

|  |
| --- |
| Documento de Análisis y Diseño |
| Nivel 14: CupiColegios  [Felipe Otálora] |
| [Sebastián Florez] |
| Estructuras de Datos  Universidad de Los Andes  [III] |
|  |

Documento de Análisis y Diseño

[N14 CupiColegios]

|  |  |  |
| --- | --- | --- |
| Versión del documento | Fecha de modificación | Responsables |
|  |  |  |
|  |  |  |
|  |  |  |

Contenido

TOC \o 2-2 \t "Heading, 3"

Requerimientos Funcionales PAGEREF \_Toc10 \h 4

Análisis del modelo del mundo PAGEREF \_Toc11 \h 7

Diseño de Interfaces PAGEREF \_Toc12 \h 8

Interfaces del mundo de la aplicación PAGEREF \_Toc13 \h 8

Interfaces de las estructuras de datos PAGEREF \_Toc14 \h 8

Diseño de estructuras de datos PAGEREF \_Toc15 \h 9

Justificación de decisiones de diseño de las estructuras de datos PAGEREF \_Toc16 \h 10

Diseño final de la aplicación PAGEREF \_Toc17 \h 11

Justificación de decisiones de diseño del mundo PAGEREF \_Toc18 \h 13

Análisis de complejidad de operaciones del mundo PAGEREF \_Toc19 \h 14

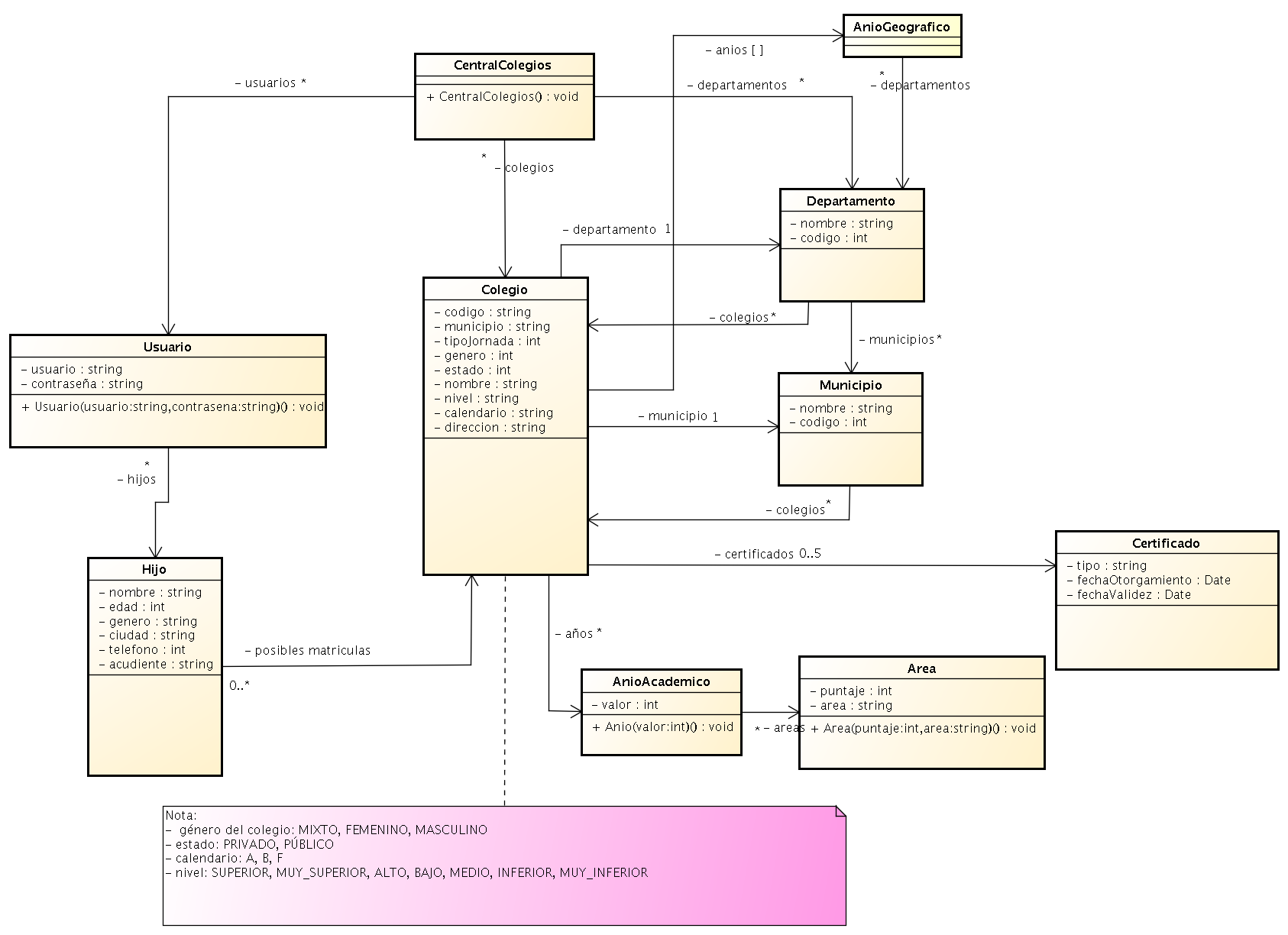
# Requerimientos Funcionales

A continuación se describen los requerimientos funcionales de la aplicación. La definición de estos es independiente de la tecnología que será usada para implementarlos.

|  |  |
| --- | --- |
| ID | <identificador único, generalmente un consecutivo> |
| Nombre | <nombre, comienza con verbo en infinitivo> |
| Resumen | <breve descripción del requerimiento> |
| Entradas | <nombre entrada i> <tipo> <descripción> |
| Resultados | <nombre resultado i> <tipo> <descripción> |

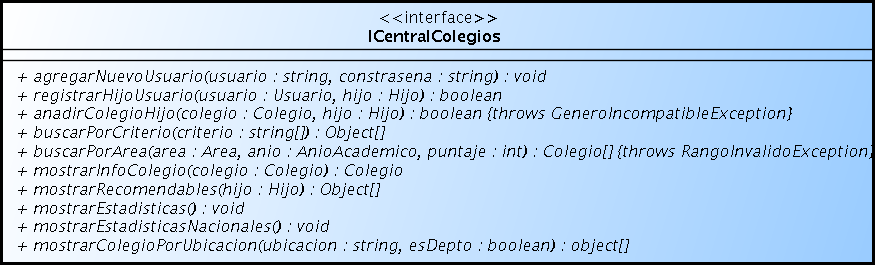
|  |  |
| --- | --- |
| ID | R1 |
| Nombre | Registrar un usuario |
| Resumen | La aplicación debe poder agregar un nuevo usuario a la lista de usuarios |
| Entradas | * usuario: El nombre del nuevo usuario * contraseña: La contraseña del usuario |
| Resultados | Se ha agregado un nuevo usuario al sistema |
| ID | R2 |
| Nombre | Registrar un hijo de un usuario |
| Resumen | La aplicación debe permitir agregar un hijo para un usuario |
| Entradas | * usuario: El nombre del usuario para agregarle el hijo * hijo: El nuevo hijo del usuario identificado con el nombre |
| Resultados | Se ha agregado un nuevo hijo al usuario dado y al sistema |
| ID | R3 |
| Nombre | Añadir un colegio a la lista de posibles matrículas |
| Resumen | La aplicación le debe permitir al usuario agregar, para cada hijo, un colegio a la lista de posibles matrículas. No debe ser posible agregar una matrícula si el género del hijo es incompatible con el colegio. |
| Entradas | * hijo: El hijo del usuario * colegio: El colegio que se quiere agregar al hijo |
