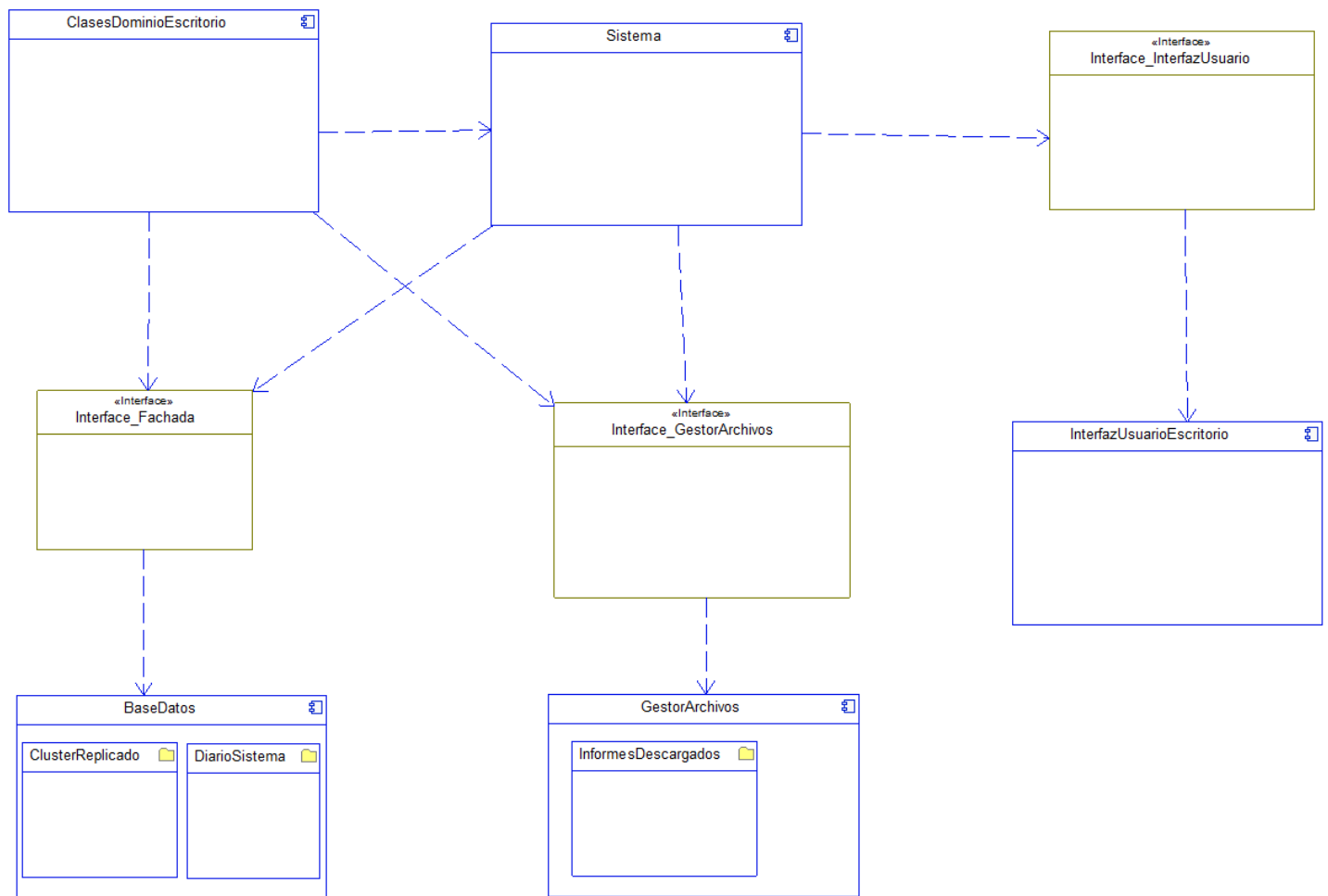
### 1.2.1 Diagrama de componentes de Cliente – SÓLO SOLUCIÓN 3

Este diagrama de componentes corresponde a la aplicación web del cliente, que pertenece a la solución 3 (en el resto de las soluciones los clientes no son actores del sistema) de nuestra aplicación. Para desarrollarla partimos del diagrama de clases de cliente. Dentro de cada componente tendremos un número de clases agrupadas en ellos (para ver las clases que están agrupadas dentro de cada componente debemos hacer click derecho en el componente -> features -> scope), por ello, las clases del dominio estarán relacionadas con las clases de sistema para que puedan acceder a ellas. Luego el sistema para relacionarse tanto con las clases de la interfaz de usuario como con las de la base de datos utiliza interfaces especializadas para la comunicación entre estos dos componentes. Luego dentro de la base de datos tenemos la carpeta cluster replicado que guarda la parte que se vaya a usar de la cluster original de la base de datos y en la carpeta diario de sistema guardaremos las modificaciones realizadas sobre el cluster.



