

Universidad ORT Uruguay
Facultad de Ingeniería
Escuela de Tecnología

OBLIGATORIO PROGRAMACION 1

APLICACION CENSO



[Anapaula Sardiña – 292575]



[Gian Franco Gatto - 248529]

[N1F]

Docente: [Mauro Nascimento]

[ATI/AP]

[Fecha de entrega del documento (22-06-2023)]

Índice

1. Resumen	3
2. Tipos de Usuarios	3
3. Listado de las funcionalidades desarrolladas	3
4. Informe testing	13
5. Código html/js.....	25
6. Lista de información precargada.....	56

1. Resumen

Nuestro enfoque será la gestión eficiente del proceso de censo de ciudadanos, crucial para obtener información precisa y actualizada sobre la población.

Dicho proceso presenta desafíos complejos. El objetivo es abordar estos desafíos desarrollando un sistema que facilite la recopilación, verificación y organización de los datos de los ciudadanos.

El sistema -desarrollado en HTML5/JavaScript- se basa en una aplicación con dos tipos de usuarios (invitado y censista) con sus respectivas funcionalidades. El objetivo principal consta de almacenar datos de personas censadas en el sistema. Esta carga de datos puede ser modificada por el censista, agregando, editando y eliminando datos de forma dinámica, permitiendo así también su visualización de manera estadística en forma de tabla. A su vez estos datos pueden ser recopilados en forma de “pendiente a validar” por el usuario invitado, en espera de validación por el censista. Cabe aclarar que el invitado también puede acceder a una tabla de información de la lista de censados donde se mostrara un reporte con todos los departamentos y la cantidad de usuarios que estudian, trabajan y no trabajan, con su porcentaje con respecto al departamento.

Tipos de usuario del sistema

- Perfil Censista
- Perfil Invitado

Listado de funcionalidades desarrolladas

A continuación, se presenta el detalle de cada una de las funcionalidades de la aplicación.

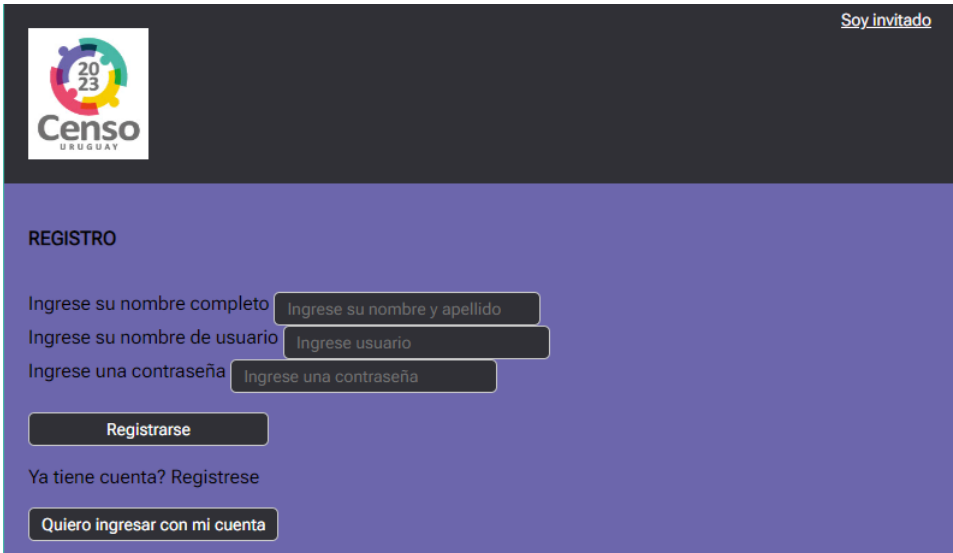
F01 – Registro en la aplicación.

Esta funcionalidad es únicamente del usuario Censista y consiste en registrar un usuario censista nuevo el cual deberá ingresar un nombre completo en el respectivo input, un nombre de usuario, el cual no debe de existir ya previamente, <en caso de ya existir el nombre de usuario se le informara con un mensaje de error: “EL NOMBRE DE USUARIO YA EXISTE”> y una contraseña la cual debe contener un mínimo de 5 caracteres, contando con una mayúscula, una minúscula y un número, en caso de que no se cumplan estos requisitos se le informara al usuario con un mensaje de error: “SU

CONTRASEÑA DEBE TENER 5 CARACTERES MINIMO Y CONTAR CON UNA MAYUSCULA Y UNA MINUSCULA”.

Estos dos campos deben contener información, de caso contrario se le informara al usuario con un mensaje de error: “LOS CAMPOS NO PUEDEN ESTAR VACIOS”. Todos los datos anteriores son obligatorios y en caso de haberlos cumplido se agregará ese censista al array de censistas.

El usuario puede saltarse esta etapa en caso de ya haberse registrado anteriormente clickeando en el botón “Quiero ingresar con mi cuenta”. El usuario puede volver al registro haciendo click en el boton “Me quiero registrar”.

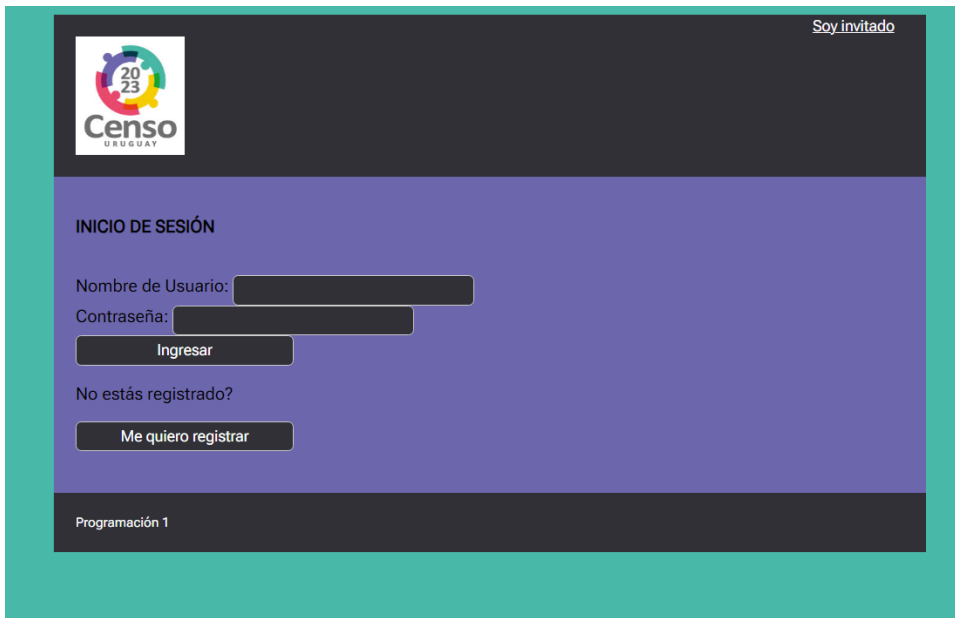
The image shows a web form for registration on the 'Censo Uruguay' website. At the top left is the logo for 'Censo 2023 URUGUAY'. At the top right is a link that says 'Soy invitado'. The main heading is 'REGISTRO'. Below this, there are three input fields with placeholder text: 'Ingrese su nombre completo' (with a sub-placeholder 'Ingrese su nombre y apellido'), 'Ingrese su nombre de usuario' (with a sub-placeholder 'Ingrese usuario'), and 'Ingrese una contraseña' (with a sub-placeholder 'Ingrese una contraseña'). Below the first two fields is a 'Registrarse' button. Below the password field is a link 'Ya tiene cuenta? Regístrese'. At the bottom is a button 'Quiero ingresar con mi cuenta'.

F02 – Ingreso a la aplicación.

El ingreso en la aplicación es una funcionalidad propia del usuario Censista y consiste en dos inputs, en primer lugar, se le pide al usuario su nombre de usuario previamente registrado, o uno de los usuarios precargados como censista. El segundo input pide una contraseña la cual debe cumplir con los mismos requisitos que se le pedía en el registro (un mínimo de 5 caracteres, contando con una mayúscula, una minúscula y un número), en caso de que no se cumplan estos requisitos se le informara al usuario con un mensaje de error: “SU CONTRASEÑA DEBE TENER 5 CARACTERES MINIMO Y CONTAR CON UNA MAYUSCULA Y UNA MINUSCULA”.

Estos dos campos deben contener información, de caso contrario se le informara al usuario con un mensaje de error: “LOS CAMPOS NO PUEDEN ESTAR VACIOS”. Si todos los datos fueron registrados o existen en la precarga se le permitirá al usuario el ingreso a la aplicación.

Si bien esta es una funcionalidad propia del censista, el usuario invitado tendra acceso parcial a la aplicación con un uso limitado de las funcionalidades. Este puede ingresar clickeando en el botón “Soy invitado”, este desplegara un mensaje de “Bienvenido :)” y se le mostrara en el navbar principal 4 funcionalidades, sujetas a su caracteristica de invitado: “Realizar Censo”, “Consulta censo pre-completado”, “Visualizar lista de censados” y “Cerrar Sesion”.



F03 – Ingresar datos de una persona.

Esta es la funcionalidad principal de la aplicación (es accesible tanto para el usuario censista como para el invitado) y consta de la recopilación de datos para agregar a una persona censada, en esta parte se le pide al usuario un “Primer nombre” y un “Primer Apellido”, luego se le pide una “Edad” la cual debe ser un valor numérico y debe estar comprendida entre 0 y 130, en caso de que no se cumpla este requisito se le informara al usuario con un mensaje de error: “EDAD INVALIDA”. Luego se le pide la “C.I”, la cual debe ser ingresada con puntos y guiones, y debe ser una cedula valida uruguaya, en caso

de que el usuario no ingrese una cedula valida o no ingrese los puntos y guiones se le informara al usuario con un mensaje de error: “CEDULA INCORRECTA, VERIFIQUE DATOS”. Después se le pide al usuario un “Departamento de residencia” y una ocupación en forma de select las cuales pasan por una validación que comprueba que el usuario no haya dejado en “blanco” esos selects.