| Resultados | Se ha agregado un nuevo colegio a la lista de colegios favoritos para el hijo |
| ID | R4 |
| Nombre | Buscar por criterio |
| Resumen | La aplicación debe poder buscar los colegios que cumplan con cierto criterio. Se puede filtrar los colegios por la jornada, el nivel del ICFES, género, calendario, y si es público o privado. |
| Entradas | * criterio: El tipo del criterio con el que se quiere buscar |
| Resultados | Se muestra el resultado de la búsqueda o vacío en caso contrario |
| ID | R5 |
| Nombre | Buscar por área |
| Resumen | Para una área determinada y un año determinado se desea buscar los colegios que obtuvieron un puntaje en un rango dado. |
| Entradas | * Area: El área de la búsqueda * Año: El año de la búsqueda * Rango: El rango de puntajes |
| Resultados | Los colegios que cumplen con los criterios |
| ID | R6 |
| Nombre | Mostrar la información del colegio |
| Resumen | La aplicación debe mostrar la información del colegio seleccionado. |
| Entradas | * Colegio: El colegio para mostrar la información |
| Resultados | Se ha mostrado la información del colegio seleccionado |
| ID | R7 |
| Nombre | Mostrar los colegios más recomendables dentro de los favoritos |
| Resumen | La aplicación debe mostrar los colegios recomendados dentro de los favoritos. Un colegio se recomienda si su nivel es SUPERIOR o MUY\_SUPERIOR y el promedio de todas sus áreas es mayor a 7. |
| Resultados | Se han mostrado los colegios recomendados dentro de los favoritos. |
| ID | R8 |
| Nombre | Mostrar estadísticas nacionales de calidad de colegios en gráficas. |
| Resumen | La aplicación debe permitir mostrar la información de los colegios en diferentes gráficas. |
| Entradas | - Gráfica: El tipo de gráfica que se quiere visualizar |
| Resultados | Se ha mostrado la gráfica que se quiere ver |
| ID | R9 |
| Nombre | Mostrar estadísticas nacionales |
| Resumen | Los usuarios debe poder ver las estadísticas nacionales de cada uno de los departamentos para sus colegios y para cada año. Y junto con el área mejor calificada para el colegio. |
| Entradas | * año: El año dado para la búsqueda |
| Resultados | Se ha mostrado la información de los departamentos para el año dado |
| ID | R10 |
| Nombre | Mostrar todos los colegios de una ubicación particular |
| Resumen | El usuario debe poder consultar todos los colegios que se encuentran en una ubicación particular. |
| Entradas | * ubicación: la ubicación sobre la que se quiere buscar |
| Resultados | Se ha mostrado la información de los colegios relevante a la búsqueda |

