

GUIA PARA UTILIZAR EL REPOSITORIO VIPERD CON NETBEANS

El repositorio VIPERD contiene un conjunto de bibliotecas y proyectos de software que implementan una arquitectura limpia (Clean Architecture) basada en el patrón VIPER. Con estos componentes se pueden desarrollar aplicaciones JAVA para entornos web, desktop y móvil que comparten un mismo diseño general de áreas / componentes y reutilizan la mayoría de su código.

El diseño de este paquete fue concebido inicialmente como una herramienta para facilitar el desarrollo ágil y adaptativo de proyectos pequeños-medianos de software, que por un lado permitiera cumplir con los principios SOLID de la programación orientada a objetos, y por otra parte ayudara a desarrollar aplicaciones modulares de calidad, fácilmente mantenibles, testeables, escalables y reutilizables, reduciendo la cantidad y mejorando la legibilidad del código de programación, y proporcionando un modelo conceptual y lógico para facilitar la división técnica del trabajo, el aprendizaje colaborativo y la comunicación entre los profesionales y equipos de desarrollo.

La distribución actual del repositorio VIPERD incluye 5 proyectos relacionados.

- **ViperdInterno:** es la biblioteca básica que se utiliza en todos los entornos de desarrollo y ejecución, programada 100% en JAVA, sin “dependencias externas” (multi-plataforma). Contiene las clases situadas en el “interior” de todas las Aplicaciones (el núcleo de dominio y los adaptadores para interfaz de usuario y de datos). Se requiere también para las Aplicaciones Móviles (Android).
- **ViperdDesktop:** es la biblioteca complementaria que se utiliza en entornos Desktop (computadoras Windows, Linux, Mac OSX), programada en JAVA usando componentes y controladores compatibles con estas plataformas. Contiene las clases situadas en el “exterior” de las Aplicaciones Desktop (interfaz de usuario e interfaz de datos).
- **ViperdWeb:** es la biblioteca complementaria que se utiliza en entornos Web (servidores de aplicaciones Java y contenedores de Servlets), programada en JAVA usando componentes y controladores compatibles con la plataforma web. Contiene las clases situadas en el “exterior” de las Aplicaciones Web (interfaz de usuario e interfaz de datos).
- **PruebaDesktop:** es una aplicación demo básica (en construcción) para implementar la arquitectura VIPERD Desktop en una pieza de software funcional (por ahora sólo probada en Windows).
- **PruebaWeb:** es una aplicación demo básica (en construcción) para implementar la arquitectura VIPERD Web en una pieza de software funcional (por ahora sólo probada en entorno de desarrollo local).

Se incluye además una carpeta “viperd”, donde se almacenan los archivos .jar, las bibliotecas de dependencias externas, recursos, bases de datos, plantillas, carpetas y archivos necesarios para poder utilizar esta distribución como un mini “kit de desarrollo” de aplicaciones. Todo lo anterior ha sido empaquetado en el repositorio VIPERD para facilitar su utilización directa en el IDE NetBeans.

Antes de comenzar a utilizar el repositorio VIPERD, sugerimos que actualice la versión de Java Runtime Environment (JRE) instalada en su computadora de desarrollo. Actualmente recomendamos instalar la versión 8, disponible en:

<https://www.java.com/es/download/>

Para ejecutar proyectos Java web recomendamos instalar la distribución completa de NetBeans (“All”, que incluye los servidores GlassFish y Apache Tomcat), aunque también sirve la distribución “Java EE”. La versión 8.2 está disponible en el sitio oficial:

<https://netbeans.org/downloads/index.html>

En la siguiente vista se muestra la página de descarga, indicando las distribuciones recomendadas.

NetBeans IDE 8.2 Download 8.1 | 8.2 | Development | Archive

Email address (optional):

Subscribe to newsletters: ☒ Monthly ☐ Weekly

☒ NetBeans can contact me at this address

IDE Language: Español Platform: Windows

Note: Greyed out technologies are not supported for this platform.

NetBeans IDE Download Bundles in community contributed languages

Supported technologies *	Java SE	Java EE	HTML5/JavaScript	PHP	C/C++	All
NetBeans Platform SDK	•	•				•
Java SE	•	•				•
Java FX	•	•				•
Java EE		•				•
Java ME						•
HTML5/JavaScript		•	•	•		•
PHP			•	•		•
C/C++					•	•
Groovy						•
Java Card™ 3 Connected						•
Bundled servers						
GlassFish Server Open Source Edition 4.1.1		•				•
Apache Tomcat 8.0.27		•				•

Download **Download** **Download x86** **Download x86** **Download x86** **Download**

