ANEXOS

Especi_caci_on de Requisitos seg_un el est_andar de IEEE 830

IEEE Std. 830-1998

22 de Octubre de 2008

Resumen

Este documento presenta, en castellano, el formato de Especi_caci_on de Requisitos Software (ERS) seg_un la _ultima versi_on del est_andar IEEE 830. Seg_un IEEE, un buen Documento de Requisitos, pese a no ser obligatorio que siga estrictamente la organizaci_on y el formato dados en el est_andar 830, s__ deber_a incluir, de una forma o de otra, toda la informaci_on presentada en dicho est_andar. El est_andar de IEEE 830 no est_a libre de defectos ni de prejuicios, y por ello ha sido justamente criticado por m_ultiples autores y desde m_ultiples puntos de vista, lleg_andose a cuestionar incluso si es realmente un est_andar en el sentido habitual que tiene el t_ermino en otras ingenier__as. El presente documento no pretende pronunciarse ni a favor ni en contra de unos u otros: tan s_olo reproduce, con prop_ositos fundamentalmente docentes, c_omo se organizar__a un Documento de Requisitos seg_un el est_andar IEEE 830.

_INDICE 2

_Indice

1. Introducci_on 3
1.1. Prop_osito 3
1.2. _Ambito del Sistema 3
1.3. De_niciones, Acr_onimos y Abreviaturas 3
1.4. Referencias 3
1.5. Visi_on General del Documento 4
2. Descripci_on General 4
2.1. Perspectiva del Producto 4
2.2. Funciones del Producto 4
2.3. Caracter__sticas de los Usuarios 5
2.4. Restricciones 5
2.5. Suposiciones y Dependencias 5
2.6. Requisitos Futuros 6
3. Requisitos Espec___cos 6
3.1. Interfaces Externas 7
3.2. Funciones 7
3.3. Requisitos de Rendimiento 9
3.4. Restricciones de Dise~no 9
3.5. Atributos del Sistema 9
3.6. Otros Requisitos 9
4. Ap_endices 9

1 INTRODUCCI _ON 3

1. Introducci_on

En esta secci_on se proporcionar_a una introducci_on a todo el documento de Especi_caci_on de Requisitos Software(ERS). Consta de varias subsecciones: prop_osito, ambito del sistema, de_niciones, referencias y visi_on general del

documento.

1.1. Prop_osito

En esta subsecci_on se de_nir_a el prop_osito del documento ERS y se especi _car_a a qui_en va dirigido el documento.

1.2. _Ambito del Sistema

En esta subsecci_on:
Se podr_a dar un nombre al futuro sistema (p.ej. MiSistema)
Se explicar_a lo que el sistema har_a y lo que no har_a.
Se describir_an los bene_cios, objetivos y metas que se espera alcanzar con el futuro sistema.
Se referenciar_an todos aquellos documentos de nivel superior (p.e. en Ingenier __a de Sistemas, que incluyen Hardware y Software, deber__a mantenerse la consistencia con el documento de especi_caci_on de requisitos globales del sistema, si existe).

1.3. De_niciones, Acr_onimos y Abreviaturas

En esta subsecci_on se de_nir_an todos los t_erminos, acr_onimos y abreviaturas utilizadas en la ERS.

1.4. Referencias

En esta subsecci_on se mostrar_a una lista completa de todos los documentos referenciados en la ERS.
2 DESCRIPCI _ON GENERAL 4

1.5. Visi_on General del Documento

Esta subsecci_on describe brevemente los contenidos y la organizaci_on del resto de la ERS.

2. Descripci_on General

En esta secci_on se describen todos aquellos factores que afectan al producto y a sus requisitos. No se describen los requisitos, sino su contexto. Esto permitir_a de_nir con detalle los requisitos en la secci_on 3, haciendo que sean m_as f_aciles de entender.
Normalmente, esta secci_on consta de las siguientes subsecciones: Perspectiva del producto, funciones del producto, caracter__sticas de los usuarios, restricciones, factores que se asumen y futuros requisitos.

2.1. Perspectiva del Producto

Esta subsecci_on debe relacionar el futuro sistema (producto software) con otros productos. Si el producto es totalemente independiente de otros productos, tambi_en debe especi_carse aqu__. Si la ERS de_ne un producto que es parte de un sistema mayor, esta subsecci_on relacionar_a los requisitos del sistema mayor con la funcionalidad del producto descrito en la ERS, y se identi_car_an las interfaces entre el producto mayor y el producto aqu__ descrito. Se recomienda utilizar diagramas de bloques.

2.2. Funciones del Producto

En esta subsecci_on de la ERS se mostrar_a un resumen, a grandes rasgos, de las funciones del futuro sistema. Por ejemplo, en una ERS para un programa de contabilidad, esta subsecci_on mostrar_a que el sistema soportar_a el mantenimiento de cuentas, mostrar_a el estado de las cuentas y facilitar_a la facturaci_on, sin mencionar el enorme detalle que cada una de estas funciones requiere.

Las funciones deber_an mostrarse de forma organizada, y pueden utilizarse gr_a_cos, siempre y cuando dichos gr_a_cos reejen las relaciones entre funciones y no el dise~no del sistema.
2 DESCRIPCI _ON GENERAL 5
2.3. Caracter__sticas de los Usuarios
Esta subsecci_on describir_a las caracter__sticas generales de los usuarios del producto, incluyendo nivel educacional, experiencia y experiencia t_ecnica.
2.4. Restricciones
Esta subsecci_on describir_a aquellas limitaciones que se imponen sobre los desarrolladores del producto
Pol__ticas de la empresa
Limitaciones del hardware
Interfaces con otras aplicaciones
Operaciones paralelas
Funciones de auditor__a
Funciones de control
Lenguaje(s) de programacion
Protocolos de comunicaci_on
Requisitos de habilidad
Criticalidad de la aplicaci_on
Consideraciones acerca de la seguridad
2.5. Suposiciones y Dependencias
Esta subsecci_on de la ERS describir_a aquellos factores que, si cambian, pueden afectar a los requisitos. Por ejemplo, los requisitos pueden presuponer una cierta organizaci_on de ciertas unidades de la empresa, o pueden presuponer que el sistema correr_a sobre cierto sistema operativo. Si cambian dichos detalles en la organizaci_on de la empresa, o si cambian ciertos detalles t_ecnicos, como el sistema operativo, puede ser necesario revisar y cambiar los requisitos.
3 REQUISITOS ESPEC_IFICOS 6
2.6. Requisitos Futuros
Esta subsecci_on esbozar_a futuras mejoras al sistema, que podr_an analizarse e implementarse en un futuro.
3. Requisitos Espec___cos
Esta secci_on contiene los requisitos a un nivel de detalle su_ciente como para permitir a los dise~nadores dise~nar un sistema que satisfaga estos requisitos, y que permita al equipo de pruebas plani_car y realizar las pruebas que demuestren si el sistema satisface, o no, los requisitos. Todo requisito aqu__ especi _cado describir_a comportamientos externos del sistema, perceptibles por parte de los usuarios, operadores y otros sistemas. Esta es la secci_on m_as larga e importante de la ERS. Deber_an aplicarse los siguientes principios:
El documento deber__a ser perfectamente legible por personas de muy distintas formaciones e intereses.
Deber_an referenciarse aquellos documentos relevantes que poseen alguna inuencia sobre los requisitos.
Todo requisito deber_a ser un__vocamente identi_cable mediante alg_un c_odigo o sistema de numeraci_on adecuado.

Lo ideal, aunque en la pr_actica no siempre realizable, es que los requisitos posean las siguientes caracter__sticas:

_ Correccion: La ERS es correcta si y s_olo si todo requisito que _gura aqu__ (y que ser_a implementado en el sistema) reeja alguna necesidad real. La correcci_on de la ERS implica que el sistema implementado ser_a el sistema deseado.

_ No ambiguos: Cada requisito tiene una sola interpretaci_on. Para eliminar la ambig• uedad inherente a los requisitos expresados en lenguaje natural, se deber_an utilizar gr_a_cos o notaciones formales. En el caso de utilizar t_erminos que, habitualmente, poseen m_as de una interpretaci_on, se de_nir_an con precisi_on en el glosario.

_ Completos: Todos los requisitos relevantes han sido incluidos en la ERS. Conviene incluir todas las posibles respuestas del sistema a los datos de entrada, tanto v_alidos como no v_alidos.

_ Consistentes: Los requisitos no pueden ser contradictorios. Un conjunto de requisitos contradictorio no es implementable.

_ Clasi_cados: Normalmente, no todos los requisitos son igual de importantes. Los requisitos pueden clasi_carse por importancia (esenciales, condicionales u opcionales) o por estabilidad (cambios que se espera que afecten al requisito). Esto sirve, ante todo, para no emplear excesivos recursos en implementar requisitos no esenciales.

_ Veri_cables: La ERS es veri_cable si y s_olo si todos sus requisitos son veri_cables. Un requisito es veri_cable (testeable) si existe un proceso _nito y no costoso para demostrar que el sistema cumple con el requisito. Un requisito ambiguo no es, en general, veri_-cable. Expresiones como a veces, bien, adecuado, etc. introducen ambig• uedad en los requisitos. Requisitos como \en caso de accidente la nube t_oxica no se extender_a m_as all_a de 25Km" no es veri_cable por el alto costo que conlleva.