Todos los datos son obligatorios y deben contener información, en caso contrario se le informara al usuario con un mensaje de error: “LOS CAMPOS NO PUEDEN ESTAR VACIOS”

Esta funcionalidad tiene una característica particular, si el usuario que ingresa la información es un invitado, la carga de datos al sistema queda en “pendiente a validar” con la propiedad “censado” en falso, para luego ser validada por el censista en la funcionalidad siguiente, (“Consulta de censos pre-completados”). En caso de que la carga de información sea por un usuario censista esta queda validada con su propiedad “censado” en verdadero y no necesita validacion siguiente.

Perfil Censista

The screenshot shows the 'Realizar Censo' (Perform Census) section of the Censo Uruguay web application. The interface is dark-themed with a teal border. At the top, there's a navigation bar with the Censo Uruguay logo and several menu items: 'Realizar Censo', 'Consulta censo', 'Reasignar persona para validar', 'Visualizar información estadística', and 'Cerrar sesión'. A welcome message 'Bienvenido Juan!' is displayed on the right. The main content area is titled 'INGRESA DATOS PARA EL CENSO' and contains a form with the following fields: 'Primer nombre' (with placeholder 'Nombre y apellido de la persona'), 'Primer Apellido' (with placeholder 'Nombre y apellido de la persona'), 'Edad' (with placeholder 'Edad debe ser ingresado en núm'), 'C.I.' (with placeholder 'Cédula de identidad de la persor'), 'Departamento de residencia' (a dropdown menu with 'Seleccione una opción...'), and 'Ocupación' (a dropdown menu with 'Seleccione...'). A 'Censar persona' button is located below the form. At the bottom left, the text 'Programación 1' is visible.

Perfil Invitado

Realizar Censo Consulta censo

Visualizar lista de censados Cerrar sesión

Bienvenido invitado :)

INGRESA DATOS PARA EL CENSO

Primer nombre:

Primer Apellido:

Edad:

C.I.:

Departamento de residencia:

Ocupación

Programación 1

F04 – Consulta de censos pre-completados / Editar y Eliminar Datos

Esta funcionalidad es accesible tanto para el usuario invitado como para el censista con ciertos cambios sujetos a su calidad de usuario.

En primer lugar, se le pide al usuario que ingrese una cedula de identidad ya censada previamente con un botón “Consultar” que busca por esa cedula si existe una persona precargada a validar o ya validada en la lista de personas censadas. En caso de que la persona ya haya sido validada se le informara al usuario con un mensaje: “EL CENSO YA FUE VALIDADO”

En caso de que la propiedad de la cedula (persona) buscada sea falso, se le cargaran todos los datos de la persona en los inputs continuos, los cuales se encuentra deshabilitados hasta que se busque una persona pendiente a validar, tanto para el usuario censista como para el invitado.

Esos cambios dependiendo de que tipo de usuario este ingresado recae en su funcionalidad, el usuario censista podrá únicamente validar el censo “clickeando” en el único botón por pantalla “Validar Censo”, a su vez el usuario censista no podrá acceder

a la información cargada en los inputs ya que estos están deshabilitados. Mientras que el usuario invitado podrá editar la información de la persona o eliminar directamente sus datos dándole click en sus respectivos botones: “Editar Censo” o “Eliminar Censo”

Perfil Censista

The screenshot displays the 'Censo Uruguay' web application interface. At the top left is the logo for the 2023 Censo Uruguay. The top navigation bar includes links for 'Realizar Censo', 'Consulta censo', 'Reasignar persona para validar', 'Visualizar información estadística', and 'Cerrar sesión'. A user greeting 'Bienvenido Juan!' is visible on the right. The main section, titled 'CONSULTA DE CENSOS PRE COMPLETADOS', contains a form for searching pre-completed censuses. The form includes a 'Cédula' field with a placeholder 'Cédula de persona a validar' and a 'Consultar' button. Below this are fields for 'Primer nombre' and 'Primer Apellido', both with placeholder text 'Nombre y apellido de la persona'. The 'Edad' field has a placeholder 'Edad debe ser ingresado en núm.'. The 'Departamento de residencia' field is a dropdown menu with the option 'Seleccione una opción...'. The 'Ocupación' field is another dropdown menu with the option 'Seleccione...'. A 'Validar censo' button is located at the bottom of the form. The footer of the page reads 'Programación 1'.

Perfil Invitado

Realizar Censo Consulta censo

Visualizar lista de censados Cerrar sesión

Bienvenido invitado :)

CONSULTA DE CENSOS PRE COMPLETADOS

Cédula:

Primer nombre:

Primer Apellido:

Edad:

Departamento de residencia:

Ocupación

Programación 1

F05 – Reasignar personas.

Esta sección es propia del usuario censista y no puede ser accedida por el invitado, y consta de dos selects. El primero de “Persona pendiente a validar” el cual incluye todas las personas que estén pendientes a validacion de su censo, es decir, que su propiedad censado se encuentre en “false”. El segundo es de los censistas cargados en el sistema, menos el que ya está logueado. La funcionalidad consta en elegir una persona pendiente a validar y un censista para reasignarle el id de ese censista.

Todos los datos son obligatorios y en caso de dejar un select por defecto en “Seleccione” se le informara al usuario por un mensaje de error “LOS CAMPOS NO PUEDEN ESTAR VACIOS”.

En caso de que la reasignación sea exitosa se le informara al usuario con un mensaje “PERSONA REASIGNADA CON EXITO”.

Censo
URUGUAY

Realizar Censo Consulta censo

Reasignar persona para validar

Visualizar información estadística Cerrar sesión

Bienvenido Juan!

Persona pendiente a validar: Seleccione una persona... ▾

Censista: Seleccione un censista... ▾

Re-asignar persona

Programación 1

F06 – Mostrar información estadística. – Censista.

Esta funcionalidad propia del usuario censista permite visualizar la información estadística de cada departamento mostrando en forma de tabla la cantidad de personas censadas en ese departamento y al lado en forma de porcentaje la cantidad de personas mayores de edad y menores de edad. El departamento es obligatorio y en caso de no estar completado se le informara al usuario con un mensaje de error: “SELECCIONE UN DEPARTAMENTO”

[Realizar Censo](#)[Consulta censo](#)[Reasignar persona para validar](#)[Visualizar información estadística](#)[Cerrar sesión](#)

Bienvenido Juan!

TOTAL DE PERSONAS CENSADAS	PORCENTAJE DE CENSOS PENDIENTES A VALIDAR:
10	70%

ELIJA UN DEPARTAMENTO PARA MOSTRAR INFORMACIÓN

Seleccione una opción... ▾

MOSTRAR

SELECCIONE UN DEPARTAMENTO

CENSOS POR DEPARTAMENTO:	MAYORES DE EDAD/ MENORES DE EDAD
TACUAREMBÓ: 3	0.0% / 100.0%

Programación 1

F09 – Visualizar lista de censados

En forma de tabla se le muestra al usuario invitado los diferentes datos de las personas censadas con respecto a su departamento, listando los departamentos, las personas censadas que estudian, que trabajan (dependientes e independientes) y que no trabajan.

Esta sección no requiere validación ni presenta posibles errores o fallas.

DEPARTAMENTO	ESTUDIAN	NO TRABAJAN	DEPENDIENTES O INDEPENDIENTES	TOTAL DE CENSOS
Artigas	1	0	1	6%
Canelones	0	1	4	15%
Cerro Largo	0	0	1	3%
Colonia	2	0	1	9%
Durazno	0	0	0	0%
Flores	0	0	0	0%
Florida	1	0	2	9%
Lavalleja	0	0	0	0%
Maldonado	0	2	1	9%
Montevideo	0	0	2	6%
Paysandú	1	0	0	3%
Río Negro	0	0	0	0%
Rivera	0	0	0	0%
Rocha	0	2	1	9%
Salto	0	0	0	0%
San José	2	0	2	12%
Soriano	0	0	0	0%
Tacuarembó	0	0	3	9%
Treinta y Tres	2	0	1	9%

Programación 1

4. Informe de testing

CASOS DE PRUEBA PARA FUNCIONALIDADES ASOCIADAS AL USUARIO CENSISTA:

F-01: REGISTRO EN LA APLICACIÓN

Precondiciones: -Para crear una cuenta en el sitio debe ingresar su nombre, crear un nombre de usuario y una contraseña (esta debe cumplir con mínimo 5 caracteres e incluir al menos una mayúscula, una minúscula y un número).

ID	Escenario de test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado (F=Falla, P=pasa)
F01-T01	Nombre de usuario repetido	1.Llena el campo de "nombre" Intenta crearse un nombre de usuario que ya fue registrado anteriormente 2.Crea una clave correcta	"juan123" Input string	Aviso de que ya existe ese nombre de usuario	Manda el aviso: "EL NOMBRE DE USUARIO YA EXISTE"	P
F01-T02	Clave incorrecta	1. Llena el campo de "nombre" 2. Ingresa un nombre de usuario único 3. Crea una contraseña que no cumple con los requerimientos	"contraseña" Input number	Aviso de que la contraseña es invalida	Manda el aviso: "SU CONTRASEÑA DEBE TENER 5 CARACTERES MINIMO Y CONTAR CON UNA MAYUSCULA, UNA MINUSCULA Y UN NUMERO"	P
F01-T03	Deja un campo vacío	1. Llena dos de los tres campos	"juan" ""	Aviso de que los campos	Manda el aviso "LOS CAMPOS NO	P

		requeridos para registrarse	"123Uno"	no pueden estar vacíos.	PUEDEN ESTAR VACIOS	
		2. Aprieta el botón "registrarse"			"	
F01-T02	Llena todos los campos con datos correctos	1. Llena el campo de "nombre" 2. Llena el campo de nombre de usuario con uno único 3. Crea una contraseña que cumple con los requisitos	Nombre, nombre de usuario, contraseña	Ingresar de manera exitosa	Ingresar de manera exitosa	P

F-02:**INICIAR****SESIÓN****Precondiciones:** - El usuario ya creó una cuenta en el sitio con su nombre de usuario y contraseña.