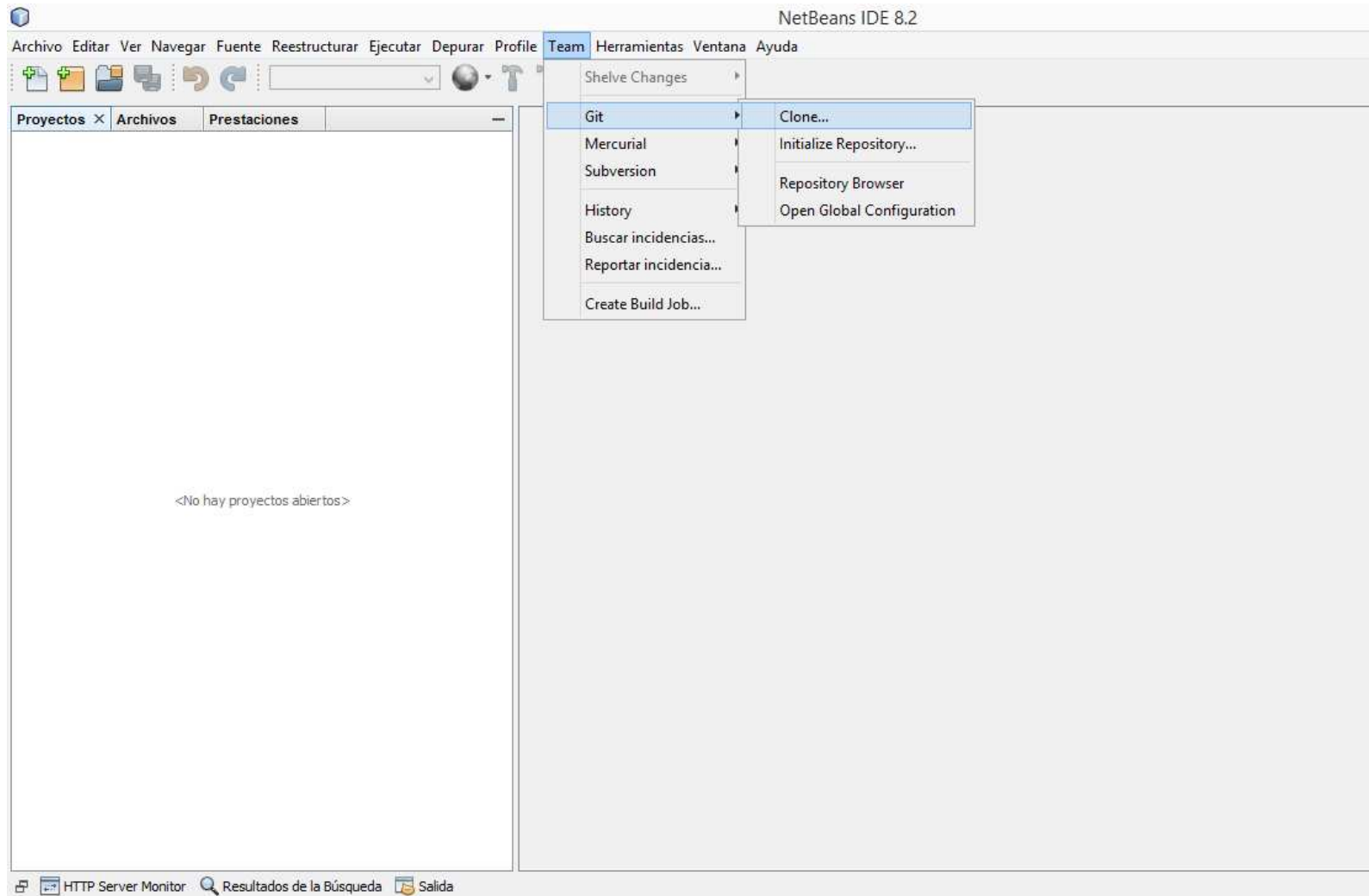
Download x64 **Download x64** **Download x64**

Free, 100 MB Free, 200 MB Free, 111 - 115 MB Free, 111 - 115 MB Free, 109 - 112 MB Free, 222 MB

