

Nombre:

kevin japa

Fecha de entrega:

Miércoles 23 de junio de 2020.

Tema:

Prueba Unidad 2 (Prueba).

Materia:

Programación Orientada a Objetos.

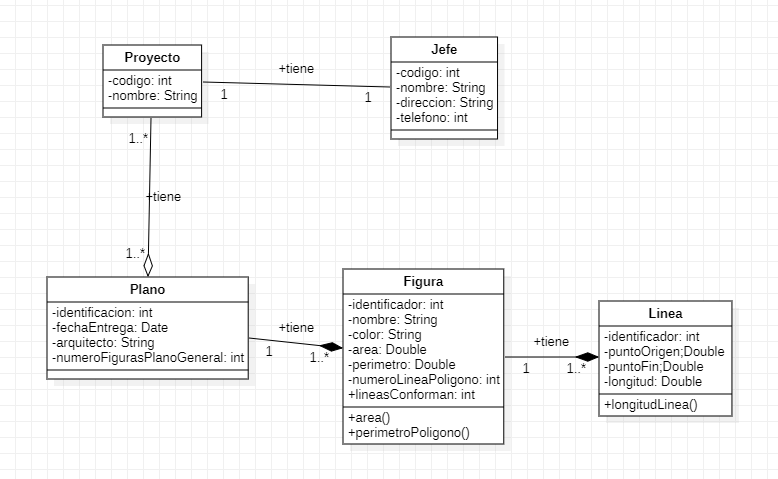
Que es el MVC

Modelo-vista-controlador es un patrón de arquitectura de software, que separa los datos y principalmente lo que es la lógica de negocio de una aplicación de su representación y el módulo encargado de gestionar los eventos y las comunicaciones.

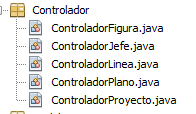
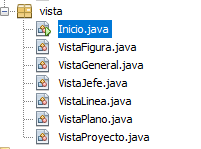
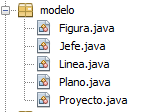
Para ello MVC propone la construcción de tres componentes distintos que son el modelo, la vista y el controlador, es decir, por un lado define componentes para la representación de la información, y por otro lado para la interacción del usuario.

Analisis

Para la implementación del MVC se utiliza el siguiente diagrama UML:

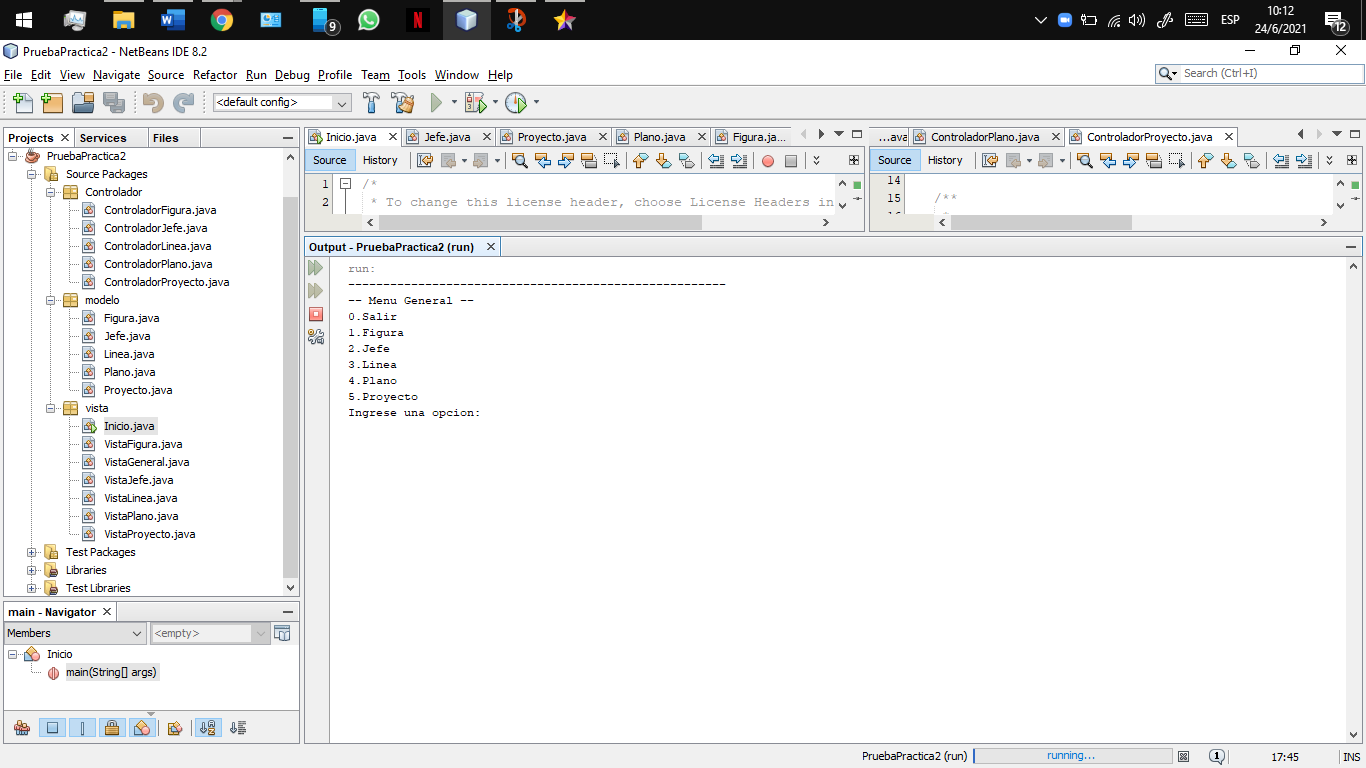


En la cual implementando el uso de relaciones de asociación y el uso patrón de diseño tenemos tres diferentes paquetes:

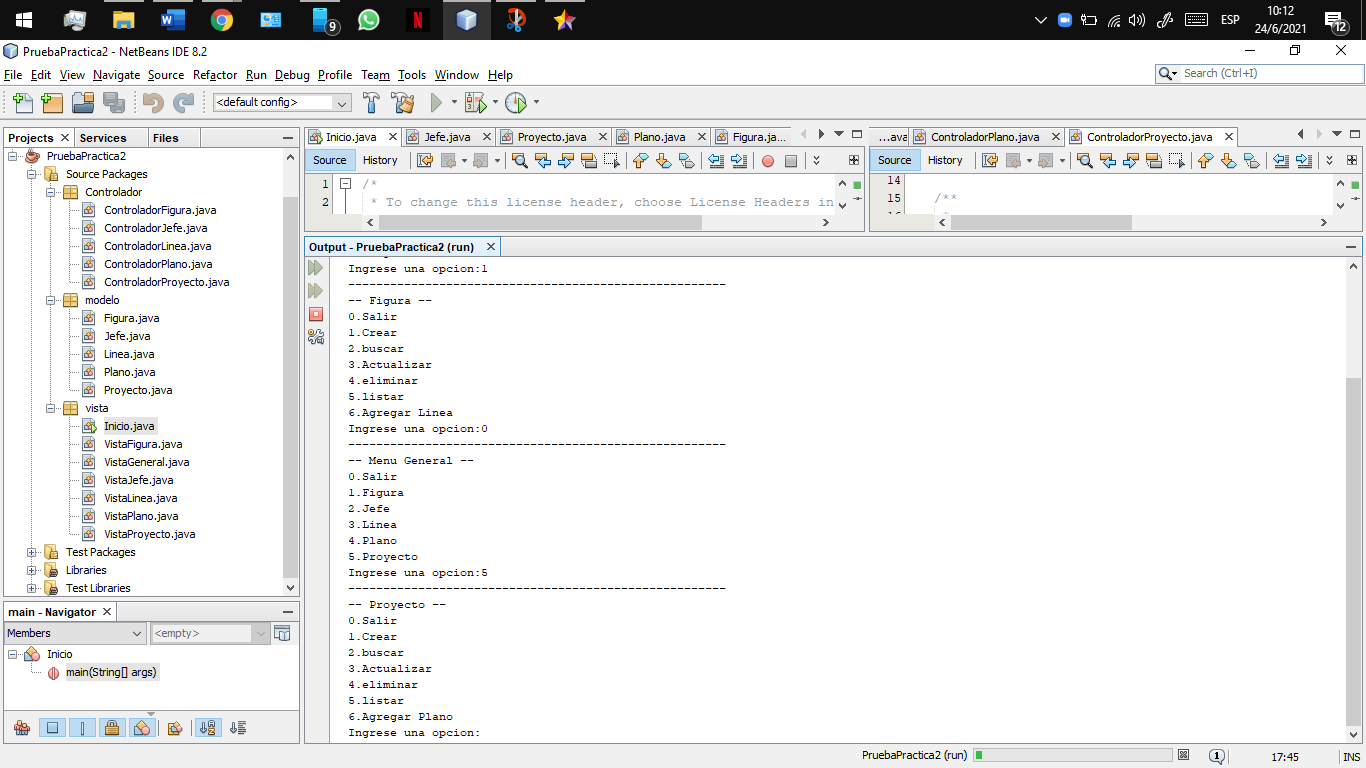


Prueba

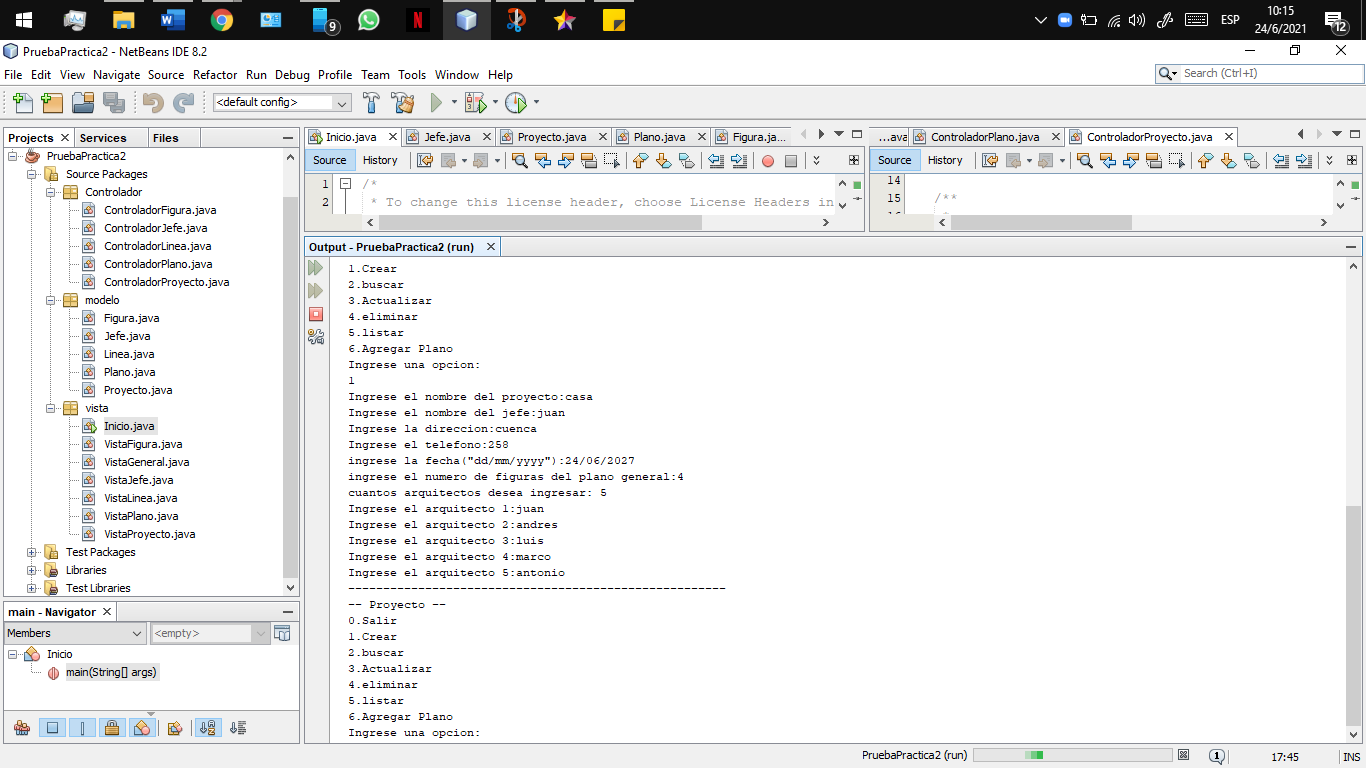
Tenemos una pantalla de Inicio en la cual nos permite crear figuras, jefes, líneas, planos y proyectos:



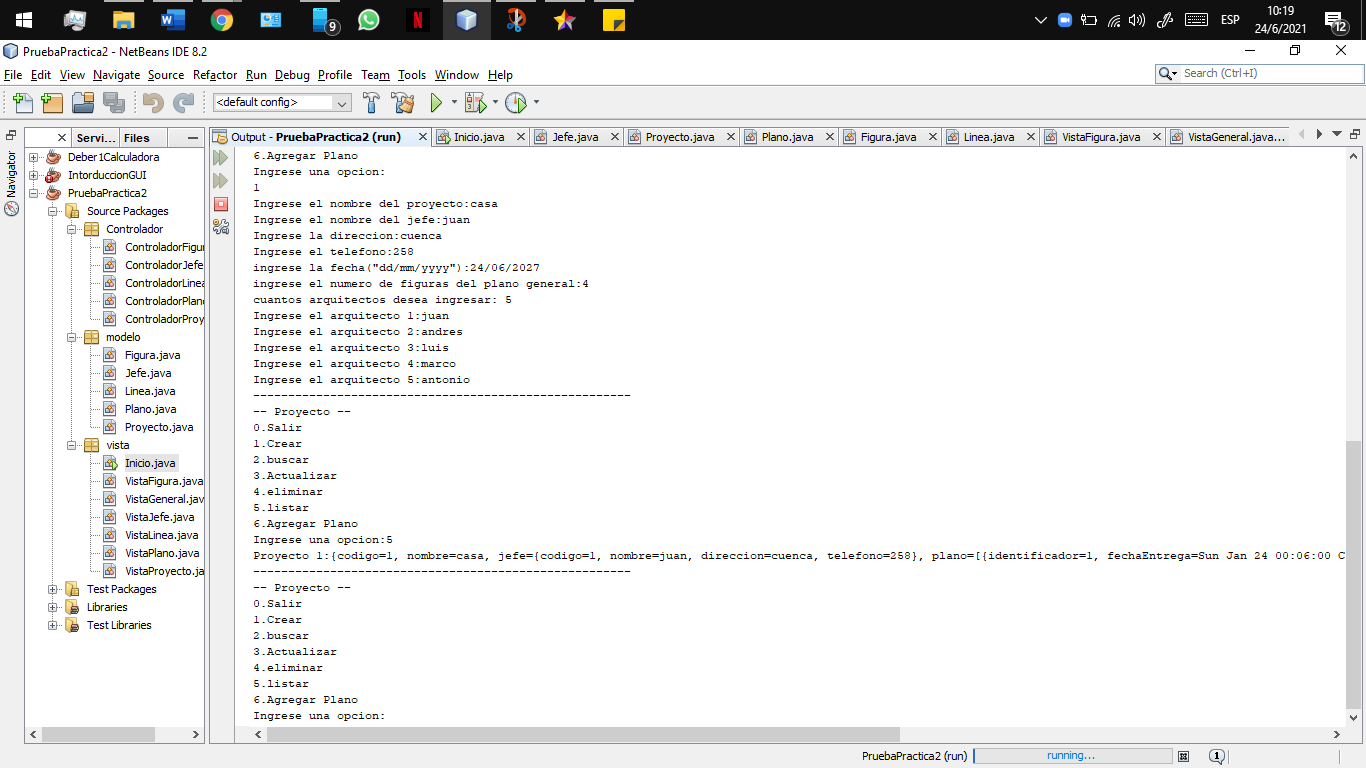
En cada vista Proyecto en todas las clases tenemos un menú similar:



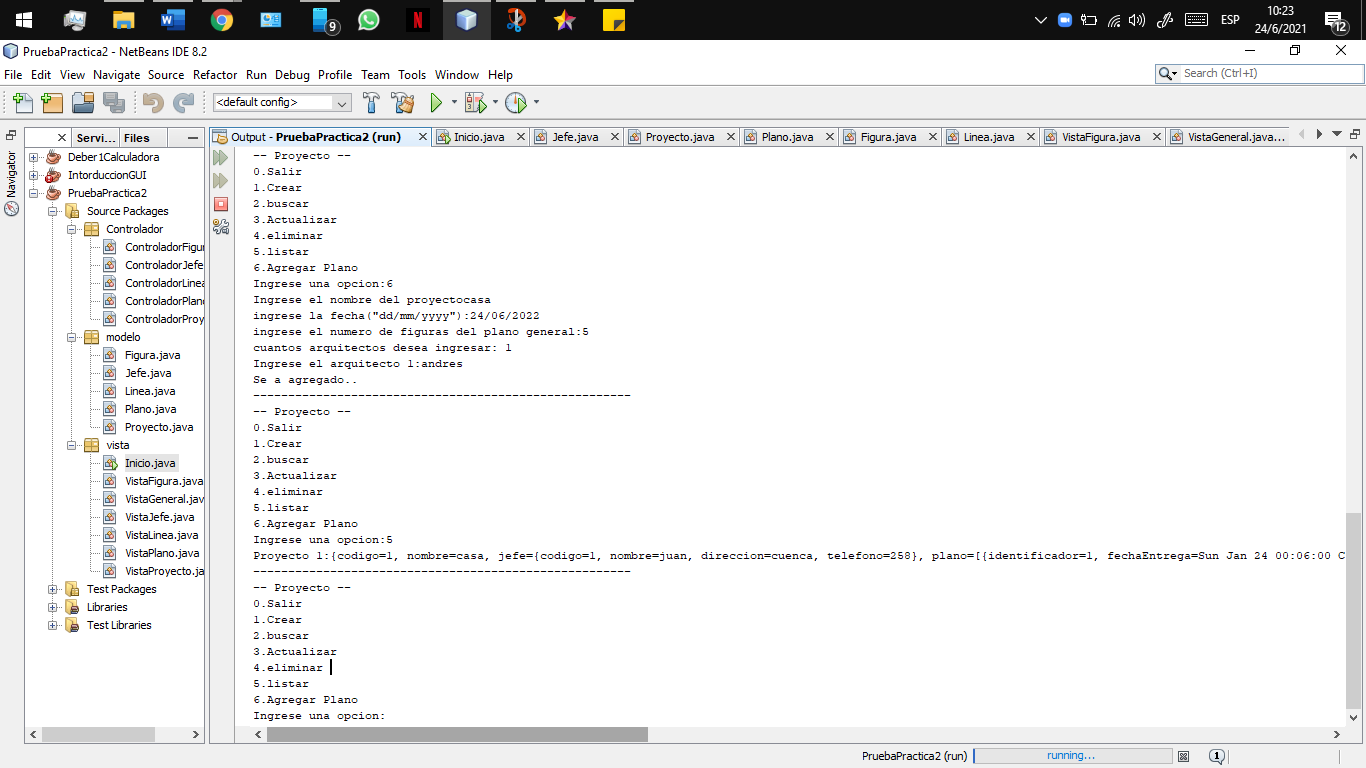
Al crear un proyecto realiza una búsqueda de la lista de jefes, si existe nos permite crear un jefe, además de agregar las Planos nos permite agregar los arquitectos que constan en el:

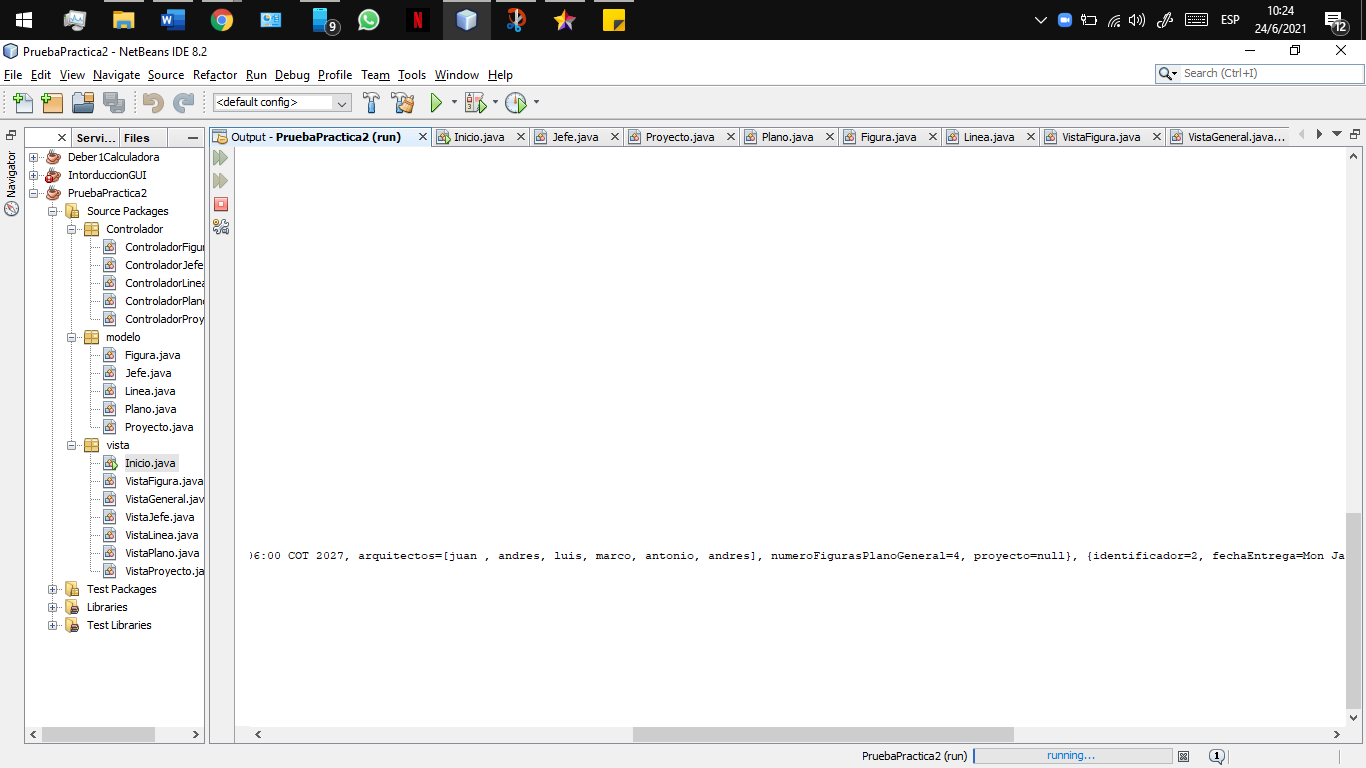


En el cual se implementa método buscar para tener las listas de los proyectos y la información contenida de acerca de los planos, arquitectos y las figuras que contienen estas;



