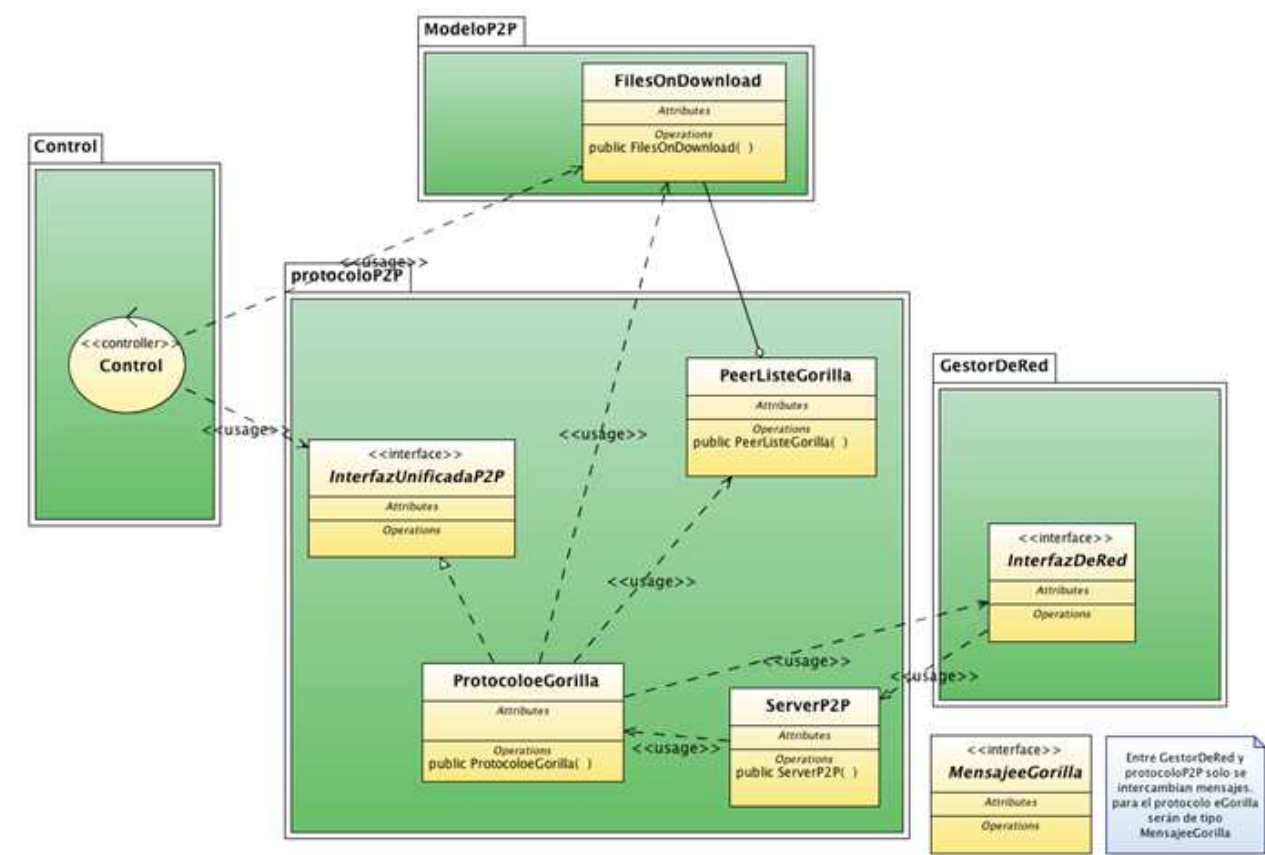




Vista Descomposición ProtocoloP2P:

1. Diagrama de Contexto:



2. Catálogo de elementos:

ProtocoloP2P	
Tipo:	Paquete
Descripción:	Contiene la lógica del protocolo p2p, ya sea eGorilla u otro implementado.
Interfaces:	Se comunica con el gestor de red mediante mensajes, recibe ordenes del control y modifica los datos de modeloP2P





ModeloP2P	
Tipo:	Paquete
Descripción:	Contiene el estado de nuestra sesión p2p, almacenando el estado individual de cada uno de los ficheros que descargamos y de los que subimos.
Interfaces:	Es modificado por el ProtocoloP2P, el control puede interrogarle para averiguar el estado de los elementos.