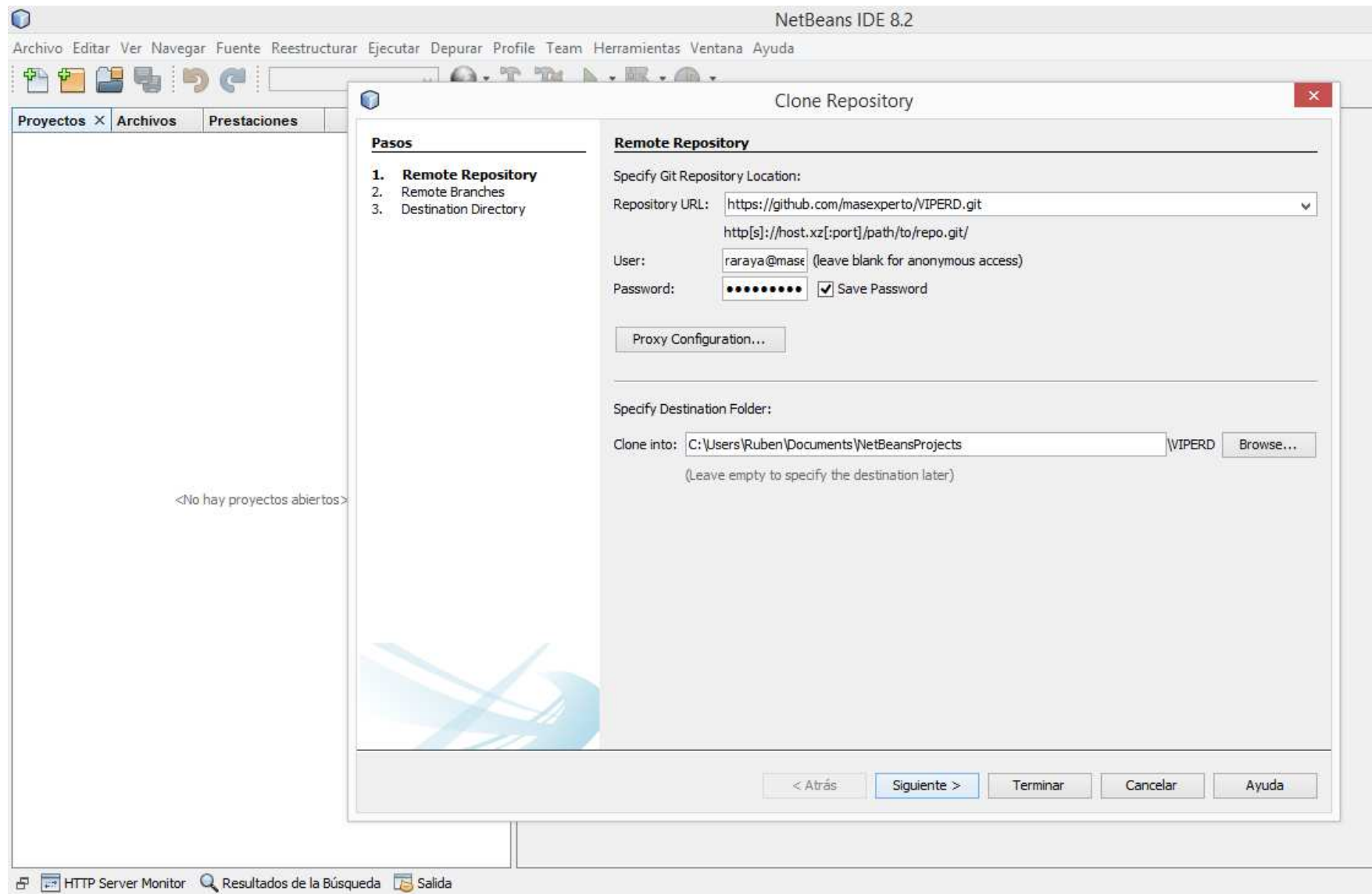
En las páginas siguientes se entregará una breve guía paso a paso para comenzar a utilizar el repositorio VIPERD con NetBeans IDE 8.2, a través de su clonación en la computadora local desde GitHub. Esto nos permitirá mantener actualizada nuestra versión de trabajo, sincronizar los cambios realizados en los archivos y trabajar en forma colaborativa con un sólido control de versiones.

INSTRUCCIONES PARA CLONAR EL REPOSITORIO VIPERD CON NETBEANS DESDE GITHUB

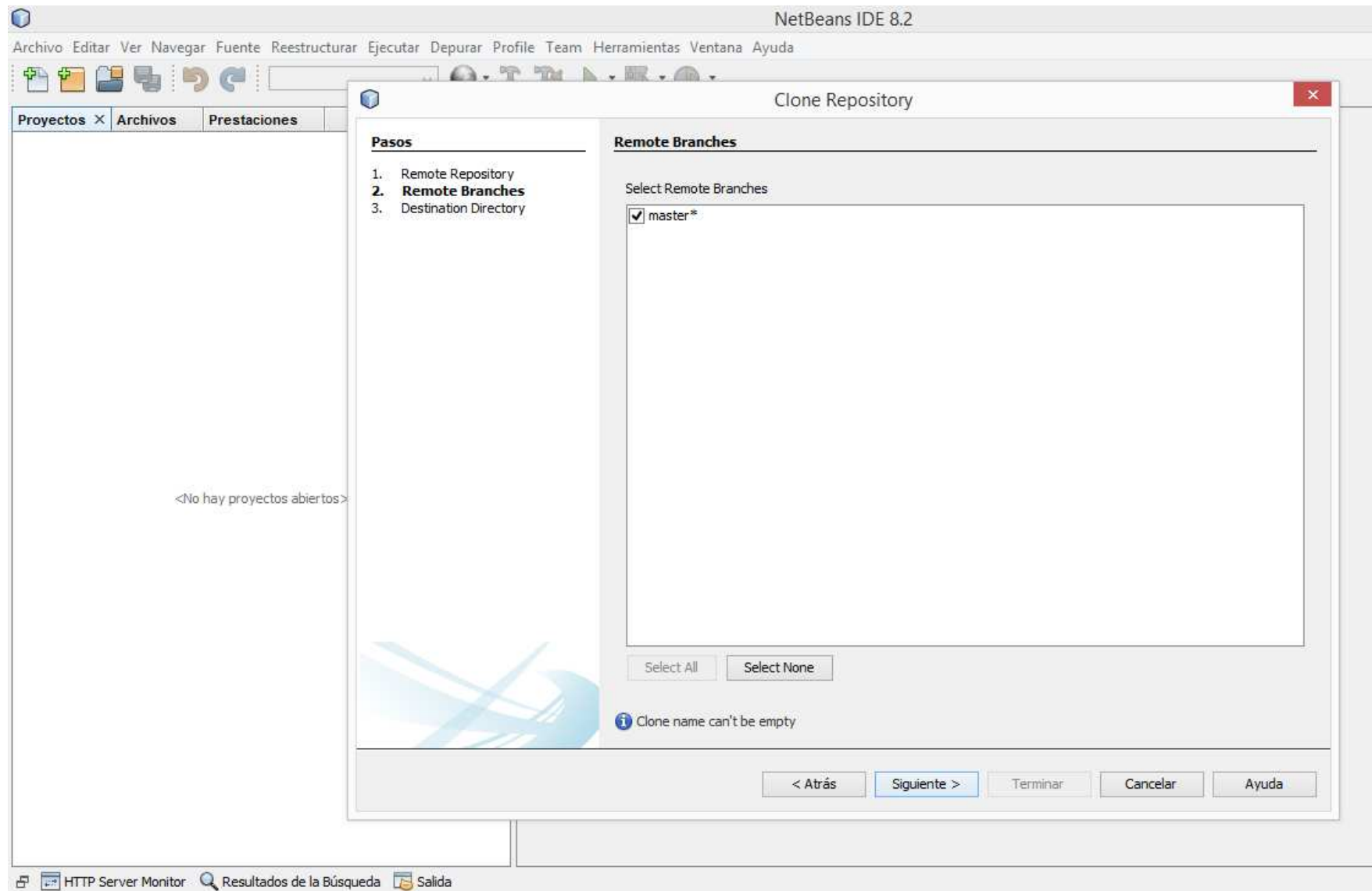
Una vez instalado NetBeans IDE, abrir el programa y proceder directamente a la clonación, sin crear ningún proyecto. Vamos a guiar la creación de un clon del repositorio en un directorio de nuestro disco local llamado “VIPERD”, el que ubicaremos en la ruta donde se almacenan los proyectos de NetBeans (usualmente sería “C:\Users\[Usuario]\Documents\NetBeansProjects”). Elegimos el menú “**Team / Git / Clone...**”:



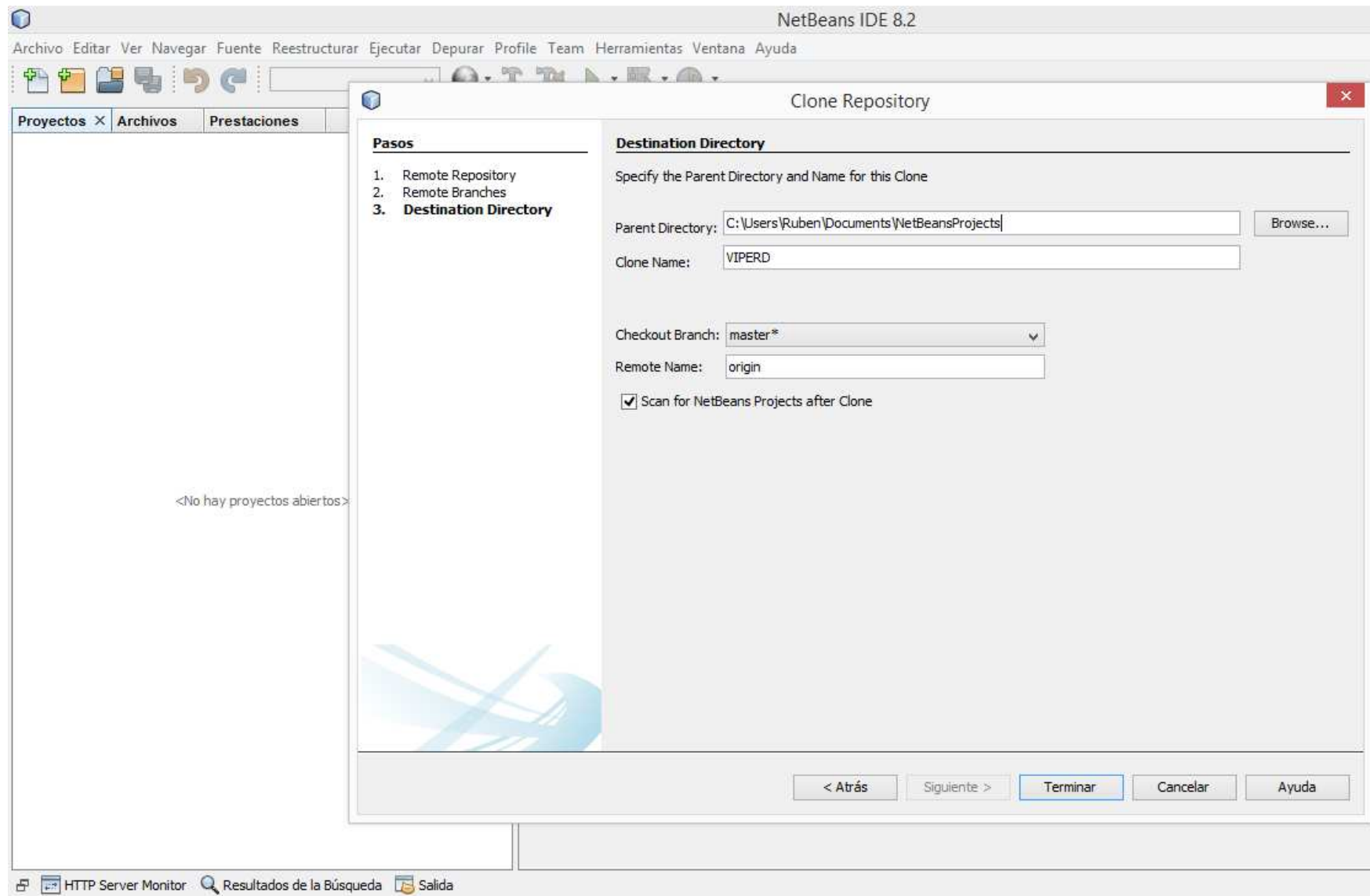
1. En la ventana emergente colocamos los datos para conectar con el repositorio remoto en GitHub. La URL del repositorio es: **<https://github.com/masexperto/VIPERD.git>**. Los datos de usuario y password corresponden a nuestra cuenta como usuarios en GitHub (que debemos tener previamente creada en <https://github.com/join>, es gratis). Luego pulsamos “Siguiente”.