# Análisis del modelo del mundo

Diagrama de clases (UML) de la abstracción del problema que se pretende solucionar. Solo se tienen en cuenta entidades, características y relaciones relevantes para el mismo. Aún no se toman decisiones de diseño.

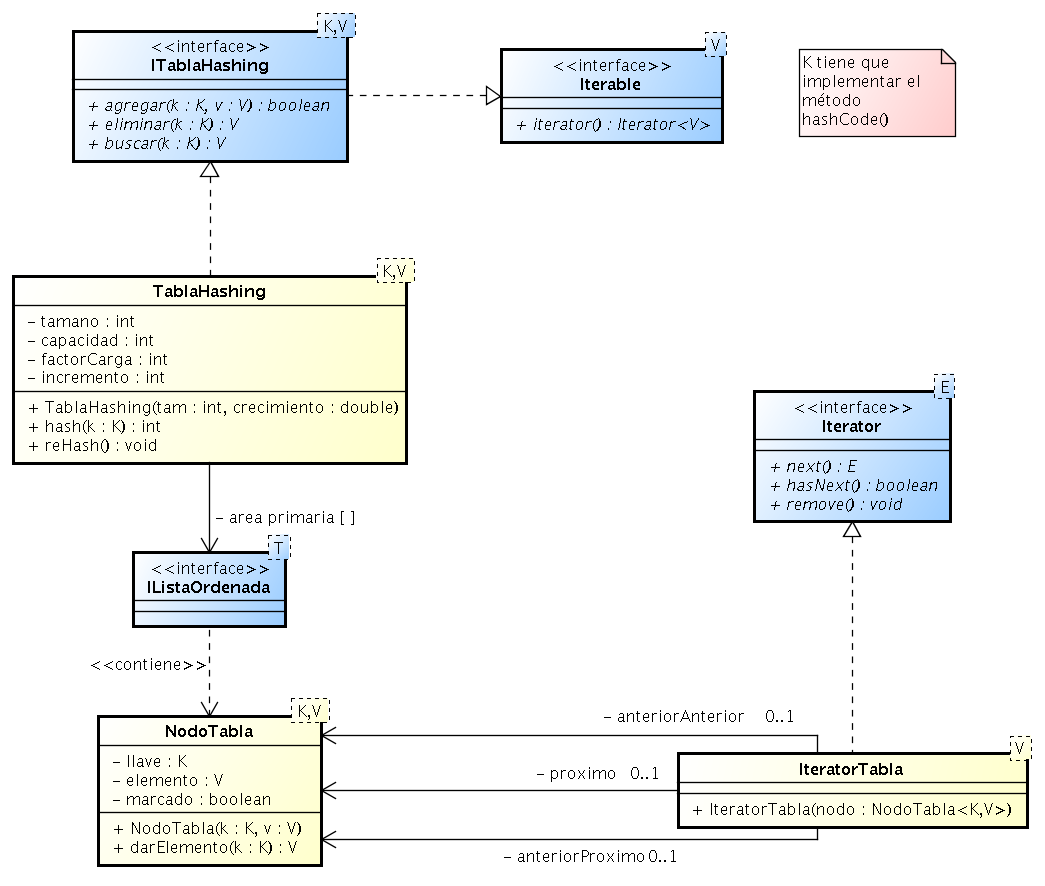
