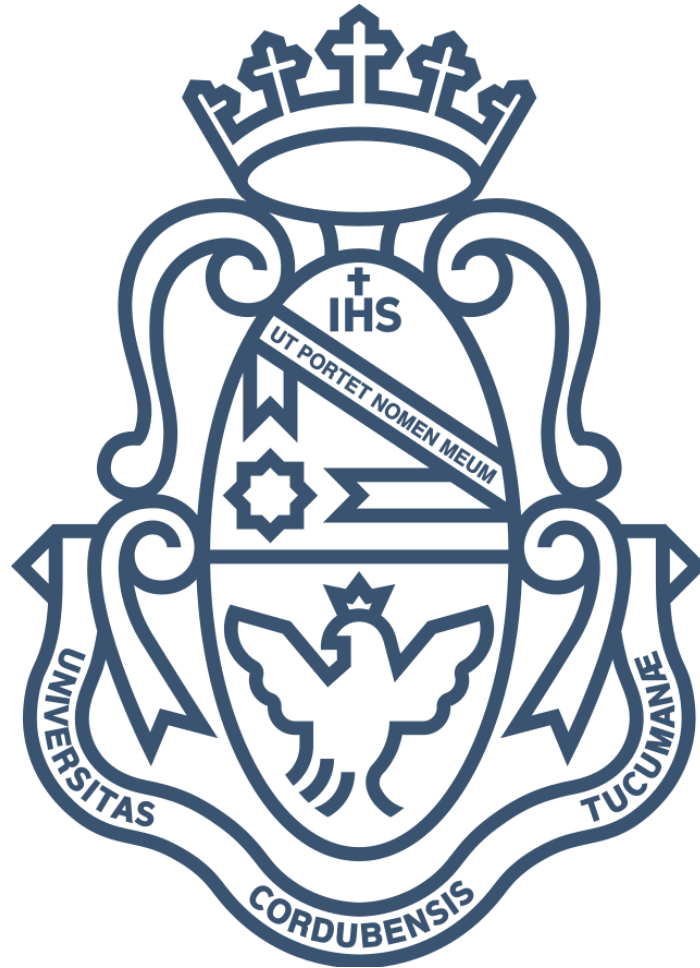


Universidad Nacional de Córdoba

Facultad de Ciencias Exactas Físicas y Naturales



Ingeniería en Software: Trabajo Práctico N° 1

Grupo: Levi

Integrantes:

- | | |
|---------------------------------|-----------------|
| • Renaudo, Bruno Agustín | DNI: 40.575.393 |
| • Moyano Seminare, Carlos Alejo | DNI: 40.802.646 |
| • Ibañez Sala, Ignacio | DNI: 42.722.786 |
| • Etchechoury, Candela | DNI: 41.711.294 |

Índice

1. Plan de Gestión de las Configuraciones

- 1.1 Introducción
- 1.2 Dirección y forma de acceso
 - 1.2.1 Control de versiones
 - 1.2.2 Integración continua
 - 1.2.3 Gestión de defectos
- 1.3 Esquema de directorios
- 1.4 Normas de etiquetado y nombramiento de archivos
- 1.5 Esquema de ramas
- 1.6 Políticas de fusión de archivos y etiquetado
- 1.7 Entrega de releases
- 1.8 Change Control Board
- 1.9 Herramienta de seguimiento de defectos

2. Documento de Requerimientos

- 2.1 Introducción
 - 2.1.1 Propósito
 - 2.1.2 Alcance
 - 2.1.3 Definiciones
- 2.2 Descripción general
 - 2.2.1 Características de los usuarios
 - 2.2.2 Restricciones Generales
 - 2.2.3 Supuestos
- 2.3 Lista de Requerimientos
 - 2.3.1 Funcionales
 - 2.3.2 No Funcionales
- 2.4 Casos de uso

3. Casos de Pruebas de Sistema y Matriz de trazabilidad

- 3.1 Introducción
- 3.2 Desarrollo de Tests
- 3.3 Matriz de Trazabilidad

1. Plan de Gestión de las Configuraciones

1.1 Introducción

El Plan de Gestión de las Configuraciones es un conjunto de políticas, procesos y herramientas para administrar los cambios en el Software a desarrollar, lo cual debe ser obligatorio para todos los equipos de trabajo.

Con el avance del proyecto y los cambios que se van realizando, pueden llegar problemas relacionados al seguimiento de los cambios, modificaciones de archivos o versiones incorrectas, perder tiempo realizando tareas ya realizadas por otros compañeros, etc. Con respecto a lo anterior se busca un control de versiones, donde se pueda seguir y controlar las múltiples versiones de cada componente del sistema asegurando que los cambios efectuados por varios desarrolladores no interfieran entre ellos.

Por otro lado, es importante incorporar un mecanismo de seguimiento de los cambios solicitados, tanto por clientes como por desarrolladores, con el fin de evaluar el impacto y el costo de estos mismos. Este proceso debe ser gestionado y apoyado por el Change Control Board, quienes documentan, aprueban o desaprueban los cambios a realizar, como así también medir el costo/beneficio de hacerlos o no.

1.2 Dirección y forma de acceso

1.2.1 Control de versiones

Como herramienta de control de versiones, se utilizará GitHub. Se accede a la misma a través del siguiente url: <https://github.com/>

Se accede con el siguiente usuario/contraseña: (***** / *****) donde se encontrará el repositorio nombrado "*****" (se puede agregar después a qué pestaña en específico se puede ver el control de versiones, o qué parte se visualiza en qué sección).

Una vez que tengamos definido bien el nombre del repositorio por ejemplo, va a ser cuestión de crear una cuenta de correo para poder crear el usuario de GitHub y ver cómo hacer lo de los permisos.

En particular el url del repositorio que se utilizara en este proyecto es: <https://github.com/alejomoyano/poss/>

1.2.2 Integración continua

Como herramienta de integración continua, se utilizará CircleCI. Se accede a la misma a través del siguiente url: <https://circleci.com/>

Se accederá con el mismo usuario y contraseña que se utiliza para acceder a la herramienta de control de versiones y gestión de defectos, GitHub, ya que ambas plataformas se encuentran conectadas.