ID	Escenario de test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado (F=Falla, P=pasa)
F02-T01	Clave incorrecta	Ingresar su nombre de usuario correctamente Ingresar una clave incorrecta.	Clave correcta: "A1aaaa" Clave utilizada: "a1aaaa"	Aviso de que la contraseña es incorrecta	Manda el aviso: "EL NOMBRE DE USUARIO O CONTRASEÑA NO COINCIDEN."	P
F02-T02	Nombre de usuario incorrecto	1. Ingresar un nombre de usuario incorrecto. 2. Ingresar su clave correctamente	Nombre de usuario: "juan123" Nombre de usuario utilizado: "juan"	Aviso de que el nombre de usuario es incorrecto	Aviso: "EL NOMBRE DE USUARIO O CONTRASEÑA NO COINCIDE"	P
F02-T04	Deja campos vacíos	1. Dejar alguno de los campos requeridos vacíos.	Nombre de usuario, contraseña	Aviso de que los campos no pueden estar vacíos	Manda el aviso "LOS CAMPOS NO PUEDEN ESTAR VACIOS" "	P

F02-T05	Se registra con un nombre de usuario en minúsculas e inicia sesión con su nombre de usuario en mayúsculas	1. Ingresar su nombre de usuario correctamente, pero en mayúsculas 2. Ingresar una clave correcta	Nombre de usuario: "JUAN123"	Ingresar de manera exitosa	Ingresar de manera exitosa	P
---------	---	--	------------------------------	----------------------------	----------------------------	---

F-03: Realizar Censo como censista

Precondiciones: - El usuario ya ingreso a la aplicación como censista.

ID	Escenario de test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado (F=Falla, P=pasa)
F02-T01	Dejar campos vacíos	No llena todos los datos requeridos para censar una persona nueva	Nombre: Juan Apellido: Edad: 19 C.I: Departamento: Ocupación:	Aviso de que los campos no pueden estar vacíos.	Aviso de que los campos están vacíos: "LOS CAMPOS NO PUEDEN ESTAR VACIOS"	P
F02-T02	Edad invalida	1. Ingresa una edad menor a 0 o mayor a 130 o 2. No ingresa un número	Edad: "ciento diez" Input number	Aviso de que la edad es inválida.	Aviso de que la edad es invalida: "EDAD INVALIDA".	P
F02-T03	Cedula Invalida	1. Ingresa una cedula sin puntos ni guiones o 2. Ingresa una cedula inventada.	"8.365.247-2" (inventada) Input string	Aviso de que la cedula es invalida.	Aviso de que la cedula es invalida: "CEDULA INVALIDA, VERIFIQUE DATOS"	P

F02-T05	Llena todos los campos correctamente	Llena todos los campos requeridos para censar una persona nueva con datos correctos.	<p>Nombre: Paula</p> <p>Apellido: Sardiña</p> <p>Edad: 19</p> <p>Cedula: "6.342.754-2"</p> <p>Departamento: "Montevideo"</p> <p>Ocupación: "Dependiente"</p>	<p>Se realiza el censo correctamente.</p> <p>Avisa que fue exitoso e internamente coloca la propiedad de la persona en true, ya que el censo fue realizado por un censista.</p>	Aviso que fue censado de manera exitosa y el censo queda validado.	P
---------	--------------------------------------	--	--	---	--	---

F-04: Consulta censo pre-completado como censista

Precondiciones: -- El usuario ya censista ya ingreso a la aplicación.

ID	Escenario de test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado (F=Falla, P=pasa)
F02-T01	Campo de la cedula vacío	Deja el campo de la cedula para buscar vacío	Cédula para buscar: ""	Aviso de que el campo está vacío.	Aviso de que el campo está vacío: "EL CAMPO NO PUEDE ESTAR VACIO"	P
F02-T02	Cedula Invalida	1. Ingresar una cedula sin puntos ni guiones	Cedula: "63427542"	Aviso de que la cedula es invalida: "DATOS NO ENCONTRADOS O CEDULA INVALIDA"	Aviso de que la cedula es invalida: "DATOS NO ENCONTRADOS O CEDULA INVALIDA"	P
F02-T03	Cedula inexistente en el sitio web	Ingresar una cedula de una persona que no ha sido censada anteriormente	Cedula: "5.278.269-4"	Aviso de que la cedula no fue encontrada.	Aviso de que la cedula no fue encontrada: "DATOS NO ENCONTRADOS O CEDULA INVALIDA"	P
F02-T04	Cedula de censo ya validado	Ingresar una cedula de una persona cuyo censo ya fue validado	Cedula: "6.342.754-2"	Aviso de que el censo ya fue validado.	Aviso de que el censo ya fue validado: "EL CENSO YA FUE VALIDADO"	P
F02-T05	Realiza la búsqueda de una persona que realizó su censo y éste está pendiente a validar.	1. Llena el campo de cédula a buscar. 2. Todos los datos recuperados son correctos así que da click al botón de validar censo.	Cedula: "5.132.213-6"	Recupera los datos de la persona buscada por su cédula, si está pendiente a validar, puede validar el censo de forma correcta.	Recupera los datos de la persona buscada por su cédula, si está pendiente a validar, puede validar el censo de forma correcta.	P

				Internamente cambia su propiedad censado de false a true.	Internamente cambia su propiedad censado de false a true.	
--	--	--	--	---	---	--

F-05: Reasignar persona para validar

Precondiciones: - El usuario ya censista ya ingreso a la aplicación.

ID	Escenario de test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado (F=Falla, P=pasa)
F02-T01	Persona pendiente a validar vacío o censista vacío	1. Deja la opción por defecto "Seleccione" en alguno, o los dos selects.	Censista: "Seleccione"	Aviso de que los campos no pueden estar vacíos.	Aviso de que los campos no pueden estar vacíos: "LOS CAMPOS NO PUEDEN ESTAR VACIOS".	P
F02-T02	Re-asigna a una persona a otro censista	1. Selecciona una persona pendiente a validar que tiene ese censista asignado 2. Selecciona a otro censista 3. Apreta el botón "reasignar"	Persona pendiente a validar: "Juan" Censista: "Carlos"	Avisa de que la persona fue reasignada y ésta desaparece de su lista de personas a validar, ya que no le pertenece al censista ingresado.	Avisa de que la persona fue reasignada y ésta desaparece de su lista de personas a validar, ya que no le pertenece al censista ingresado.	P

F-06: Visualizar información estadística

Precondiciones: - El usuario ya censista ya ingreso a la aplicación.

ID	Escenario de test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado (F=Falla, P=pasa)
F02-T01	Elija un departamento para mostrar información: vacío	1. Deja el campo de departamento vacío 2. Da click al botón "Mostrar"	Select: "Seleccione una opción"	Aviso de que el campo está vacío.	Aviso de que el campo está vacío: "ELIJA UN DEPARTAMENTO"	P

F02-T01	Elige un departamento para mostrar información	1. Elige un departamento. 2. Da click al botón "Mostrar"	Select: "Montevideo"	Muestra una tabla que carga la cantidad de censos realizados en ese departamento junto con el porcentaje de personas censadas mayores de edad y menores de edad.	Muestra una tabla que carga la cantidad de censos realizados en ese departamento junto con el porcentaje de personas censadas mayores de edad y menores de edad.	P
---------	--	---	-------------------------	--	--	---

CASOS DE PRUEBA PARA FUNCIONALIDADES ASOCIADAS AL USUARIO INVITADO:

F-01: Realizar Censo como invitado

Precondiciones: - El usuario ya ingreso a la aplicación como invitado.

ID	Escenario de test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado (F=Falla, P=pasa)
F02-T01	Dejar campos vacíos	1. No llena todos los campos requeridos para realizar un censo.	Primer nombre: "Paula" Primer apellido: "Sardiñha" Edad: Cedula: Departamento: "Montevideo" Ocupación: "estudiante"	Aviso de que los campos no pueden estar vacíos.	Aviso de que los campos no pueden estar vacíos: "LOS CAMPOS NO PUEDEN ESTAR VACIOS"	P
F02-T02	Edad invalida	1. Ingresar una edad menor a 0 o mayor a 130 2. No ingresar un numero	Primer nombre: "Paula" Primer apellido: "Sardiñha" Edad: 400 Cedula: "6.342.754-2" Departamento: "Montevideo" Ocupación: "estudiante"	Aviso de que la edad es invalida.	Aviso de que la edad es invalida: "EDAD INVALIDA".	P
F02-T03	Cedula Invalida	1. Ingresar una cedula sin	Cedula: 63427542	Aviso de que la	Aviso de que la cedula es invalida:	P

		puntos ni guiones 2. Ingresar una cedula inventada		cedula es invalida.	“CEDULA INVALIDA, VERIFIQUE DATOS”	
--	--	--	--	------------------------	---	--

F03- T04	Completa todos los datos de forma correcta	1. Ingresar cada uno de los datos requeridos de forma correcta para realizar el censo.	Primer nombre: “Paula” Primer apellido: “Sardiña” Edad: 19 Cedula: “6.342.754-2” Departamento: “Montevideo” Ocupación: “estudiante”	Aviso de que el censo fue exitoso e internamente el censo queda pendiente a validar, ya que no fue realizado por un censista.	Aviso de que el censo fue exitoso e internamente el censo queda pendiente a validar, ya que no fue realizado por un censista.	P
-------------	--	---	---	--	--	---