_ Modi_cables: La ERS es modi_cable si y s_olo si se encuentra estructurada de forma que los cambios a los requisitos pueden realizarse de forma f_acil, completa y consistente. La utilizaci_on de herramientas autom_aticas de gesti_on de requisitos (por ejemplo RequisitePro o Doors) facilitan enormemente esta tarea.

_ Trazables: La ERS es trazable si se conoce el origen de cada requisito y se facilita la referencia de cada requisito a los componentes del dise~no y de la implementaci_on. La trazabilidad hacia atr_as indica el origen (documento, persona, etc.) de cada requisito. La trazabilidad hacia delante de un requisito R indica qu_e componentes del sistema son los que realizan el requisito R.

3.1. Interfaces Externas

Se describir_an los requisitos que afecten a la interfaz de usuario, interfaz con otros sistemas (hardware y software) e interfaces de comunicaciones.

3.2. Funciones

Esta subsecci_on (quiz_a la m_as larga del documento) deber_a especi_car todas aquellas acciones (funciones) que deber_a llevar a cabo el software. Nor3

malmente (aunque no siempre), son aquellas acciones expresables como \el sistema deber_a . . . ". Si se considera necesario, podr_an utilizarse notaciones gr_a_cas y tablas, pero siempre supeditadas al lenguaje natural, y no al rev_es.
Es importante tener en cuenta que, en 1983, el Est_andar de IEEE 830 establec__a que las funciones deber__an expresarse como una jerarqu__a funcional (en paralelo con los DFDs propuestos por el an_alisis estructurado). Pero el Est_andar de IEEE 830, en sus _ultimas versiones, ya permite organizar esta subsecci_on de m_ultiples formas, y sugiere, entre otras, las siguientes:
Por tipos de usuario: Distintos usuarios poseen distintos requisitos. Para cada clase de usuario que exista en la organizaci_on, se especi_car_an los requisitos funcionales que le afecten o tengan mayor relaci_on con sus tareas.
Por objetos: Los objetos son entidades del mundo real que ser_an reejadas en el sistema. Para cada objeto, se detallar_an sus atributos y sus funciones. Los objetos pueden agruparse en clases. Esta organizaci_on de la ERS no quiere decir que el dise~no del sistema siga el paradigma de Orientaci_on a Objetos.
Por objetivos: Un objetivo es un servicio que se desea que ofrezca el sistema y que requiere una determinada entrada para obtener su resultado. Para cada objetivo o subobjetivo que se persiga con el sistema, se detallar_an las funciones que permitan llevarlo a cabo.
Por est__mulos: Se especi_car_an los posibles est__mulos que recibe el sistema y las funciones relacionadas con dicho est__mulo.
Por jerarqu__a funcional: Si ninguna de las anteriores alternativas resulta de ayuda, la funcionalidad del sistema se especi_car_a como una jerarqu __a de funciones que comparten entradas, salidas o datos internos. Se detallar_an las funciones (entrada, proceso, salida) y las subfunciones del sistema. Esto no implica que el dise~no del sistema deba realizarse seg_un el paradigma de Dise~no Estructurado.
Para organizar esta subsecci_on de la ERS se elegir_a alguna de las anteriores alternativas, o incluso alguna otra que se considere m_as conveniente. Deber_a, eso s__, justi_carse el porqu_e de tal elecci_on.

3.3. Requisitos de Rendimiento

Se detallar_an los requisitos relacionados con la carga que se espera tenga que soportar el sistema. Por ejemplo, el n_umero de terminales, el n_umero esperado de usuarios simultaneamente conectados, n_umero de transacciones por segundo que deber_a soportar el sistema, etc.
Tambi_en, si es necesario, se especi_car_an lo requisitos de datos, es decir, aquellos requisitos que afecten a la informaci_on que se guardar_a en la base de datos. Por ejemplo, la frecuencia de uso, las capacidades de acceso y la cantidad de registros que se espera almacenar (decenas, cientos, miles o millones).

3.4. Restricciones de Dise~no

Todo aquello que restrinja las decisiones relativas al dise~no de la aplicaci _on: Restricciones de otros est_andares, limitaciones del hardware, etc.

3.5. Atributos del Sistema

Se detallar_an los atributos de calidad (las \ilities") del sistema: Fiabilidad, mantenibilidad, portabilidad, y, muy importante, la seguridad. Deber_a especi _carse qu_e tipos de usuario est_an autorizados, o no, a realizar ciertas tareas, y c_omo se implementar_an los mecanismos de seguridad (por ejemplo, por medio de un login y una password).

3.6. Otros Requisitos

Cualquier otro requisito que no encaje en otra secci_on.

4. Ap_endices

Pueden contener todo tipo de informaci_on relevante para la ERS pero que, propiamente, no forme parte de la ERS. Por ejemplo:

1. Formatos de entrada/salida de datos, por pantalla o en listados.
2. Resultados de an_alisis de costes.
3. Restricciones acerca del lenguaje de programaci_on.

2016 Trends in NoSQL: Next Stages of Development

By Charles Roe / January 19, 2016 / 0 Comments

The myriad NoSQL database offerings in the marketplace today are testament to the fact that such systems, along with their diverse architectures, are here to stay. Such a statement was not the case only a few years ago when such "trends" were still the purview of industry thought leaders, innovative startups, and apocalyptic harbingers proclaiming the end of the SQL-relational database world. Data Management and its many associated technologies and practices – from Data Governance to Data Modeling, Enterprise Information Management to Customer Relationship Management – have entered a new age. There is no denying that any longer. Big Data, no matter what definition someone chooses to use, has changed everything, and with it the growth of NoSQL platforms and the consequences of such adoptions at the enterprise level will push the envelope on how data is collected, stored, and analyzed far into the future.

In a recent DATAVERSITY® interview with Dr. Vladimir Bacvanski, the co-founder of SciSpike, we discussed such industry-wide changes and where NoSQL is going in 2016. Dr. Bacvanski commented:

"The main three trends right now happening in the NoSQL and Big Data space are the development of microservices in the Cloud, which lead to polyglot databases; the growth of the Internet of Things that is pushing many companies into Big Data (SPARK is increasingly important for this trend, since it is providing both batch and streaming potential to many companies); and the final thing is the fact that Big Data and NoSQL have become the new normal. There is now a recognition that, in certain areas, NoSQL is better than relational systems, while in others relational databases are better."

1. **Microservices and Polyglot Persistence**

The term "microservice architecture," or just "microservices" for short, is a term that has been discussed since the early 2000s, but it really started to gain industry popularity around 2013-14. Martin Fowler and James Lewis wrote a Microservices Resource Guide in mid-2014, and their work (along with many others) has helped push the swift growth of microservices to a point that they are becoming a necessity for any enterprise that wants low cost, agile, massively scalable software development and Data Management options outside of traditional, monolithic databases. According to Dr. Bacvanski, microservices allow for "really nice modularity. They are typically deployed on many different machines,

so the application is broken into many small services that run in the Cloud (either public or private), with so many benefits." A short list of those benefits include:

- The ability to use many data models for different services, rather than one single data model that the data must be forced into
- Small services and databases, which allow for powerful distributed and massively scalable architectures
- A natural platform for deployment in the Cloud
- The possibility of Agile Software Development. A single developer can rewrite the code for a microservice in two weeks. Compare that to gargantuan databases with millions of lines of code
- Microservices can be planned around modular business needs and single operations, thus aiding in faster deployment and, in theory, lowering development costs

One case study for the many benefits of running microservices is an eCommerce site, Dr. Bacvanski said. Rather than employing a huge relational database, or even a massive single NoSQL system, an eCommerce company can use microservices to create greater speed and flexibility within all their systems. They can serve their product catalog with a document database. "Since descriptions of different products vary, you would want to accommodate that," he said. Then they could implement a key-value store for their user sessions, while a graph store would probably be best for their internal supply chains and relationships between producers, suppliers, and so on, because graph stores are built to handle such relationships. And finally, for click analytics they could dump all that data into Hadoop or SPARK and store it for later analysis.

The disengagement of all these processes into separate data stores and services is also referred to as polyglot persistence, an impressive sounding term for a concept with complex ramifications for data professionals. Where once a data modeler, data architect, software developer, system administrator or any other person working within the data stream (including business-minded professionals) only had to worry about one type of system – the RDBMS – they now have to learn multiple platforms.

"It is no longer just a 'this or that' option," remarked Dr. Bacvanski. "Instead, there are many options and new challenges with polyglot persistence and microservices. You now have many more databases to deal with, much more training to be done. But, the overall execution and performance tend to be much better. So for most companies, the payoffs are well worth the time and effort put into implementing such options."

2. **The Internet of Things Forces the Issue**

The rapid growth of the Internet of Things (IoT) has been nothing short of remarkable. “These systems produce tremendous amounts of data,” said Dr. Bacvanski. “More than anything we’ve seen so far, and it’s still small compared to the amount of data these sensors *can* produce.” Enterprises across a multitude of industries, including (but not limited to) healthcare, manufacturing, energy, transportation and travel, retail, and so many others are just beginning to comprehend the benefits of collecting and analyzing the sensor data available to them. Many are only beginning to understand what they can do with such sensors and so they are installing them everywhere.

Only a few years ago there were many companies that claimed they didn’t need to worry about Big Data or NoSQL because their industries didn’t necessitate such systems. Dr. Bacvanski gave the example of auto insurance: “They told us the only events that happen for them is that people buy cars, they collide, they have insurance claims, we investigate those claims, we pay them or we don’t. There are not that many things that change.”