1.2.3 Gestión de defectos

La herramienta está contenida dentro de la web de control de versiones que elegimos, por ende usaremos las mismas credenciales que en "Dirección y forma de acceso a la

herramienta de control de versiones”. Para la gestión de defectos se accederá a la ventana de “Issues”.

1.3 Esquema de directorios

```
nombre
|
|__ src __ scripts_firebase
|         |__ styles
|
|__ test
|
|__ node_modules
```

src/scripts: archivos .js
src/scripts/firebase: archivos .js de firebase
src/styles: archivos .css

node_modules: módulos de node, que no se exportan al repo.

test: guardamos todo lo que tenga que ver con tests

1.4 Normas de etiquetado y nombramiento de archivos

Los archivos tendrán nombres cortos y en minúsculas. Si se compone por muchas palabras, deberán estar separadas por “_”. No se debe utilizar cualquier otro tipo de carácter.

Los commits deberán ser nombrados siguiendo una estructura como la siguiente: **<descripción>-<dev>-<fecha>**. La descripción debe ser concreta para comprender a grandes rasgos qué cambios se han realizado. El formato de fecha de la forma: dd_mm_yy.

1.5 Esquema de ramas

El esquema de ramas que se va a utilizar en este proyecto es Promotion Branching. Particularmente se compone de una rama Master, que es la branch de release, una rama por Feature (Tareas, Timer, Chat), una rama Develop, donde se realizarán los merges desde las Features y luego una branch por desarrollador.

1.6 Políticas de fusión de archivos y etiquetado

Cada vez que creamos un branch, deberíamos utilizar un tag estilo **<feature><version>** o **<developer>**, una vez que avancemos y realicemos cambios en el código, hay que fijar que determinados test o que porcentaje de los mismos debe pasar satisfactoriamente ese

branch para luego ser mergeado al Develop con un tag **from-<dev>-<feature><version>** y luego a Master. En este caso, es útil tener doble tag, el tag inicial que sería **<feature><versión>** y el tag final que sería **merge-from-<feature><version>-<dev>** (el día puede ser opcional, podemos poner otro tipo de información que nos sirva más, puede ser nuestro nombre para saber y preguntar los cambios por si no se entiende algo). En estos casos, el tag final pasaría a ser el inicial en el próximo merge al Máster si se sigue trabajando en el mismo branch. (Esto parecido al ejemplo CM que está en el grupo PAGE 19).

1.7 Entrega de releases

Los releases serán deployed tomando la versión que se encuentra en la rama Master. Si se realizó algún cambio que amerite notificar al usuario, como alguna funcionalidad, entonces se generará un popup dentro de la página principal que indique dichos cambios.

1.8 Change Control Board

Se debe convocar a una reunión de todo el grupo, donde se debe revisar los distintos cambios y debatir si serán definitivos o se debe cambiar algo. En este caso, todos deberán estar de acuerdo para que los cambios en el código sean definitivos. La frecuencia de las reuniones va de acuerdo a la disponibilidad de los miembros, pero se fijará los días Lunes como obligatorios para planificar la semana. En estas reuniones se deben debatir las modificaciones a futuro para el proyecto, y ver si son posibles o no, teniendo en cuenta el tiempo necesario y la capacidad de llevarlas a cabo correctamente. También se discutirá acerca del estado del producto, bugs a arreglar y demás.

Nombre	CCB Rol	SME
Renaudo, Bruno Agustín	Engineering Manager	Test Expert
Moyano Seminare, Carlos Alejo	Engineering Director	Backend Expert
Ibañez Sala, Ignacio	Release Manager	Design Expert
Etchechoury, Candela	Issues Coordinator	Frontend Expert

1.9 Herramienta de seguimiento de defectos

Para realizar el seguimiento de defectos usaremos la sección de GitHub que tiene como nombre **Issues**, dentro del menú de los repositorios. Para poder ingresar a dicha herramienta hay que dirigimos a la siguiente web: <https://github.com/>.

2. Documento de Requerimientos

2.1 Introducción

En esta sección se introducen las especificaciones de los requerimientos para nuestro proyecto POSS (Pomodoro Study Sessions). Se presentarán el propósito y el alcance del producto además de cualquier definición o referencia necesaria para comprender el desarrollo del mismo.

2.1.1 Propósito

Los requerimientos de este documento tienen como objetivo conformar como un todo nuestra idea. La cual decidimos llamar POSS, abreviatura de Pomodoro Study Session. Como se puede suponer por su nombre, planeamos hacer una aplicación web que responda a la necesidad de trabajar y estudiar en grupo, que se vio reforzada por el contexto en el que nos encontramos a nivel mundial durante estos últimos tres años.

2.1.2 Alcance

Nuestra aplicación debe ayudar al usuario a organizarse para estudiar. Debe facilitar la comunicación del grupo de estudio. Debe fomentar el uso de la técnica de estudio Pomodoro. Debe ayudar al usuario a familiarizarse con esta técnica y hacer que el tiempo de estudio rinda más y provea la menor cantidad de distracciones posible. Por lo que el acceso a la plataforma debe ser rápido, limpio y claro, completamente accesible para cualquier usuario que quiera hacer uso de tal.

2.1.3 Definiciones

User: Usuario normal de la aplicación.

Admin: Usuario con más permisos que el normal. Creador de la sala.

Pomodoro: La técnica Pomodoro es una solución de administración del tiempo desarrollada por un estudiante universitario, Francesco Cirillo, en la década de 1980. Formada por numerosos intervalos de tiempo espaciados a lo largo del día, la técnica recibe su nombre de la palabra italiana para “tomate”, basada en el temporizador de cocina en forma de tomate que Cirillo utilizó al perfeccionar su método de administración del tiempo. La Técnica Pomodoro proporciona una simple herramienta/proceso para mejorar la productividad (tuya o de los miembros de tu equipo) para hacer lo siguiente:

- Alivia la ansiedad vinculada a comenzar.
- Aumenta la concentración y el enfoque disminuyendo las interrupciones.
- Te hace más consciente de tus decisiones.
- Impulsa tu motivación y la mantiene constante.
- Refuerza tu determinación para alcanzar tus metas.
- Refina el proceso de estimación, tanto en términos cualitativos como cuantitativos.
- Mejore su trabajo o proceso de estudio.