F-02: Consulta censo pre-completado como invitado

Precondiciones: -- El usuario ya ingreso a la aplicación como invitado

ID	Escenario de test	Pasos	Datos utilizados	Resultado esperado	Resultado obtenido	Estado (F=Falla, P=pasa)
F02-T01	Campo de la cedula vacío	Deja el campo de la cedula para buscar vacío	Cédula para buscar: ""	Aviso de que el campo está vacío.	Aviso de que el campo está vacío: "EL CAMPO NO PUEDE ESTAR VACIO"	P
F02-T02	Cedula Invalida	1. Ingresar una cedula sin puntos ni guiones	Cedula: "63427542"	Aviso de que la cedula no fue encontrada y da la opción a la persona a que complete sus datos de censo habilitando un botón "ir a realizar censo".	Aviso de que la cedula no fue encontrada y da la opción a la persona a que complete sus datos de censo habilitando un botón "ir a realizar censo"	P
F02-T03	Cedula inexistente en el sitio web	Ingresar una cedula de una persona que no ha sido censada anteriormente	Cedula: "5.278.269-4"	Aviso de que la cedula no fue encontrada y da la opción a la persona a que complete sus datos de censo habilitando un botón "ir a realizar censo".	Aviso de que la cedula no fue encontrada: "DATOS NO ENCONTRADOS O CEDULA INVALIDA" y da la opción a la persona a que complete sus datos de censo habilitando un botón "ir a realizar censo".	P
F02-T04	Cedula de censo ya validado	Ingresar una cedula de una persona cuyo censo ya fue validado	Cedula: "6.342.754-2"	Aviso de que el censo ya fue validado.	Aviso de que el censo ya fue validado: "EL CENSO YA FUE VALIDADO"	P

F02-T05	Realiza la búsqueda de una persona que realizó su censo y éste está pendiente a validar.	<p>1. Llena el campo de cédula a buscar.</p> <p>2. Se recuperan los datos de la persona a la que pertenece la cedula buscada y se le habilitan los botones “editar censo” que le permite editar datos anteriormente cargados, y “eliminar datos” que le permite a la persona eliminar todos sus datos.</p>	Cedula: “5.132.213-6”	Recupera los datos de la persona buscada por su cédula, si está pendiente a validar, puede editar o eliminar el censo.	Recupera los datos de la persona buscada por su cédula, si está pendiente a validar, puede editar o eliminar el censo.	P
---------	--	--	-----------------------	--	--	---

5. Código completo

HTML:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Censo 2023</title>

  <script src="/js/sistema.js"></script>
  <script src="/js/clases.js"></script>
  <script src="/js/codigo.js"></script>

  <link rel="stylesheet" href="css/styles.css">
</head>
<body>

  <div id="contenedor">
    <div id="cabezal" class="clearfix">
      

      <ul id="navPrincipal">

        <li id="btnAgregarPersona" class="btnFuncionalidades censista
invitado" >
          <a>Realizar Censo</a>
        </li>

        <li id="btnConsultaPreCompletados" class="btnFuncionalidades
censista invitado">
          <a>Consulta censo</a>
        </li>

        <li id="btnReasignarPersonaParaValidar"
class="btnFuncionalidades censista">
          <a>Reasignar persona para validar</a>
        </li>
      </ul>
    </div>
  </div>
</body>
```

```

        <li
id="btnVisualizarInfoEstadistica" class="btnFuncionalidades censista">
        <a>Visualizar información estadística</a>
    </li>

    <li id="btnVerListaCensados" class="btnFuncionalidades
invitado">
        <a>Visualizar lista de censados</a>
    </li>

    <li id = "cerrarSesion">
        <a>Cerrar sesión</a>
    </li>
    <div id="aux">
        <p id="nombreUsuarioLogeado"></p>
    </div>

</ul>

<div id="aux">
    <a id="btnSoyInvitado">Soy invitado</a>
    <p id="logueado"></p>
</div><!-- FIN AUX -->
</div><!-- FIN CABEZAL -->

<div id="contenido"> <!--INICIO CONTENIDO CSS-->

<div id="registro" class="seccion"> <!--REGISTRO-->

<P><strong>REGISTRO</strong></P><br>

<label for="registroNombre">Ingrese su nombre completo</label>
<input type="text" id="registroNombre" placeholder="Ingrese su
nombre y apellido"><br>

<label for="registroUsuario">Ingrese su nombre de usuario</label>
<input type="text" id="registroUsuario" placeholder="Ingrese
usuario"><br>

<label for="registroContraseña">Ingrese una contraseña</label>
<input type="text" id="registroContraseña" placeholder="Ingrese
una contraseña"><br>

<p id="pRegistro"></p>

<input type="button" id="btnRegistro" value="Registrarse"><br>

```

```

        <p>Ya tiene cuenta? Regístrese</p>
        <input type="button" id="btnIrAlLogin" value="Quiero ingresar con
mi cuenta"><br>

    </div><!--FIN ID REGISTRO-->

    <div id="login" class="seccion"> <!--INICIO -->

        <p><strong>INICIO DE SESIÓN</strong></p><br>

        <label for="LoginNUusuario">Nombre de Usuario:</label>
        <input type="text" id="loginNUusuario"><br>

        <label for="LoginContraseña">Contraseña:</label>
        <input type="text" id="loginContraseña"><br>

        <input type="button" id="btnLogin" value="Ingresar">

        <p id="pLogIn"></p>

        <p>No estás registrado?</p>
        <input type="button" id="btnIrAlRegistrar" value="Me quiero
registrar">

    </div><!--FIN ID LOGIN-->

    <div id="funcionalidades"><!--FUNCIONALIDADES-->
        <div id="agregarPersona" class="seccion"><!--AGREGAR PERSONA-->
            <p><strong>INGRESA DATOS PARA EL CENSO</strong></p>

            <label for="pNombre">Primer nombre: </label>
            <input type="text" id="pNombre" placeholder="Nombre y apellido de
la persona"><br>

            <label for="pApellido">Primer Apellido: </label>
            <input type="text" id="pApellido" placeholder="Nombre y apellido
de la persona"><br>

            <label for="pEdad">Edad: </label>
            <input type="text" id="pEdad" placeholder="Edad debe ser
ingresado en números"><br>

            <label for="pCedula">C.I: </label>
            <input type="text" id="pCedula" placeholder="Cédula de identidad
de la persona"><br>

```

```

<label>Departamento de residencia:</label>
<select id="slcDepto" class="deptos"></select><br>

<label>Ocupación</label>
<select id="slcOcupacion">
  <option value="-1">Selecione...</option>
  <option value="1">Dependiente</option>
  <option value="2">Independiente</option>
  <option value="3">No trabaja</option>
  <option value="4">Estudiante</option>
</select><br><br>

<input type="button" id="pCensoFinalizado" value="Censar
persona">

<p id="pDatosCenso"></p>
</div> <!--FIN DIV AGREGAR PERSONA-->

<div id="consultaPreCompletados" class="seccion"><!--CONSULTA PRE
COMPLETADOS -->
  <p><strong>CONSULTA DE CENSOS PRE COMPLETADOS</strong></p>

  <label for="consultaCedula">Cédula: </label>
  <input type="text" id="consultaCedula" placeholder="Cédula de
persona a validar">

  <input type="button" id="consultaCensoAValidar"
value="Consultar"><br><br>

  <hr>

  <label for="cNombre">Primer nombre: </label>
  <input type="text" id="cNombre" placeholder="Nombre y apellido de
la persona"><br>

  <label for="cApellido">Primer Apellido: </label>
  <input type="text" id="cApellido" placeholder="Nombre y apellido
de la persona"><br>

  <label for="cEdad">Edad: </label>
  <input type="text" id="cEdad" placeholder="Edad debe ser
ingresado en números"><br>

  <label>Departamento de residencia:</label>
  <select id="slcDeptoC" class="deptos"></select><br>

```

```

<label>Ocupación</label>
<select id="slcOcupacionC" >
  <option value="-1">Selecione...</option>
  <option value="1">Dependiente</option>
  <option value="2">Independiente</option>
  <option value="3">No trabaja</option>
  <option value="4">Estudiante</option>
</select><br><br>

<input type="button" id="btnValidarCenso" value="Validar censo">

<input type="button" id="irARealizarCenso" value="Ir a realizar
censo">

<input type="button" id="btnEditarCenso" value="Editar censo">

<input type="button" id="btnEliminarDatos" value="Eliminar
datos">

<p id="pMensaje"></p >

</div><!--FIN CONSULTA PRE COMPLETADOS-->

<div id="reasignarPersonaParaValidar" class="seccion"> <!--REASIGNA
PERSONAS-->

<label>Persona pendiente a validar: </label>
<select id="slcPersonasPendientesValidar" class="selectPersonas">
  <!--CARGAR DINAMICAMENTE LAS PERSONAS PENDIENTES A VALIDAR-->
</select><br>

<label>Censista:</label>
<select id="slcCensistas" class="censistas">
  <!--CARGAR DINAMICAMENTE LOS CENSISTAS, MENOS EL QUE ESTA
LOGUEADO-->
</select><br>

<input type="button" id="rAsignarCensista" value="Re-asignar
persona">

<p id="rMensajeReasignado"></p>
</div><!--FIN REASIGNA PERSONAS-->

<div id="visualizarInfoEstadistica" class="seccion"><!--INFO
ESTADISTICA-->

<table border="1">

```

```

        <thead>
            <tr>
                <th>TOTAL DE PERSONAS CENSADAS</th>
                <th>PORCENTAJE DE CENSOS PENDIENTES A VALIDAR:</th>
            </tr>
        </thead>
        <tbody id="tblInfoEstadistica"></tbody>
    </table><br><br>

    <label><strong>ELIJA UN DEPARTAMENTO PARA MOSTRAR
    INFORMACIÓN</strong></label><br>
    <select id="slcDeptosInfoEstadistica" class="deptos">
    </select>
    <input type="button" id="btnCensosPorDepto"
    value="MOSTRAR"><br><br>

    <p id="pMensajeTbl"></p>

    <table border="1">
        <thead>
            <tr>
                <th>CENSOS POR DEPARTAMENTO:</th>
                <th>MAYORES DE EDAD/ MENORES DE EDAD</th>
            </tr>
        </thead>
        <tbody id="tblInfoEstadisticaDepto"></tbody>
    </table>