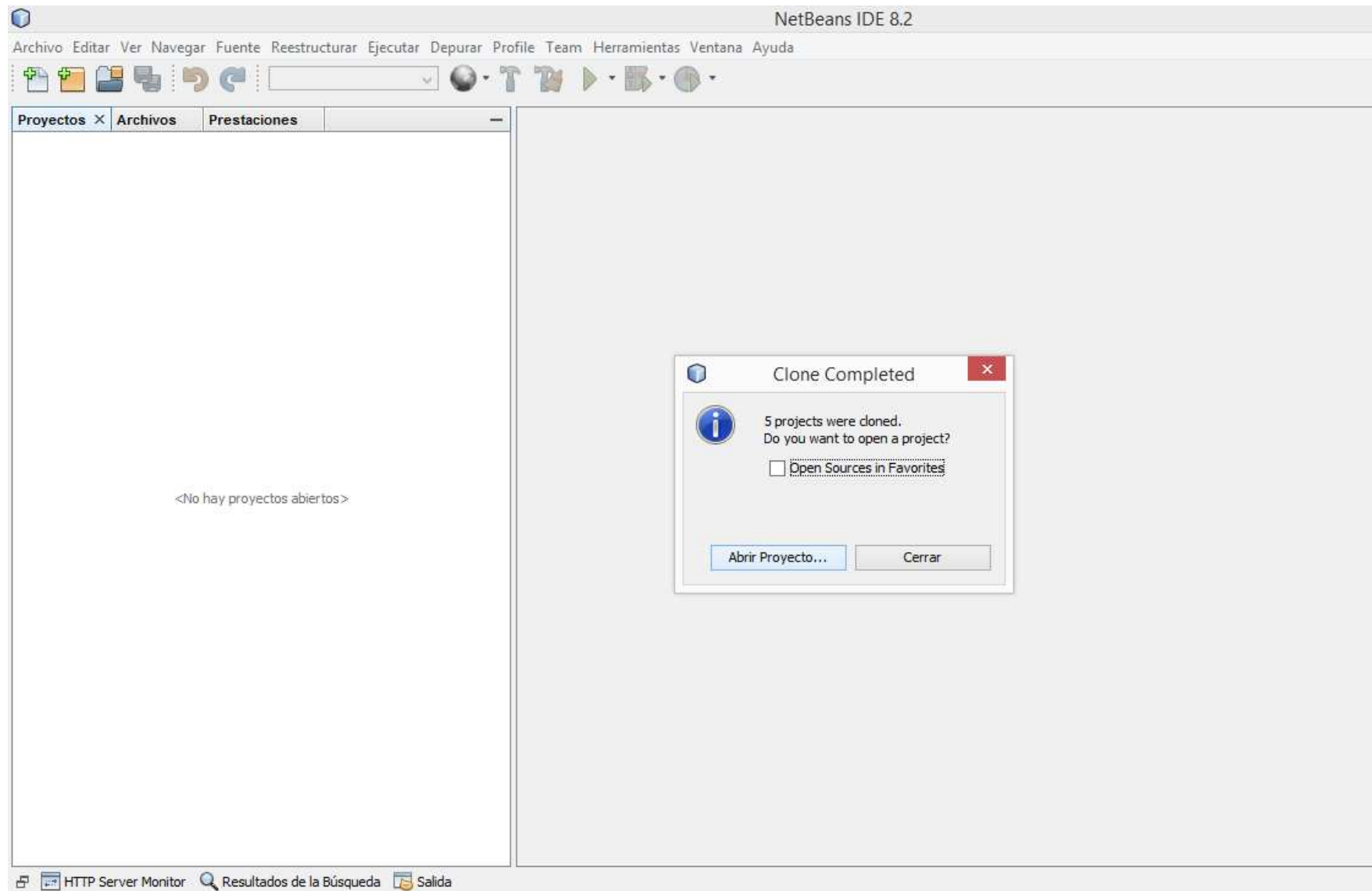
2. En este paso elegimos la rama “**master**” para clonarla. Luego pulsamos “Siguiente”.