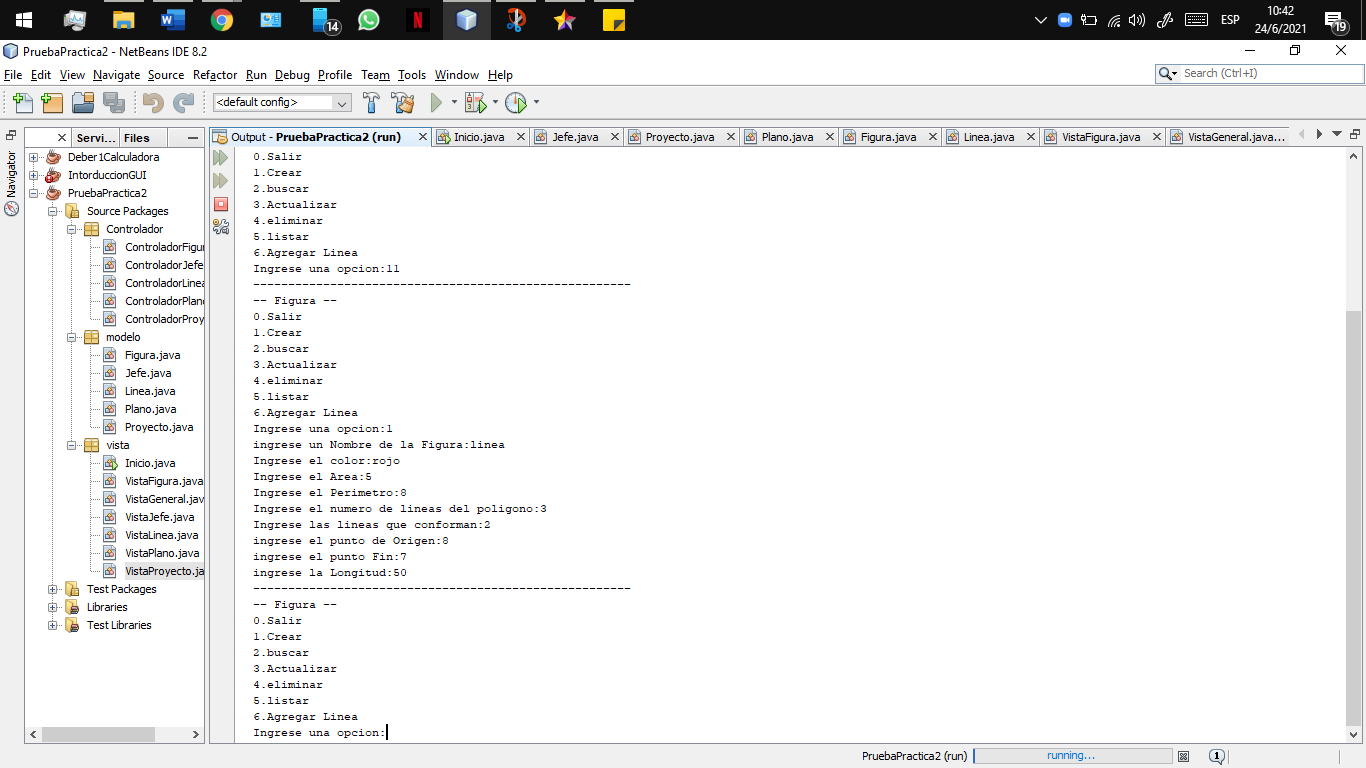
En la cual al agregar un nuevo plano se agrega, se realiza una lista de esta y se demuestra que fue agregado con éxito:



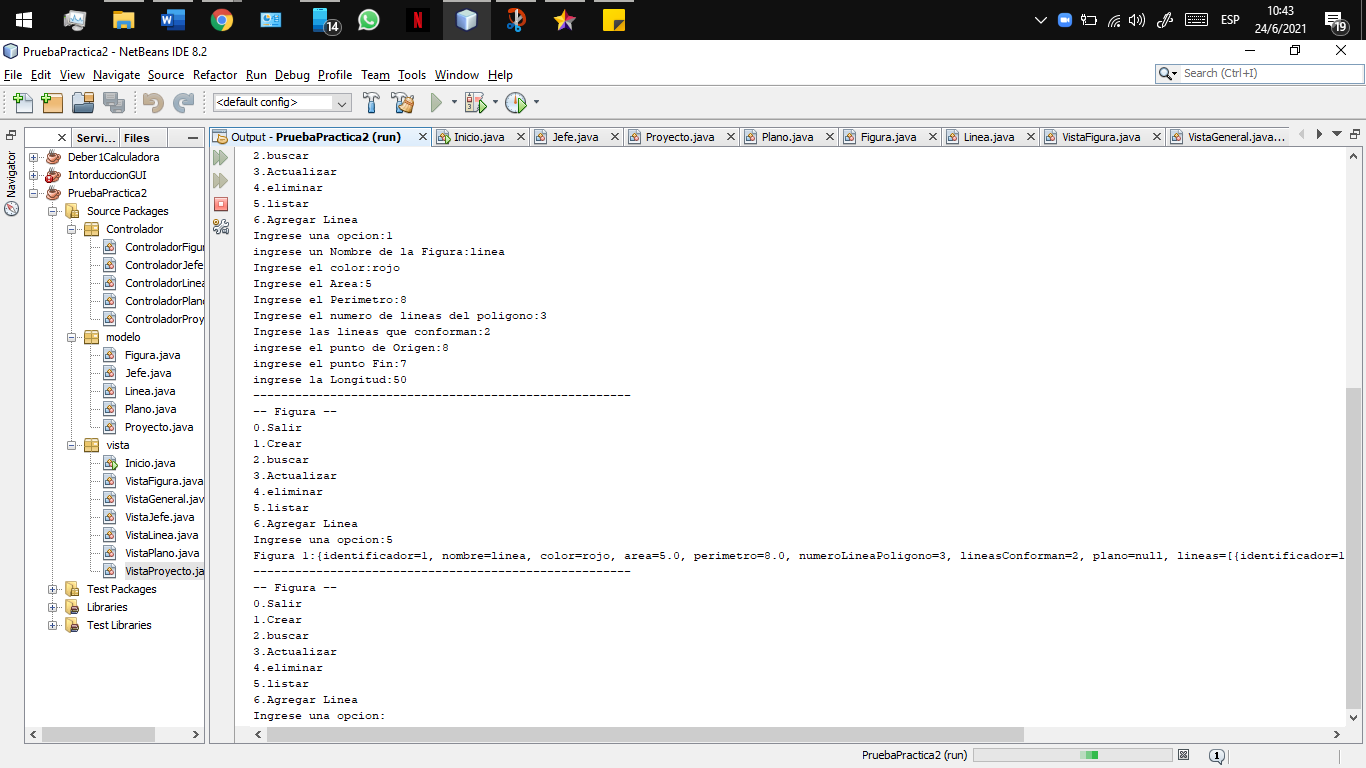


Vista Figura

En la vista figura tenemos inicio común este nos permite crear las figuras

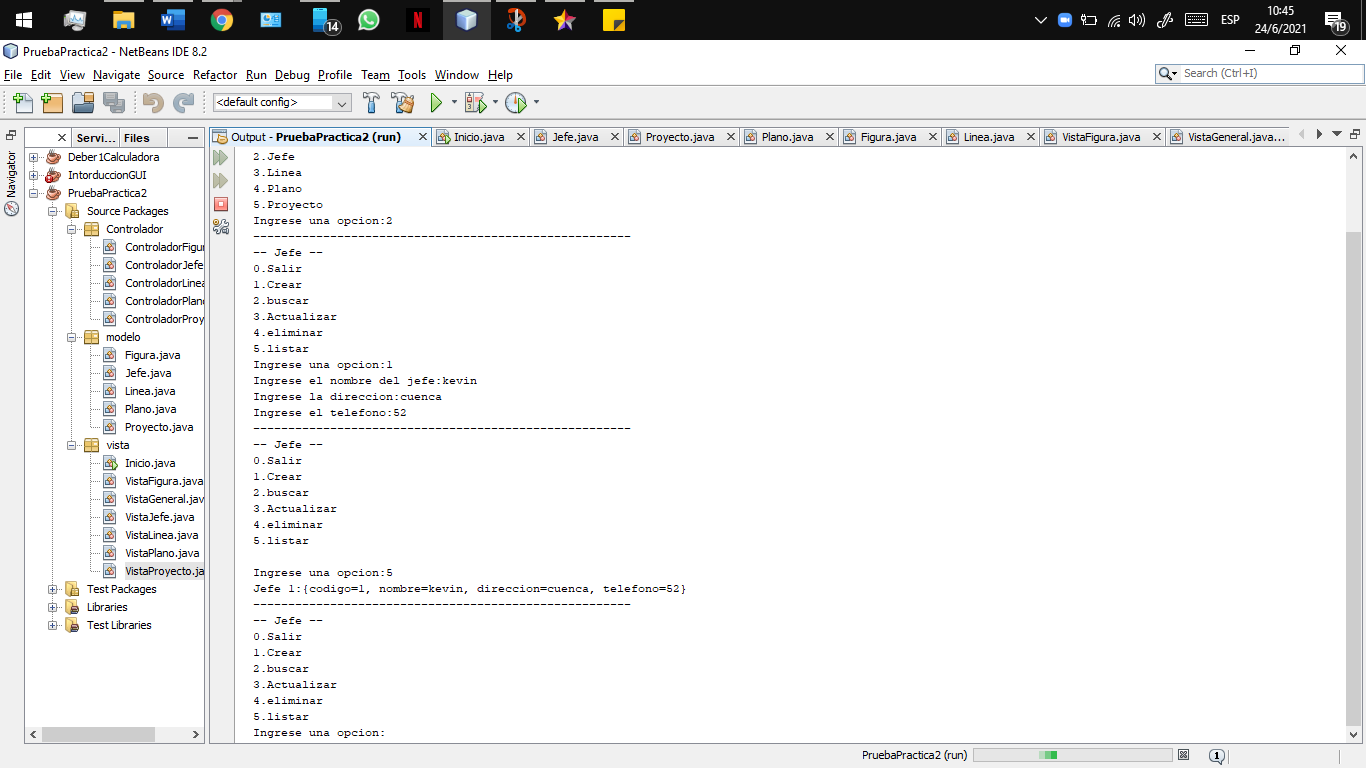


En la cual tenemos un método para buscar las figuras preestablecidas

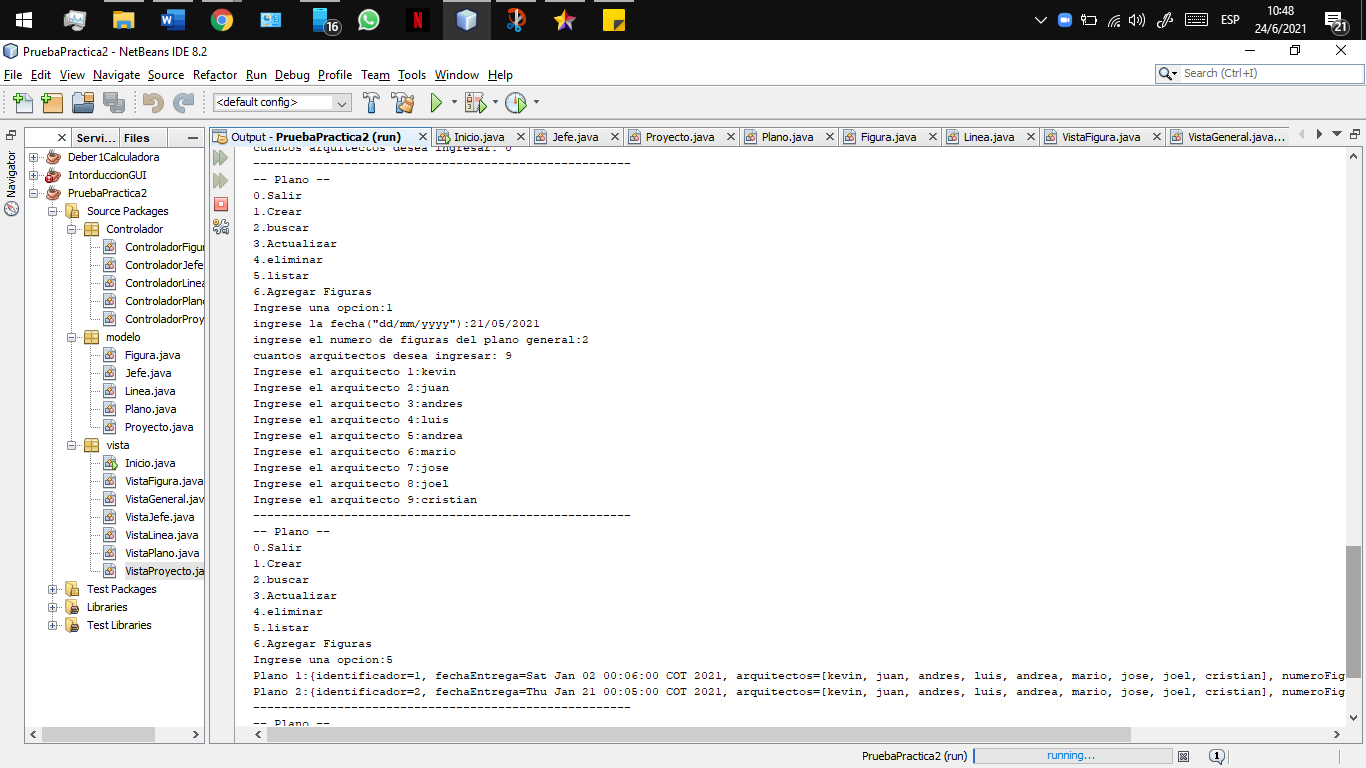


Vista jefe

Esta vista nos permite crear, listar , actualizar los jefes de proyecto



Visat de Planos



En la cual todos tienen su respectiva relación en la cual son almacenad en su respectivo modelo y son enviados a la clase del proyecto en la cual controla la mayor cantidad de datos .