These companies are beginning to see the benefits of IoT, of sensors, and of tracking exactly what their customers are doing. They’ve figured out that they can put sensors in cars and get reports on what kind of a driver their customer is, how they accelerate, how they brake, how fast they go, what sort of risky driving they do. This type of data collection and analysis allows them to get a much clearer picture of what is going on and possibly adjust their rates. “So now, all of a sudden, these companies are in the Big Data business,” he said. “The old argument that only huge companies like Yahoo! or Google needed Big Data – this was a common argument – is no longer valid. Everyone has to deal with Big Data.”

The same is true in retail and healthcare. Brick-and-mortar and online retail stores both want to maximize their customers’ experiences to drive more sales. If a customer walks into a store, traditional retailers want to be able to track their movements, buying practices, and time spent at various points in the store in real time. They want to be able to recommend that someone buy one product while they’re looking at another. The same is true of the online experience.

In healthcare, physicians and nurses want to be able to track a given patient’s vital signs throughout the day, rather than checking on them only when they come in for a visit. They want to be able to collect all that data to give better recommendations to that one patient, but also analyze millions of patients with similar symptoms to get more comprehensive health plans to all their patients.

Even smaller enterprises want to leverage their social media, click stream, and other types of unstructured data, including data collected from real or virtual sensors. The amounts might not be at the petabyte scales of some of the monoliths in the Big Data sphere, but nonetheless, there are many benefits for moving into such systems.

"Even if they don't have much data to necessitate Hadoop or SPARK," Dr. Bacvanski said, "applying these technologies can lead to cheaper IT. Instead of buying a $2 million system, you can just dump this data onto a smaller cluster and see significant decreases in data and analytics costs."

3. **NoSQL and Big Data are Here to Stay**

The adoption of NoSQL systems has been slow for many, but these systems have gained so much traction that fighting the tide is no longer possible, or in most instances, no longer a wise business decision. "The growing acceptance of these systems has become significant," said Dr. Bacvanski. "From the beginning it wasn't 'data people' pushing for these technologies in most companies, but instead it was software developers." Developers require agility, and so there have long been internal struggles between data people and software people. Such struggles are well-documented in conference presentations, white papers, and blogs. Data people want consistency; they want their data assets safe and reliable. Software developers want agility; they want their projects done quickly and in iterations (at least in Agile camps).

"They would like to have free hands to change the databases, change the schema, do whatever they want with the data," he said. "That led, in many organizations, to developers adopting NoSQL stores just to move faster."

Such changes have, in terms of data and software development history, happened quite suddenly. NoSQL systems inevitably led to the creation of many silos for organizations, where there are developers on one side just "picking up a database and completing a project on their own," while data people push back against such rapid changes on the other side. Such silos have been created in companies that lack strong Data Governance. "There is a conflict between Data Governance people and developers in terms of adopting such technologies," he remarked. "Their points of view don't often match." Or we should say, their points of view didn't used to match. They are now beginning to meet somewhere in the middle.

Luckily, the early antagonism caused by the disparate adoption of NoSQL systems into the larger Enterprise Information Management structures of many organizations has lessened, and in some cases,

that antagonism has become a point of congruence between both sides. Such changes are due to many factors:

- Better education on the distinctive value of NoSQL stores
- Better understanding of the necessity of teamwork in terms of Data Governance and software development
- The creation of tools that provide SQL-like capabilities within NoSQL platforms
- Numerous case studies that have clearly demonstrated the benefits of NoSQL adoption
- A clear understanding that relational systems are still important and have certain operations that will continue to be needed going forward

"The data side has realized that they need to take care of these new systems somehow," said Dr. Bacvanski. "And the developer side understands that these new systems certainly need better Data Governance. The benefits are finally much clearer to everyone." Such claims are certainly not true everywhere. Many companies are still dealing with these struggles and will continue to far into the future.

NoSQL, Big Data, the Internet of Things, microservices, Hadoop, SPARK, and an ever-growing assortment of new trends and technologies are here to stay. "They are no longer viewed as antagonists to relational systems, but as systems with special, unique capabilities. There is a peaceful co-existence forming, but it will still take some time. 2016 will be a big year for such changes."

Come meet Dr. Vladimir Bacvanski at the Enterprise Data World 2016 Conference. He will be doing a Tutorial on Thursday, April 21st at 1.15pm titled: "Building a Modern Big Data Enterprise: Hadoop, Spark, and Beyond"

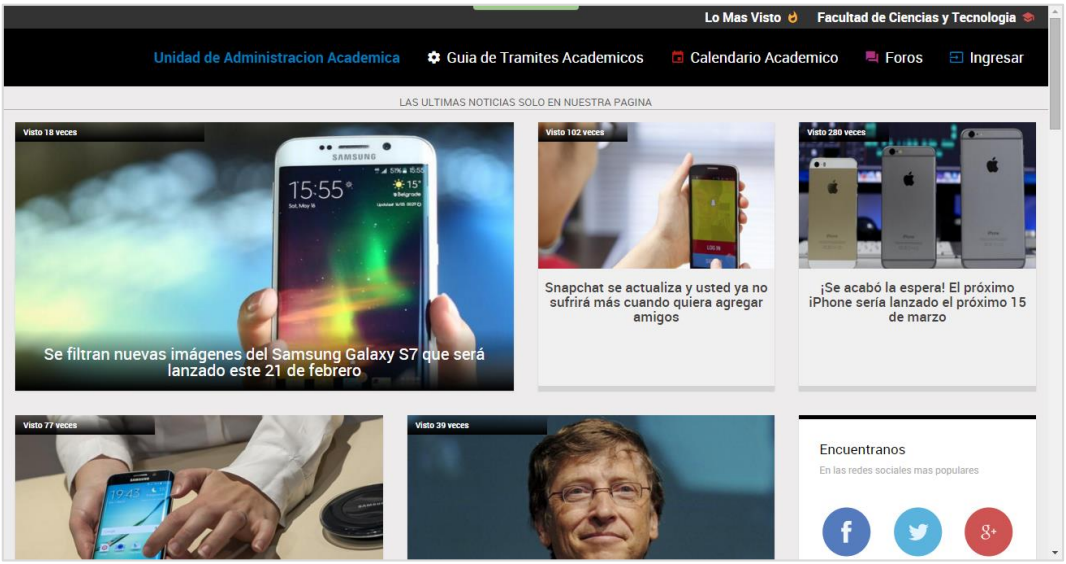


- By: Charles Roe

- Filed Under: Big Data Articles, Big Data News, Articles, & Education, Data Articles | Data Science, Business Intelligence, & More, Data Education, Education Resources For Use & Management of Data, NoSQL Articles, NoSQL News, Articles, & Education
- Tagged With: big data, internet of things, microservices, NoSQL News, Articles, & Education, polyglot persistence

MANUAL DE USUARIO

1. Página principal



En esta pantalla usted podra realizar las siguientes tareas:

En la parte superior esta el enlace “Lo Mas Visto”, que ordena las noticias que se le muestran por el numero de visitas que tengan.

El enlace “Facultad de ciencias y tecnologia”, lo llevara a la pagina oficial de la facultad

Por debajo de estos enlaces se encuentra el enlace “Guia de tramites academicos”, la cual lo envia donde estan las guias de los tramites que realiza UNADA, para que usted sepa como hacerlos de manera mas rapida.

El enlace “Calendario Academico” lo lleva a lo que son los eventos del mes en la facultad, como ser fecha de programaciones, cursos de veranos, etc.

El enlace foros lo llevara a los que son el listado de foros en la facultad, con las preguntas que tengan los usuarios, las cuales entre toda la comunidad se responde.

Por ultimo el enlace “Ingresar” nos llevara a la pantalla de ingreso al panel de administracion del sistema.

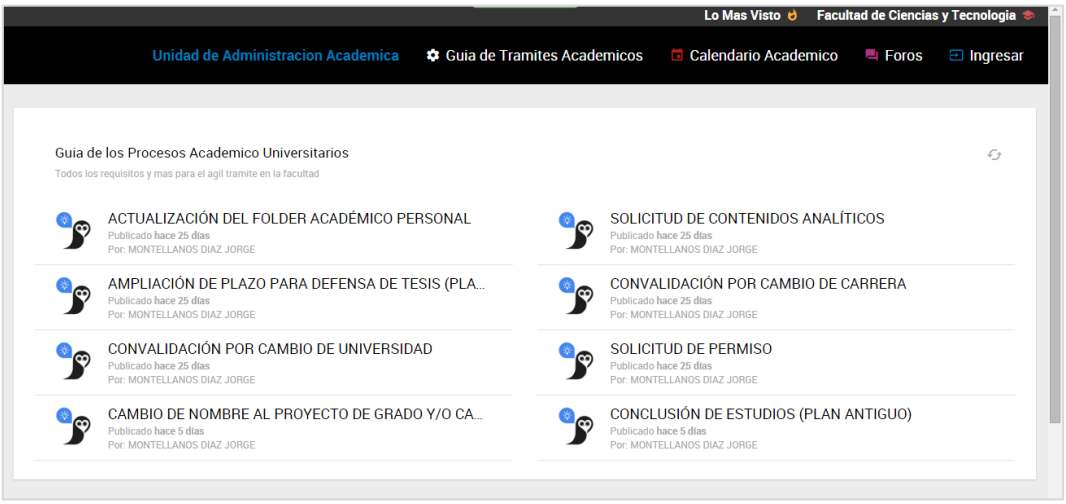
Por debajo de estas opciones, tenemos lo que son las noticias relevantes a la facultad, a las cuales se puede ingresar y verlas a detalles, pero eso lo explicamos en la siguiente pantalla.