# Diseño de Interfaces

## Interfaces del mundo de la aplicación

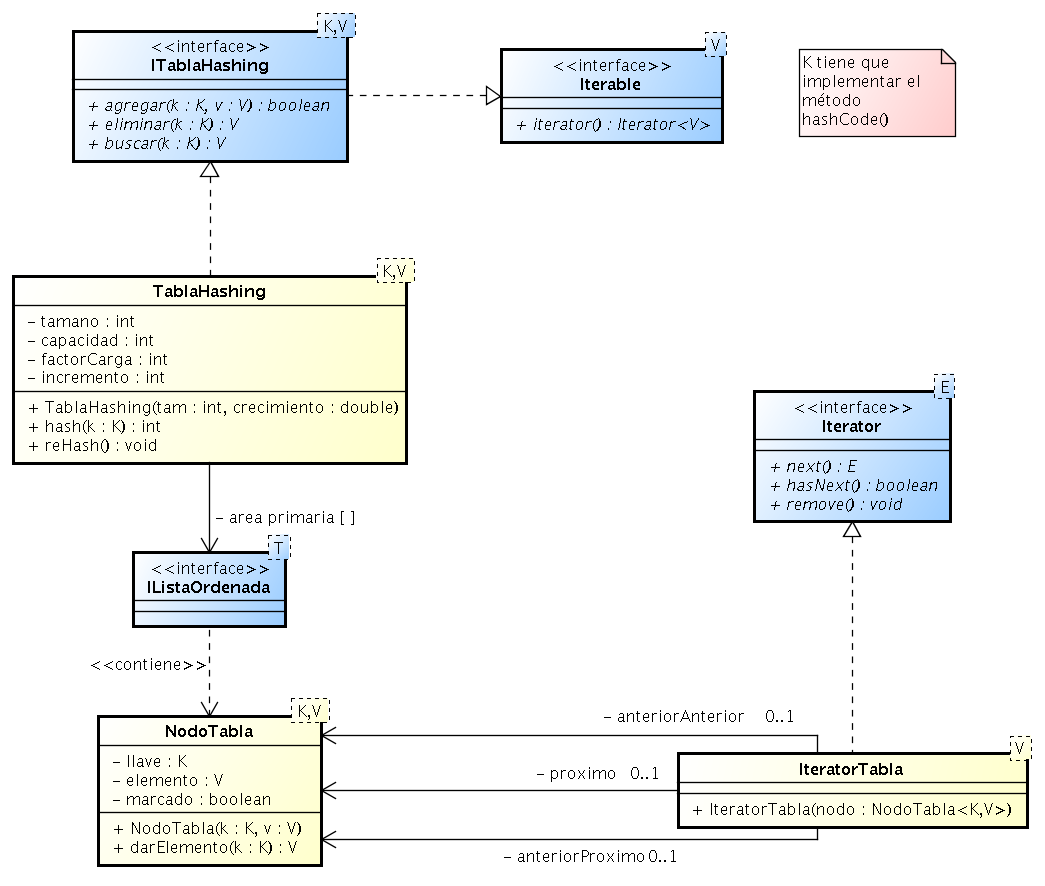
A partir de los requerimientos funcionales, defina las operaciones (a través de métodos) que debe proveer el mundo del problema. Incluya parámetros, tipos de retorno y documentación completa de cada operación (incluyendo precondiciones).