Distribución: La técnica se divide en ciclos que están compuestos por 4 subciclos pomodoro, 3 descansos cortos y por último, 1 descanso largo. Los tiempos del pomodoro

clásico son 25 min por subciclo, 5 min los descansos cortos y 15 el largo. Sin embargo, estos se pueden modificar de acuerdo a lo que sea más útil para cada individuo o grupo.

2.2 Descripción General

El proyecto consiste en una aplicación web que provee una herramienta de estudio basada en el método de organización de tiempo Pomodoro y el trabajo y la dinámica de equipo. Es por eso que las principales funcionalidades que debe tener nuestra app es la de proveer un timer totalmente configurable, o con configuraciones predeterminadas para las sesiones pomodoro. El cual podrá ser seteado solo por el administrador de la sala y estará disponible para todos los usuarios miembros de la misma. Otra funcionalidad clave es la capacidad de crear y administrar tareas, como parte de las facilidades para aplicar pomodoro. Por último, como la otra base de nuestro producto va a ser el trabajo en equipo, la comunicación se verá implementada mediante un chat, presente en cada sala de estudio.

2.2.1 Características de los Usuarios

En nuestra aplicación se podrán distinguir dos tipos de usuario: el usuario standard (user) y el usuario admin.

Comenzaremos por describir el **usuario standard** de la app para luego diferenciar al admin, que tendrá los mismos permisos que el standard, más los que se listen luego.

El primer tipo de usuario, contará con acceso ilimitado al chat y a las tareas de la sala. Con respecto al chat, podrá observarlo, enviar mensajes, reaccionar a mensajes y silenciarlo.

Las tareas podrá modificarlas, agregar nuevas, marcar tareas como listas o eliminar tareas.

También podrá observar el timer y recibirá notificaciones cuando concluya o comience algún ciclo pomodoro.

Por otro lado, el **usuario admin** contará con todo lo descripto anteriormente. Se distingue principalmente porque este será el creador de la sala de estudio y además tendrá la capacidad de configurar el timer pomodoro.

Diagrama de casos de uso figura 4.

2.2.2 Restricciones Generales

Limitaciones de Software

Al ser una app web, se requiere conexión a internet para hacer uso de esta.

Por el mismo motivo también es necesario un navegador compatible con la versión 5 de HTML.

Limitaciones de Hardware

Para acceder a la web se requiere el uso de un dispositivo que pueda ejecutar un navegador y que pueda acceder a internet.

2.2.3 Supuestos

Se asume que la gran mayoría de los posibles usuarios tienen en su poder por lo menos una computadora personal o teléfono móvil que le permita acceder a la aplicación web.

También se supone que los usuarios tienen el conocimiento suficiente para manejar dichos dispositivos y por ende saben acceder a una web.

2.3 Lista de Requerimientos

En esta sección se listarán todos los requerimientos de nuestro proyecto. Cada requerimiento tiene un número y nombre únicos y una descripción. En la sección siguiente se desarrollarán las descripciones de los casos de uso.

2.3.1 Funcionales

RF1 - Nombre de Usuario

Descripción: Cada persona, ingrese o cree una sala, deberá registrarse con un nombre que será utilizado solo durante la existencia de esa sala para distinguirlo en el chat.

En la página inicial es necesario que haya un área dedicada al ingreso del nombre de usuario. Se debe asegurar que el texto ingresado no contenga contenido sensible. Si no es aceptable entonces se debe notificar al usuario con un texto que diga “nombre de usuario no disponible”.

RF2 - Creación de una sala de estudio

Descripción: Una persona podrá crear una sala. Automáticamente obtendrá el título de “Admin” en dicha sala.

Caso de uso: figura 1.

En la página principal se debe disponer de un botón “Crear sala” que al accionarlo nos permita, valga la redundancia, crear la sala. Dicho botón debe poderse accionar recién cuando el nombre haya sido verificado y aceptado. En caso contrario no debe figurar o debe estar desactivado.

Se debe poner un nombre para la sala, si el usuario no la especifica se usa uno genérico como “sala1” o lo que sea. Luego se genera el link y el código para la sala.

Habrà un máximo de integrantes por sala (para facilitar el chat).

RF3 - Ingreso a una sala de estudio

Descripción: Luego de ser creada la sala, el usuario creador y tantos como reciban el link o la clave deben poder ingresar a la misma. No va a poder ingresar a una sala si no se obtiene el código o link.

Caso de uso: figura 1.

En la página principal se debe disponer de un botón “Ingresar a una sala” que al accionarlo nos permita ingresar el código de la sala. Dicho botón debe poderse accionar

recién cuando el nombre haya sido verificado y aceptado. En caso contrario no debe figurar o debe estar desactivado.

Si se eligió el ingreso mediante link, se pega el link en el navegador, se accede y luego se elige el nombre de usuario, para ese caso una vez que esté validado el nombre de usuario, se accede directamente a la sala.

RF4 - Creación de Tareas

Descripción: Particularidad del método pomodoro. El usuario debe crear tareas que hagan referencia a lo que se realizará durante los ciclos pomodoro.

Debe haber una sección que sea exclusiva para las tareas. Dentro de dicha sección debe haber un botón para crear tareas que nos genere una en blanco.

Cada tarea no podrá contener más de 256 caracteres.

RF5 - Envío y recepción de mensajes

Descripción: Los usuarios podrán enviar mensajes en el chat que podrán ser observados por los otros usuarios presentes en la sala.

Es necesario que dentro de la sala haya un área dedicada para el chat, el cual consiste de un visor de mensajes (donde se imprimen los mensajes enviados por los integrantes), una sección donde escribir los mensajes y un botón de “enviar”.

Además cada usuario debe estar representado, dentro del chat, con su nombre y con un color único. Dicho color debe ser elegido aleatoriamente por el sistema.

RF6 - Acceso a la configuración del Timer

Descripción: El usuario administrador debe configurar los tiempos de los ciclos y descansos del timer. Se debe hacer obligatoriamente apenas se crea la sala y se puede modificar en cualquier momento, solo por el usuario administrador.

Caso de uso: figura 2.

Debe haber un botón que nos permita acceder a las configuraciones del timer. Dicho botón solo podrá ser presionado sólo por el administrador de la sala, el resto de los usuarios lo verán pero no lo tendrán habilitado.

El botón de configuraciones abrirá una ventanita con todas las cosas necesarias para modificar los ciclos.