2. Interfaz 1.1 Ver noticia



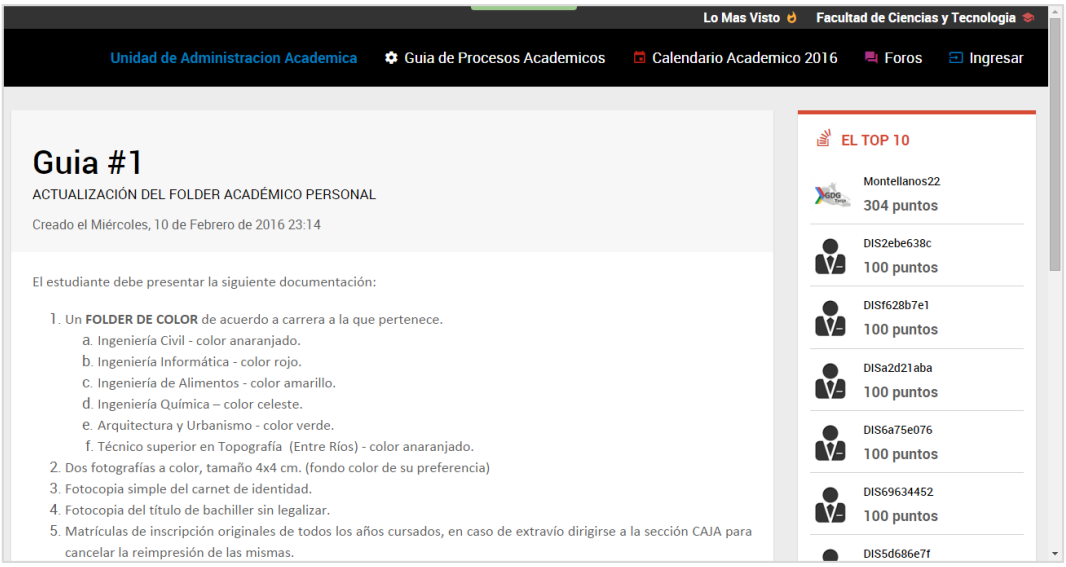
Esta interfaz es consecuencia de ingresar en una de las noticias que salen en la pantalla principal, contiene todo lo referido a la noticia como ser: contenido, imágenes y comentarios, también ofrece lo que son los enlaces a otras noticias como las más leída.

3. Interfaz 2 Guía de trámites académicos



La presente interfaz contiene los que son todas las guías para los tramites académicos, que realiza la UNADA, estos mismos fueron previamente cargados y extraídos de la base de datos.

4. Interfaz 2.1 (Ver guía de tramite)



Se ve en detalle los pasos a seguir para realizar con éxito un trámite, el mismo contiene la información detallada de los requisitos de cada tramite, también ofrece enlaces a foros por si alguno se relaciona con el trámite.

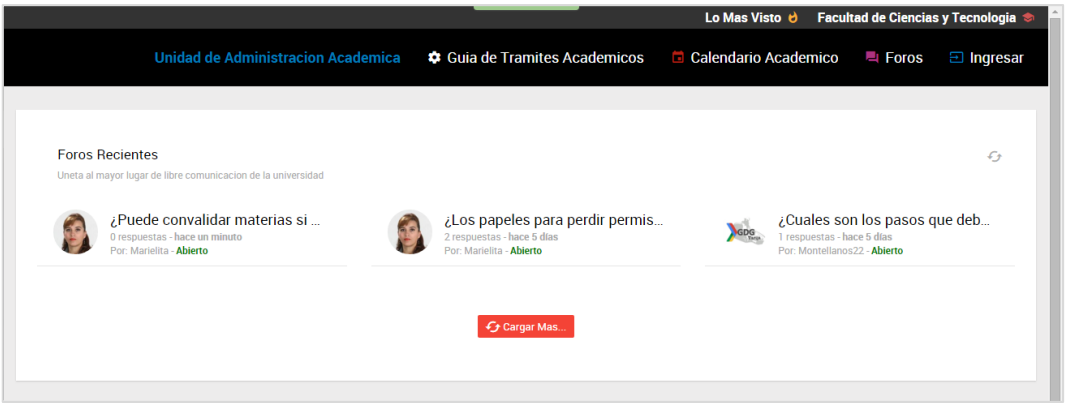
5. Interfaz 3 (Calendario Académico)



Presenta el calendario académico, correspondiente a la gestión, muy gráfico lo cual da una perspectiva amigable al usuario, se carga con todas las actividades que planifica la facultad si éstas son cargadas previamente

.

6. Interfaz 4 (Foros)



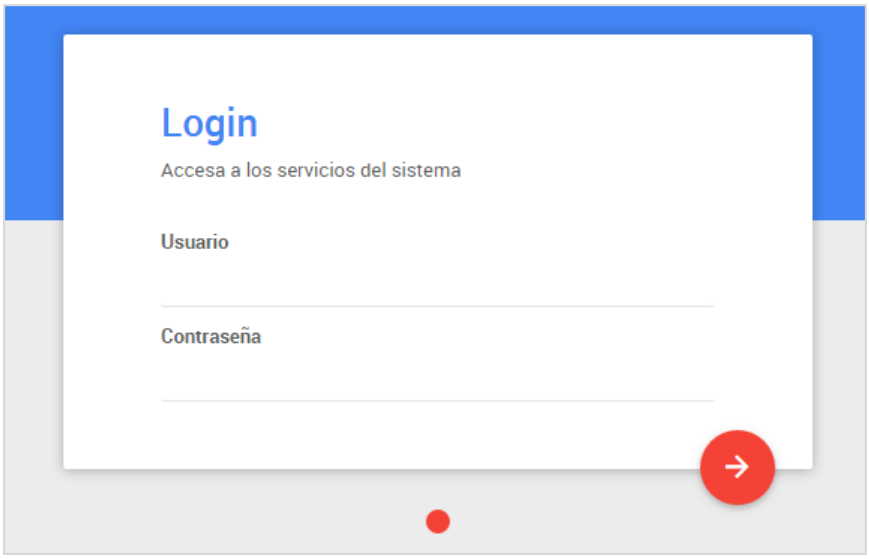
Carga lo que son los últimos foros creados por los usuarios, en la primera interfaz se cargan los últimos 24 foros, con el botón “cargar más”, se muestran 24 foros más en caso de existir.

6.1.1.1. Interfaz 4.1 (Ver foro)



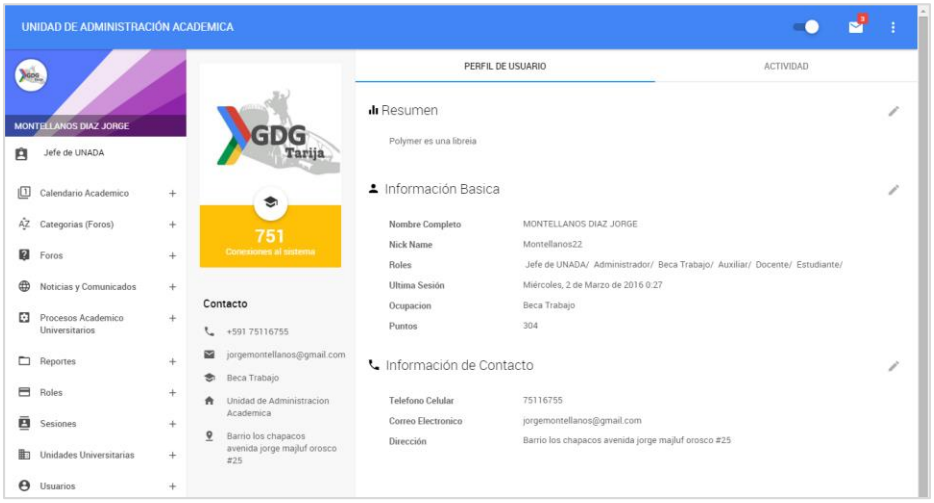
Se muestra a detalle el foro, con la pregunta y su descripción en caso de tenerla, también están los comentarios de otros usuarios sobre el tema comentado por el moderador, para poder comentar los mismos de se debe iniciar sesión, en caso de estar cerrado, no se podrán aumentar más comentarios.