## Interfaces de las estructuras de datos

Seleccione cuáles son las estructuras de datos genéricas que debe utilizar y las operaciones que estas deben proveer para minimizar la complejidad temporal de las operaciones definidas en el numeral 3.1.



# Diseño de estructuras de datos

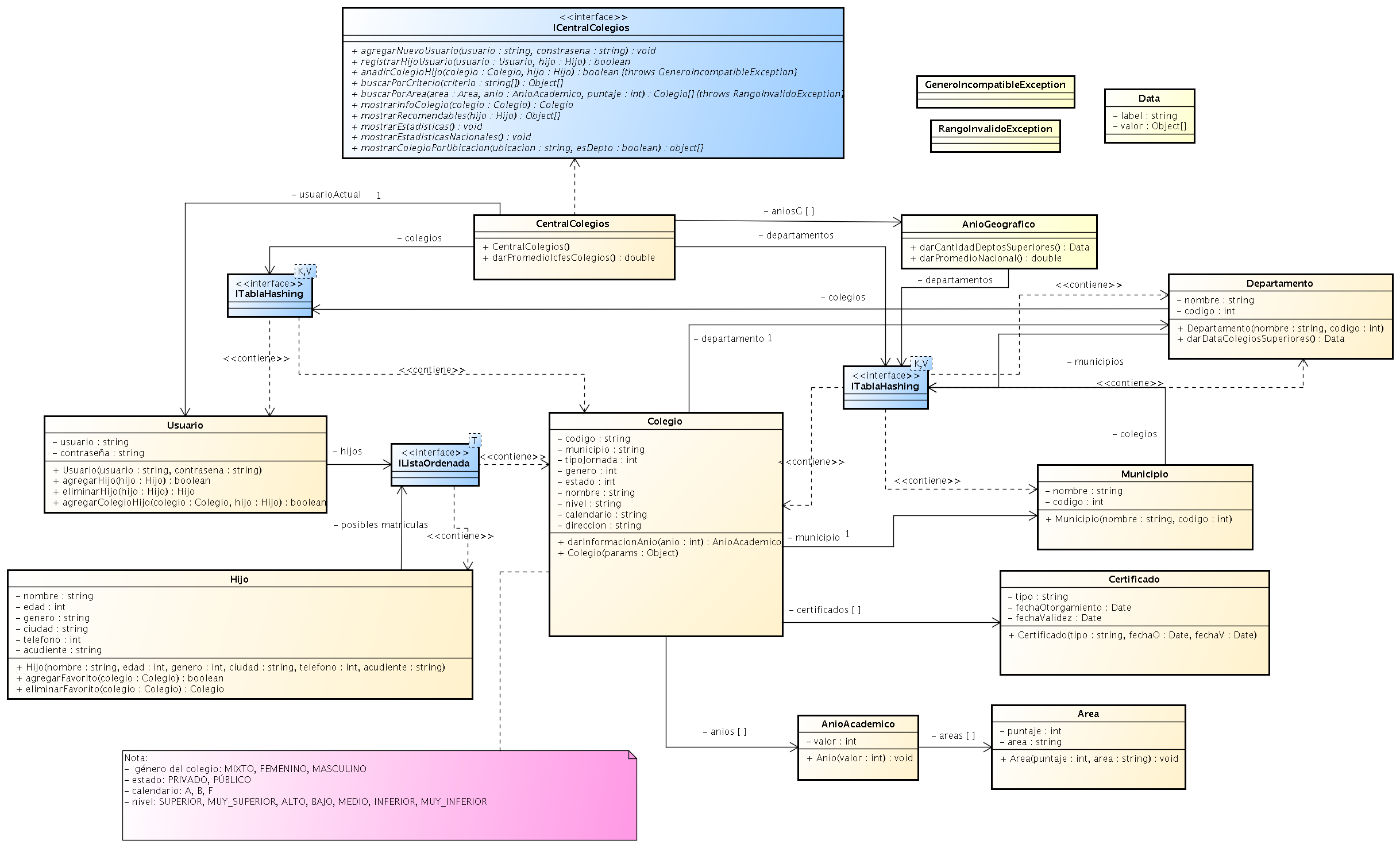
Describa el detalle de las Estructuras de Datos que va a utilizar en su solución, a través de un diagrama UML de las mismas. No olvide incluir invariantes si las requiere.