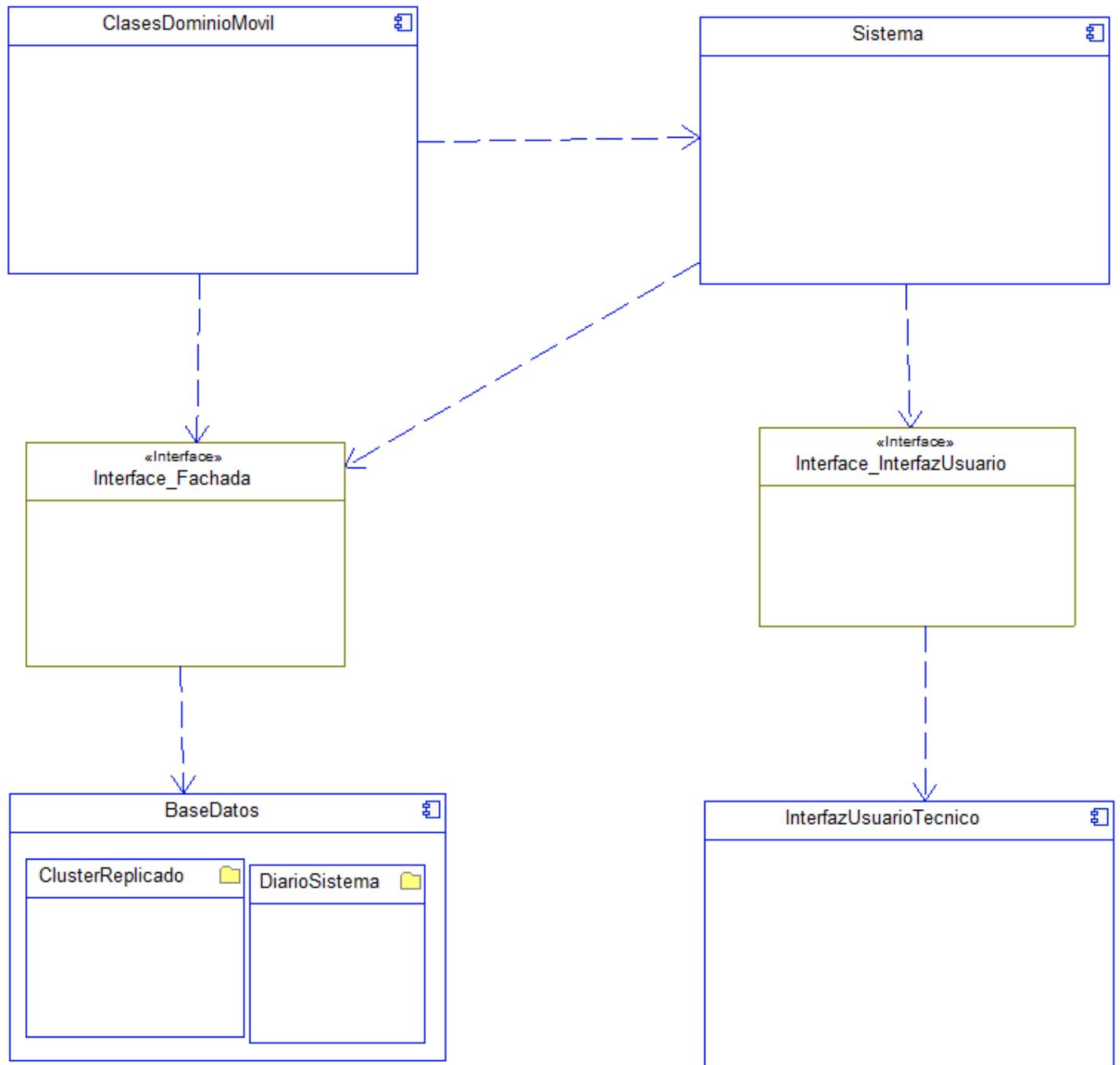
### 1.2.2 Diagrama de componentes de Escritorio

Dentro de cada componente nos encontraremos con las clases correspondientes, basándonos en el diagrama de clases de la aplicación de escritorio. En este caso nos encontramos con un gestor de archivos, por el cual mediante la interfaz gestor archivos accedemos a las clases que controlan el uso de los informes, los informes están almacenados en el servidor Apache en una carpeta llamada Informes. Con estas clases analizaremos una conexión FTP al servidor de modo que podremos descargarnos dichos archivos y la guardaremos en la carpeta InformesDescargados.



