

Test Suit	Test Scenario	Test Case	TC Description	Step	Action	Test Data	Expect Result	Actual Result	Pass/Fail
TS001	Should return initial state on first run	TC001	Verifica que el programa haya pasado al estado inicial.	1	Declara el siguiente estado como el estado Inicial	initialState	TRUE	TRUE	PASS
				2	Se carga los datos que serán necesarios para pasar al siguiente estado	PayloadAction	-	-	
				3	Se ejecuta la función con los datos cargados y se compara con el resultado esperado	undefined, action	nextState	nextState	
TS002	Should add note if there are no notes in draft state on addNote	TC001	Verifica el borrador para añadir una nota nueva	1	Se ejecuta el constructor de nota	id, category	-	-	PASS
				2	Se declara el siguiente estado como nota creada.	nextState	notes[payload]	notes[payload]	
				3	Se ejecuta el programa con los datos de la nueva nota.	addNote	nextState	nextState	
TS003	Should update the category name on updateCategory	TC001	Verifica el nombre de la categoría se actualice	1	Se crea una nota llamada 'note 0'	id, name, draggeOver	-	-	PASS
				2	Se declara un el siguiente estado con los datos cargados y el nuevo nombre de la categoría 'note o renamed'	nextState	categories[payload]	categories[payload]	
				3	Se compara el estado obtenido con el estado esperado.	updateCategory(payload)	nextState	nextState	
TS004	Renders the ContextMenuOptions	TC001	Habilita las opciones del menú de contexto	1	Escribe la interfaz ContextMenuOptions	id, text, created, lastUpdated	-	-	PASS
				2	Se obtiene el componente creado y se ejecuta la función para verificar su validez.	component	TRUE	TRUE	
TS005	Should return 45 characters	TC001	Escribe una nota y verifica el valor devuelto.	1	Se define la nota con el contenido o mensaje.	note	-	-	PASS
				2	Se ejecuta la función designada para devolver el título de la nota	note	45	45	
TS006	Should display the category name followed by the app name	TC001	Verifica que se muestre el nombre de la app después de la categoría	1	Se escribe la categoría	id, name, draggeOver	-	-	PASS
				2	Se ejecuta la función encargada de obtener el nombre de categoría.	category	Recipes Takenote	Recipes Takenote	
TS007	Should see empty editor if there are no active notes	TC001	Verifica que se muestre el editor vacío si no hay notas activas	1	Se ejecuta las funciones mock, se inicializa las categorías, notas vacías y las configuraciones por defecto	activeCategoryId, activeFolder, activeNoteId, SelectNotesId, error, loading, notes, searchValue	-	-	PASS
				2	Se renderiza el editor.	component	TRUE	TRUE	
TS008	Renders the PreviewEditor component	TC001	Habilita la vista previa del editor	1	Escribe el editor de texto y los valores default.	noteText, directionText, notes	-	-	PASS
				2	Verifica que la vista previa del editor no sea nula o vacía.	component	TRUE	TRUE	
TS009	Should set the loading true on login	TC001	Cambia a estado a loading en el login	1	Se declara el siguiente estado con el loading.	nextState	loading: true	loading: true	PASS
				2	Se ejecuta la función y se verifica si el valor ha sido establecido correctamente.	login	nextState	nextState	
TS010	should set the error to payload, isAuthenticated to false and loading to false on loginError	TC001	Verifica que los datos de autenticación sean correctos y establece el error de no serlo.	1	Se escribe el siguiente estado con la autenticación y el loading en falso	initialState, login, isAuthenticated, error	-	-	PASS
				2	Se cargan los datos del error y se ejecuta el siguiente estado	loginError(payload)	nextState	nextState	
TS011	Should display Unsaved change	TC001	Verifica que se muestre el mensaje de alerta del respectivo caso	1	Escribe la última notificación sincronizada, con valores pendientes por guardar	datetime, pending, syncing	-	-	PASS
				2	Se ejecuta la función para mostrar el mensaje de notificación sincronizada.	LastSyncedNotification	Unsaved Change	Unsaved Change	
	Should display date	TC002	Muestra la fecha de la última sincronización hecha	1	Escribe la última notificación sincronizada con la fecha actual y sin cambios a guardar	datetime, pending, syncing	-	-	PASS
				2	Se ejecuta la función de renderizado.	LastSyncedNotification	Date	Date	
TS012	Should return initial state on first run	TC001	Verifica que el programa haya pasado al estado inicial.	1	Declara el siguiente estado como el estado Inicial	initialState	TRUE	TRUE	PASS
