**Informe de entrega  
Hito N°2**

Integrantes:

* Vicente Diaz
* Cristóbal Olave
* Aldana Bravo
* Ernesto Lagos

Profesor:

* Pablo Schwarzenberg

Índice

[Introducción](#_gjdgxs) **3**

[Backlog de Historias](#_30j0zll) **4**

[Mapa de historia](#_1fob9te) 4

[Modelo 4+1](#_3znysh7) **5**

[Diagrama de Clases](#_2et92p0) 5

[Diagrama de Estados](#_3dy6vkm) 7

[Diagrama de componentes](#_17dp8vu) 8

[Diagrama de despliegue](#_1ksv4uv) 9

[Diagrama Base de Datos](#_44sinio) 9

1. Introducción

En el presente informe se hará un desglose con toda la información de cómo se va a estructurar el sistema a desarrollar para el proyecto de la academia “As your wish”, con la finalidad de que se pueda tener una mejor comprensión de cómo funcionará y por qué tiene la construcción señalada.

1. Backlog de Historias
   1. Mapa de historia

Dado que la Historia de Usuario Principal es:

**“Como integrante de la Academia, en calidad de administrador, quiero ser capaz de gestionar una o varias Listas, con el fin de tener todo organizado de manera eficiente y óptima”**

De lo citado anteriormente, se deduce que el mapa de historia debe tener una lógica determinada a la hora de ingresar al Sistema para realizar algún tipo de actividad. Para ello, lo primero en considerar es *Ingresar al Sistema*, una vez realizado los detalles, se puede optar por dos patrones de orden distintos. Se optó por seleccionar primeramente el patrón de *escoger la Lista deseada o Escoger gestion de Usuario* para luego *decidir qué hacer con ella*, teniendo en cuenta que las opciones son: modificar, eliminar y agregar. Una vez realizadas las actividades deseadas, se da la opción de *guardar los cambios,* y luego *cerrar el Sistema* para que ningún usuario que no tenga los accesos permitidos pueda realizar algún tipo de acción no deseada.

 La Historia de Usuario estaría conformada de la siguiente manera:

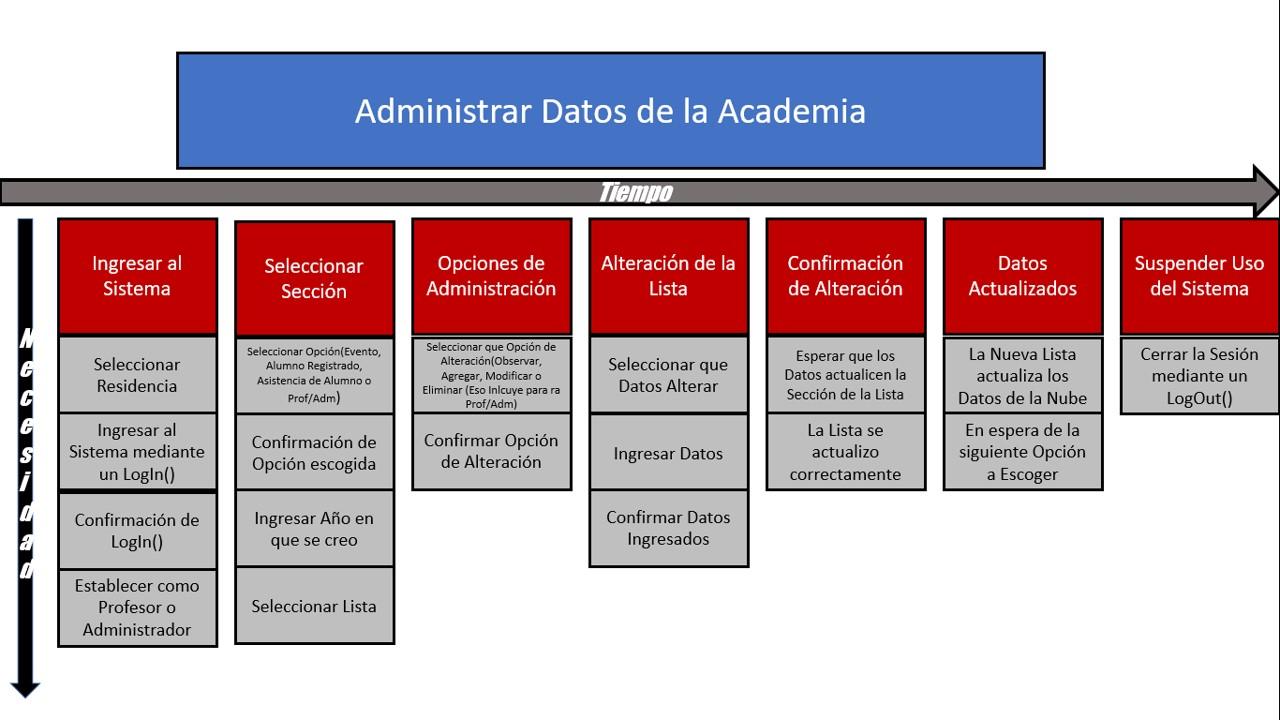


Figura 2.1.1 Mapa de Historias

1. Modelo 4+1
   1. Diagrama de Clases

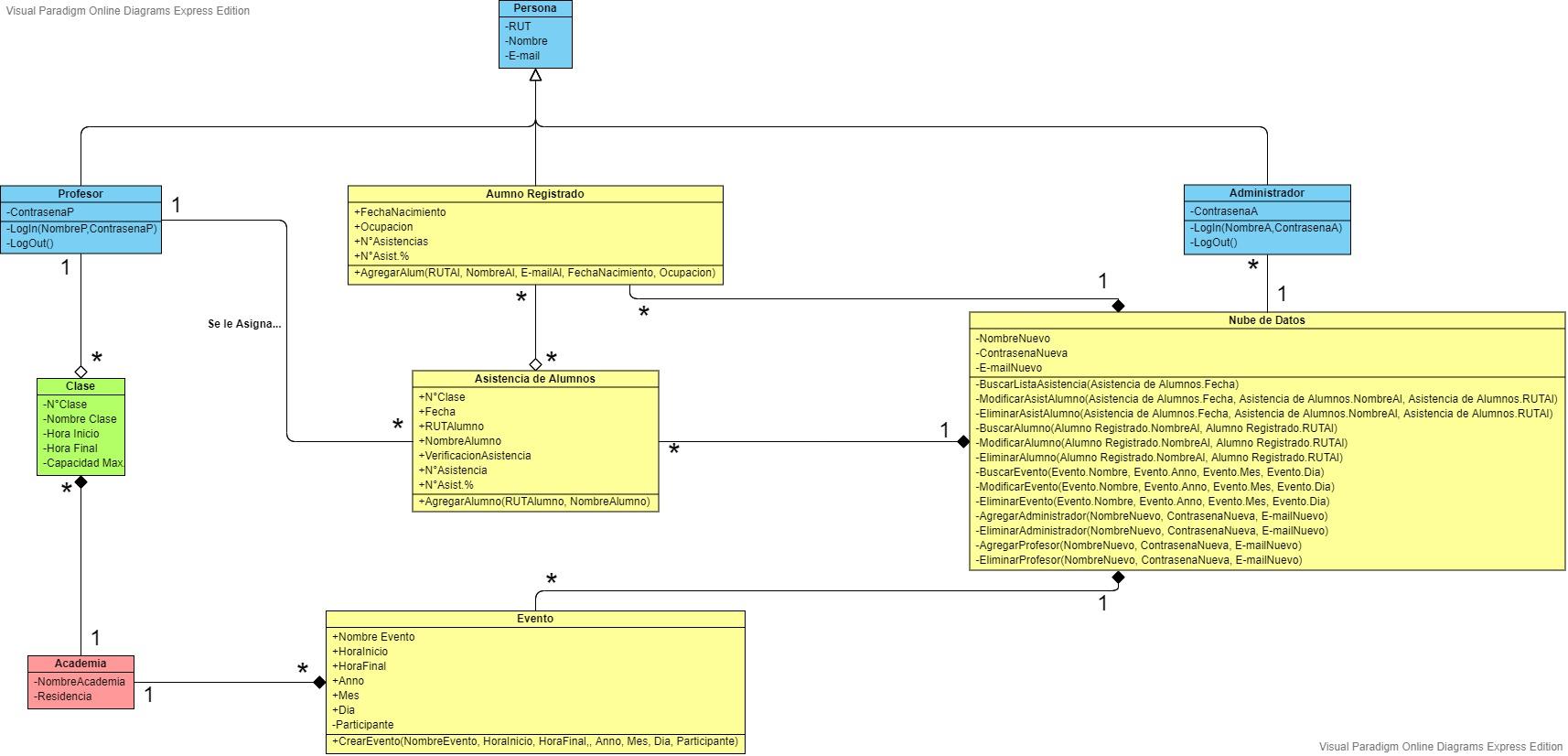


Figura 3.1.1 Diagrama de clases

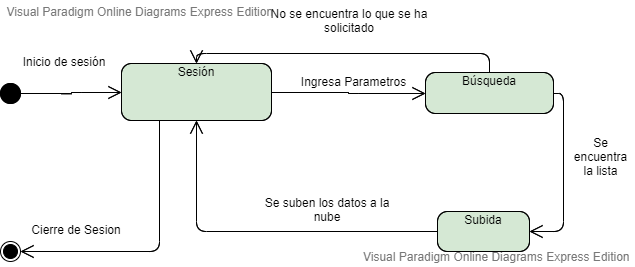
Para Nuestro Diag. De Clases “As Your Wish”, se puede mostrar que 3 Clases presentes heredan de un Clases de Atributos comunes, de esta forma uno ahorrará líneas de código al no repetir Atributos fundamentales, para la creación de Objetos de alguno de las 3 Clases.

A la Clase *“Profesor”*, posee asignado **una o más** “Asistencia de Alumnos” (De su correspondiente Clase), esto se debe a que debe tener una conexión vaga pero No Inexistente a la hora de registrar a los Alumnos, pues en caso de que un Profesor se ausente por ciertas razones, existe la posibilidad que otro reemplace su Lugar, por ende, la Clase *“Profesor”* necesita una asignación para así mantener una conexión entre CLASES-ALUMNOS.