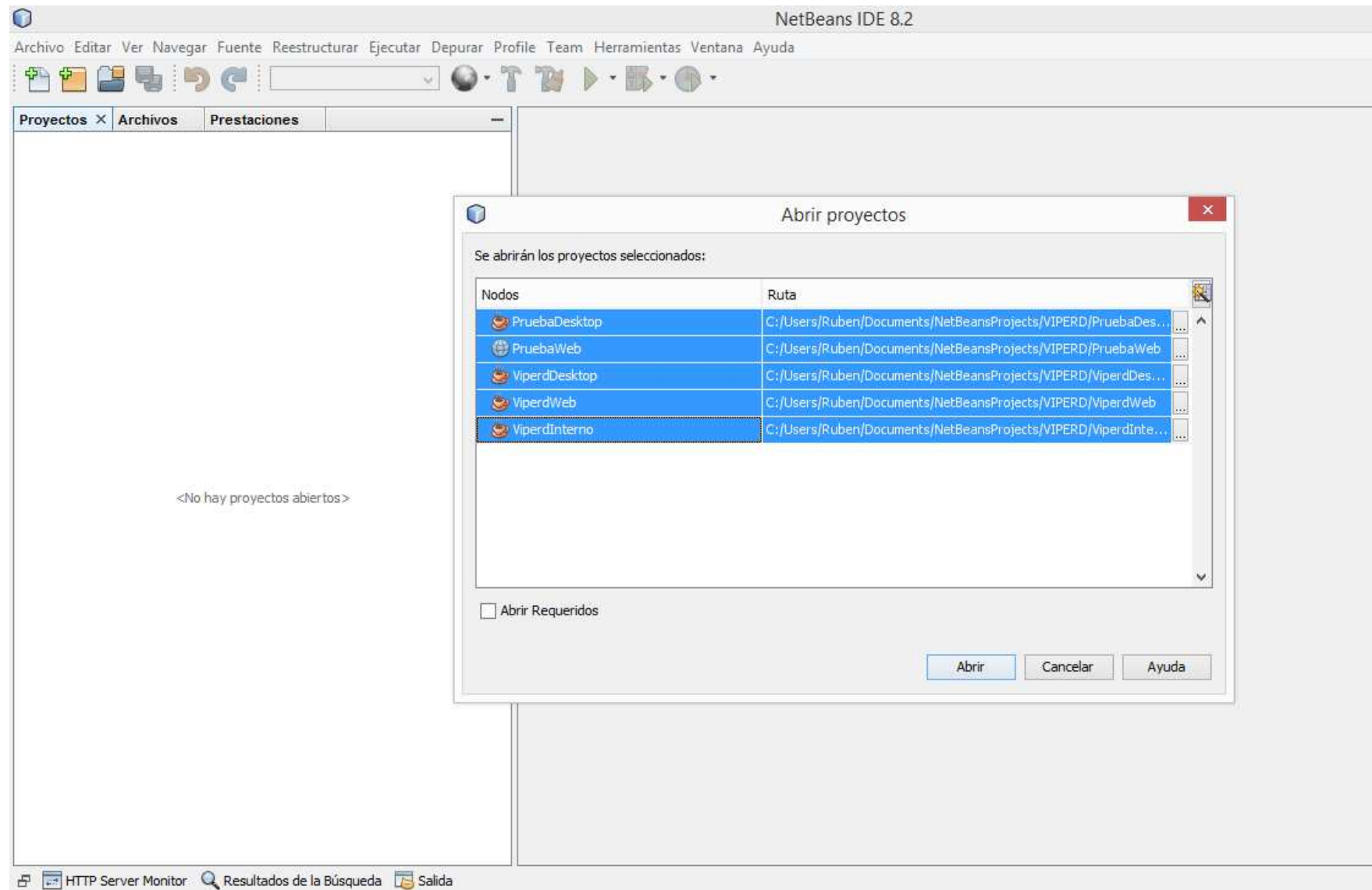
3. En este paso confirmamos la ubicación donde se almacenará la copia local del repositorio. Por defecto usará la ruta de proyectos de NetBeans y el nombre “VIPERD”. Debemos MARCAR la casilla que dice “**Scan for NetBeans Projects after Clone**” para asegurar que busque y configure los 5 proyectos que vienen en el repositorio. Confirmamos estos valores y pulsamos “Terminar”.