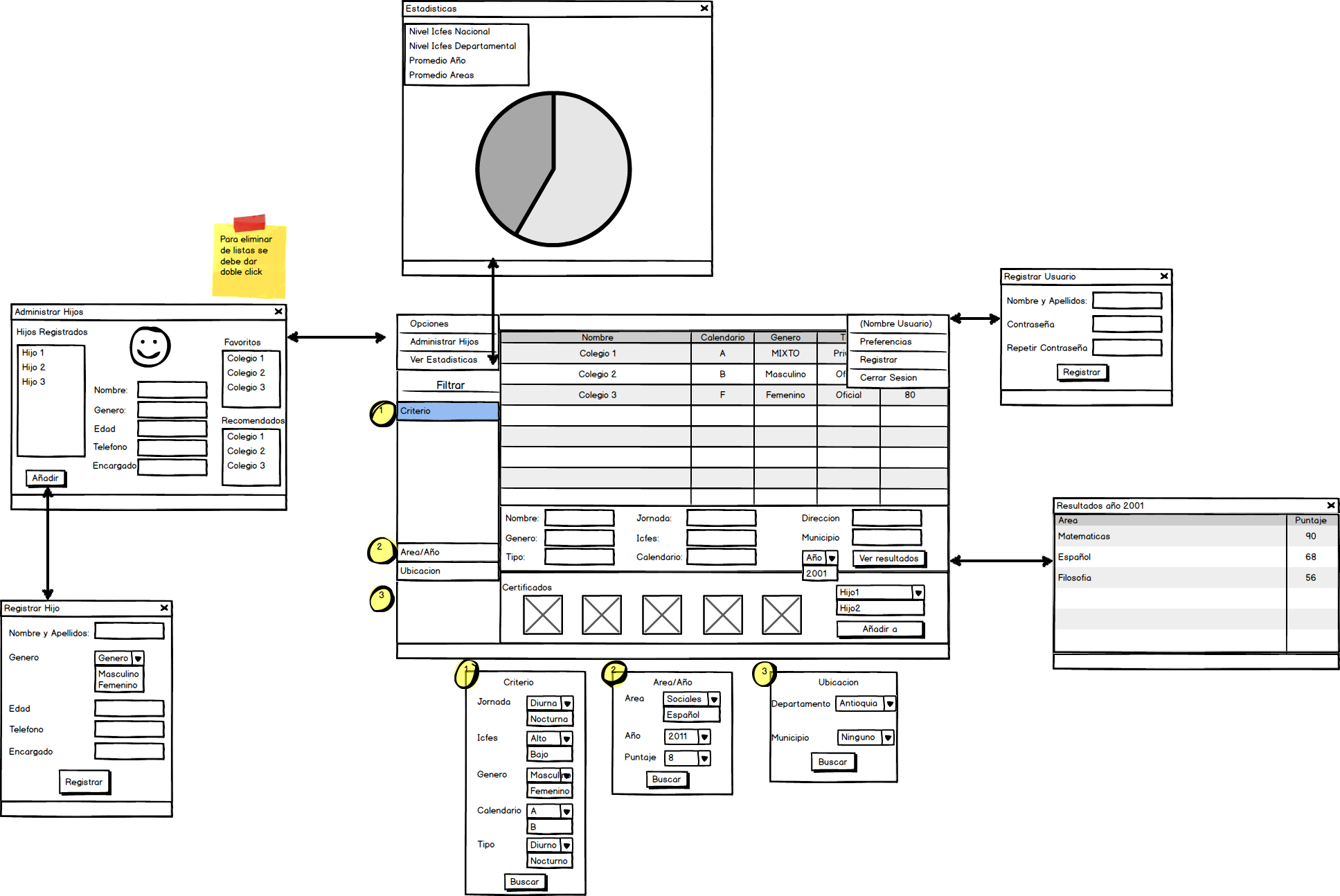
# Justificación de decisiones de diseño de las estructuras de datos

Describa en detalle los aspectos que tuvo en cuenta para seleccionar y diseñar las estructuras de datos descritas en el punto anterior.