</div><!--FIN VISUALIZAR INFO ESTADISTICA-->

<div id="verListaCensados" class="seccion"><!--LISTA DE CENSADOS-->
    <table border="1">
        <thead>
            <tr>
                <th>DEPARTAMENTO</th>
                <th>ESTUDIAN</th>
                <th>NO TRABAJAN</th>
                <th>DEPENDIENTES O INDEPENDIENTES</th>
                <th>TOTAL DE CENSOS</th>
            </tr>
        </thead>
        <tbody id="tblListaCensados"></tbody>
    </table>
</div><!--FIN LISTA DE CENSADOS-->
</div><!--FUNCIONALIDADES-->
    <!--FIN ELEMENTOS-->
</div><!-- FIN CONTENIDO -->

```

```
<div id="pie">
  <a href="https://www.ort.edu.uy">Programación 1</a>
</div><!-- FIN PIE -->
</div><!-- FIN CONTENEDOR -->

</body>
</html>
```

Js/código.js:

```
window.addEventListener("load", inicio);

function inicio() {
    document.querySelector("#btnRegistro").addEventListener("click",
    registrarUsuario);
    document.querySelector("#btnLogin").addEventListener("click",
    loginCensista);
    document.querySelector("#btnIrAlLogin").addEventListener("click",
    irALogin);
    document.querySelector("#btnIrAlRegistrar").addEventListener("click",
    irARegistro);
    document.querySelector("#cerrarSesion").addEventListener("click",
    cerrarSesion);
    document.querySelector("#btnSoyInvitado").addEventListener("click",
    abrirInvitado);
    //-----FUNCIONALIDADES
    document.querySelector("#pCensoFinalizado").addEventListener("click",
    ingresoDatosParaCenso);
    document.querySelector("#consultaCensoAValidar").addEventListener("click",
    consultaCensosPreCompletados);
    document.querySelector("#btnEditarCenso").addEventListener("click",
    editarDatosCenso);
    document.querySelector("#btnEliminarDatos").addEventListener("click",
    eliminarDatos);
    document.querySelector("#btnValidarCenso").addEventListener("click",
    validarCenso);
    document.querySelector("#irARealizarCenso").addEventListener("click",
    irAlIngresarDatos);
    document.querySelector("#rAsignarCensista").addEventListener("click",
    reAsignarPersona);
    document.querySelector("#btnVerListaCensados").addEventListener("click",
    mostrarListaDeCensados);
    document.querySelector("#btnVisualizarInfoEstadistica").addEventListener("click",
    mostrarTabla);
    document.querySelector("#btnCensosPorDepto").addEventListener("click",
    mostrarTablaDepartamento);
    //-----CARGA SELECTS
    document.querySelector("#btnReasignarPersonaParaValidar").addEventListener("click",
    cargarSelectsReasignar);
    cargarDeptos();
    //-----ESTO ES PARA LA PARTE DE "CONSULTA CENSOS"
    deshabilitaDatos();
    //-----NAVEGACIÓN
```



```

    let botonesFuncionalidades =
document.querySelectorAll(".btnFuncionalidades");

    for (let i = 0; i < botonesFuncionalidades.length; i++) {
        botonesFuncionalidades[i].addEventListener("click", mostrarSeccion);
    }

ocultarSecciones(); // oculta todo lo que tiene class="seccion"

    document.querySelector("#navPrincipal").style.display = "none";
    irARegistro();
}
//-----ACA TERMINA EL INICIO
let sistema = new Sistema();
let idCensista= 4;
let idInvitado= 31;
let perfilIniciadoSesion = false;
let usuarioLogueado= null;

function abrirInvitado(){
    mostrarFuncionalidades();
    mostrarBotones("invitado");
    mostrarBotonesConsultaPInvitado();
    document.querySelector("#consultaPreCompletados").style.display=
"block";
}

function irALogin(){
    document.querySelector("#registro").style.display = "none";
    document.querySelector("#login").style.display = "block";
}

function irARegistro(){
    document.querySelector("#registro").style.display = "block";
    document.querySelector("#login").style.display = "none";
}

function irAlIngresarDatos(){
    document.querySelector("#agregarPersona").style.display = "block";
    document.querySelector("#consultaPreCompletados").style.display=
"none";
}

function mostrarSeccion(){
    ocultarSecciones();

    let idBoton= this.getAttribute("id");

```

```

    let idSeccion= idBoton.charAt(3).toLowerCase() + idBoton.substring(4);
    document.querySelector("#" + idSeccion).style.display = "block";
}

function mostrarBotones(tipo){
    ocultarBotones();
    let botonesMostrar = document.querySelectorAll("." + tipo);
    for (let i = 0; i < botonesMostrar.length; i++) {
        botonesMostrar[i].style.display = "block";
    }
}

function ocultarBotones() {
    let botonesOcultar = document.querySelectorAll(".btnFuncionalidades");
    for (let i = 0; i < botonesOcultar.length; i++) {
        botonesOcultar[i].style.display = "none";
    }
}

function ocultarSecciones(){ //oculta todo (funcionalidades)
    let secciones= document.querySelectorAll(".seccion");
    for (let i= 0; i < secciones.length; i++) {
        secciones[i].style.display= "none";
    }
}

function cargarDeptos() { //carga departamentos
    const deptos = document.querySelectorAll(".deptos");

    for (let i = 0; i < deptos.length; i++) {
        deptos[i].innerHTML = `<option value="-1">Seleccione una
opción...</option>`;

        for (let j = 0; j < sistema.departamentos.length; j++) {
            const depto = sistema.departamentos[j];
            deptos[i].innerHTML += `<option
value="${depto.idDeppto}">${depto.depto}</option>`;
        }
    }
}

function cargarSelectsReasignar(){
    cargarCensistas();
    cargarPendientesAValidar();
}

function cargarPendientesAValidar(){

```

```

let personas= document.querySelectorAll(".selectPersonas");

for (let i = 0; i < personas.length; i++) {
    personas[i].innerHTML = `<option value="-1">Seleccione una
persona...</option>`;

    for(let j= 0; j < sistema.censados.length; j++){
        const persona= sistema.censados[j];
        if(persona.censado === false && usuarioLogueado.id ===
persona.idCensista){
            personas[i].innerHTML += `<option
value="${persona.id}">${persona.primerNombre}
${persona.apellido}</option>`;
        }
    }
}

function cargarCensistas(){

    const cargaCensistas = document.querySelectorAll(".censistas");

    for (let i = 0; i < cargaCensistas.length; i++) {
        cargaCensistas[i].innerHTML = `<option value="-1">Seleccione un
censista...</option>`;

        for (let j = 0; j < sistema.censistas.length; j++) {
            const censista = sistema.censistas[j];
            const personaLogueada= usuarioLogueado.id;

            if (personaLogueada !== censista.id) {
                cargaCensistas[i].innerHTML += `<option
value="${censista.id}">${censista.nombre}</option>`;
            }
        }
    }
}

function cerrarSesion(){
    ocultarSecciones();
    irARegistro();
    usuarioLogueado = null;
    perfilIniciadoSesion = false;
    document.querySelector("#navPrincipal").style.display = "none";
    document.querySelector("#nombreUsuarioLogeado").innerHTML = "none";
    document.querySelector("#btnSoyInvitado").style.display = "block";
    limpiarInputsLogIn();
}

```

```

limpiarInputsRegistro();
limpiarInputsRealizarCenso();
limpiarInputsConsultaCenso();
document.querySelector("#tblInfoEstadisticaDepto").innerHTML =
`<tr></tr>`;
document.querySelector("#rMensajeReasignado").innerHTML= "";
document.querySelector("#slcDeptosInfoEstadistica").value= -1;
document.querySelector("#pMensajeTbl").innerHTML= "";
}

function mostrarFuncionalidades(){
    document.querySelector("#navPrincipal").style.display = "block";
    document.querySelector("#registro").style.display = "none";
    document.querySelector("#login").style.display = "none";
    if(perfilIniciadoSesion){
        document.querySelector("#nombreUsuarioLogeado").innerHTML =
"Bienvenido " + usuarioLogeado.nombre + "!";
    }else{
        document.querySelector("#nombreUsuarioLogeado").innerHTML =
"Bienvenido invitado :>";
    }

    document.querySelector("#btnSoyInvitado").style.display = "none";
}

//-----REGISTRO E INICIO

function registrarUsuario() {
    let mensaje = "";
    let rNombre = document.querySelector("#registroNombre").value;
    let rUsuario = document.querySelector("#registroUsuario").value;
    let rContraseña = document.querySelector("#registroContraseña").value;

    rNombreUsuario= rUsuario.toLowerCase();

    rNombreCampoLleno = sistema.validarCampoVacio(rNombre);
    rUsuarioCampoLleno = sistema.validarCampoVacio(rNombreUsuario);
    rContraCampoLleno = sistema.validarCampoVacio(rContraseña);

    if (rNombreCampoLleno === true && rUsuarioCampoLleno === true &&
rContraCampoLleno === true) {

        let rContraseñaValidada = sistema.validarContraseña(rContraseña);

        if (rContraseñaValidada === true ) {

```

```

        let rUsuarioExiste =
sistema.buscarElemento(sistema.censistas,"nombreDeUsuario",rNombreUsuario
);

        if (!rUsuarioExiste) {
            let rUsuarioNuevo = new Censista(idCensista, rNombre,
rNombreUsuario, rContraseña);

            sistema.agregaCensista(rUsuarioNuevo);
            idCensista++;
            usuarioLogueado= rUsuarioNuevo.id;
            irALogin();
        }else{
            mensaje = "EL NOMBRE DE USUARIO YA EXISTE";
        }
        }else{
            mensaje = "SU CONTRASEÑA DEBE TENER 5 CARACTERES MINIMO Y CONTAR
CON UNA MAYUSCULA, UNA MINUSCULA Y UN NUMERO";
        }
        }else{
            mensaje = "LOS CAMPOS NO PUEDEN ESTAR VACIOS";
        }
        document.querySelector("#pRegistro").innerHTML = mensaje;
    }
}

function loginCensista() {
    let lNombre = document.querySelector("#loginNUsuario").value;
    let lContraseña = document.querySelector("#loginContraseña").value;
    let mensaje= "";

    let lNombreCI= lNombre.toLowerCase()

    let lNombreCampolLeno = sistema.validarCampoVacio(lNombreCI);
    let lContraseñaCampolLeno = sistema.validarCampoVacio(lContraseña);

    if(lNombreCampolLeno=== true && lContraseñaCampolLeno=== true){

        let verificarDatos= sistema.verificarLogin(lNombreCI, lContraseña);

        if(verificarDatos === true){
            usuarioLogueado= sistema.obtenerObjeto(sistema.censistas,
"nombreDeUsuario", lNombre);
            perfilIniciadoSesion= true;
            mostrarBotonesConsultaPCensista();
            mostrarFuncionalidades();
            mostrarBotones("censista");

```