7. Interfaz 5 (Ingresar o Iniciar Sesión)



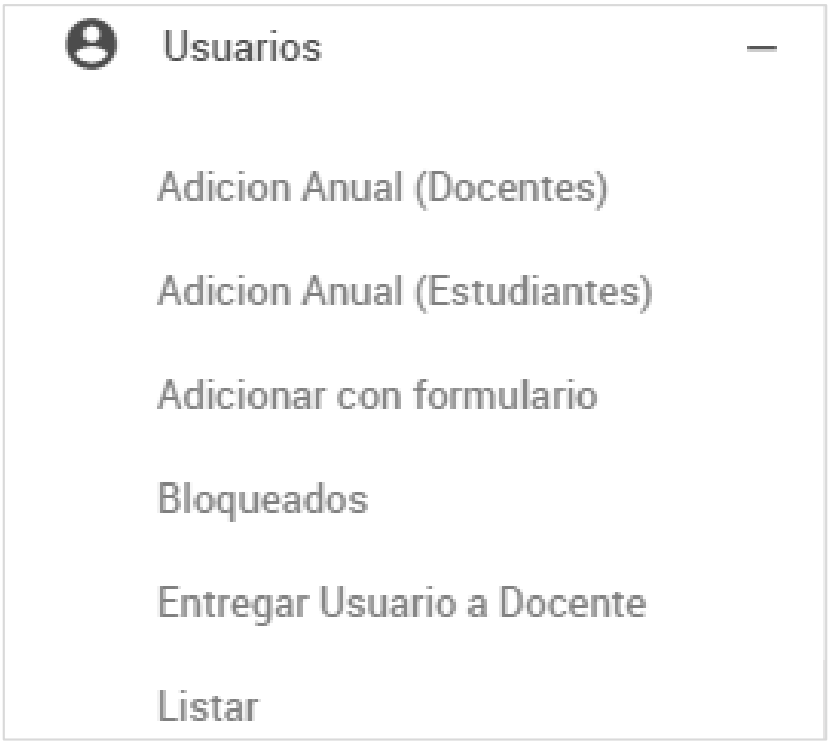
Esta interfaz provee lo que es el formulario de acceso al sistema, el cual solicita el usuario y la contraseña para ingresar, también tiene la opción de recuperar contraseña vía correo electrónico, ingresando en el botón rojo que tiene por debajo, el cual despliega otro formulario, el mismo es necesario para la restitución de contraseña, en caso de ingresar la contraseña errónea, o estar Bloquedo esta misma mostrara el mensaje pertinente.

8. Interfaz 6 (Panel de administración u Escritorio)



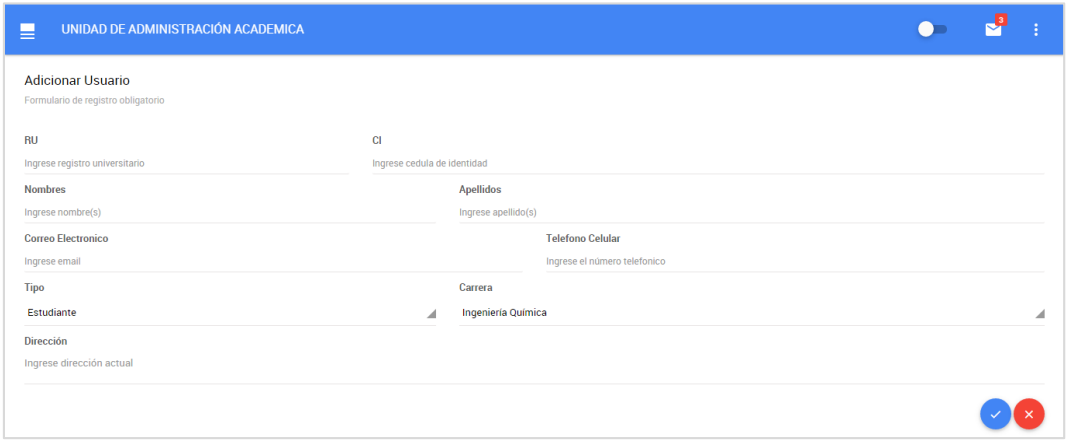
Esta interfaz se muestra si se ingresó de forma correcta los datos en la interfaz 5, nos muestra lo que es el perfil de usuario mas todos los menús correspondientes al usuario, según el rol que tenga asignado.

9. Interfaz 7 (Menú de usuarios)



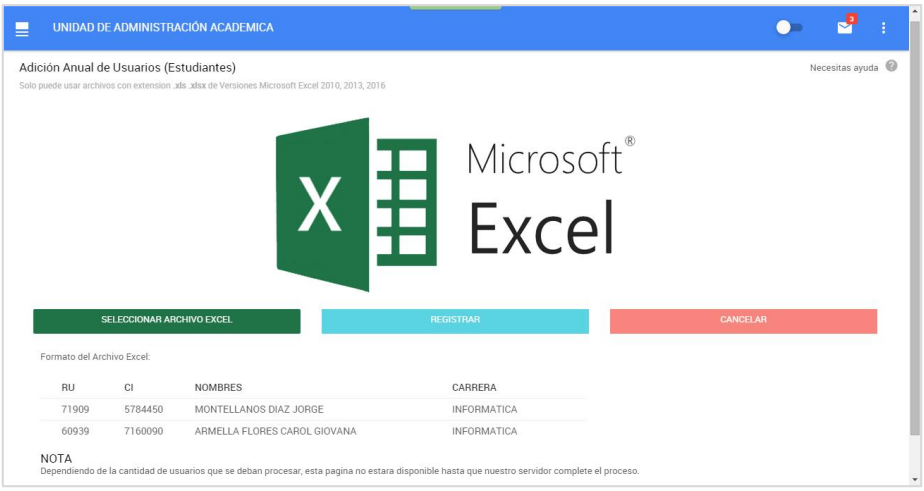
Menú con las opciones para los usuarios, se carga si se tiene el rol asignado.

10. Interfaz 7.1 (Adicionar con formulario)



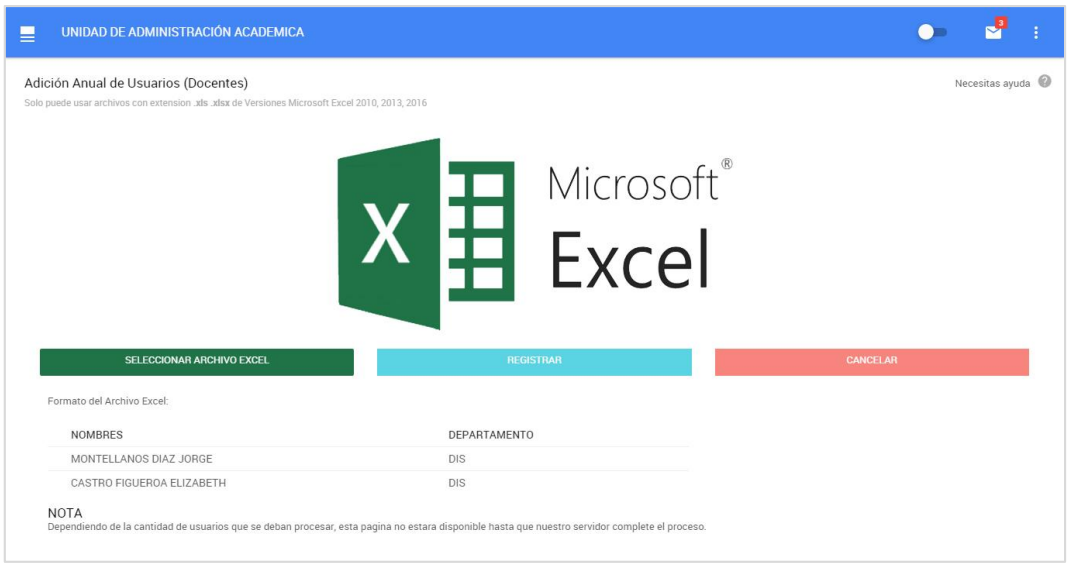
La interfaz que se muestra es para los casos que algún usuario no fue ingresado en la adiciona anual con Excel que se extrae del sistema Tariquia

11. Interfaz 7.2 (Adición Anual-Estudiantes)



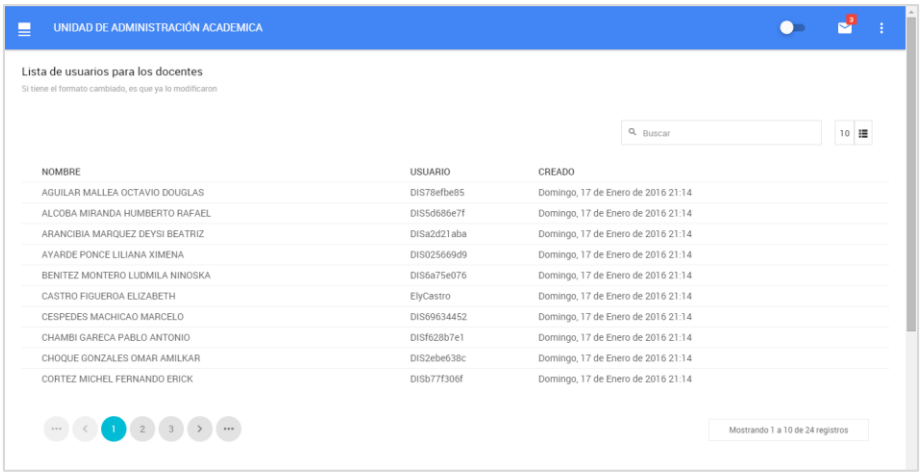
Esta interfaz, cuenta con lo que es el cargado de archivos Excel, la misma especifica en que caso debe encontrarse la información dentro del archivo, en el caso de los estudiantes se toma en cuenta que la columna carrera deberá ser la misma abreviación que se encuentra registrada con la unidad, caso contrario no la registraría, también cuenta con la ayuda asistida, la cual colabora mucho al usuario.

