PONTIFICIA UNIVERSIDAD CATÓLICA DE VALPARAÍSO

FACULTAD DE INGENIERÍA

ESCUELA DE INGENIERÍA INFORMÁTICA

**Gran Minería**

Entrega Final

**Profesor: Claudio Cubillos**

**Asignatura: Programación Avanzada ICI 3241**

**Participantes:**

* Cristofer Pizarro
* Donovan Aguilar
* Alexis Armijo

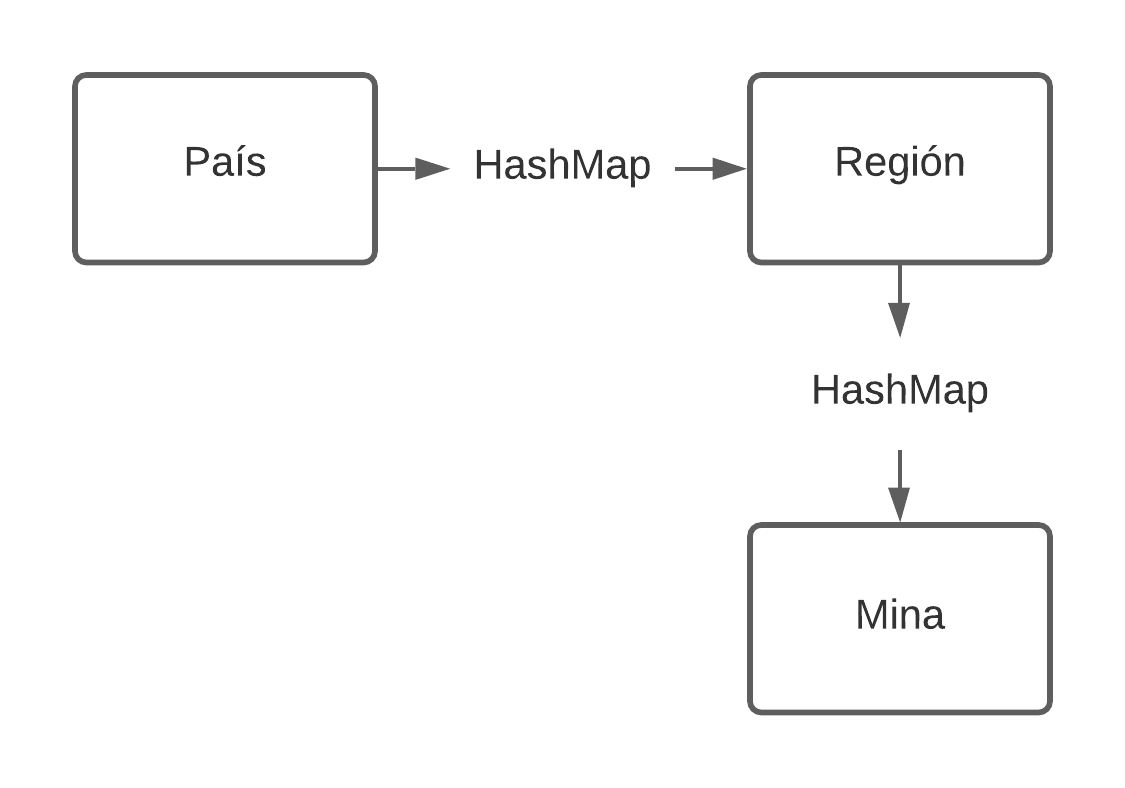
**Julio2021**

**1.1 Análisis de los datos a utilizar y principales funcionalidades**

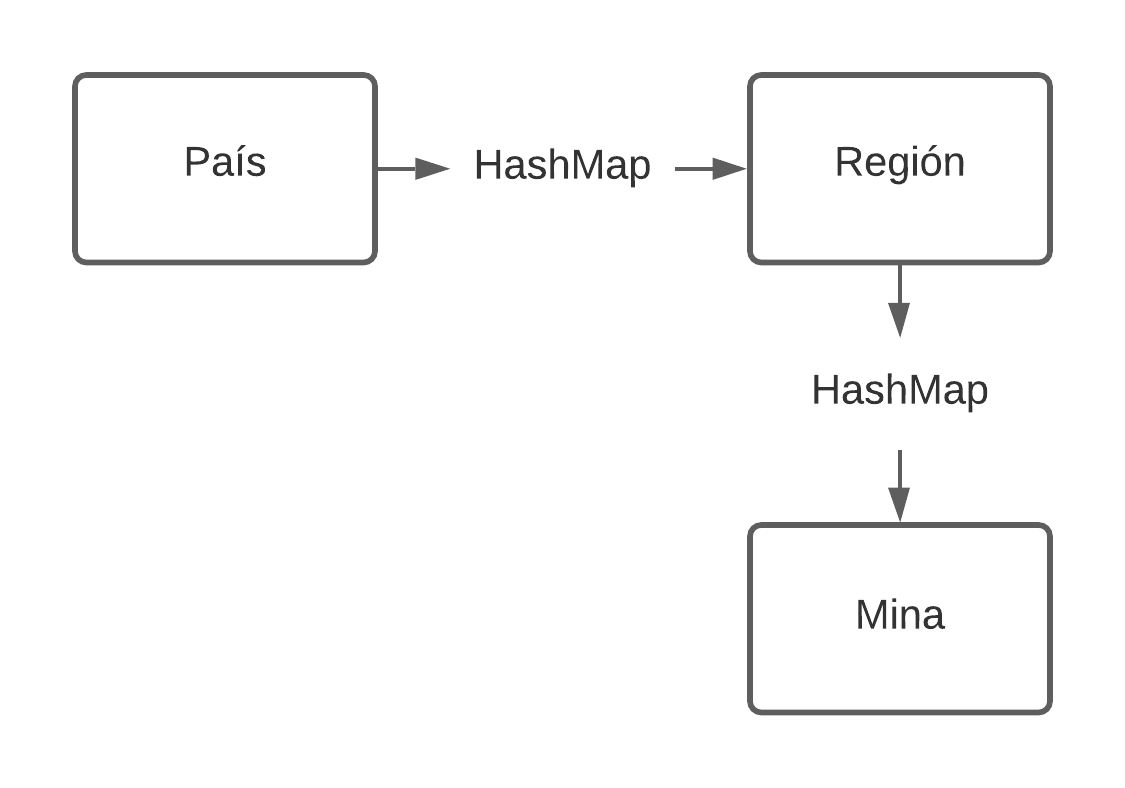
La Minería es un campo de trabajo muy amplio y diverso alrededor del mundo, incluso nuestro país contiene todo tipo de Minas: Cobre, plata, zinc, oro, etc. Con Gran Minería tendremos una herramienta que facilita visualizar los datos sobre una Mina, por ejemplo; ubicación, tipo de material con el que se trabaja, cantidad de material que se extrae, cantidad de personal. Otras funcionalidades que tendrá el proyecto son, ver minas por país, ver minas por región, agregar minas a los diferentes tipos de datos con los cuales manejamos la información (función que lee datos de un CSV), como también eliminar minas, buscar que mina tiene más producción en una región determinada, ver datos sobre una mina en específico, generar un reporte con las minas que contiene el programa en un archivo txt, busque por rangos de producción. Esto se podrá ver y se podrá interactuar con la aplicación mediante un menú implementado con ventanas graficas.

**1.2** **Diseño Conceptual de las Clases de Dominio**

* País: clase que contiene un ArrayList con las regiones de cada pais.
* Región: contiene lista de minas la cual pertenece a la región.
* Mina: Clase que tiene nombre de la mina, mineral que extrae, su producción y la región y el país al que pertenece junto con el personal.



**2.1. Diseño conceptual y codificación de 2 (dos) niveles de anidación de colecciones de objetos**



Las clases que se utilizarán para el proyecto serán: Mina, Región y País.

