# **Formulario: Vista Lógica del Proyecto de Software**

**Proyecto:** SOSERVICE  
**Grupo:** GRUPO 5  
**Fecha:** 23/10/2024

**1. Descripción General del Sistema**

Proporciona una descripción general del sistema y los principales componentes lógicos que lo forman.

**Descripción del sistema:**

Nuestro sistema se basa en Ionic Angular + Firebase , y este se basa en una serie de páginas que cuentan con Login , páginas para registrarse , recuperar contraseña , una página de home donde se puede agregar una serie de “Servicios” que el usuario puede tomar para trabajar con la persona que lo sube.

**2. Principales Clases y Entidades del Sistema**

Describe las clases o entidades principales del sistema, sus atributos y métodos principales, y su propósito dentro del sistema.

**Clase 1 (Usuario):   
Atributos: Nombres, Apellido M, Apellido P, contraseña, email, número celular, descripción.  
Métodos: Login,recuperar contraseña, crear usuario, crear trabajo, aceptar trabajo.  
Propósito: Poder tener una cuenta para el uso de la app**

**Clase 2 (Trabajo):   
Atributos: Categoría, creator Id, descripcion Trabajo, image Url, requerimientos, timestamp.  
Métodos: Actualizar estado, calificación, detalle trabajo, message creador.  
Propósito: Crear publicaciones con objetivos claros mostrando una recompensa para aquellos que estén interesados en aceptar el trabajo y crear una conexión entre el solicitante y trabajador por medio de funciones de llamada o mensaje.**

**3. Relaciones entre clases**

Especifica cómo están relacionadas las clases entre sí, indicando relaciones de herencia, composición, agregación o dependencias.

**Relaciones entre clases:**

La clase usuario es la entidad principal la cual interactúa con toda la aplicación, en ella se registra, se loguea y procede a crear un trabajo o ver las opciones que le brinda la aplicación en tiempo real, por otro lado la clase trabajo hereda la id del creador de la misma para tener más información dentro de la publicación como el nombre completo y crear métodos como mensajería o llamada de acuerdo a la necesidad del creador.

**4. Interfaces y dependencias**

Indica si alguna de las clases o entidades tiene interfaces y cómo interactúan con otros módulos del sistema a través de ellas.

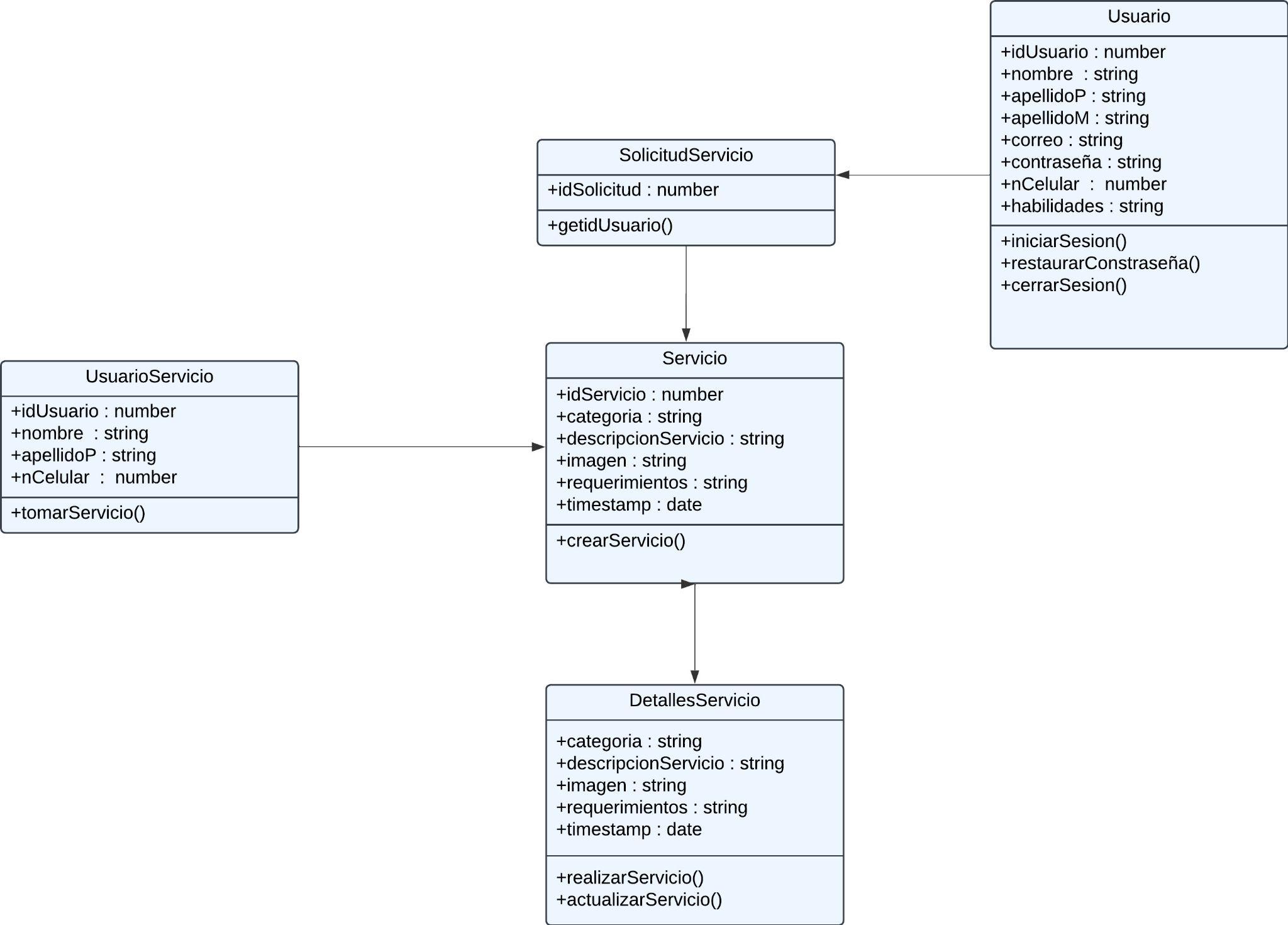
**Descripción de interfaces y dependencias:**

Todas las clases cuentan con su propia interfaz. Por ejemplo, la clase 1 tiene su interfaz, al igual que la clase 2, donde se establece el servicio deseado. Estas interfaces permiten interactuar de diversas formas, como completar formularios, reproducir cierta información, crear publicaciones, y acceder a secciones específicas del sistema.

**5. Diagrama de Clases del Sistema**

Adjunta un diagrama de clases que representa la estructura lógica del sistema, las clases principales y las relaciones entre ellas.

**Diagrama de clases:***(Adjunta el diagrama en esta sección)*



**6. Diagrama de componentes (opcional)**

Si su sistema lo requiere, adjunte un diagrama de componentes que represente cómo los componentes principales del sistema están organizados y cómo interactúan.

**Diagrama de componentes:***(Adjunta el diagrama en esta sección)*

**7. Comentarios adicionales**

Incluye cualquier comentario adicional o aclaración relevante sobre la vista lógica de tu sistema.

**Comentarios:**

Ya que trabajamos con una metodología ágil se podría realizar cambios futuros