12. Interfaz 7.3 (Adición Anual-Docentes)



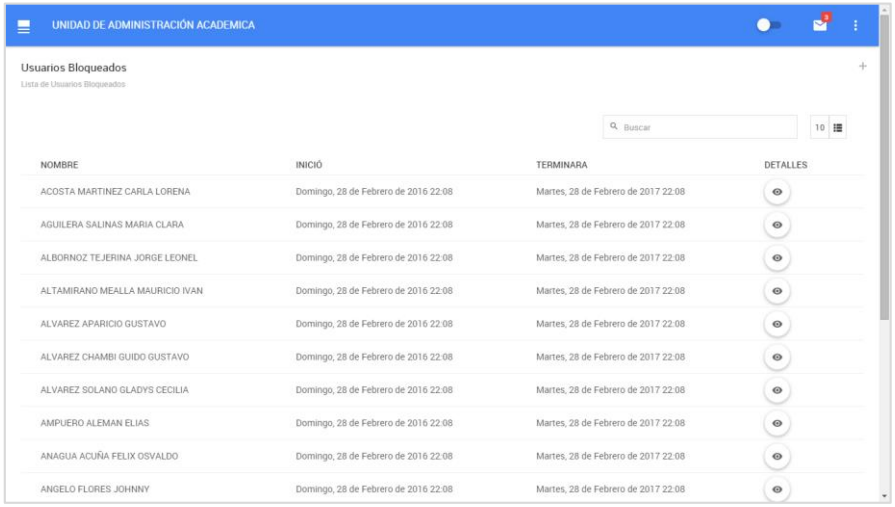
Sigue los mismos principios que la interfaz anterior, con diferencia en la información que solicita contenga el archivo Excel.

13. Interfaz 7.4 (Entregar usuario a docente)



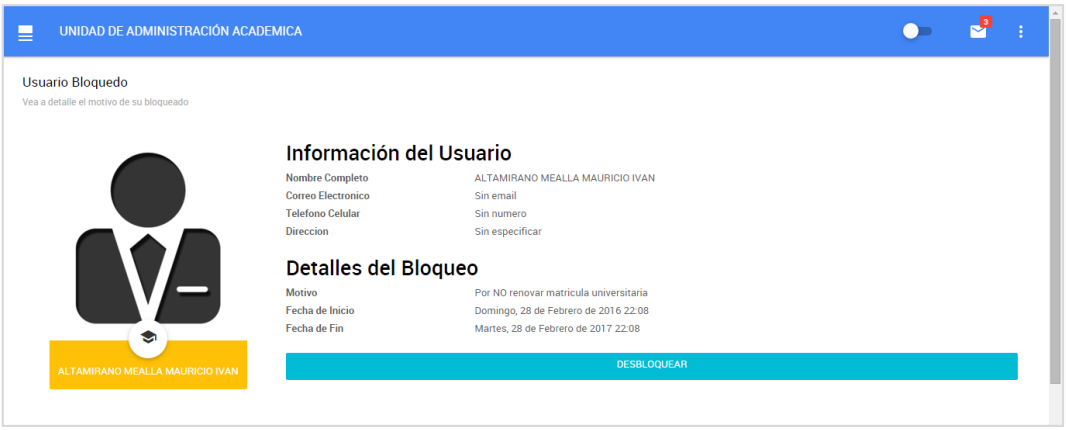
La interfaz muestra lo que son los usuarios con rol docente, cuenta con el filtro correspondiente, el usuario que se deberá entregar al docente, es el de la columna USUARIO, si tiene un formato diferente al de los demás es que ya se modificó al ingresar, la contraseña es la misma que el usuario.

14. Interfaz 7.5 (Bloqueados)



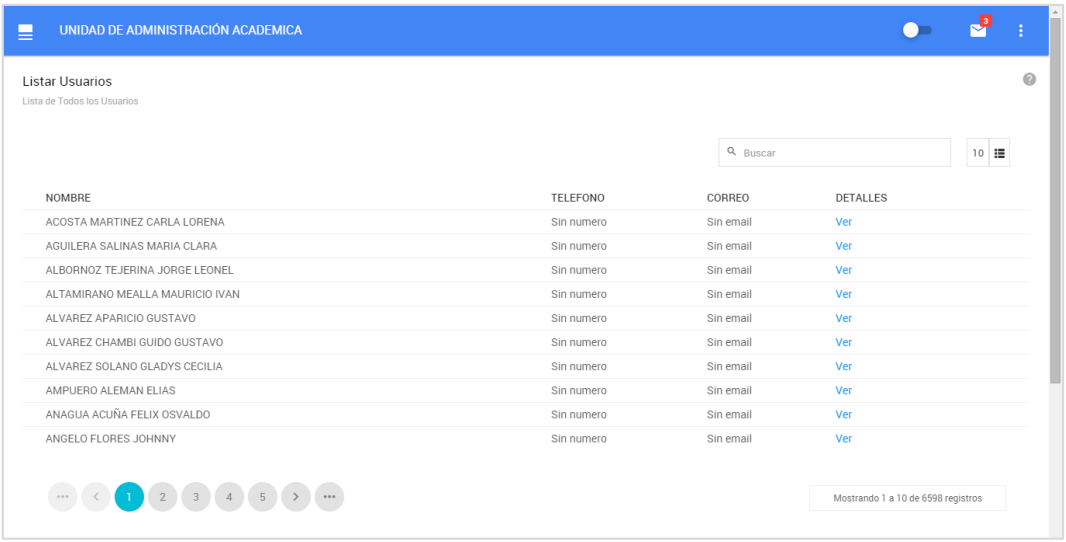
La interfaz muestra los que son, los usuarios que ya no tienen acceso al sistema mediante el usuario y contraseña, para ver el motivo del bloqueo, se deberá ingresar en la interfaz 7.5.1 que se explicará a continuación.

15. Interfaz 7.5.1 (Ver detalle de bloqueo)



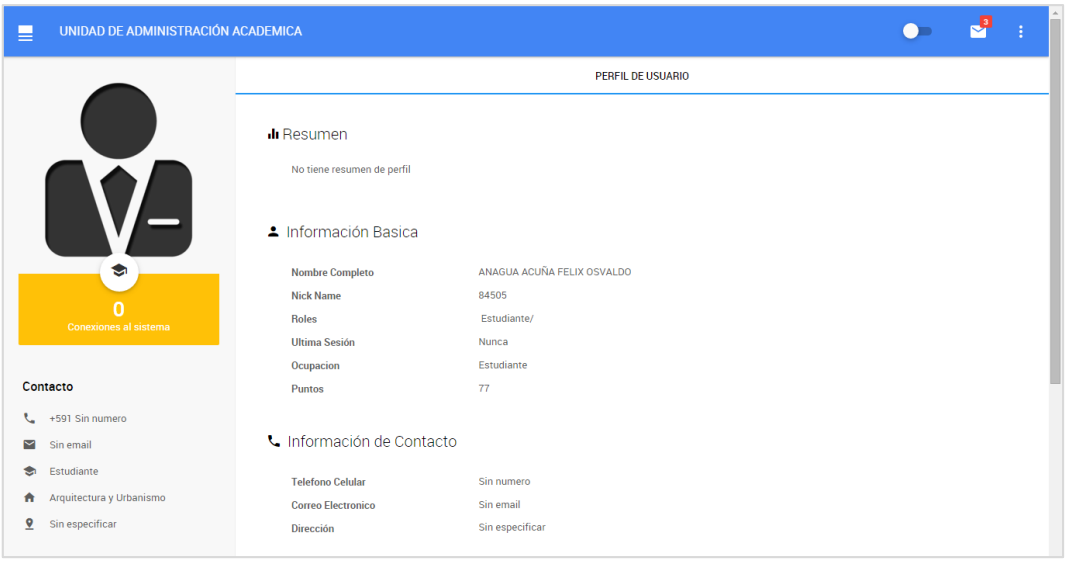
Se muestra a detalle el motivo por el cual el usuario ya no tiene acceso al sistema, puede ser desbloqueado, por seguridad en la base de datos se guarda quien lo desbloquea, para posibles casos de auditoria.

16. Interfaz 7.6 (Listar)



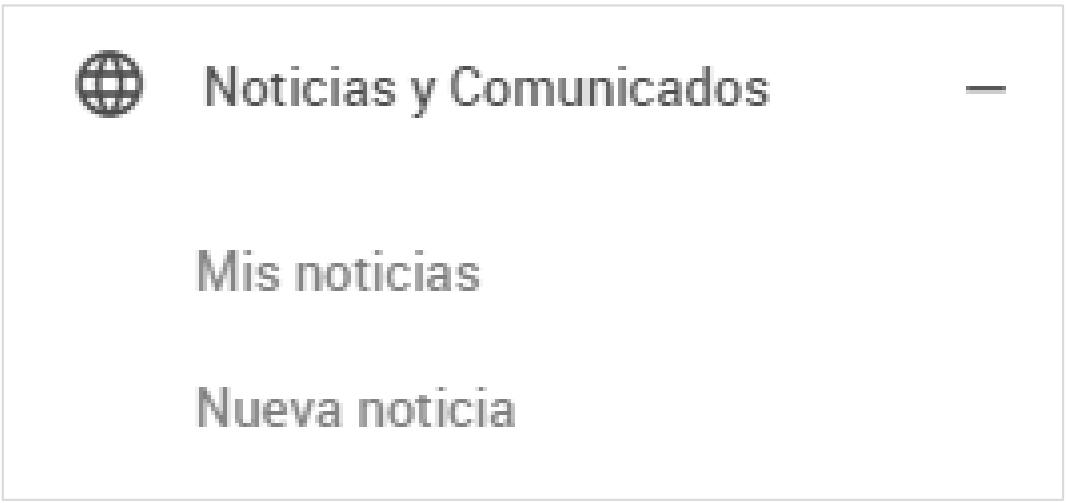
Se muestra en lista todos los usuarios activos registrados en el sistema, contiene el filtro correspondiente, también la opción de ver a detalle el usuario.