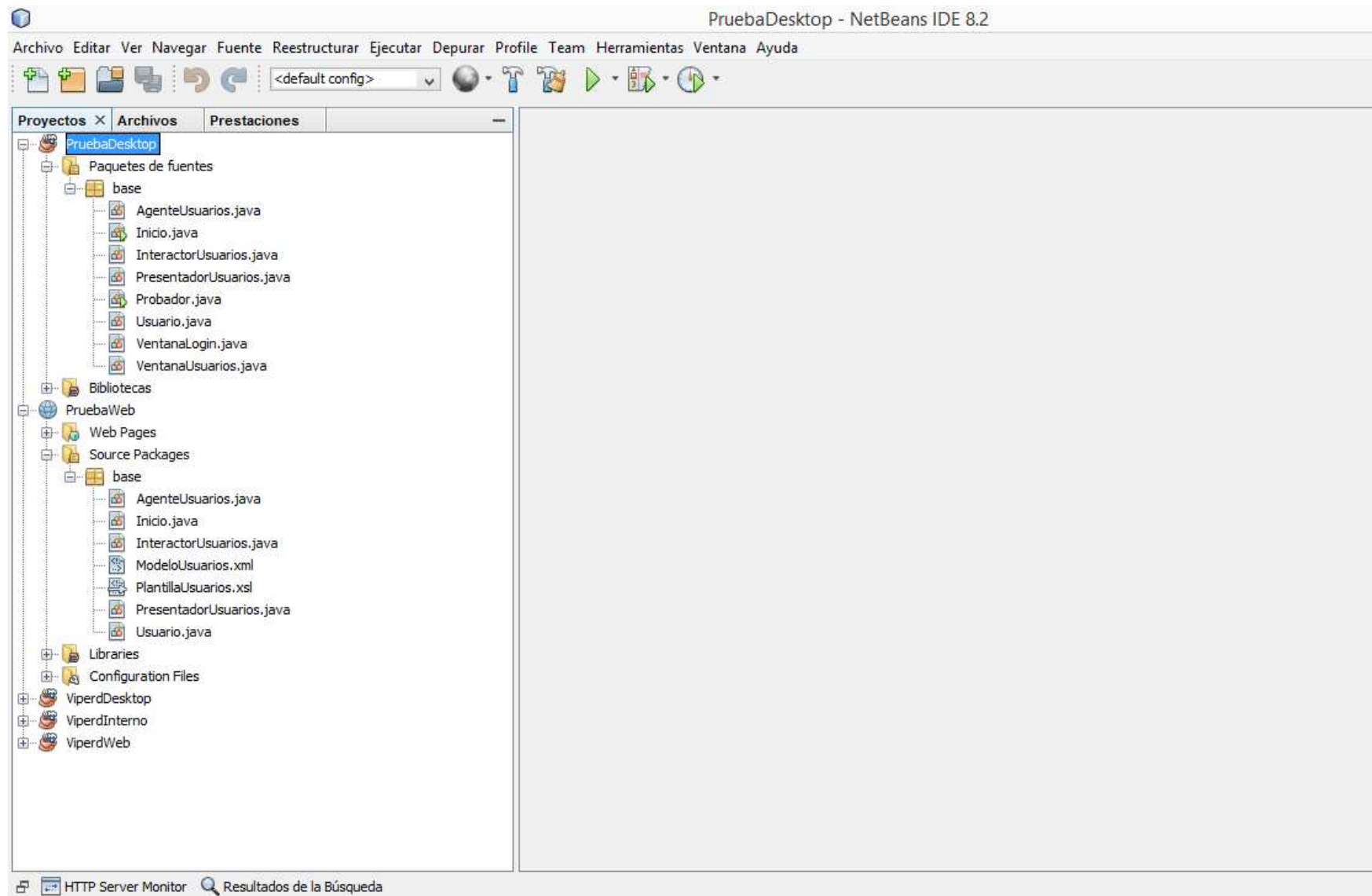
4. Una vez terminada la clonación aparecerá un mensaje de confirmación. Allí pulsamos el botón “Abrir Proyecto...”.



5. Seleccionamos los 5 proyectos que nos ofrece para abrir, luego pulsamos el botón “Abrir”.



6. Al finalizar la clonación y abrir los proyectos, se mostrarán en el árbol de “Ventana de Proyectos” los 5 proyectos del repositorio.



En este punto ya estamos en condiciones de comenzar a trabajar en el IDE con VIPERD y ejecutar las aplicaciones de prueba.

CONCEPTOS BASICOS DE LA ARQUITECTURA VIPERD

El patrón de arquitectura VIPER divide la lógica de las aplicaciones de software en capas que tienen responsabilidades diferentes:

V: Vista (interfaz de usuario y sus componentes).

I: Interactor (lógica de negocio y sus casos de uso).

P: Presentador (diseño de interacción y acciones).

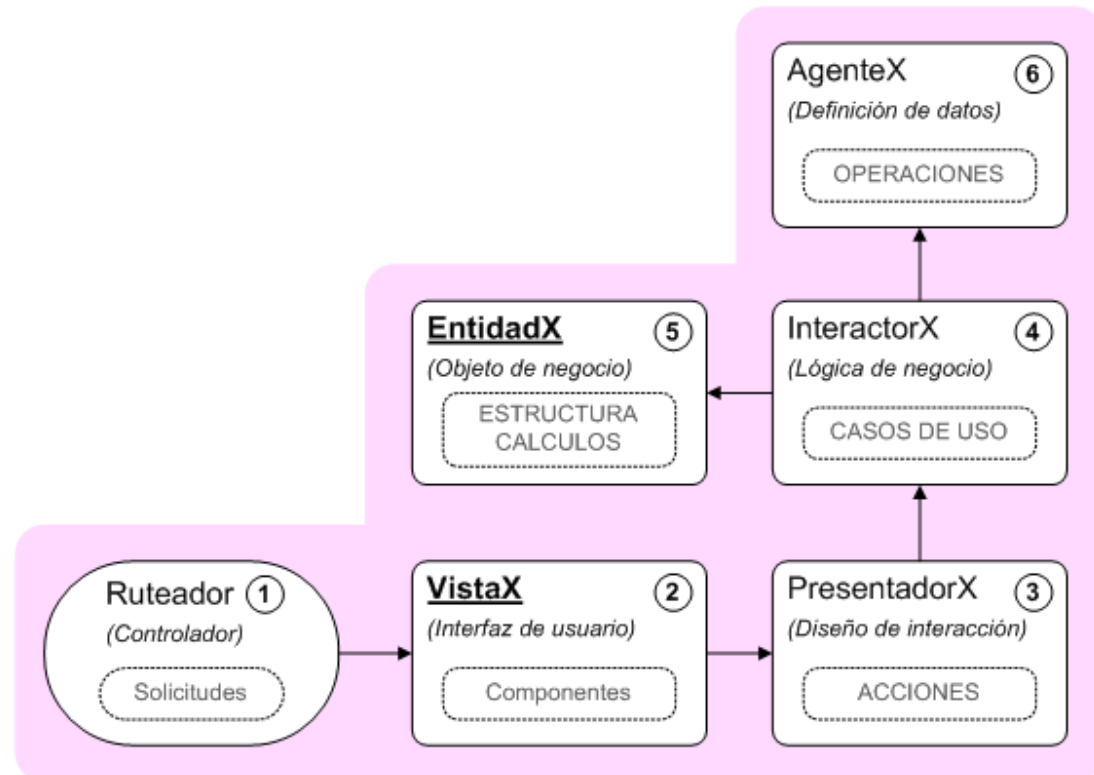
E: Entidad (objetos de negocio y sus reglas).