RF7 - Cambio de estado de Tareas

Descripción: Los usuarios podrán cambiar el estado de las tareas a terminada, suspendida, pendiente, actual.

Dentro de cada tarea debe encontrarse un **checkbox/combo** que al seleccionarlo cambie el estado de la tarea. Cuando esto ocurra, el color y la fuente deberían cambiar indicando el cambio de estado.

RF8 - Manejo de la Sala

Descripción: Cuando la sala queda con 0 usuarios después de un determinado tiempo, el sistema procederá a eliminarla.

RF9 - Configuración del Timer

Descripción: Luego de acceder al menú de configuración del timer podremos hacer los ajustes deseados.

Dentro del menú de configuración deben figurar varios parámetros modificables: duración del descanso corto, duración del descanso largo y duración de estudio. Cada uno podrá ser cambiado a gusto. Además debe incluir un botón de “Aplicar” para los cambios.

RF10 - Eliminar Tareas

Descripción: Un usuario cualquiera debe poder eliminar tareas. Cada tarea deberá contener un botón que al presionarlo elimine la tarea de la lista de tareas. Cuando una tarea sea eliminada, las otras deben re ordenarse.

RF11 - Edición de Tareas

Descripción: El usuario debe poder cambiar el contenido de una tarea.

Cada tarea debe contar con un botón que le permite editar el contenido de una tarea. Además, al presionar el botón, se desplegará otra opción para poder aplicar los cambios realizados. o podemos poner un botón y al presionarlo la primera vez editamos y cuando le damos de nuevo salimos del modo edición y quedan los cambios guardados.

RF12 - Visualización de la Sala

Descripción: aca deberíamos poner el layout que va a tener la sala.Diseño gráfico, interfaz.Whatever.

RF13 - Modos de la Sala

Descripción: La sala tiene 2 modos, modo estudio y modo descanso. La sala toma el modo estudio cuando el timer se encuentra en un subciclo de estudio. La sala debe modificar su comportamiento. Se minimiza el chat, se silencia y no se pueden enviar

mensajes. Además no se pueden modificar las tareas. Solo observarlas. Ocurren cambios de estética.

El modo de descanso es el que toma durante los descansos (cortos y largos) la sala recupera su comportamiento normal: el chat funciona normalmente y los usuarios tienen acceso completo a la modificación de tareas.

RF14 - Memoria de los Ciclos

Descripción: El sistema debe mantener la cuenta del ciclo y subciclo de Pomodoro en el que se encuentra. Dicha cuenta debe ser accesible a los usuarios de la sala, en el mismo display del timer. Se podrá ver el número de ciclo y el de subciclo.

RF15 - Organización de las Tareas según su Estado

Descripción: Si la tarea pasa a **estado=actual**, su color cambia a **verde** y pasa a estar al principio de la fila. Significa que es la tarea sobre la que se está trabajando en ese momento (a la que le corresponden los current ciclos de pomodoro).

Si la tarea que está en **estado=actual** pasa a **estado=terminada**, cambia a color **azul**(u otro) y pasa a un costado de la lista (o no se lo que quieran); y además la tarea que le sigue en la lista (que tendrá **estado=pendiente**) se actualizará automáticamente a **estado=actual**. El color de las tareas con **estado=pendiente** es **amarillo**.

Cuando a una tarea se le asigna **estado=suspendida (color gris)**, la tarea se mueve al final de la lista, después de todas las que tienen **estado=pendiente**. Esto significa que la tarea se suspende por x motivo pero por alguna razón no se quiere eliminar todavía, capaz hay esperanza de que se pueda hacer o algo.

terminada, suspendida, pendiente, actual.

2.3.2 No funcionales

RNF1 - Tiempo en línea

La aplicación debería mantenerse en pie durante el 99,9 % del día.

RNF2 - Tiempo de creación y entrada de sala

El tiempo máximo que debe tardar la página para crear la sala debe ser de 5 seg y para entrar a la sala de 2 seg.

RNF3 - Máxima ocupación de la sala

La sala tiene un límite máximo de 5 usuarios, una vez hayan entrado dicha cantidad de participantes a una sala el sistema debe impedir la entrada de más.

RNF4 - Accesibilidad por motor de búsqueda

Debe poder ser accedido desde cualquier motor de búsqueda navegador compatible con la versión 5 de HTML.

RNF5 - Simplicidad de acceso y creación de salas

Un usuario promedio sin importar sus conocimientos sobre operación del sitio debería ser capaz de crear una sala en no más de 5 clicks y acceder a una en no más de 3. Para ello la UI de inicio debe ser diseñada con fin de ser intuitiva de tal manera que apoye este requerimiento.

RNF6 - Adaptabilidad del layout

El layout de la página debe ser responsive, para que pueda adaptarse a todos los dispositivos.

RNF7 - Eficiencia

No debemos sobrepasar los límites establecidos por los servicios utilizados para el mantenimiento del proyecto (Firebase, CircleCI, etc)

RNF8 - Encriptación de datos personales

Se deben encriptar los datos personales que son almacenados en la base de datos. Los datos almacenados no deben ser accesibles para nadie.

RNF9 - Sobre el nombre de Usuario

No se deben usar palabras ofensivas como nombre de usuario, así como caracteres especiales.

RNF10 - Uso de palabras ofensivas

No se debe permitir el uso de palabras ofensivas dentro de la aplicación.

RNF11 - Uso de código abierto

Las librerías o fragmentos de códigos utilizados para el desarrollo del software deben ser de código abierto

RNF12 - Uso de datos

Los datos no deben tener ningún otro uso mas que el que propone la aplicación.

RNF13 - Lenguajes y herramientas a utilizar para el desarrollo del programa

El programa debe estar hecho en JavaScript junto con HTML y CSS. Además se deberá hacer uso de herramientas como Firebase para contener una pequeña base de datos.

RNF14 - Idioma de la aplicación web

Todo texto encontrado en la aplicación y disponible a los usuarios deberá estar en idioma inglés.

2.4 Casos de Uso

Caso de uso 1 - C1: Nombre de usuario correcto.

En la página principal de la web el usuario debe ingresar su nombre (alias). En este caso el nombre ingresado cumple con todas las condiciones impuestas, así que el sistema pasa directamente a la interfaz de creación de / entrada a una sala.

Figura 3.

Caso de uso 2 - C2: Nombre de usuario incorrecto.