17. Interfaz 7.6.1 (Ver detalles de usuario)



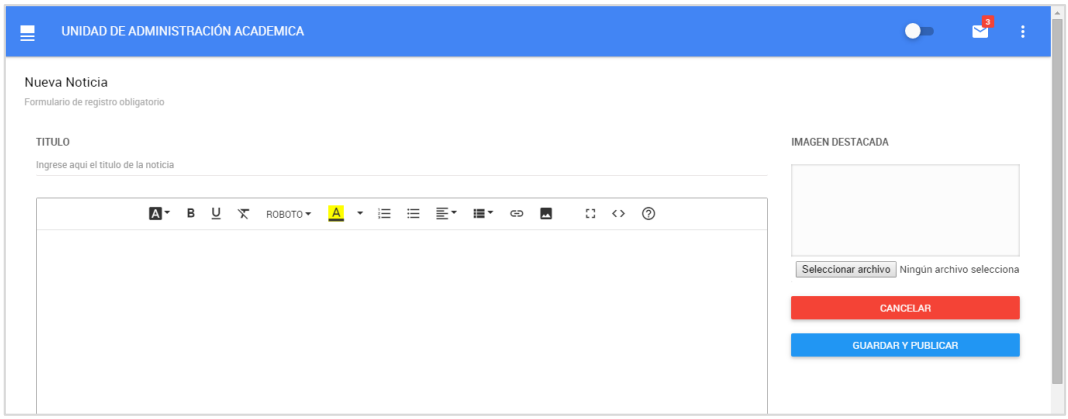
Detalles de información del usuario, un reporte por si se necesita alguna información de él.

18. Interfaz 8 (Noticias y comunicados)



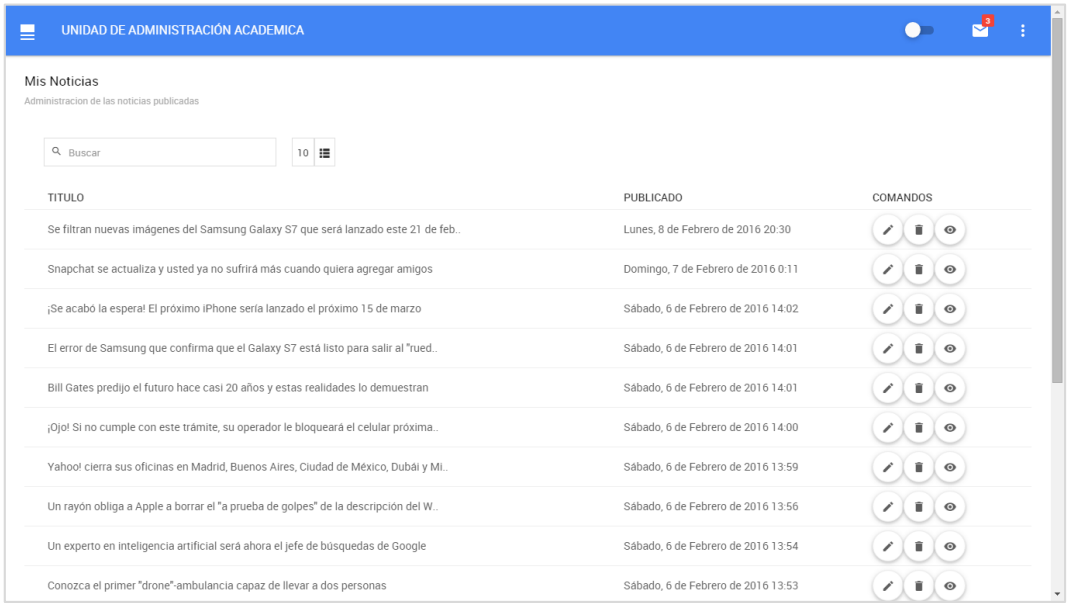
Menú con las opciones, si se tiene el rol asignado.

19. Interfaz 8.1 (Adicionar noticia)



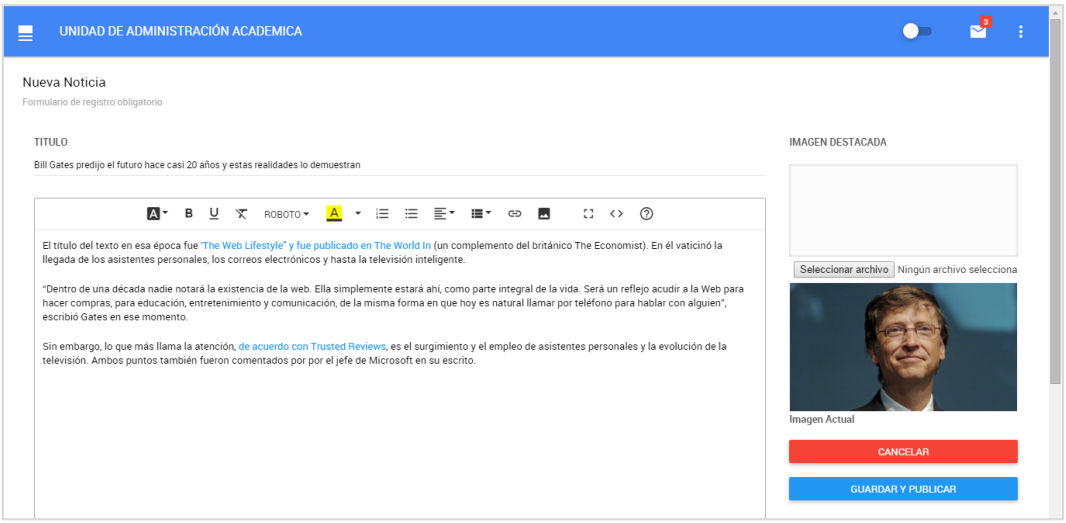
Interfaz intuitiva para el usuario ingrese el título de la noticia, una imagen obligatoria para la mejor publicación de la misma, y el contenido que se deja a la imaginación del autor, se ha proporcionado un panel de personalización para que cada noticia sea personalizada.

20. Interfaz 8.2 (Mis Noticias)



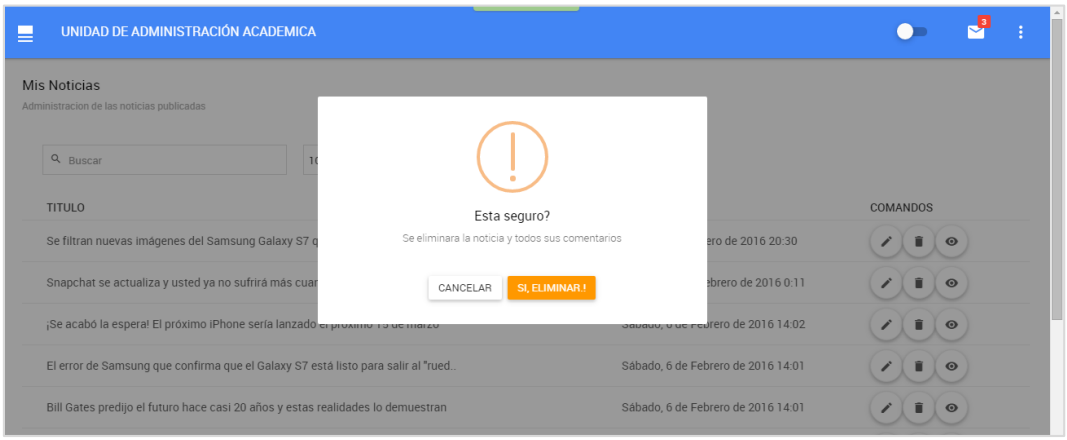
Panel de admiración de las noticas que se encuentran actualmente registradas, los comandos son las opciones que abren las demás interfaces, cuenta con filtro por si se busca una noticia específica.

21. Interfaz 8.2.1 (Modificar noticia)



La interfaz se muestra al acceder al icono de editar de la interfaz 8.2, se carga toda la información actual para poder modificar donde sea que se hayan equivocado, o en caso de solo querer mejorar el contenido de la noticia

22. Interfaz 8.2.2. (Eliminar noticia)



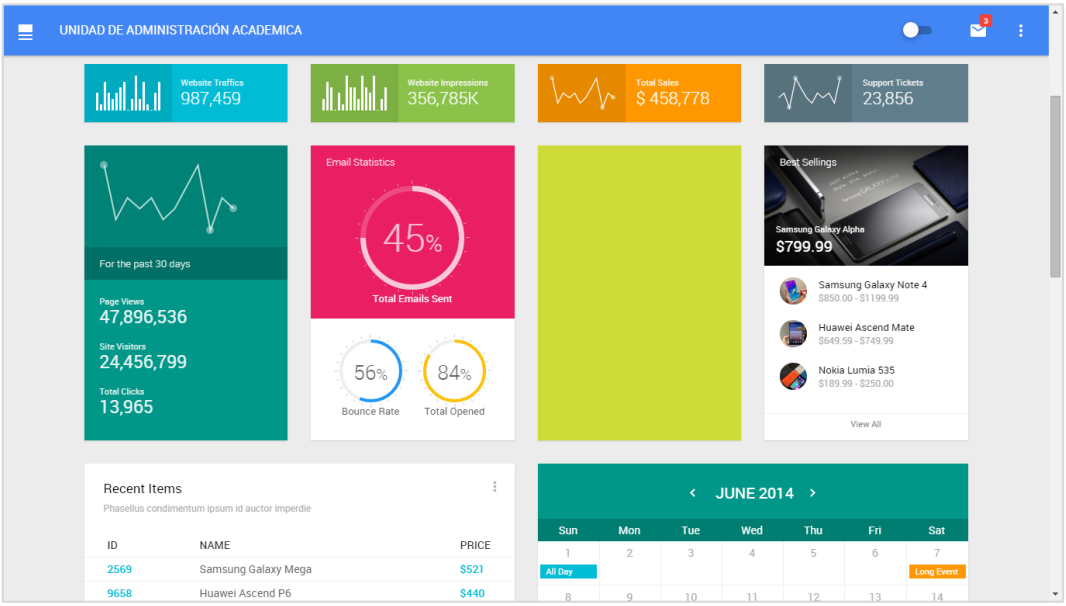
Interfaz donde se despliega el mensaje de confirmación, para la eliminación de la noticia seleccionada, previamente se debe tocar el icono de borrar.