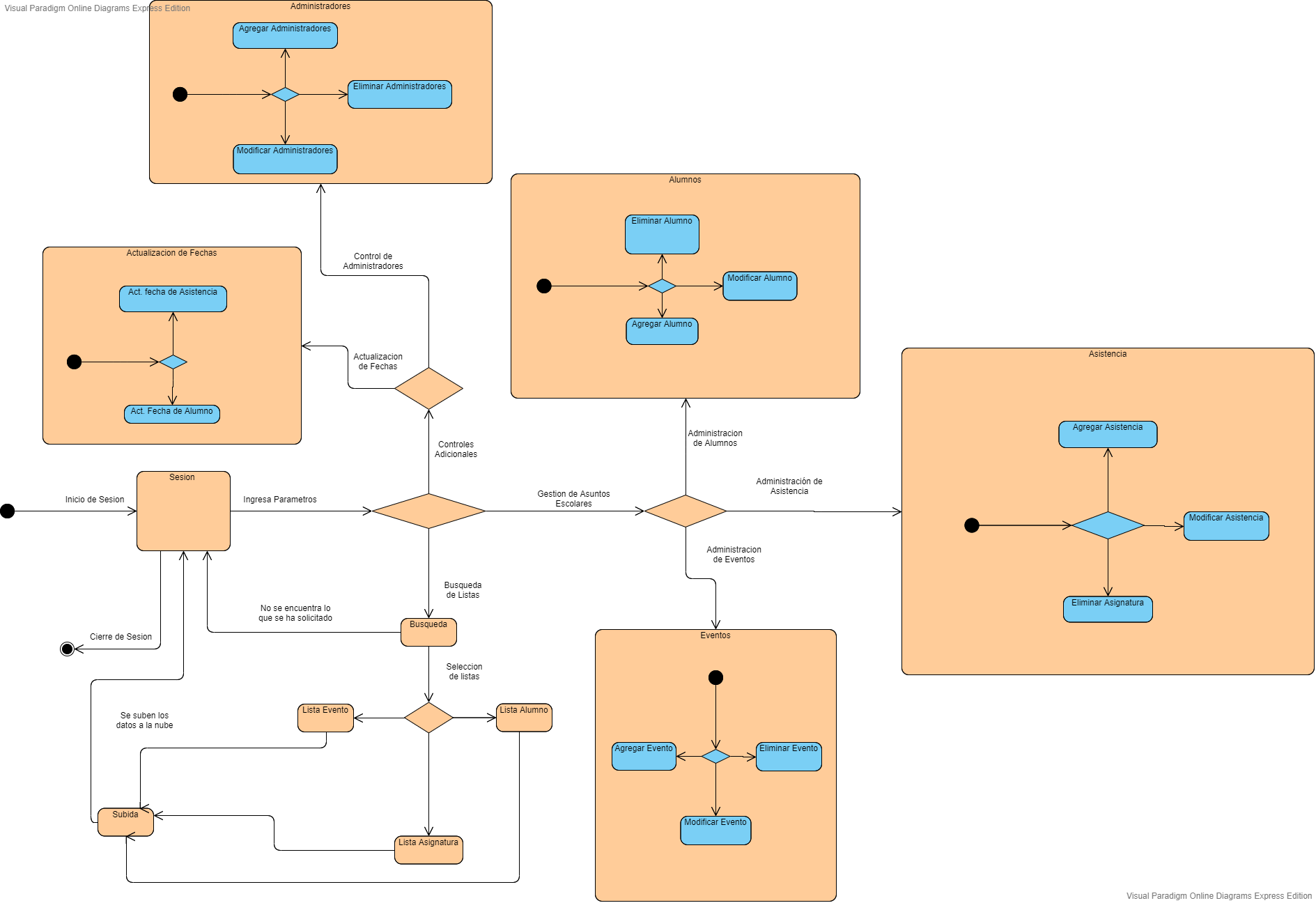
A la Clase *“Alumno registrado”* únicamente sirve para ser asignado en “Asistencia de Alumno” & ser parte importante de la “Nube de Datos”, otorgando una Lista de Alumnos con sus respectivos Datos o Atributos específicos.

Por último, la Clase *“Administrador”* será el único que tendrá asignado la Clase *“Nube de Datos”*, la cual será la composición de las 3 Clases que sirven para hacer una Lista, “Alumno Registrado, Asistencia de Alumno & Evento”, dentro de esta Clase se tendrá una gran variedad de Operaciones que permitirá gestionar una diversa cantidad de Atributos o Objetos, incluso Lista. Se optó por dejar estas operaciones en la “Nube de Datos” pues únicamente se puede acceder si se es Administrador, restringiendo cualquier ingreso por otro medios

* 1. Diagrama de Estados



Esta imagen corresponde al diagrama de estados del profesor, en el que describe como el profesor interactúa con el sistema, principalmente lo hace mediante la búsqueda de la lista de los estudiantes y la subida de ésta.



