

## RELEASE NOTES VERSIÓN 1

### Catálogo de Aplicaciones

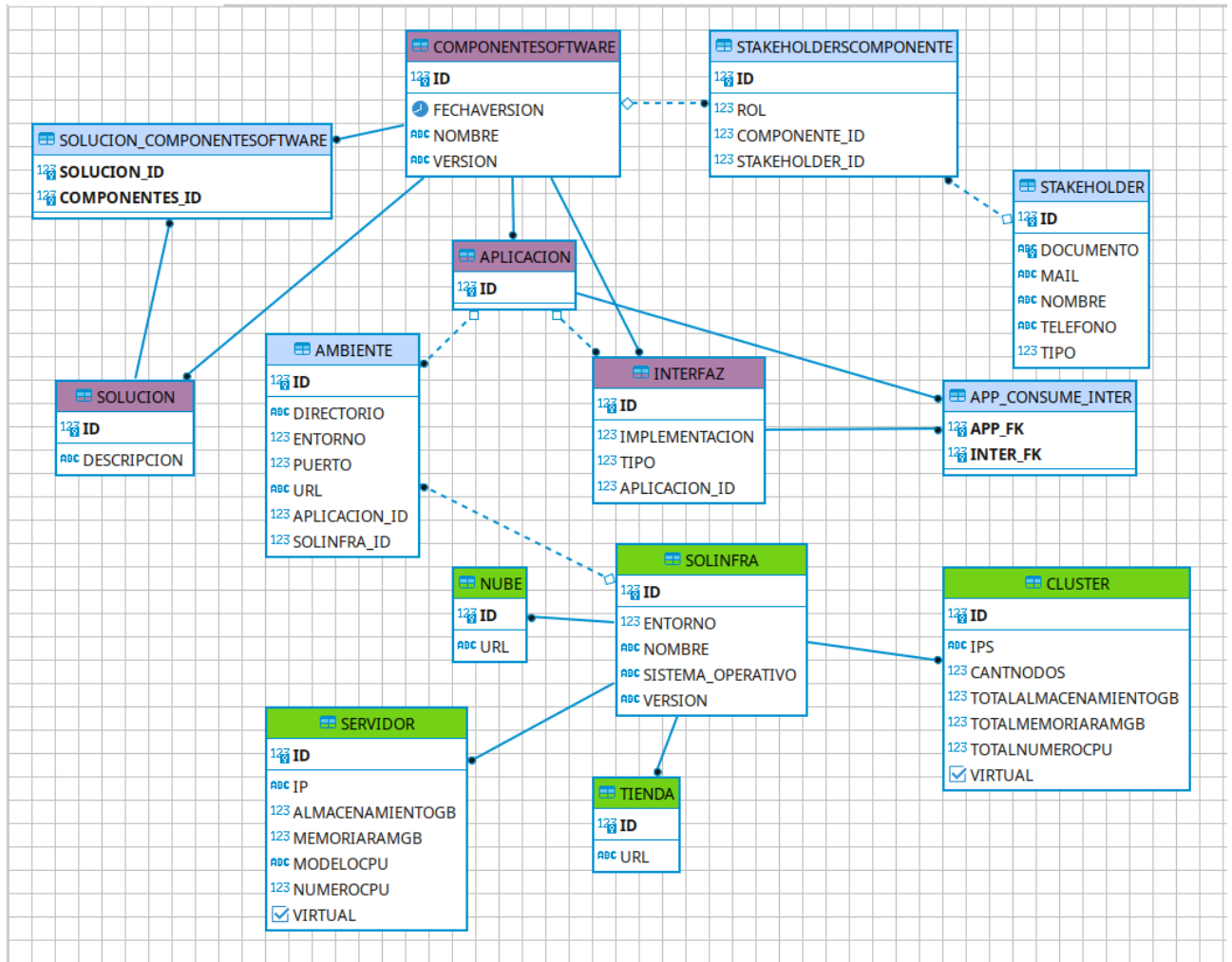
**Sergio Azcurrain; Alvaro Menéndez; Patricia Montaña**