Una vez en la página principal de la web el usuario debe ingresar un nombre. En este caso el nombre contiene símbolos que no son permitidos por la aplicación entonces se observa un texto que dice “El nombre contiene símbolos no permitidos, por favor ingrese uno diferente”.

Figura 2.

Caso de uso 3 - C3: Creación de Sala exitosa.

Considerando que se cumple el Caso de uso 1, el usuario debe ingresar un nombre para la sala. En caso de ser un nombre aceptable, el botón de crear sala quedará activado permitiendo al usuario presionarlo para luego pasar al siguiente paso.

Figura 2.

Caso de uso 4 - C4: Creación de Sala fallido.

Considerando que se cumple el Caso de uso 1, el usuario debe ingresar un nombre para la sala. Si este nombre no cumple con las condiciones impuestas, entonces se podrá ver un mensaje que diga lo siguiente “El nombre contiene símbolos no permitidos, por favor ingrese uno diferente”.

Figura 2.

Caso de uso 5 - C5: Ingreso a una sala - código correcto.

En la interfaz de creación de / entrada a una sala, se pone la opción para ingresar a una sala (puede ser un botón, una pestaña, no se) y aparece una casilla para poner el código de la sala y se clickea el botón Ingresar. En este caso el código es correcto así que luego de presionar “Ingresar” vamos a pasar a la interfaz de la sala.

Figura 1.

Caso de uso 6 - C6: Ingreso a una sala - código incorrecto.

En la interfaz de creación de / entrada a una sala, se pone la opción para ingresar a una sala (puede ser un botón, una pestaña, no se) y aparece una casilla para poner el código de la sala y se clickea el botón Ingresar. Como el código ingresado será incorrecto, aparecerá un cartel que diga “Código Incorrecto: Sala de estudio inexistente”. El cartel tiene un botón de OK, lo clickeo y se volverá a la vista anterior.

Figura 1.

Caso de uso 7 - C7: Ingreso a una sala - mediante link (vincular con casos 1 y 2).

A la persona le llega el link de una sala de estudio. Lo copia y lo pega en su explorador de preferencia y se abre nuestra web en la interfaz para ingresar un nombre de usuario. Luego de ahí se replica lo que pasa en los casos 1 y 2, pero en vez de abrirse la

vista de creación o entrada a una sala, se abre directamente la sala a la que corresponde ese link. Si la sala expiró porque pasó determinado tiempo desde que se quedó sin usuarios, al abrir el link debe aparecer un cartel que diga “La Sala Expiró” y un botón para volver al inicio.

Figura 1.

Caso de uso 8 - C8: Creación de tareas .

En la sección de Tareas el usuario presiona un botón que generará una tarea nueva en limpio. Luego podrá ingresar el contenido que sea necesario dentro de la tarea. Cuando la tarea se crea toma inicialmente el estado de Pendiente. Si es la primera tarea que se crea, se le asigna automáticamente el estado Actual.

Caso de uso 9 - C9: Edición de tarea.

El usuario aprieta el botón de edición de una tarea, cambia el contenido a gusto y luego vuelve a presionarlo para salir del modo edición.

Caso de uso 10 - C10: Eliminar tarea.

El usuario aprieta el botón de eliminar de la tarea. Aparece un cartel para confirmar la acción. El usuario la acepta y la tarea desaparece.

Caso de uso 11 - C11: Cambio de estado de la tarea.

El usuario aprieta el menú desplegable de estados de la tarea y puede cambiar el estado a cualquiera de los disponibles. En base al que elija, la tarea va a moverse en la lista y cambiar de color.

Caso de uso 12 - C12: Uso del chat.

El usuario siempre podrá ver el chat de la sala (una vez que esté dentro de la misma). Durante los ciclos de descanso podrá enviar nuevos mensajes.

Caso de uso 13 - C13: Acceso a la configuración del timer - Usuario admin

El usuario *admin* podrá acceder a la configuración del timer tocando el botón de configuración del mismo. Este botón abrirá otra ventana, la ventana de configuración. Dicho botón está visible para todos los usuarios.

Caso de uso 14 - C14: Acceso a la configuración del timer - Usuario standard

Si un usuario *standard* toca el botón de configuración, salta una ventana pop up que diga que no está disponible esta funcionalidad porque no es admin.

Caso de uso 15 - C15: Configurar el timer.

Una vez abierta la ventana de configuración, el usuario *admin* podrá setear los tiempos de subciclo, descanso corto y descanso largo.

En la ventana habrá 3 recuadros uno para cada tiempo seteable, con un botón “+” y uno “-” para ir sumando o restando 5 minutos respectivamente.

Caso de uso 16 - C16: Iniciar el timer de ciclo de estudio - para ver los cambios de modo.

Cuando el usuario inicia el timer de un ciclo pomodoro la sala pasa a modo estudio, en este modo se desactivan algunos accesos para promover la concentración del equipo.

Cuando termina el ciclo de estudio, comienza automáticamente el descanso corto, o largo, según corresponda y el modo de la sala pasa a modo descanso.

Caso de uso 17 - C17: Visualización del contador de ciclos y subciclos.

El usuario, dentro de la sección del timer, observará el contador de ciclos y subciclos

3. Casos de Pruebas de Sistema y Matriz de trazabilidad

3.1 Introducción

En esta sección se introducen los Test a realizar para los casos de prueba de sistema. Estos test sirven para identificar posibles defectos al obtener como resultado una salida incorrecta o inesperada, pero no confirma su ausencia.

Por otro lado, la Matriz de Trazabilidad nos ayuda a visualizar fácilmente la relación entre los Requerimientos y los Test antes mencionados, por ende si falla algun test, podremos vincularlo al requerimiento que no se cumple. Esto nos ayuda a asegurarnos que partes críticas o prioritarias del sistema funcionen como se debe o logremos identificar errores de manera temprana.

Por lo antes mencionado, optamos por seguir la estrategia de Testeo Incremental, la cual consiste en realizar los tests correspondientes a cada porción de código, conforme avance el coding del programa.

3.2 Desarrollo de Tests

TEST #01RF1

Descripción del test: Verificar que el input ingresado como 'Nombre de Usuario' no contenga caracteres especiales.