Esta imagen corresponde al diagrama de estados del administrador, en donde se aprecia la diferencia de funciones que se pueden hacer en el sistema en comparación con el profesor, entre estas funciones se encuentran:

* Asignación de Fechas.
* Organización de Eventos.
* Administración de Asistencia.
* Gestión de Alumnos.
* Gestión de Administradores.
* Subida de Datos.
  1. Diagrama de componentes

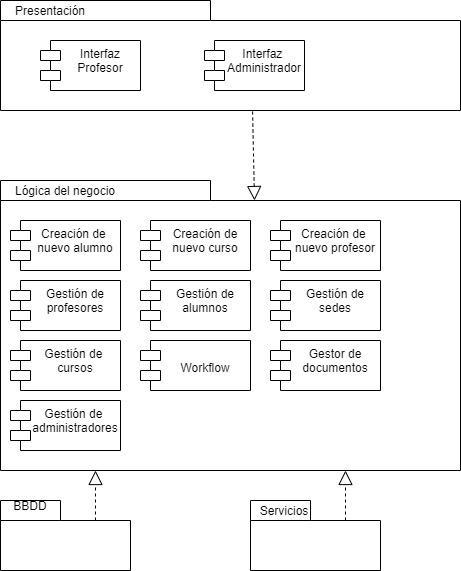


Figura 3.3.1 Diagrama de componentes

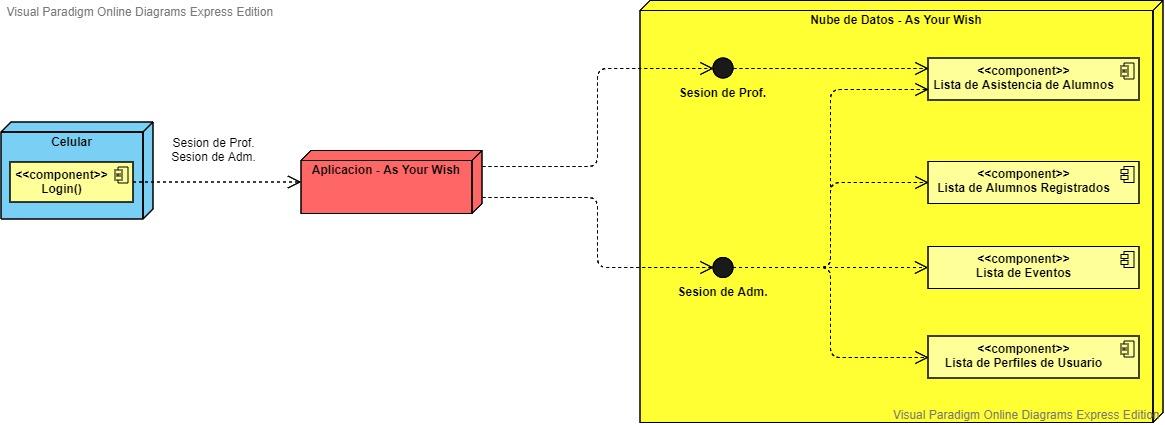
En el diagrama de componentes se pueden destacar dos grandes packages. En el primero tenemos la presentación, la cual apunta a las interfaces que acceden los distintos tipos de usuarios. Luego, se puede observar la lógica del negocio, donde tenemos las distintas acciones que se pueden realizar.

De los usuarios, el de mayor dominio es el Administrador, el cual puede gestionar los distintos tipos de listas, ya sea de alumnos, sedes, profesores, cursos, gestionar información a través de distintos

documentos e incluso administrar otros administradores. Estas gestiones incluyen modificar, agregar o eliminar algún elemento de dichas listas. El usuario Profesor, podrá realizar algunas gestiones sobre el listado de alumnos, por ejemplo, podrá registrar la asistencia de estos a las distintas clases impartidas, además de visualizar la información de ellos.

Por último, se tienen los packages de los cuales dependerá el sistema, como la base de datos respectiva que contiene toda la información a manejar y los distintos servicios que se puedan necesitar.