```

        document.querySelector("#agregarPersona").style.display= "block";
    }else{
        mensaje= "EL NOMBRE DE USUARIO O CONTRASEÑA NO COINCIDEN. <br>
VERIFIQUE SUS DATOS O CREÉSE UNA CUENTA";
    }
    }else{
        mensaje= "LOS CAMPOS NO PUEDEN ESTAR VACIOS";
    }
document.querySelector("#pLogIn").innerHTML = mensaje;
}

//-----FUNCIONALIDADES
function generarIDCensista(){
    let arrayCensista = sistema.censistas;
    let idAleatorio = Math.floor(Math.random() * arrayCensista.length);
    return idAleatorio;
}

function ingresoDatosParaCenso(){
    let iPrimerNombre= document.querySelector("#pNombre").value;
    let iApellido = document.querySelector("#pApellido").value;
    let iEdad= Number(document.querySelector("#pEdad").value);
    let iCedula= document.querySelector("#pCedula").value;
    let iDepto= Number (document.querySelector(".deptos").value);
    let iOcupacion = Number(document.querySelector("#slcOcupacion").value);
    let mensaje= "";
    let iCensado = false; //propiedad censado

    if(perfilIniciadoSesion) { //si el usuario es censista, guarda el censo
con la propiedad censado en true
        iCensado = true;
    }

    // validaciones
    let iPrimerNombreCompCampoLleno=
sistema.validarCampoVacio(iPrimerNombre);
    let iApellidoCompCampoLleno = sistema.validarCampoVacio(iApellido);
    let iEdadCompCampoLleno = sistema.validarCampoVacio(iEdad);
    let iCedulaCompCampoLleno = sistema.validarCampoVacio(iCedula);
    let iDeptoCompCampoLleno = sistema.validarSelects(iDepto);
    let iOcupacionCompCampoLleno = sistema.validarSelects(iOcupacion);

    if(iPrimerNombreCompCampoLleno=== true &&
        iApellidoCompCampoLleno=== true &&
        iEdadCompCampoLleno=== true &&
        iCedulaCompCampoLleno=== true &&

```

```

iDeptoCompCampolleno === true &&
iOcupacionCompCampolleno === true){

    let iEdadValida= sistema.verificaEdad(iEdad);

    if(iEdadValida===true){

        let validaCedula= sistema.verificaCedula(iCedula);

        if(validaCedula=== true){

            let buscaSiCedulaExiste= sistema.buscarElemento(sistema.censados,
"cedula", iCedula);

            if(buscaSiCedulaExiste=== false){
                let idCensistaAsignado;
                if(usuarioLogueado !== null){
                    idCensistaAsignado= usuarioLogueado.id;
                }else{
                    idCensistaAsignado= generarIDCensista();
                }
                let censistaAsignado = sistema.obtenerObjeto(sistema.censistas,
"id", idCensistaAsignado);

                let nuevaPCensada= new Censado(idInvitado, iPrimerNombre,
iApellido, iEdad, iCedula, iDepto, iOcupacion, idCensistaAsignado,
iCensado);

                sistema.agregaCensado(nuevaPCensada);
                idInvitado++;
                mostrarTabla();           //para que se actualice la información
de las tablas
                mostrarListaDeCensados();
                mensaje= "PERSONA CENSADA CON EXITO <br><br> Se le asigno el
censista: " + censistaAsignado.nombre;
            }else{
                mensaje = "LA PERSONA YA ESTÁ CENSADA";
            }
        }else{
            mensaje= "CEDULA INCORRECTA, VERIFIQUE DATOS";
        }
    }else{
        mensaje= "EDAD INVALIDA";
    }
}else{
    mensaje = "LOS CAMPOS NO PUEDEN ESTAR VACIOS";
}
}

```

```

    document.querySelector("#pDatosCenso").innerHTML = mensaje;
}

function consultaCensosPreCompletados(){
    deshabilitaDatos();
    let preCargaCedula = document.querySelector("#consultaCedula").value;
    let preCargaCedulaValidada = sistema.verificaCedula(preCargaCedula);

    let mensaje = "";

    let preCargaCedulaCampoLleno =
sistema.validarCampoVacio(preCargaCedula);

    if(preCargaCedulaCampoLleno === true){
        let busquedaAValidar = sistema.buscarElemento(sistema.censados,
"cedula", preCargaCedula);

        if(busquedaAValidar === true &&
preCargaCedulaValidada === true){
            let persona = sistema.obtenerObjeto(sistema.censados, "cedula",
preCargaCedula);

            if(persona.censado === true){
                mensaje = "EL CENSO YA FUE VALIDADO";
            }else{
                recuperaDatosInput(preCargaCedula);
            }
        }else {
            if(!perfilIniciadoSesion){
                let btnIrARealizarCenso=
document.querySelector("#irARealizarCenso");
                btnIrARealizarCenso.removeAttribute("disabled");
                mensaje= `DATOS NO ENCONTRADOS O CEDULA INVÁLIDA <br>
SI ANTERIORMENTE NO INGRESÓ SUS DATOS, PUEDE IR A REALIZAR
CENSO`;
            }else{
                mensaje = "DATOS NO ENCONTRADOS O CEDULA INVÁLIDA";
            }
        }
    }
    } else {
        mensaje = "EL CAMPO NO PUEDE ESTAR VACIO";
    }
    document.querySelector("#pMensaje").innerHTML = mensaje;
}

```



```

//-----INTERFAZ
FUNCIONALIDAD CONSULTA CENSO
// deshabilitamos los inputs precompletados
function deshabilitaDatos(){
    let cNombreInput = document.querySelector("#cNombre");
    let cApellidoInput = document.querySelector("#cApellido");
    let cEdadInput = document.querySelector("#cEdad");
    let slcDeptoSelect = document.querySelector("#slcDeptoC");
    let slcOcupacionSelect = document.querySelector("#slcOcupacionC");
    let btnEliminarDatos = document.querySelector("#btnEliminarDatos");
    let btnEditarCenso = document.querySelector("#btnEditarCenso");
    let btnValidarCenso = document.querySelector("#btnValidarCenso");
    let btnIrARealizarCenso= document.querySelector("#irARealizarCenso");

    cNombreInput.setAttribute("disabled", "disabled");
    cApellidoInput.setAttribute("disabled", "disabled");
    cEdadInput.setAttribute("disabled", "disabled");
    slcDeptoSelect.setAttribute("disabled", "disabled");
    slcOcupacionSelect.setAttribute("disabled", "disabled");
    btnEliminarDatos.setAttribute("disabled", "disabled");
    btnEditarCenso.setAttribute("disabled", "disabled");
    btnValidarCenso.setAttribute("disabled", "disabled");
    btnIrARealizarCenso.setAttribute("disabled", "disabled");
}

// habilitamos los inputs precompletados
function habilitaDatos(){
    let cNombreInput = document.querySelector("#cNombre");
    let cApellidoInput = document.querySelector("#cApellido");
    let cEdadInput = document.querySelector("#cEdad");
    let slcDeptoSelect = document.querySelector("#slcDeptoC");
    let slcOcupacionSelect = document.querySelector("#slcOcupacionC");
    let btnEliminarDatos = document.querySelector("#btnEliminarDatos");
    let btnEditarCenso = document.querySelector("#btnEditarCenso");
    let btnValidarCenso = document.querySelector("#btnValidarCenso");

    cNombreInput.removeAttribute("disabled");
    cApellidoInput.removeAttribute("disabled");
    cEdadInput.removeAttribute("disabled");
    slcDeptoSelect.removeAttribute("disabled");
    slcOcupacionSelect.removeAttribute("disabled");
    btnEliminarDatos.removeAttribute("disabled");
    btnEditarCenso.removeAttribute("disabled");
    btnValidarCenso.removeAttribute("disabled");
}

function mostrarBotonesConsultaPCensista(){

```

```

    if(perfilIniciadoSesion === true){
        document.querySelector("#btnValidarCenso").style.display = "block";
        document.querySelector("#btnEditarCenso").style.display = "none";
        document.querySelector("#btnEliminarDatos").style.display = "none";
        document.querySelector("#irARealizarCenso").style.display = "none";
    }
}

function mostrarBotonesConsultaPInvitado(){
    if(perfilIniciadoSesion === false){
        document.querySelector("#btnValidarCenso").style.display = "none";
        document.querySelector("#btnEditarCenso").style.display = "block";
        document.querySelector("#btnEliminarDatos").style.display = "block";
        document.querySelector("#irARealizarCenso").style.display = "block";
    }
}

function recuperaDatosInput(cedula){

    let personaCedula = sistema.obtenerObjeto(sistema.censados, "cedula",
cedula);

    if(personaCedula !== null){
        habilitaDatos();
        document.querySelector("#cNombre").value =
personaCedula.primerNombre;

        document.querySelector("#cApellido").value =
personaCedula.apellido;

        document.querySelector("#cEdad").value = personaCedula.edad;

        document.querySelector("#slcDeptoC").value = personaCedula.depto;

        document.querySelector("#slcOcupacionC").value =
personaCedula.ocupacion;

        alert("Datos recuperados");
    }
}

//-----FIN INTERFAZ
FUNCIONALIDAD CONSULTA CENSO

function editarDatosCenso(){

    let preCargaCedula = document.querySelector("#consultaCedula").value;

```

```

let primerNombre = document.querySelector("#cNombre").value;
let apellido = document.querySelector("#cApellido").value;
let edad = Number(document.querySelector("#cEdad").value);
let depto = Number(document.querySelector("#slcDeptoC").value);
let ocupacion = Number(document.querySelector("#slcOcupacionC").value);
let mensaje = "";