Los 2 niveles de anidación son:

* País tiene una ArrayList de Regiones con las regiones del país.
* Región tiene una ArrayList que contiene todas las minas de la región y sus datos

**2.2. Diseño conceptual y codificación de 2 (dos) clases que utilicen sobrecarga de métodos**

Las clases con sobrecarga son:

Diagrama

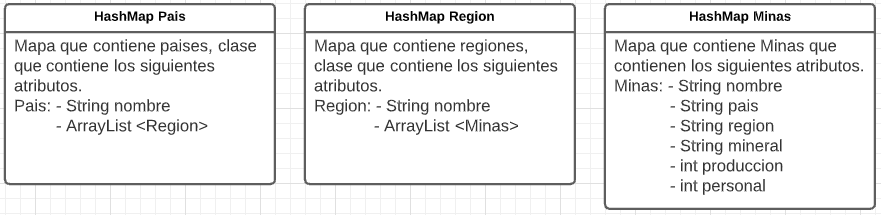
Descripción generada automáticamente

En el caso de public void addMina (Minas mina), se agrega una mina ya existente al mapa de minas, por otro lado, public void addMina (String nombre, Minas mina, HashMap< String, Region> mapaRegion, HashMap< String, Pais> mapaPaises) crea una mina con los datos que se entregan por parámetros y luego se agrega al mapa de minas.

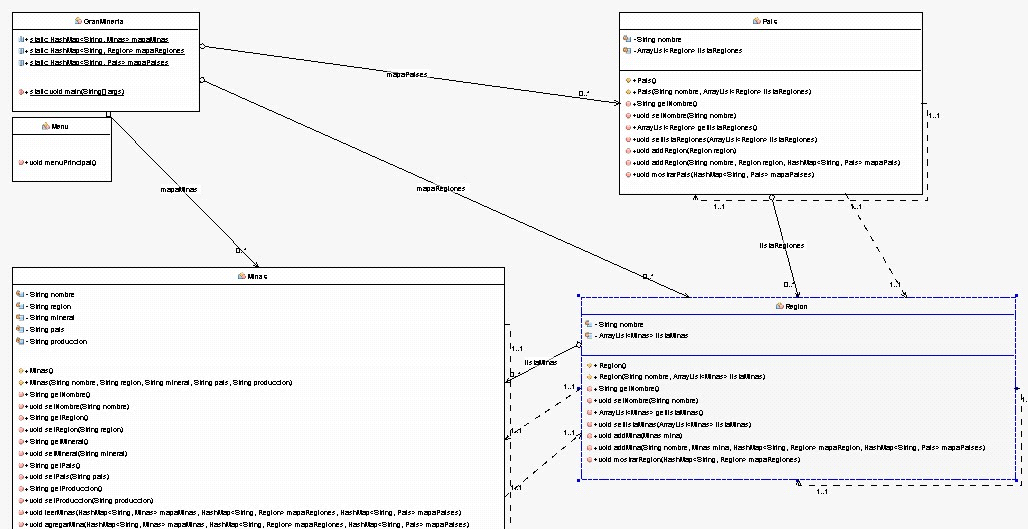
En el caso de public void addRegion (Region region), se agrega una región ya existente al mapa de regiones, por otro lado, public void addRegion (String nombre, Region región, HashMap< String, Pais> mapaPais) crea una nueva región con los datos que se entregan por parámetro y luego se agrega al mapa de regiones.

**2.3. Diseño conceptual y codificación de al menos 1 clase mapa del JCF.**

Se utilizarán diferentes mapas que contengan todos los datos del país, región y mina para su rápida búsqueda, cuya clave será el nombre.

****

**3.1. Diseño de diagrama de clases UML**



Se adjuntará la imagen para mejor visualización, en la carpeta del proyecto.

**PA.2. Se debe hacer un menú para el Sistema donde ofrezca las funcionalidades de: 1) Edición o modificación del elemento y 2) Eliminación del elemento. Esto para cada una de las 2 colecciones anidadas.**

Se muestra un menú con distintas opciones en el que se solicita ingresar el número de la acción deseada, dichas acciones serán:

1. Añadir una nueva Mina

2. Mostrar datos de una Mina

3. Mostrar todas las minas de una región

4. Mostrar todas las minas de un país

5. Modificar datos de una Mina

6. Eliminar una mina

7. Salir

En el caso de que se ingrese un valor no valido, se volverá a solicitar un número de los que se solicita.

Todo esto se agrega al menú echo por ventanas graficas que se solicita en la parte B.

**PA.3. Se debe generar un reporte en archivo txt que considere mostrar datos de las 2 colecciones anidadas.**

Se genera un reporte de las colecciones anidadas, con la información de cada mina que hay en el programa en un archivo llamado “Reporte.txt”.