* 1. Diagrama de Despliegue



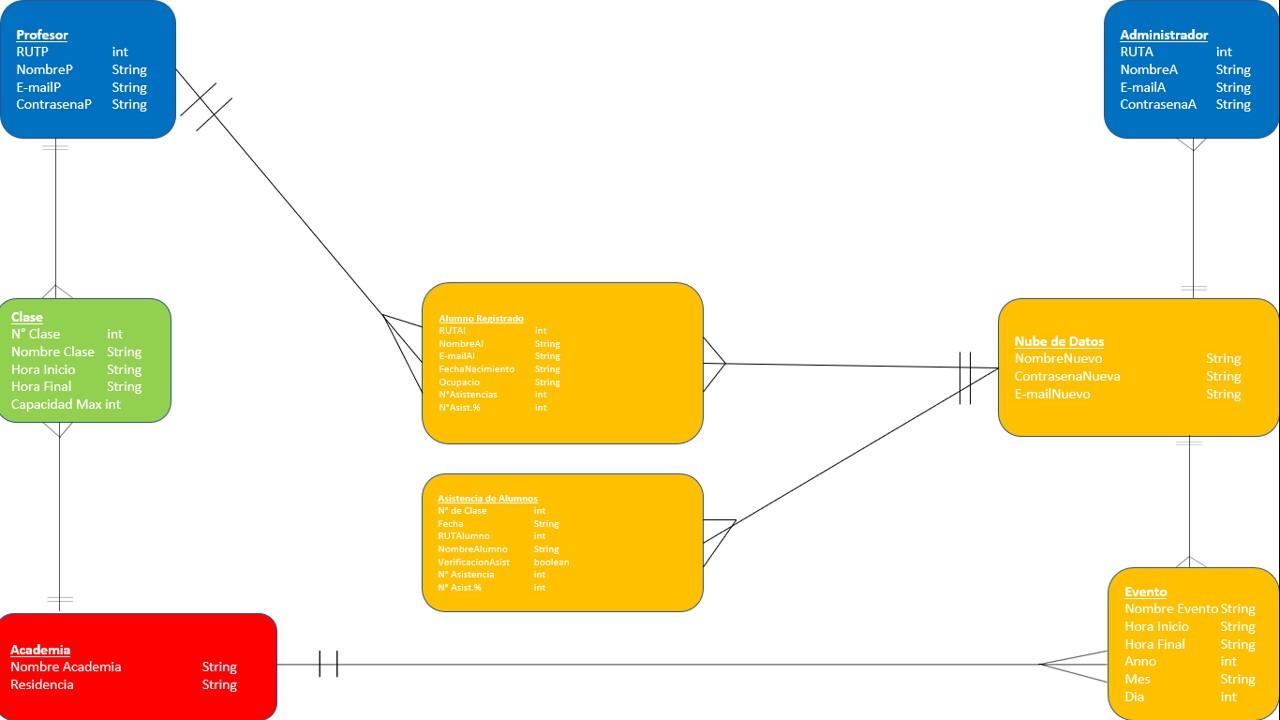
Para la Arquitectura del Sistema, se tendrán dos Interfaces que especificara como el Sistema se debe comportar dependiendo del Inicio de Sesión que se ingrese.

Ingresando la “Sesión de Prof.” a través de Nodo Celular este ira al Nodo de la Aplicación que lo llevara a la Interfaz de Sesión de Prof. donde únicamente podrá interactuar con la Lista de Asistencia de Alumnos.

Si en caso contrario este Ingreso con “Sesión de Adm.” el Nodo de la Aplicación lo llevara a la Interfaz de Sesión de Adm. donde tendrá acceso total a todos los Componentes de la Nube de Datos.

Básicamente mediante un Inicio de Sesión en la Aplicación se tendra acceso a la Nube de Datos del Sistema.

* 1. Diagrama Base de Datos



En el Diagrama de B.Datos, se detalla de qué tipo será cada Atributo utilizado para el Sistema, así el como se relacionan entre ellos

* 1. Trazabilidad Historia - Componente