Ejecución del test

1. Entrar al sitio Web
2. Seleccionar el área designada para el ingreso del nombre de usuario
3. Ingresar un nombre de usuario correcto
4. Click en registrar

Resultados esperados: True, se espera una notificación que asegure que el nombre de usuario fue creado.

TEST #02RF1

Descripción del test: Verificar que el input ingresado como 'Nombre de Usuario' no contenga palabras sensibles.

Ejecución del test

1. Ingresar un nombre de usuario con determinada palabra sensible
2. Click en registrar

Resultados esperados: False, se espera una notificación de error donde se explicite que el 'Nombre de usuario no disponible'

TEST #01RF2

Descripción del test: Verificar que el estado del usuario es 'Admin' luego de crear una sala de estudio.

Ejecución del test

1. Registrar un usuario correctamente
2. Crear una sala de estudio
3. Verificar que el usuario pasa a ser Admin

Resultados esperados: True si el status del usuario es Admin

TEST #02RF2

Descripción del test: Verificar el seteo de nombre de la sala

Ejecución del test

1. Clickear en el botón 'Crear sala'
2. Ingresar nombre de la sala
3. Crear sala de estudio

Resultados esperados: True si el usuario ingresa el nombre de la sala, False si no se ingresó ningún valor, setear el mismo por default y notificar al usuario mediante un cartel 'Nombre de la sala no ingresado'.

TEST #01RF3

Descripción del test: Comprobar el ingreso correcto a una sala de estudio mediante código

Ejecución del test

1. Registrar un usuario correctamente
2. Obtener la clave de la sala de estudio
3. Rellenar el campo correspondiente a 'Ingresar a una sala'
4. Clickear en 'Ingresar a una sala'

Resultados esperados: True si se puede ingresar a esa sala, False si no es posible ingresar, se espera una notificación explicitando 'Clave incorrecta o Sala de Estudio inexistente'

TEST #02RF3

Descripción del test: Comprobar el ingreso correcto a una sala de estudio mediante link

Ejecución del test

1. Registrar un usuario correctamente
2. Obtener el link de ingreso a la sala de estudio
3. Pegar dicho link en la barra del navegador
4. Ingresar a la dirección

Resultados esperados: True si se puede ingresar a esa sala, False si no es posible ingresar, se espera una notificación explicitando 'Link incorrecto o Sala de Estudio inexistente'

TEST #01RF4

Descripción del test: Creación de tarea

Ejecución del test

1. Ingresar a una sala de estudio
2. Click en 'Crear Tarea'

Resultados esperados: True, se espera notificación 'Tarea creada exitosamente'

TEST #02RF4

Descripción del test: Verificar que el máximo de la tarea sea de 256 caracteres

Ejecución del test

1. Click en 'Crear Tarea'
2. Tipear 270 caracteres
3. Crear la tarea

Resultados esperados: False, no se crea la tarea y se espera una notificación que explicita 'Máximo de caracteres superado'

TEST #01RF5

Descripción del test: Envío de mensaje

Ejecución del test

1. Clickear en el input del chat para enviar el mensaje
2. Escribir un mensaje
3. Click en 'Enviar'

Resultados esperados: True si se logró enviar el mensaje

TEST #02RF5

Descripción del test: Intento de enviar un mensaje con palabra sensible

Ejecución del test

1. Escribir un mensaje con palabra sensible
2. Click en 'Enviar'

Resultados esperados: False, no se envía el mensaje y se espera una notificación avisando 'Mensaje inapropiado'.

TEST #01RF6

Descripción del test: Verificar acceso a la Configuración del Timer

Ejecución del test

1. Crear un usuario válido
2. Crear una sala
3. Verificar que el status del usuario sea 'Admin'
4. Click en 'Configuraciones'

Resultados esperados: True si se logra acceder a la ventana de configuraciones

TEST #01RF7

Descripción del test: Suspender una tarea

Ejecución del test

1. Click en 'Modificar Tareas'
2. Click en 'Suspender'
3. Click en 'Aplicar'

Resultados esperados: True si el status de la tarea cambió a 'Suspendida'

TEST #02RF7

Descripción del test: Terminar una tarea

Ejecución del test

1. Click en 'Modificar Tareas'
2. Click en 'Terminar'
3. Click en 'Aplicar'

Resultados esperados: True si el status de la tarea cambió a 'Terminada'

TEST #03RF7

Descripción del test: Verificar el estado pendiente de una tarea

Ejecución del test

1. Click en 'Modificar Tareas'
2. Click en 'Pendiente'
3. Click en 'Aplicar'

Resultados esperados: True si el status de la tarea cambió a 'Pendiente'

TEST #04RF7

Descripción del test: Verificar que el estado de la tarea sea actual

Ejecución del test

1. Click en 'Modificar Tareas'
2. Click en 'Actual'
3. Click en 'Aplicar'

Resultados esperados: True si el status de la tarea cambio a 'Actual'

TEST #01RF8

Descripción del test: Eliminación de sala vacía

Ejecución del test

1. Verificar el tiempo desde cuando la sala está vacía
2. Eliminar la sala

Resultados esperados: True si el tiempo es el necesario y se elimina la sala

TEST #01RF9

Descripción del test: Configuración del descanso corto del Timer

Ejecución del test

1. Verificar que el status del usuario es 'Admin'
2. Crear una sala de estudio
3. Click en 'Configuraciones'
4. Setear la duración del descanso corto
5. Click en 'Aplicar'

Resultados esperados: True si el número ingresado es correcto

TEST #02RF9

Descripción del test: Configuración del descanso largo del Timer

Ejecución del test

1. Verificar que el status del usuario es 'Admin'
2. Crear una sala de estudio
3. Click en 'Configuraciones'
4. Setear la duración del descanso largo
5. Click en 'Aplicar'

Resultados esperados: True si el número ingresado es correcto

TEST #03RF9

Descripción del test: Configuración de la duración del tiempo de estudio del Timer

Ejecución del test

1. Verificar que el status del usuario es 'Admin'
2. Crear una sala de estudio
3. Click en 'Configuraciones'
4. Setear la duración del tiempo de estudio
5. Click en 'Aplicar'