**PA.4. El código fuente debe estar bien modularizado de acuerdo con lo descrito en el informe además de seguir las buenas prácticas de documentación interna y legibilidad.**

Para lo que se nos solicita en este punto, disponemos de varias Clases para distribuir las diversas tareas que lleva a cabo el proyecto:

1. Clase GranMineria, es la clase Main del proyecto, es una ventana grafica donde se encuentran todas las funciones del programa, cada función es una ventana grafica dedicada a la función solicitada.
2. Clase Minas, contiene los todos los métodos de para poder manejar datos de tipo Mina
3. Clase Pais, contiene una lista de regiones y métodos para operar esta lista
4. Clase Regiones, contiene una lista de minas y métodos para operar con minas y la lista de minas
5. Tenemos Varias clases tipo JFrame, donde se codifican las funciones solicitadas para el proyecto; AñadirMina, MinasPais, MinasRegion, ModificarDatos y así otras más.

**EP4.1 Se deben incluir al menos 2 funcionalidades propias que sean de utilidad para el negocio (distintas de la inserción, edición, eliminación y reportes)**

* Seleccionar un objeto por criterio: Se implemento un método que busca la mina que tiene la mayor producción dentro de una región, se pide el nombre de la región y se entrega la mina que tiene mayor producción.
* Subconjunto filtrado por criterio: Dentro de todas las regiones aplicamos un filtrado por la producción de minas, al usuario se le solicita que ingrese los limites inferiores y superiores para que se muestren todas las minas que cumplen con esta condición.

**EP4.2. Diseño y codificación de 2 (dos) clases que utilicen sobreescritura de métodos**

* Tenemos la clase país que tiene el método addRegion y su clase extendida que es region que también tiene un método llamado addRegion, region sobre escribe el método heredado de país
* Tenemos la interfaz LimpiarVista, que se sobrescribe en todas las ventanas graficas

**EP4.3. Diseño y codificación de 1 (una) clase abstracta que sea padre de al menos 2 (dos) clases. La clase abstracta debe ser utilizada por alguna otra clase (contexto).**

Tenemos la clase abstracta país, que extiende a region

**EP4.4. Diseño y codificación de 1 (una) interfaz que sea implementada por al menos 2 (dos) clases. La interfaz debe ser utilizada por alguna otra clase (contexto)**

Se creo la interfaz LimpiadorVentanas, que se utiliza en la mayoría de las Clases JFrame con el de limpiar los datos que queden en los diferentes elementos de las ventanas, es una interfaz muy útil ya que cada ventana se limpia de forma diferente.

**PB.2. Se deben implementar al menos 3 ventanas gráficas (GUIs en AWT o SWING): 1 ventana de menú, 1 ventana de agregar elemento y 1 ventana de listar elementos.**

Se crea la ventana grafica GranMineria la cual posee un menu de opciones; Ingresar nueva mina, mostrar datos de una mina, mostrar todas las minas de un país, mostrar minas de una region, eliminar mina, búsqueda por rango, modificar datos de una mina, generar reporte. Cada opción esta implementado mediante ventanas graficas GUIs, en total son alrededor de 10 ventanas graficas.

**EF.2. Implementar el manejo de excepciones capturando errores potenciales específicos mediante Try-catch.**

Se tiene que en las ventanas graficas se una un bloque try-catch para asegurar que se ingresen los datos de forma correcta dentro de un ciclo for, esto es así en la materia de las ventanas graficas donde se usan ciclos for.

**EF.3. Crear 2 clases que extiendan de una Excepción y que se utilicen en el programa.**

**EF.4. Aplicación el patrón de diseño Strategy (Estrategia)**

Se tiene un grupo de ventanas graficas; BuscarPorNombre, BuscarPorPais, BuscarPorRangos, BuscarPorRegion. En las cuales se reutiliza código para facilitar su implementación y aplicar el patrón de diseño Strategy.

**EF.4. (opcional) Reemplazar los datos iniciales en el código por una conexión con base de datos local (MySQL), archivo de texto CSV o Excel, para la persistencia de datos.**

Se llena la base de datos del programa a través de la lectura de información de un archivo CSV llamado “Minas.csv”.