R: Ruteador (controlador de solicitudes).

Nuestro diseño incorpora además una capa de gestión de datos (D), en la cual un proxy (agente) opera como intermediario con los componentes de acceso a datos (DAO):

D: Agente (definiciones de datos y operaciones).

Esta arquitectura es independiente de lenguajes de programación, frameworks y bibliotecas, bases de datos, servicios externos e interfaces de usuario.



La implementación para Java que hemos construido separa la estructura de clases de cualquier proyecto de aplicación en dos niveles:

- **La infraestructura general:** corresponde a las bibliotecas VIPERD “internas” (ViperdInterno) y “externas” (ViperdDesktop, ViperdWeb y ViperdAndroid) que sirven como base para la construcción de las aplicaciones Java desktop, web y móvil. Estas bibliotecas proporcionan una estructura lógica de componentes abstractos y un modelo funcional de cooperación entre ellos, definidos en un conjunto de interfaces por capas. Adicionalmente, proveen algunas superclases abstractas que implementan en forma genérica las funciones, y facilitan la creación de componentes modulares con una alta cohesión y bajo acoplamiento.
- **El contenido específico:** corresponde a los componentes reales (clases / objetos) que se desarrollan en un proyecto de software concreto para construir una solución funcional, y que en su codificación implementan las interfaces de los componentes del modelo VIPERD y/o consumen sus superclases. Estos componentes son los que se muestran en el diagrama de arriba sobre fondo rosado, y representan la unidad básica del desarrollo: un módulo conformado por seis piezas básicas de software (como mínimo, ya que puede haber varias Entidades o Vistas en un módulo).

La composición por capas se organiza en primer lugar con los **Interactores** y **Entidades** en el núcleo de la aplicación. En la capa de adaptadores se encuentran los **Presentadores** y los **Agentes de Datos**. Y en la capa de interfaz tenemos a las **Vistas** y los **Ruteadores**, además de los **Objetos de Acceso a Datos** (DAO) que ya vienen incluidos en la infraestructura general.

Desde el punto de vista de la división técnica del trabajo se pueden diferenciar dos ámbitos que operan con cierta autonomía dentro del proceso de desarrollo y que se asocian a roles distintos: el **diseño de front-end**, que incluye creación de Ruteador, Presentadores y Vistas enfocados en la experiencia del usuario (1, 2 y 3); y el **diseño de back-end**, que incorpora la construcción de Interactores, Entidades y Agentes enfocados en la gestión de los datos y servicios (4, 5 y 6).

De esta forma, el procedimiento general de construcción de software basado en VIPERD se puede resumir estructuralmente como la creación de **módulos**, integrados principalmente por **componentes** de contenido específico de estos seis tipos básicos. Para facilitar el desarrollo de aplicaciones basadas en VIPERD con la IDE de NetBeans, el repositorio incluye plantillas (templates) de clases para estos componentes que aplican diversos patrones de diseño:

- **Agente:** definir orígenes de datos y credenciales de acceso y gestionar operaciones y servicios de datos (“DAO”, “proxy”).
- **Entidad:** definir estructuras, cálculos y reglas de los objetos de negocio de la solución (“value object”, “entity”).
- **Inicio:** controlar las solicitudes, el acceso y la navegación dentro de la aplicación (“front controller”, “singleton”).
- **Interactor:** implementar los casos de uso y aplicar procedimientos con la lógica de negocio.
- **Presentador:** diseñar interacciones (acciones y reacciones) intermediando entre la vista y el interactor (“MVP”, “adapter”).
- **Ventana:** implementar interfaz de usuario y componentes (formularios, listados, diálogos, notificaciones, etc.) (“composite”).
- **Probador:** clase especial para probar aplicaciones sin interfaz de usuario, consumiendo directamente interactores.

INSTRUCCIONES PARA AGREGAR PLANTILLAS DE COMPONENTES VIPERD EN NETBEANS

Las plantillas de componentes VIPERD se encuentran en la ruta “**VIPERD / viperd / plantillas**” del repositorio. Para poder utilizarlas en el IDE de NetBeans es necesario agregarlas manualmente a su gestor de plantillas:

1. Elija el menú “**Herramientas / Plantillas**”.
2. En la ventana emergente del “Gestor de plantillas” pulse el botón “**Nueva Carpeta**”.
3. En el diálogo siguiente “Cambiar nombre de Plantilla”, coloque el nombre “VIPERD” y presione el botón “Aceptar”.
4. Seleccione la carpeta VIPERD recién creada y pulse el botón “Agregar...”. En el diálogo siguiente “Agregar Plantilla existente” localice la carpeta donde vienen las plantillas, seleccione todos los archivos (7 en total) y pulse el botón “Agregar”.

Una vez realizado lo anterior, podrá comenzar a crear clases en sus proyectos usando las plantillas VIPERD, mediante “Archivo / Nuevo” y luego eligiendo en la categoría “VIPERD” cuál de los siete tipos de plantilla desea crear.