Resultados esperados: True si el número ingresado es correcto

TEST #01RF10

Descripción del test: Eliminar tareas

Ejecución del test

1. Crear una tarea
2. Click en 'Eliminar'

Resultados esperados: True si se elimina la tarea

TEST #01RF11

Descripción del test: Edición de tareas

Ejecución del test

1. Click en 'Crear Tarea'
2. Tipear "Esto se debería editar correctamente"
3. Crear tarea
4. Click en 'Modificar'
5. Borrar "correctamente"
6. Click en 'Aplicar'

Resultados esperados: True si el string final es "Esto se debería editar"

TEST #01RF13

Descripción del test: Modificar tareas en tiempo de estudio

Ejecución del test

1. Crear una sala
2. Inicializar el timer
3. Click en 'Modificar'
4. Modificar el contenido
5. Click en 'Aplicar'

Resultados esperados: False y notificar 'La tarea no puede ser modificada en tiempo de estudio'

TEST #01RF14

Descripción del test: Verificar el contador de ciclos del programa

Ejecución del test

1. Inicializar el programa
2. Verificar que la variable 'ciclos' se incrementó

Resultados esperados: True si la variable tiene el valor de '1'

TEST #01RF15

Descripción del test: Verificar el 'color verde' de la tarea actual

Ejecución del test

1. Click en 'Modificar'
2. Click en 'Actual'
3. Click en 'Aplicar'
4. Verificar que la variable 'color' está seteado en 'verde'
5. Verificar que la tarea es la primera del stack

Resultados esperados: True y notificar 'Actualización exitosa'

TEST #02RF15

Descripción del test: Verificar el 'color azul' de la tarea terminada

Ejecución del test

1. Click en 'Modificar'
2. Click en 'Terminada'
3. Click en 'Aplicar'
4. Verificar que la variable 'color' está seteado en 'azul'

Resultados esperados: True y notificar 'Actualización exitosa'

TEST #03RF15

Descripción del test: Verificar el 'color amarillo' de la tarea pendiente

Ejecución del test

1. Click en 'Modificar'
2. Click en 'Pendiente'
3. Click en 'Aplicar'
4. Verificar si la variable 'color' está seteado en 'amarillo'

Resultados esperados: True y notificar 'Actualización exitosa'

TEST #04RF15

Descripción del test: Verificar el 'color gris' de la tarea suspendida

Ejecución del test

1. Click en 'Modificar'
2. Click en 'Suspendida'
3. Click en 'Aplicar'
4. Verificar si la variable 'color' está seteado en 'gris'
5. Verificar que la tarea es la última del stack

Resultados esperados: True y notificar 'Actualización exitosa'

3.3 Matriz de Trazabilidad:

Id de Requerimientos	Titulo	Id de Test	Numero de Caso de uso
RF1	Nombre de Usuario	T#01RF1	C1
		T#02RF1	C2
RF2	Creación de una sala de estudio	T#01RF2	C3
		T#02RF2	C4
RF3	Ingreso a una sala de estudio	T#01RF3	C5
		T#02RF3	C6
			C7
RF4	Creación de Tareas	T#01RF4	C8
		T#02RF4	
RF5	Envío y recepción de mensajes	T#01RF5	C12
		T#02RF5	
RF6	Acceso a la configuración del Timer	T#01RF6	C13
			C14
RF7	Cambio de estado de Tareas	T#01RF7	C11
		T#02RF7	
		T#03RF7	
		T#04RF7	
RF8	Manejo de la Sala	T#01RF8	
RF9	Configuración del Timer	T#01RF9	C15
		T#02RF9	
		T#03RF9	
RF10	Eliminar Tareas	T#01RF10	C10
RF11	Edición de Tareas	T#01RF11	C9
RF12	Visualización de la Sala		
RF13	Modos de la Sala	T#01RF13	C16
RF14	Memoria de los Ciclos	T#01RF14	C17
RF15	Organización de las Tareas según su Estado	T#01RF15	C11
		T#02RF15	
		T#03RF15	
		T#04RF15	

Figura 1.