let primerNombreCampolleno = sistema.validarCampoVacio(primerNombre);

let apellidoCampolleno = sistema.validarCampoVacio(apellido);

let edadCampolleno = sistema.validarCampoVacio(edad);
let edadValidada = sistema.verificaEdad(edad);
let deptoCampolleno = sistema.validarSelects(depto);
let ocupacionCampolleno = sistema.validarSelects(ocupacion);

if(primerNombreCampolleno && apellidoCampolleno && edadCampolleno &&
edadValidada&& deptoCampolleno && ocupacionCampolleno){
    let persona = sistema.obtenerObjeto(sistema.censados, "cedula",
preCargaCedula);

    if(persona !== null){
        sistema.sobreEscribirDatosPersona(preCargaCedula, primerNombre,
apellido, edad, depto, ocupacion);
        mensaje = `SE HA EDITADO LOS DATOS CON EXITO`;
    }
    else {
        mensaje = `LOS CAMPOS NO PUEDEN ESTAR VACIOS`;
    }
    document.querySelector("#pMensaje").innerHTML = mensaje;
}

function eliminarDatos() {
    let preCargaCedula = document.querySelector("#consultaCedula").value;
    let persona= sistema.obtenerObjeto(sistema.censados, "cedula",
preCargaCedula);
    let propiedadCensado= persona.censado;
    let personasCensadas = sistema.censados;
    let index = -1;
    let mensaje= "";

    for (let i = 0; i < personasCensadas.length; i++) {
        if (personasCensadas[i].cedula === preCargaCedula) {
            index = i;
            break;
        }
    }
}

```

```

if (index !== -1) {
  if(propiedadCensado !== true){
    if (confirm("¿Estás seguro de eliminar todos los datos?")) {
      personasCensadas.splice(index, 1);
      alert("DATOS ELIMINADOS CORRECTAMENTE.");
      document.querySelector("#cNombre").value = "";
      document.querySelector("#cApellido").value = "";
      document.querySelector("#cEdad").value = "";
      document.querySelector("#slcDeptoC").value = "";
      document.querySelector("#slcOcupacionC").value = "";
    }
  }else{
    mensaje= "EL CENSO YA HA SIDO VALIDADO";
  }
} else {
  mensaje = "PERSONA NO ENCONTRADA"
}
document.querySelector("#pMensaje").innerHTML = mensaje;
}

function validarCenso(){
  editarDatosCenso();
  let cedulaObtenida = document.querySelector("#consultaCedula").value;
  let primerNombre = document.querySelector("#cNombre").value;
  let apellido = document.querySelector("#cApellido").value;
  let edad = Number(document.querySelector("#cEdad").value);
  let depto = Number(document.querySelector("#slcDeptoC").value);
  let ocupacion = Number(document.querySelector("#slcOcupacionC").value);

  let primerNombreCampoLleno = sistema.validarCampoVacio(primerNombre);
  let apellidoCampoLleno = sistema.validarCampoVacio(apellido);

  let edadCampoLleno = sistema.validarCampoVacio(edad);
  let edadValidada = sistema.verificaEdad(edad);
  let deptoCampoLleno = sistema.validarSelects(depto);
  let ocupacionCampoLleno = sistema.validarSelects(ocupacion);
  let mensaje = "";

  if(primerNombreCampoLleno && apellidoCampoLleno && edadCampoLleno &&
  edadValidada && deptoCampoLleno && ocupacionCampoLleno){

    let persona = sistema.obtenerObjeto(sistema.censados, "cedula",
    cedulaObtenida);

    if(!persona.censado){

```

```

    persona.censado = true;
    mensaje = "PERSONA VALIDADA CON EXITO";

    } else{
        mensaje = "LA PERSONA YA HA SIDO VALIDADA";
    }

    }else{
        mensaje = "LOS CAMPOS NO PUEDEN ESTAR VACIOS";
    }

    document.querySelector("#pMensaje").innerHTML = mensaje
}

function reAsignarPersona(){
    let idPersonaAREasignar= Number
(document.querySelector("#slcPersonasPendientesValidar").value);
    let idNuevoCensistaSlc= Number
(document.querySelector("#slcCensistas").value);
    let mensaje= "";

    let idPersonaCampoLleno= sistema.validarSelects(idPersonaAREasignar);
    let idCensistaCampoLleno= sistema.validarSelects(idNuevoCensistaSlc);

    if(idPersonaCampoLleno && idCensistaCampoLleno){
        sistema.sobreEscribirPropiedadCensado(idPersonaAREasignar,
idNuevoCensistaSlc);
        cargarSelectsReasignar();
        mensaje="PERSONA REASIGNADA CON ÉXITO";
    }else{
        mensaje= "LOS CAMPOS NO PUEDEN ESTAR VACIOS";
    }
    document.querySelector("#rMensajeReasignado").innerHTML= mensaje;
}

function mostrarTabla(){
    document.querySelector("#tblInfoEstadistica").innerHTML= "";

    let censosValidados = 0;
    let censosNoValidados = 0;

    for (let i = 0; i < sistema.censados.length; i++) {
        const unCensado = sistema.censados[i];

        if(unCensado.censado == true){

```

```

        censosValidados++;
    }else{
        censosNoValidados++;
    }
    let total = censosValidados + censosNoValidados;
    let porcentaje = (censosNoValidados / total)* 100;

    if(total > 0){
        document.querySelector("#tblInfoEstadistica").innerHTML =
        `<tr>
            <td>${censosValidados}</td>
            <td>${porcentaje.toFixed(0)}%</td>
        </tr>`
    }
}
}

function mostrarTablaDepartamento(){

document.querySelector("#tblInfoEstadisticaDepto").innerHTML = "";
let deptoSlc =
Number(document.querySelector("#slcDeptosInfoEstadistica").value);
let validarDepto= sistema.validarSelects(deptoSlc);
let mensaje= "";
let contadorDeptos = 0;
let contadorMayores=0;
let contadorMenores=0;
for (let i = 0; i < sistema.censados.length; i++) {
    const unCensado = sistema.censados[i];
    if(deptoSlc == unCensado.depto){
        contadorDeptos++;
    }
    if(unCensado.edad >= 18 && deptoSlc == unCensado.depto){
        contadorMayores++;
    }
    if(unCensado.edad < 18 && deptoSlc == unCensado.depto){
        contadorMenores++;
    }
}

let total = contadorMayores + contadorMenores;

let porcentajeMenores = 0;
let porcentajeMenoresFinal = 0;

```

```

let porcentajeMayores = 0;
let porcentajeMayoresFinal = 0;

if (total !== 0) {
    porcentajeMenores = contadorMenores / total;
    porcentajeMenoresFinal = porcentajeMenores * 100;

    porcentajeMayores = contadorMayores / total;
    porcentajeMayoresFinal = porcentajeMayores * 100;
}

if (validarDepto) {
    let departamentos = sistema.obtenerObjeto(sistema.departamentos,
    "idDepto", deptoSlc);
    let nombreDelDepto = departamentos.depto;
    document.querySelector("#tblInfoEstadisticaDepto").innerHTML =
    `|
        <td>${nombreDelDepto.toUpperCase()}: ${contadorDeptos}</td>
        <td>${porcentajeMayoresFinal.toFixed(1)}% /
    ${porcentajeMenoresFinal.toFixed(1)}%</td>
    </tr>`;
    }else{
        mensaje= "SELECCIONE UN DEPARTAMENTO";
    }
    document.querySelector("#pMensajeTbl").innerHTML= mensaje;
}

function mostrarListaDeCensados() {
    let censados = sistema.censados;
    let departamentos = sistema.departamentos;

    for (let i = 0; i < departamentos.length; i++) {
        let nombreDeptos = departamentos[i].depto;
        let censosRealizados = 0;
        let censosPorDepto = 0;
        let estudiantes = 0;
        let noTrabajan = 0;
        let trabajan = 0;

        for (let j = 0; j < censados.length; j++) {
            const unCensado = censados[j];
            censosRealizados++;

            if (departamentos[i].idDepto === unCensado.depto) {
                censosPorDepto++;

                if (unCensado.ocupacion === 4) {

|  |

```

```

        estudiantes++;
    }
    if (unCensado.ocupacion === 3) {
        noTrabajan++;
    }
    if (unCensado.ocupacion === 1 || unCensado.ocupacion === 2) {
        trabajan++;
    }
}
}

let porcentaje = (censosPorDepto / censosRealizados) * 100;

document.querySelector("#tblListaCensados").innerHTML +=
`<tr>
    <td>${nombreDeptos}</td>
    <td>${estudiantes}</td>
    <td>${noTrabajan}</td>
    <td>${trabajan}</td>
    <td>${porcentaje.toFixed(0)}%</td>
</tr>`;
}
}

//-----LIMPIAR INPUTS

function limpiarInputsRealizarCenso(){
    document.querySelector("#pNombre").value= "";
    document.querySelector("#pApellido").value= "";
    document.querySelector("#pEdad").value= "";
    document.querySelector("#pCedula").value= "";
    document.querySelector(".deptos").value= "-1";
    document.querySelector("#slcOcupacion").value= "-1";
    document.querySelector("#pDatosCenso").innerHTML = "";
    document.querySelector("#pMensaje").innerHTML= "";
}

function limpiarInputsConsultaCenso(){
    document.querySelector("#consultaCedula").value= "";
    document.querySelector("#cNombre").value= "";
    document.querySelector("#cApellido").value= "";
    document.querySelector("#cEdad").value= "";
    document.querySelector("#slcDeptoC").value= "-1";
    document.querySelector("#slcOcupacionC").value= "-1";
    document.querySelector("#consultaCedula").value= "";
}

```