Las tablas de hashing permiten maximizar la eficiencia a la hora de realizar búsquedas debido a que se reduce la complejidad de la tarea por ser una estructura de acceso directo y ya que la cantidad de datos que se manejan es considerablemente alta.

# Diseño final de la aplicación

Describa en detalle el diseño final de su aplicación a través de un diagrama UML, incluya atributos y métodos. Cuando se refiera a una estructura de datos no es necesario que incluya toda su definición dentro del diagrama. Basta con referenciar su interfaz.



# Justificación de decisiones de diseño del mundo

Describa en detalle los aspectos que guiaron el diseño final de su aplicación.

Se tendrá la clase CentralCupiColegios, la cual contara con las referencias a todos los usuarios, departamentos, municipios, usuarios y años. Esto permite el fácil acceso a los distintos posibles aspectos para la búsqueda.

La estructura de datos de tabla de hashing se empleara principalmente en objetos de los cuales se sepa se tendrán cantidades importantes de instancias, así se lograra aumentar la eficiencia en especial para los requerimientos de búsqueda.

Se decidió implementar una clase año que puede ser geográfico o académico, esta distinción se realizo principalmente porque las gráficas exigen ciertos criterios para la realización de las gráficas. La geográfica se encargara de saber las instancias de departamentos en cada año, y la académica las instancias de notas de áreas de un colegio, dado el mismo. Estas por su cuenta, sabrán reportar su información.

La utilización de lista ordenada será para fines de acceso de información de menor magnitud que un departamento, o una lista de usuarios. Esta principalmente será para el uso de los usuarios con sus hijos y de los hijos con sus colegios agregados a favoritos.

# Análisis de complejidad de operaciones del mundo

Basado en los diseños de las estructuras de datos y en el mundo de la aplicación, defina la complejidad de cada una de las operaciones. Utilice la notación O(f(n)). Justifique por qué considera que esta es la mínima posible para el problema que está resolviendo.

* R1: Agregar un usuario O(1) La tabla de hashing permitirá el almacenamiento directo según la dirección dada por el método hash. En el peor de los casos ocurrirán colisiones que harán que se almacene en la respectiva lista ordenada, sin embargo se reduce drásticamente la cantidad de objetos en ella.
* R2: Registrar Hijo O(n) Debido al uso de la lista ordenada su complejidad será de este orden, aun así, se sabe que una persona no posee gran cantidad de hijos así que en términos generales su eficiencia será alta.
* R3: Añadir colegio a favoritos O(n) se cumple el mismo caso que el de registrar un hijo, solo que este es ,as abierto a que se tenga una mayor cantidad de colegios almacenados.
* R4: Buscar por criterio O(n) Los criterios se encuentran presentes en los colegios que estarán almacenados en tablas de hashing. Sin embargo, se debe recorrer toda la lista.
* R5: Buscar año y área O(n): Se deben recorrer todos los colegios para saber sus resultados en el año dado.
* R6 Mostrar la información del colegio O(1): Cada colegio tiene su información como parámetros. También para la búsqueda del colegio se cuenta con la tabla de hashing.
* R7 Mostrar recomendados O(n): Se deben recorrer todos los colegios para averiguar su nivel de icfes de cada uno.
* R8 Mostrar estadísticas nacionales O(n): La clase relacionada a la estadística deberá generar el respectivo reporte con etiqueta y valor.
* R9 Mostrar Estadísticas O(n) : La clase relacionada a la estadística deberá generar el respectivo reporte con etiqueta y valor.
* R10 Mostrar colegios por departamento O(1): Cada departamento y/o municipio tiene acceso a sus colegios respectivos, el acceso será directo gracias a la estructura, pero se deberán recorren aun así los colegios contenidos.