

Número de test: Demo con StakeHolder

Fecha: 16/04/2018

Nombre del participante: David, Manuel, Daniel

Heurística	Descripción	Comentarios
Visibilidad del estado del sistema	El sistema siempre debería mantener informados a los usuarios de lo que está ocurriendo, a través de retroalimentación apropiada dentro de un tiempo razonable.	
Relación entre el sistema y el mundo real	El sistema debería hablar el lenguaje de los usuarios mediante palabras, frases y conceptos que sean familiares al usuario, más que con términos relacionados con el sistema. Seguir las convenciones del mundo real, haciendo que la información aparezca en un orden natural y lógico.	Ellos registran la unidad al registrar el material en el catálogo, en lugar de registrarlo en cada compra. Cambiar el nombre ID a Número Orden de compra (OC), (OT), (OV), (AI). Agregar estatus a orden de venta. En detalle de materia prima, tiene que existir tanto orden de venta, de compra y de trabajo y ajuste inventario (Historia de movimiento). Agregar un código a las recetas. Quiere ver código en TODOS lados. Checar costo unitario por orden de compra para ver qué proveedor le vende más barato. Agregar materia prima al producto terminado, para agregar material de empaque.

Control y libertad del usuario	Hay ocasiones en que los usuarios elegirán las funciones del sistema por error y necesitarán una “salida de emergencia” claramente marcada para dejar el estado no deseado al que accedieron, sin tener que pasar por una serie de pasos. Se deben apoyar las funciones de deshacer y rehacer.	
Consistencia y estándares	Los usuarios no deberían cuestionarse si acciones, situaciones o palabras diferentes significan en realidad la misma cosa; siga las convenciones establecidas.	El campo de duración no le agrega valor al cliente. Algunos nombres en el nav bar son diferentes al nombre del encabezado de la página a la que te llevan. Quiere recetas en TODO, no lo quiere separado de receta a producto semiterminado Recetas se debe llamar “Catalogo producto semiterminado”
Prevención de errores	Mucho mejor que un buen diseño de mensajes de error es realizar un diseño cuidadoso que prevenga la ocurrencia de problemas.	
Reconocimiento antes que recuerdo	se deben hacer visibles los objetos, acciones y opciones, El usuario no tendría que recordar la información que se le da en una parte del proceso, para seguir adelante. Las instrucciones para el uso del sistema deben estar a la vista o ser fácilmente recuperables cuando sea necesario.	No sabe la unidad de la cantidad que se necesita a la hora de hacer una receta. El costo del material es por kilo no por orden de compra. Cambiar nombre de “Agregar Orden” a “Agregar Orden de Trabajo”. Quitarle el nombre “Catálogo” a todo lo que

		<p>sea inventario, en este caso “Catálogo inventario materia prima”.</p> <p>Una orden de trabajo se encarga de crear todo, producto semiterminado y terminado tambien.</p> <p>Nav Bar</p> <ul style="list-style-type: none"> ● Catálogos <ul style="list-style-type: none"> ○ Clientes ○ Proveedores ○ Unidades ○ Materia prima ○ Producto termi ● Órdenes ● Inventario <ul style="list-style-type: none"> ○ Materia prima ○ Producto semi ○ Producto termi ● Ajuste Inventario <p>cambiar “Seleccionar receta” a “Seleccionar producto semiterminado”</p>
Flexibilidad y eficiencia de uso	<p>la presencia de aceleradores, que no son vistos por los usuarios novatos, puede ofrecer una interacción más rápida a los usuarios expertos que la que el sistema puede proveer a los usuarios de todo tipo. Se debe permitir que los usuarios adapte el sistema para usos frecuentes.</p>	

Estética y diseño minimalista	los diálogos no deben contener información que es irrelevante o poco usada. Cada unidad extra de información en un diálogo, compite con las unidades de información relevante y disminuye su visibilidad relativa.	
Ayudar a los usuarios a reconocer, diagnosticar y recuperarse de errores	los mensajes de error se deben entregar en un lenguaje claro y simple, indicando en forma precisa el problema y sugerir una solución constructiva al problema.	
Ayuda y documentación	incluso en los casos en que el sistema pueda ser usado sin documentación, podría ser necesario ofrecer ayuda y documentación. Dicha información debería ser fácil de buscar, estar enfocada en las tareas del usuario, con una lista concreta de pasos a desarrollar y no ser demasiado extensa.	

- **Un cotizador**