				2	Se carga los datos que serán necesarios para pasar al siguiente estado	PayloadAction	-	-	
				3	Se ejecuta la función con los datos cargados y se compara con el resultado esperado	undefined, action	nextState	nextState	
TS013	Should return initial state on first run	TC001	Verifica que el programa haya pasado al estado inicial.	1	Declara el siguiente estado como el estado Inicial	initialState	TRUE	TRUE	PASS
				2	Se carga los datos que serán necesarios para pasar al siguiente estado	PayloadAction	-	-	
				3	Se ejecuta la función con los datos cargados y se compara con el resultado esperado	undefined, action	nextState	nextState	
TS014	Renders the SearchBar component	TC001	Habilita los accesorios de la barra de búsqueda	1	Escribe la interfaz SearchBarProps.	-	-	-	PASS
				2	Prepara un elemento de entrada para la interfaz de la barra de búsqueda.	React.RefObject<HTMLInputElement>	TRUE	TRUE	
				3	Renderiza el componente.	<SearchBar {...enabledProps} />	Valor TRUE en contexto booleano	Valor TRUE en contexto booleano	
TS015	Renders the IconButton component	TC001	Habilita los accesorios del botón de iconos.	1	Escribe la interfaz IconButtonProps.	-	-	-	PASS
				2	Establece el valor del texto e icono del botón, así como su ID.	'takeNote', Camera, TestID.ICON_BUTTON	TRUE	TRUE	
				3	Renderiza el componente.	<IconButton {...enabledProps} />	Valor TRUE en contexto booleano	Valor TRUE en contexto booleano	
TS016	Renders the NoteListButton component	TC001	Habilita los accesorios del botón que se encarga de la lista de notas.	1	Escribe la interfaz NoteListButtonProps.	-	-	-	PASS
				2	Establece el valor de su label y su ID.	'Test', TestID.EMPTY_TRASH_BUTTON	TRUE	TRUE	
				3	Renderiza el componente.	<NoteListButton {...enabledProps} />	Valor TRUE en contexto booleano	Valor TRUE en contexto booleano	
TS017	Should pass saved cookies to locals	TC001	Las cookies se almacenan localmente en la memoria caché	1	Se prepara un objeto requestMock con un token de acceso en su propiedad cookies.	'test access token'	TRUE	TRUE	PASS
				2	Verifica el token de acceso y lo escribe en un objeto responseMock.	requestMock, responseMock, nextMock	-	-	
				3	Almacena las cookies.	-	-	-	
TS018	Renders the EmptyEditor component	TC001	Habilita los accesorios del editor vacío.	1	Renderiza el componente React.JSX.EmptyEditor	<EmptyEditor />	Valor TRUE en contexto booleano	Valor TRUE en contexto booleano	PASS
TS019	Renders the AddCategoryForm component	TC001	Habilita los accesorios del formulario que se encarga de añadir categorías.	1	Escribe la interfaz AddCategoryFormProps.	-	-	-	PASS
				2	Establece su nombre, ID y algunas funciones mock para los handlers.	'Category-id', 'Category-id', TestID.NEW_CATEGORY_INPUT, jsf.fn	TRUE	TRUE	
				3	Renderiza el componente.	<AddCategoryForm {...enabledProps} />	Valor TRUE en contexto booleano	Valor TRUE en contexto booleano	
TS020	Renders the CollapseCategoryButton component	TC001	Habilita los accesorios del botón que se encarga de colapsar la categoría.	1	Escribe la interfaz CollapseCategoryListButton.	-	-	-	PASS
				2	En la interfaz los booleanos deben estar en true.	true, true	TRUE	TRUE	
				3	Renderizar la interfaz.	<CollapseCategoryListButton {...enabledProps} />	Valor TRUE en contexto booleano	Valor TRUE en contexto booleano	
TS021	Renders the FolderOption component	TC001	Habilita los accesorios de apoyo de carpetas.	1	Escribe la interfaz FolderOptionProps.	-	-	-	PASS
				2	En la interfaz se le tiene que declarar un booleano en true y le pasamos el valor CATEGORY del objeto Folder.	true, Folder.CATEGORY	TRUE	TRUE	
				3	Renderizar la interfaz	<FolderOption {...enabledProps} />	Valor TRUE en contexto booleano	Valor TRUE en contexto booleano	

TS022	Renders the ScratchpadOption component	TC001	Habilita los accesorios de apoyo para la opción de bloc de notas	1 Escribe la interfaz ScratchpadOptionProps.	-	-	-	PASS
				2 En la interfaz se declara el booleano en true y se le pasa un objeto Folder vacío.	true, Folder	TRUE	TRUE	
				3 Renderizar la interfaz.	<ScratchpadOption {...enabledProps} />	Valor TRUE en contexto booleano	Valor TRUE en contexto booleano	
TS023	Renders the Switch component	TC001	Habilita los accesorios del componente switch.	1 Escribe la interfaz SwitchProps.	-	-	-	PASS
				2 En la interfaz se declara el booleano en false y se le pasa una función toggle vacía.	false, empty	TRUE	TRUE	
				3 Renderizar la interfaz.	<Switch {...enabledProps} />	Valor TRUE en contexto booleano	Valor TRUE en contexto booleano	