GestorDeRed	
Tipo:	Paquete
Descripción:	Contiene el bajo nivel de las comunicaciones de red, UDP o TCP.
Interfaces:	Se comunica con el protocoloP2P mediante mensajes.

ProtocoloP2P	
Tipo:	Paquete
Descripción:	Contiene la lógica del protocolo p2p, ya sea eGorilla u otro implementado.
Interfaces:	Se comunica con el gestor de red mediante mensajes, recibe ordenes del control y modifica los datos de modeloP2P

ProtocoloP2P	
Tipo:	Paquete
Descripción:	Contiene la lógica del protocolo p2p, ya sea eGorilla u otro implementado.
Interfaces:	Se comunica con el gestor de red mediante mensajes, recibe ordenes del control y modifica los datos de modeloP2P

MensajeeGorilla	
Tipo:	Interfaz





Descripción:	Es un mensaje de los diferentes que puede haber en el protocolo p2p.
Interfaces:	Sirve de comunicación entre el protocoloP2P y el modulo de red.

InterfazUnificadaP2P	
Tipo:	Interfaz
Descripción:	Interfaz de ordenes para interactuar con cualquier protocolo p2p.
Interfaces:	Recibe ordenes del control de la aplicación. Convierte estas en un estado y comienza mecanismos para satisfacerlas.

ProtocoloeGorilla	
Tipo:	Clase-implementación
Descripción:	Satisface la interfazUnificadaP2P para poder comunicarnos con otros clientes y con servidores eGorilla.
Interfaces:	Recibe ordenes del Control y las resuelve mediante el protocolo eGorilla.

PeerListeGorilla	
Tipo:	Clase
Descripción:	Lista los clientes con los que tenemos algún tipo de transito. Ya sea cliente-servidor o servidor cliente. Para cada cliente se le vinculará con los archivos involucrados.
Interfaces:	Lo usa el ProtocoloeGorilla para almacenar los extremos de las transferencias eGorilla.

FilesOnDownload	
Tipo:	Clase Almacenamiento.





Descripción:	Conservará el estado de todos los archivos con los que se relaciona este cliente, tanto los compartidos como los que están en proceso de descarga. Será capaz de ser almacenado de forma no volátil para continuar en diferentes ejecuciones.
Interfaces:	Se comunica con ProtocoloeGorilla, control y con los peer almacenados en PeerListeGorilla

3. Información sobre la arquitectura

a) Elementos y propiedades:

La arquitectura parece un poco complicada para lo que podria ser el sistema, pero de esta manera nos aseguramos la compatibilidad con futuros cambios.

El diseño se ha hecho pensando en transferencias aisladas y no en conexiones establecidas, de forma que se permite el uso de otros protocolos para poder utilizar otras técnicas.

Ademas se añade la posibilidad de añadir compatibilidad con otras arquitecturas de p2p al solo tener que implementar la interfazUnificadaP2P.

4. Guia de variabilidad

El protocolo aquí dibujado hace referencia EXCLUSIVAMENTE al protocolo eGorilla, de forma que se establecera una interfaz de comunicación con el resto de paquetes y control de aplicación generico. Para que pueda ser añadida compatibilidad con cualquier otro protocolo.

Añadir nuevos diálogos al protocolo no debere modificar el comportamiento de los ya establecidos, ya que cada uno de los mensajes será procesado de forma independiente.





5. Vistas relacionadas:

a) Vista Cliente Servidor:

b) Modelos de actividad:

i) *holaquiero.png*

ii) *tengo.png*

iii) *dame.png*

iv) *toma.png*

v) *Altooo.png*