```

function limpiarInputsLogIn(){
    document.querySelector("#loginNUusuario").value= "";
    document.querySelector("#loginContraseña").value="";
}

function limpiarInputsRegistro(){
    document.querySelector("#registroNombre").value= "";
    document.querySelector("#registroUsuario").value= "";
    document.querySelector("#registroContraseña").value= "";
}

```

Js/clases.js:

```

class Censista {
    constructor(idCensista, nombre, nombreDeUsuario, contrasena) {

        this.id= idCensista;
        this.nombre = nombre;
        this.nombreDeUsuario = nombreDeUsuario;
        this.contrasena= contrasena;
    }
}

class Censado {
    constructor(idCensado, primerNombre, apellido, edad, cedula, depto,
ocupacion, idCensista, censado) {
        this.id = idCensado;
        this.primerNombre = primerNombre;
        this.apellido = apellido;
        this.edad = edad;
        this.cedula = cedula;
        this.depto = depto;
        this.ocupacion = ocupacion;
        this.idCensista= idCensista; //este es el censista asignado
        this.censado = censado;
    }
}

class Departamento {
    constructor(idDepto, nombreDepto){
        this.idDepto = idDepto;
        this.depto = nombreDepto;
    }
}

```

Js/sistema.js:

```
class Sistema {
  constructor() {
    this.censistas = [
      new Censista(1, "Juan", "juan123", "A1aaaa"),
      new Censista(2, "Maria", "maria123", "B2bbbb"),
      new Censista(3, "Carlos", "carlos123", "C3ccc"),
    ];

    this.censados = [
      new Censado(1, "Ben", "Diez", 10, "6.342.754-2", 4, 4, 1, false),
      new Censado(2, "Goku", "Goku", 30, "5.132.213-6", 2, 2, 1, false),
      new Censado(3, "Dora", "la Exploradora", 7, "4.765.339-3", 3, 1, 2,
true),
      new Censado(4, "Bob", "Esponja", 8, "4.189.432-5", 7, 4, 1, false),
      new Censado(5, "Pikachu", "Pikachu", 30, "7.010.924-2", 16, 1, 1,
false),
      new Censado(6, "Mickey", "Mouse", 5, "8.032.312-7", 1, 4, 3, true),
      new Censado(7, "Pluto", "Fernandez", 4, "5.192.995-4", 7, 1, 2,
false),
      new Censado(8, "Masha", "Martinez", 6, "3.858.201-0", 11, 4, 3,
false),
      new Censado(9, "Nobita", "Nobi", 9, "3.180.659-4", 19, 4, 1, false),
      new Censado(10, "Elmo", "Martinez", 3, "5.989.149-8", 7, 1, 2,
false),
      new Censado(11, "Scooby", "Doo", 6, "6.286.290-5", 4, 4, 3, true),
      new Censado(12, "Diego", "Marquez", 8, "8.666.859-1", 4, 1, 3, true),
      new Censado(13, "Bugs", "Bunny", 7, "3.787-440-2", 9, 3, 1,
false),
      new Censado(14, "Timmy", "Turner", 10, "7.113.583-2", 19, 4, 2,
true),
      new Censado(15, "Heidi", "Martinez", 7, "7.233.151-8", 16, 1, 1,
false),
      new Censado(16, "Naruto", "Shipuden", 11, "7.614.592-1", 1, 1, 3,
false),
      new Censado(17, "Tweety", "Fernandez", 5, "5.013.139-0", 2, 3, 2,
false),
      new Censado(18, "Pucca", "Pucci", 18, "6.664.842-0", 10, 1, 2,
true),
      new Censado(19, "Oliver", "Atom", 13, "7.672.799-7", 14, 3, 1,
false),
      new Censado(20, "Lilo", "Pelekai", 42, "2.389.888-8", 2, 1, 3, false),
      new Censado(21, "Dumbo", "Perez", 4, "4.467.144-5", 16, 4, 3,
true),
    ];
  }
}
```

```

        new Censado(22,"Pinky","Dinky Doo",6, "3.426.777-9" ,9 ,3 ,2,
true),
        new Censado(23,"Beyblade","Rodriguez",11, "8.457.734-2" , 19 ,2 ,
2, false),
        new Censado(24,"Kim","Possible",14,"9.891.450-2", 10 ,2 , 1, true),
        new Censado(25,"Pato","Aventuras",50,"7.318.850-2",2 ,1 , 2,
false),
        new Censado(26, "Popeye", "Martinez", 45, "6.901.814-3", 14 , 3 ,
1, false),
        new Censado(27,"George","Pájaro",7,"4.842.763-2",9 ,2 , 2, false),
        new Censado(28, "Dino", "Fernandez", 4, "4.576.260-9", 16 , 4 , 2,
false),
        new Censado(29,"Donatello","Tortuga",59,"3.409.788-7",14 ,2 , 2,
true),
        new Censado(30,"Ryder","Paw Patrol",6,"7.695.216-6",2 ,2 , 1,
false),
    ];

    this.departamentos= [
        new Departamento(1, "Artigas"),
        new Departamento(2, "Canelones"),
        new Departamento(3, "Cerro Largo"),
        new Departamento(4, "Colonia"),
        new Departamento(5, "Durazno"),
        new Departamento(6, "Flores"),
        new Departamento(7, "Florida"),
        new Departamento(8, "Lavalleja"),
        new Departamento(9, "Maldonado"),
        new Departamento(10, "Montevideo"),
        new Departamento(11, "Paysandú"),
        new Departamento(12, "Río Negro"),
        new Departamento(13, "Rivera"),
        new Departamento(14, "Rocha"),
        new Departamento(15, "Salto"),
        new Departamento(16, "San José"),
        new Departamento(17, "Soriano"),
        new Departamento(18, "Tacuarembó"),
        new Departamento(19, "Treinta y Tres")
    ]
}

agregaCensista(censista) {
    this.censistas.push(censista);
}

agregaCensado(personaCensada) {
    this.censados.push(personaCensada);
}

```

```

obtenerObjeto(arrElementos, propiedad, busqueda) {
  let objeto = null;
  for (let i = 0; i < arrElementos.length; i++) {
    const unElemento = arrElementos[i];
    if(unElemento[propiedad] === busqueda){
      objeto = unElemento;
      break;
    }
  }
  return objeto;
}

buscarElemento(array, propiedad, busqueda) {
  let existe = false;
  for (let i = 0; i < array.length; i++) {
    const elementoEncontrado = array[i];
    if (elementoEncontrado[propiedad] === busqueda) {
      existe = true;
      break;
    }
  }
  return existe;
}

validarCampoVacio(valor) {
  let campoValido = false;
  if(valor !== ""){
    campoValido= true;
  }
  return campoValido;
}

validarContrasena(clave) {
  let contadorMinusculas = 0;
  let contadorMayusculas = 0;
  let contadorNumeros = 0;
  let contrasenaValida = false;

  for (let i = 0; i < clave.length; i++) {
    let codigoDeLetraActual = clave.charCodeAt(i);
    if (codigoDeLetraActual >= 48 && codigoDeLetraActual <= 57) {
      contadorNumeros++;
    }
    if (codigoDeLetraActual >= 65 && codigoDeLetraActual <= 90) {
      contadorMayusculas++;
    }
    if (codigoDeLetraActual >= 97 && codigoDeLetraActual <= 122) {

```

```

        contadorMinusculas++;
    }
}

if (clave.length >= 5 && contadorNumeros >= 1 && contadorMayusculas >=
1 && contadorMinusculas >= 1) {
    contrasenaValida = true;
}
return contrasenaValida;
}

verificarLogin(nombreUsuario, clave) {
    let resultado = false;
    let unUsuario = this.obtenerObjeto(this.censistas,
"nombreDeUsuario", nombreUsuario);

    if (unUsuario !== null) {
        if (unUsuario.contrasena === clave) {

            resultado = true;
        }
    }
    return resultado;
}

verificaEdad(edad){
let resultado= false;
if(!isNaN(edad)){
    if(edad > 0 && edad <= 130){
        resultado = true;
    }
}
return resultado
}

//esta la usamos para eliminar guiones y puntos de la cedula
eliminarCaracter(txt, caracter){
let textoSustituido = "";

for (let i = 0; i < txt.length; i++) {
    if (txt.charAt(i) !== caracter) {
        textoSustituido += txt.charAt(i);
    }
}
return textoSustituido;
}

```

```

verificaCedula(cedula){
  let valida = false;
  let CIgion = this.eliminarCaracter(cedula, "-");
  let CI = this.eliminarCaracter(CIgion, ".");

  if(!isNaN(CI)){
    if (CI.length === 7) {
      CI = "0" + CI;
    }

    let multiplicador= "2987634";
    let digitoAVerificar= CI.charAt(CI.length -1);
    let acumulador= 0;

    for(let i= 0; i < CI.length -1; i++){
      acumulador+= Number(CI.charAt(i) *
Number(multiplicador.charAt(i)));
    }

    let digitoVerificador = (10 - (acumulador % 10)) % 10;

    if(digitoVerificador === Number(digitoAVerificar)){
      valida= true;
    }
    return valida;
  }
}

validarSelects(value){
  let existe = false;
  if(value !== -1){
    existe = true
  }
  return existe
}

sobreEscribirDatosPersona(cedula, primerNombre, apellido, edad, depto,
ocupacion){
  let persona = this.obtenerObjeto(this.censados, "cedula", cedula);
  persona.primerNombre = primerNombre;
  persona.apellido = apellido;
  persona.edad = edad;
  persona.depto = depto;
  persona.ocupacion = ocupacion;
}

```

```
sobreEscribirPropiedadCensado(idPersona, idCensistaNuevo){  
  let personaAValidar= this.obtenerObjeto(this.censados, "id",  
idPersona);  
  personaAValidar.idCensista= idCensistaNuevo;  
}  
}
```

LISTAS DE INFORMACIÓN PRECARGADA:

CENSISTAS					
ID	NOMBRE	NOMBRE USUARIO		CONTRASEÑA	
1	Juan	juan123		A1aaaa	
2	Maria	maria123		B2bbbb	
3	Carlos	carlos123		C3ccc	

[illegible]