---

1. A continuación, se listan los cambios realizados en la construcción de la aplicación:
  - Nomenclatura: se cambió **Usuario** por **Stakeholder**, para diferenciar los usuarios del login, de los interesados en los componentes registrados en la aplicación.
  - Clases: la clase Solución hereda de la clase ComponenteSoftware, para facilitar las consultas JPA al devolver las Aplicaciones, Soluciones e Interfaces relacionadas. Además permite que una Solución esté compuesta de otras soluciones (soluciones anidadas).
2. A continuación, se listan las mejoras implementadas no requeridas por la letra del obligatorio:
  - Se agrega en Consulta Simple y Avanzada: carga lazy paginada de datos desde el backend para mejorar la performance.
  - Se agrega en Consulta Simple y Avanzada: el paginado configurable.
  - Se agrega en ABM de Soluciones, las funcionalidades:
    - Modificar una existente
    - Eliminar una Solución.
  - Se agrega ABM de Aplicaciones, con las funcionalidades:
    - Insertar una nueva Aplicación
    - Modificar una existente
    - Eliminar una Aplicación.
  - Se agrega ABM de Interfaces, con las funcionalidades:
    - Insertar una nueva Interfaz (desde el ABM de Aplicación; Interfaces Provistas)
    - Modificar una existente
    - Eliminar una Interfaz.
  - Se agrega ABM de Infraestructura, con las funcionalidades:
    - Insertar una nueva Infraestructura
    - Modificar una existente
    - Eliminar una Infraestructura.

- En el servicio Alta de Usuario, se definen además, los protocolos get, put y delete, para soportar las operaciones CRUD (create, read, update, delete).

### 3. Modelo de Persistencia:



### 4. Notas para el uso de la aplicación:

- Usuarios (idem username y pass):
  - o usuario
  - o usuario1
  - o usuario2
  - o usuario3
  - o usuario4
  - o paty
  - o aem
  - o sergio

- Los Identificadores de las aplicaciones pre existentes son 7 y 9.
- Las URL de los servicios definidos son:
  - Para recuperar (get), modificar (put) y borrar (delete) stakeholders:

<http://localhost:8080/soluciones/servicios/rest/stakeholders/11>

Estructura para la operación put (body), donde son requeridos todos los campos:

```
{
  "nombre": "Pepe",
  "mail": "cacho@gmail.com",
  "telefono": "999999999",
  "tipo": 1,
  "documento": "2931343-3"
}
```

Para crear un nuevo stakeholder (post):

<http://localhost:8080/soluciones/servicios/rest/stakeholders>

Estructura para la operación post (body), donde son requeridos todos los campos:

```
{
  "nombre": "Pepe",
  "mail": "cacho@gmail.com",
  "telefono": "999999999",
  "tipo": 1,
  "documento": "2931343-3"
}
```

- Para devolver todos los componentes donde la persona identificada por esa CI, tiene rol Desarrollador:
 

<http://localhost:8080/soluciones/servicios/rest/stakeholders/<CI>/desarrollos>
5. A continuación, se listan los requisitos (funcionalidades) de la aplicación que se cumplen:
- Permite listar todas las Aplicaciones registradas, se hace a través de:
    - Menú Consulta, Consulta simple (se destacan los filtros de la búsqueda)
    - Menú Consulta avanzada (se destaca el paginado)
    - Menú Soluciones; Ver aplicaciones
  - Se respeta el diagrama y propósito de las Clases, acotado al alcance consensuado en clase, con una mejora ya explicada en el punto 1 (Clases).

- En cuanto a los atributos de cada clase, fueron contemplados las sugerencias por la letra del obligatorio.
- Requerimientos funcionales que se cumplen en su totalidad:

Módulo de Front End:

Página de bienvenida

Detalle de los componentes

Módulo de Back End:

Login de administrador

ABM de usuarios

ABM de soluciones

Interoperabilidad:

Consulta desarrollador

Consulta interfaces

Alta de usuario

**Git:**

<https://github.com/sazcurrain/soluciones-bse.git>