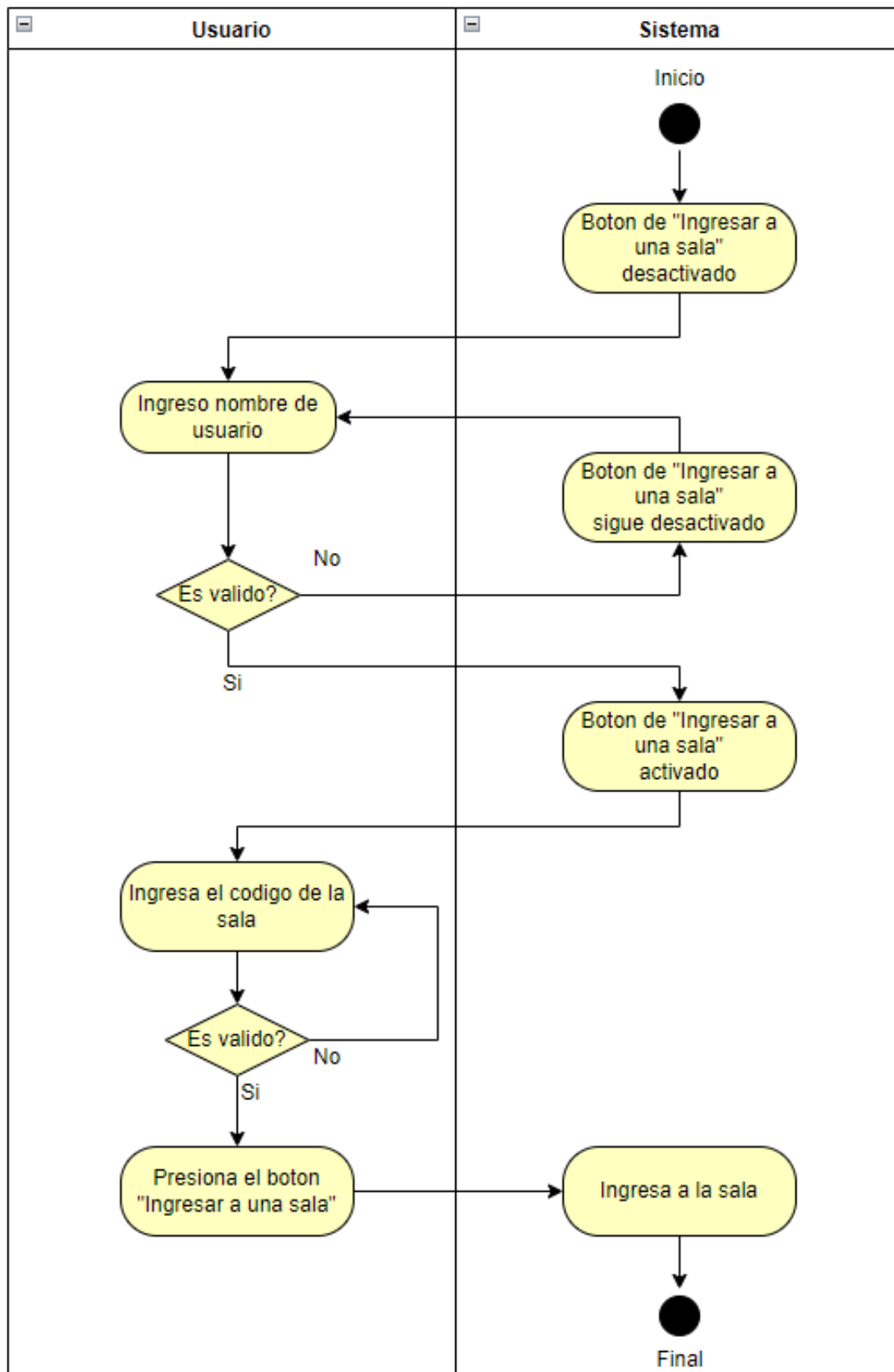


Figura 2.

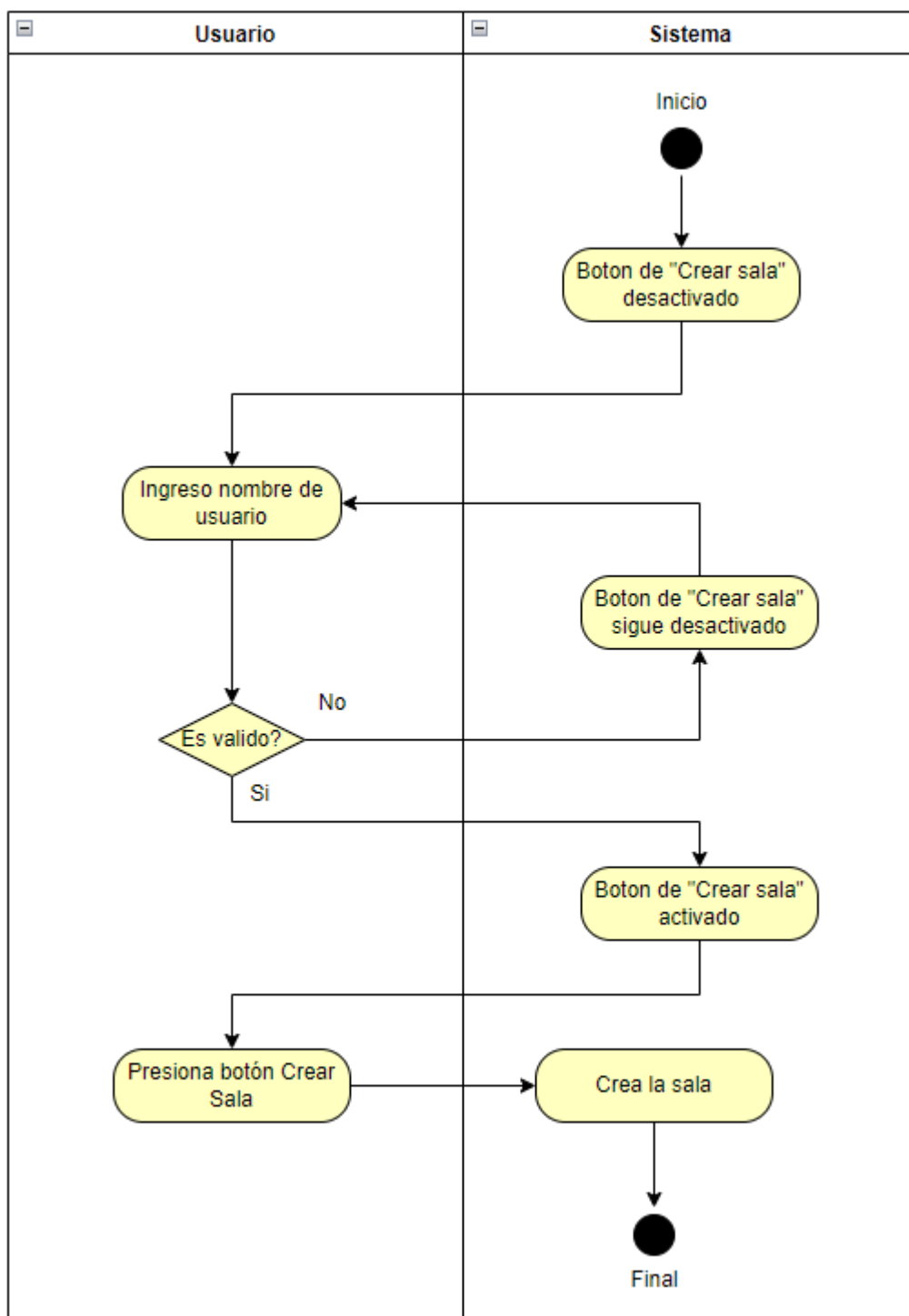


Figura 3.

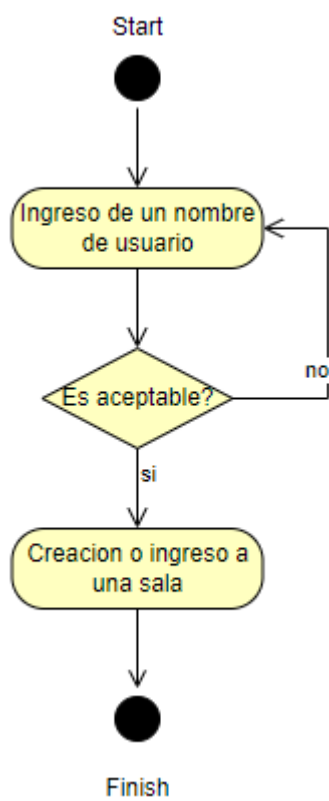


Figura 4. Se puede distinguir la diferencia entre los dos tipos de usuarios.