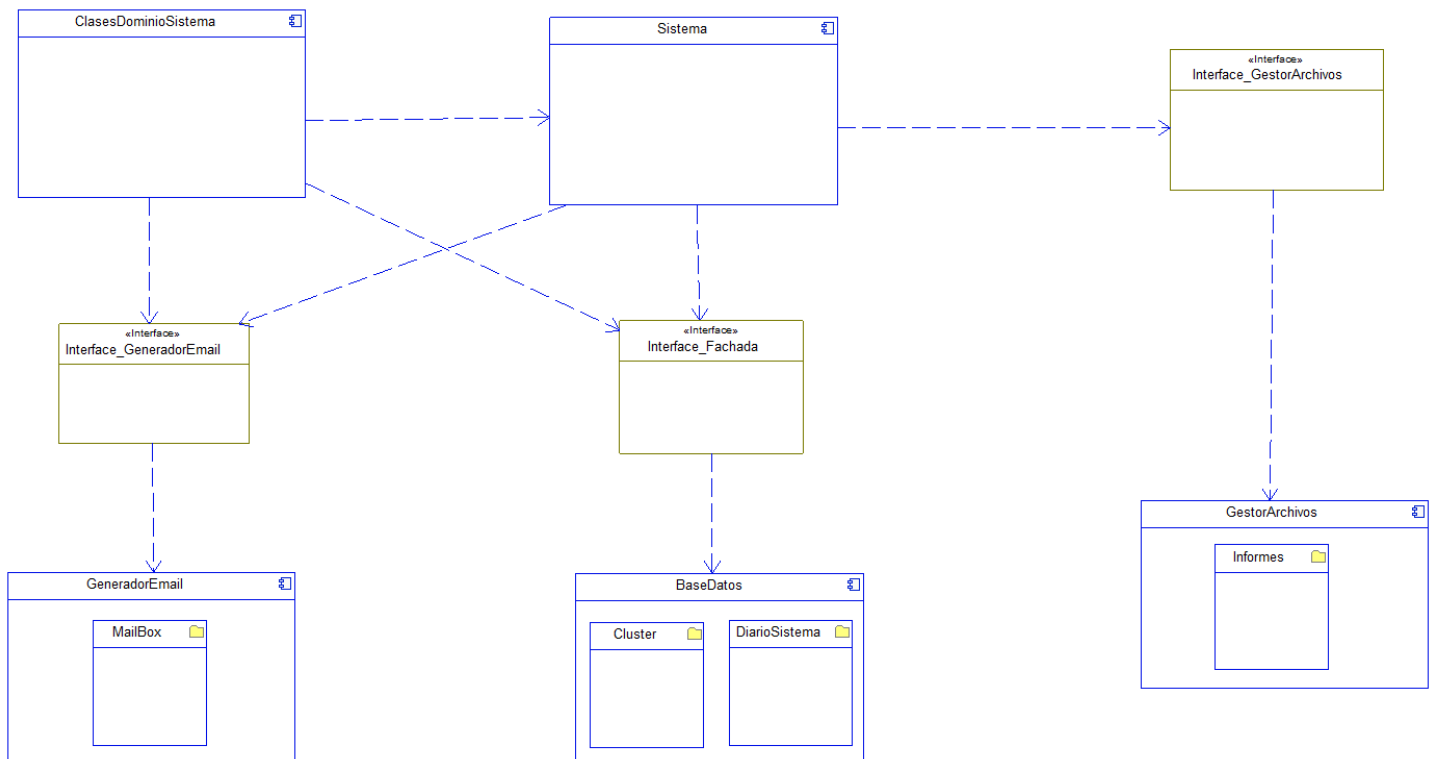
### 1.2.3 Diagrama de componentes de Móvil

Para realizar este diagrama nos basamos en el diagrama de clases de la aplicación móvil, y agrupamos las clases utilizadas en el diagrama de clases agrupándola en los distintos componentes aquí reflejados.



#### 1.2.4 Diagrama de componentes de Sistema

Para realizar este diagrama de componentes nos hemos basado en el diagrama de clases del Sistema. Aquí nos encontramos con la carpeta MailBox dentro de GeneradorEmail en la cual guardamos los correos tanto creados como recibidos. Y la carpeta Informes del gestor de archivos será publica para las conexiones FTP realizadas por la aplicación de escritorio del sistema.



### 1.2.5 Diagrama de componentes de Desarrollo

Aquí reflejamos como vería el programador los paquetes del desarrollo donde estarían cada una de las clases a utilizar.

A la hora de lanzar una aplicación se seleccionarían los archivos que van a formar parte de ella y se compilarían para formar la aplicación y así solo utilizar lo que vayamos a utilizar, que es lo que exponemos en el resto de los diagramas.