TS024	Renders the IconButtonUploader component	TC001	Habilita los accesorios del cargador del botón de icono.	1 Escribe la interfaz IconButtonUploaderProps.	-	-	-	PASS
				2 En la interfaz se declara dos cadenas, en la variable icono se le pasa la función cámara y en toggle se le pasa el objeto File.	takeNote', 'takeNote', Camera.File	TRUE	TRUE	
				3 Renderizar la interfaz.	<IconButtonUploader {...enabledProps} />	Valor TRUE en contexto booleano	Valor TRUE en contexto booleano	
TS025	Renders the ActionButton component	TC001	Habilita los accesorios del botón de acción.	1 Escribir la interfaz ActionButtonProps.	-	-	-	PASS
				2 En la interfaz se declara dos cadenas, en la variable icono se le pasa la función Cámara y en handler se le agrega el MouseEventHandler.	text', Camera	TRUE	TRUE	
				3 Renderizar la interfaz.	<ActionButton {...enabledProps} />	Valor TRUE en contexto booleano	Valor TRUE en contexto booleano	
TS026	Renders the AddCategoryButton component	TC001	Habilita los accesorios de la interfaz que se encarga de agregar botones.	1 Escribe la interfaz AddCategoryButtonProps.	-	-	-	PASS
				2 En la interfaz se establece en el handler un parámetro booleano.	handler: (boolean)	TRUE	TRUE	
				3 Renderizar la interfaz.	<AddCategoryButton {...enabledProps} />	Valor TRUE en contexto booleano	Valor TRUE en contexto booleano	
TS027	Should open settings menu	TC001	Abrir el menú de configuración.	1 Visita la carpeta /app		-	-	PASS
				2 Espera que se realice la igualdad para luego afirmarlo.		TRUE	TRUE	
				3 Después de afirmar la respuesta de espera				
				4 Busca la función de dar click en button	{name: '/settings/!}	Valor TRUE en contexto booleano	Valor TRUE en contexto booleano	
TS028	Should close settings menu on clicking X	TC001	Debería cerrar el menú de configuración al hacer clic en X	1 Busca la función de darle click al button	{name: '/settings/!}	-	-	PASS
TS029	Should close settings menu on clicking outside of window	TC001	Debería cerrar el menú de configuración al hacer clic fuera de la ventana	1 Busca la función de darle click al button	{name: '/settings/!}	-	-	PASS
TS030	Should toggle preferences: active line highlight [off]	TC001	Debería alternar las preferencias: resaltado de línea activa [desactivado]	1 Busca el rol de button name:/setting/! le da click	name:/setting/!	-	-	PASS
				2 Desactiva la línea activa, alternando las alternativas		TRUE	TRUE	
TS031	should hide the category list on click of category	TC001	Debería alternar las preferencias: resaltado de línea activa [desactivado]	1 Busca el rol de button name:/setting/! le da click	name:/setting/!	-	-	PASS
				2 Desactiva la línea activa, alternando las alternativas		TRUE	TRUE	
TS032	should show category list on add new category	TC001	Debería alternar las preferencias: resaltado de línea activa [desactivado]	1 Busca el rol de button name:/setting/! le da click	name:/setting/!	-	-	PASS
				2 Desactiva la línea activa, alternando las alternativas		TRUE	TRUE	
TS033	creates a new category with the current time	TC001	Debería alternar las preferencias: resaltado de línea activa [desactivado]	1 Busca el rol de button name:/setting/! le da click	name:/setting/!	-	-	PASS
				2 Desactiva la línea activa, alternando las alternativas		TRUE	TRUE	
TS034	should add a note to new category	TC001	Debería alternar las preferencias: resaltado de línea activa [desactivado]	1 Busca el rol de button name:/setting/! le da click	name:/setting/!	-	-	PASS
				2 Desactiva la línea activa, alternando las alternativas		TRUE	TRUE	
TS035	should rename existing category after defocusing edit state	TC001	Debería alternar las preferencias: resaltado de línea activa [desactivado]	1 Busca el rol de button name:/setting/! le da click	name:/setting/!	-	-	PASS
				2 Desactiva la línea activa, alternando las alternativas		TRUE	TRUE	
TS036	should change category order	TC001	Debería alternar las preferencias: resaltado de línea activa [desactivado]	1 Busca el rol de button name:/setting/! le da click	name:/setting/!	-	-	PASS
				2 Desactiva la línea activa, alternando las alternativas		TRUE	TRUE	
TS037	should add a note to new category	TC001	Debería alternar las preferencias: resaltado de línea activa [desactivado]	1 Busca el rol de button name:/setting/! le da click	name:/setting/!	-	-	PASS
				2 Desactiva la línea activa, alternando las alternativas		TRUE	TRUE	