23. Interfaz 8.2.3 (Ver noticia)



Interfaz diferente de la 1.1, este tiene formato diferente que solo el creador puede ver, también no cuenta con los comentarios, ni con los enlaces de la 1.1

24. Interfaz 9 (Reporte General)



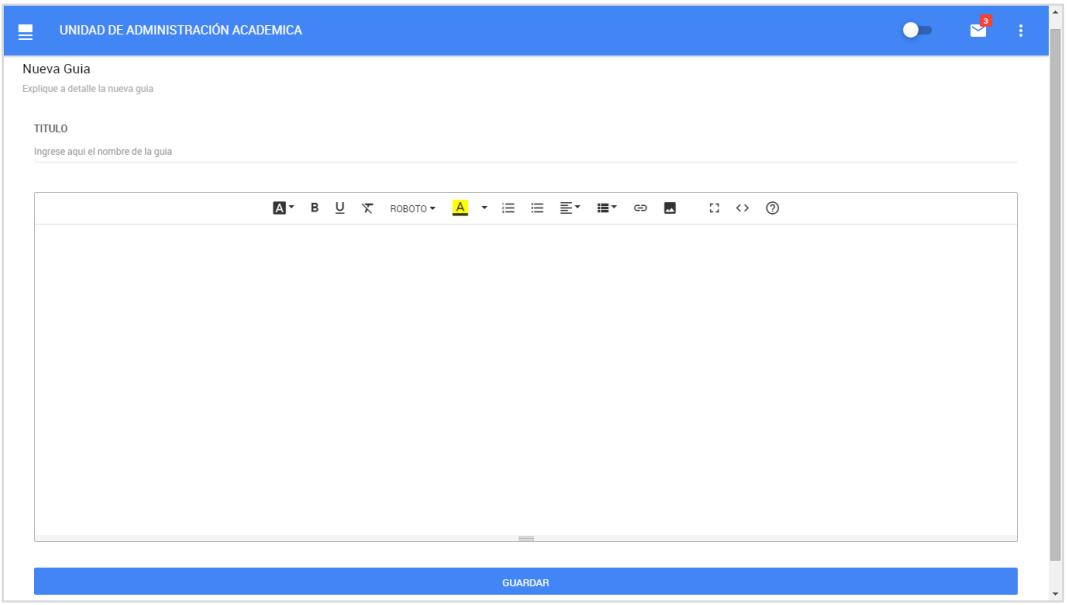
La presente interfaz, informa lo que es el uso del sistema, por parte de los usuarios, todo esto para poder tomar decisiones donde mejorar para que sea más usada y conocido.

25. Interfaz 10 (Guía de trámites académicos)



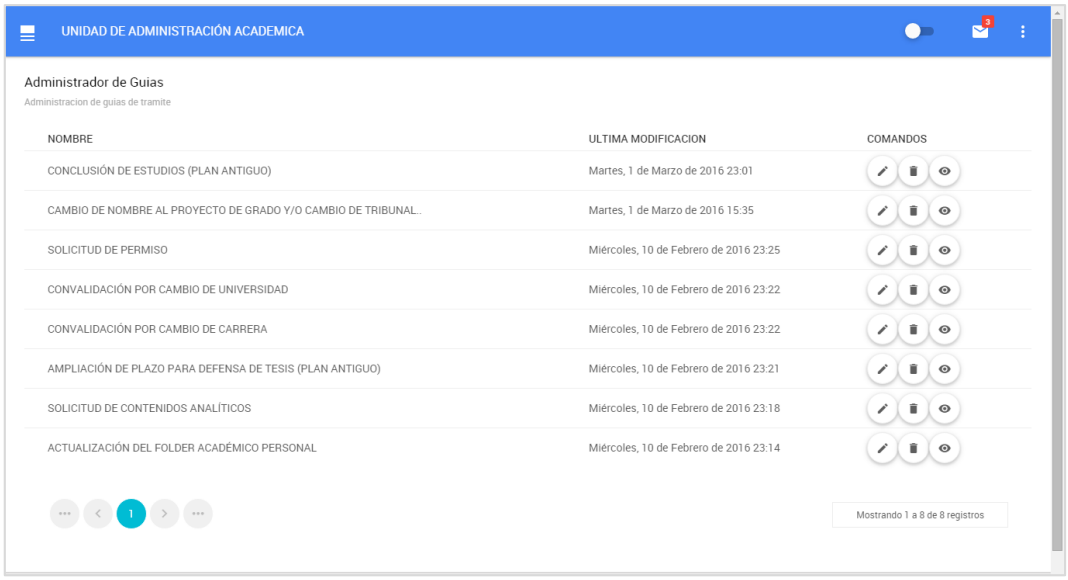
Esta interfaz muestra las opciones que tienen las guías de trámites académicos, la cuales son dos, el agregar guía y administrar las que ya existen

26. Interfaz 10.1 (Agregar guía)



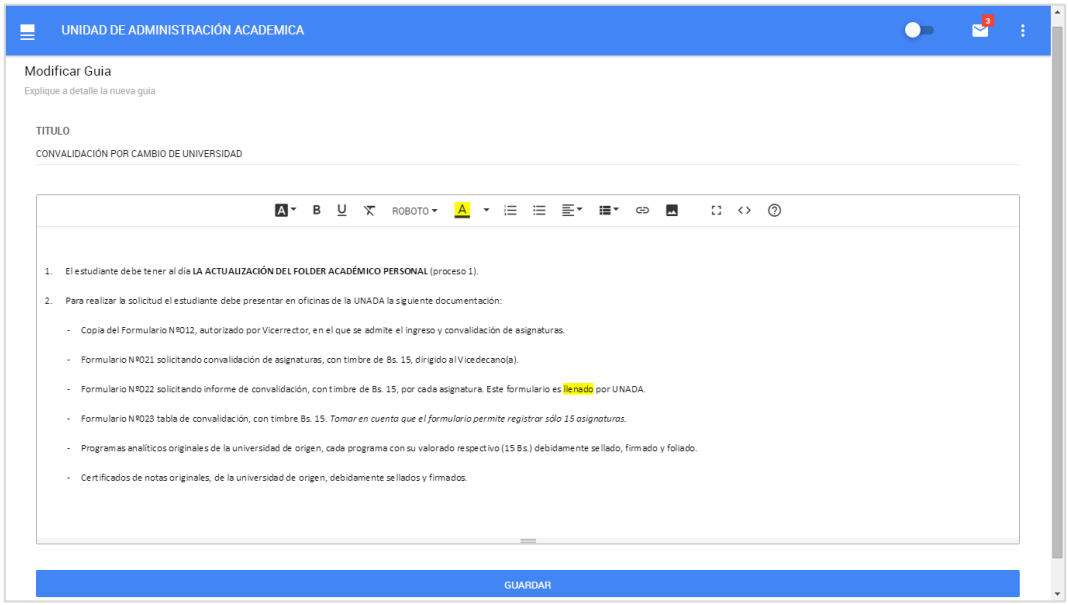
Esta interfaz presenta al usuario, un formulario dinámico, para que el contenido de la guía sea lo más agradable posible.

27. Interfaz 10.2 (Administrar guías)



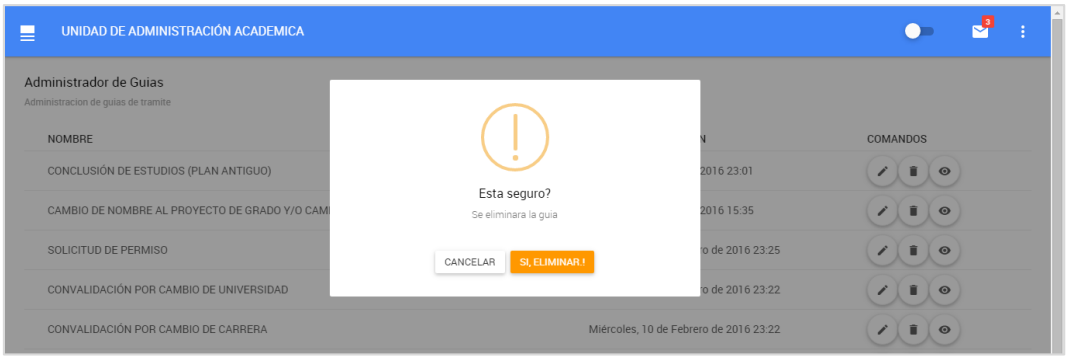
Esta interfaz proporciona al usuario un panel de administración sobre las guías que ya existen, las mismas que contiene otras interfaces, las que son editar, eliminar y ver, se describirán a continuación.

28. Interfaz 10.2.1 (Modificar guía)



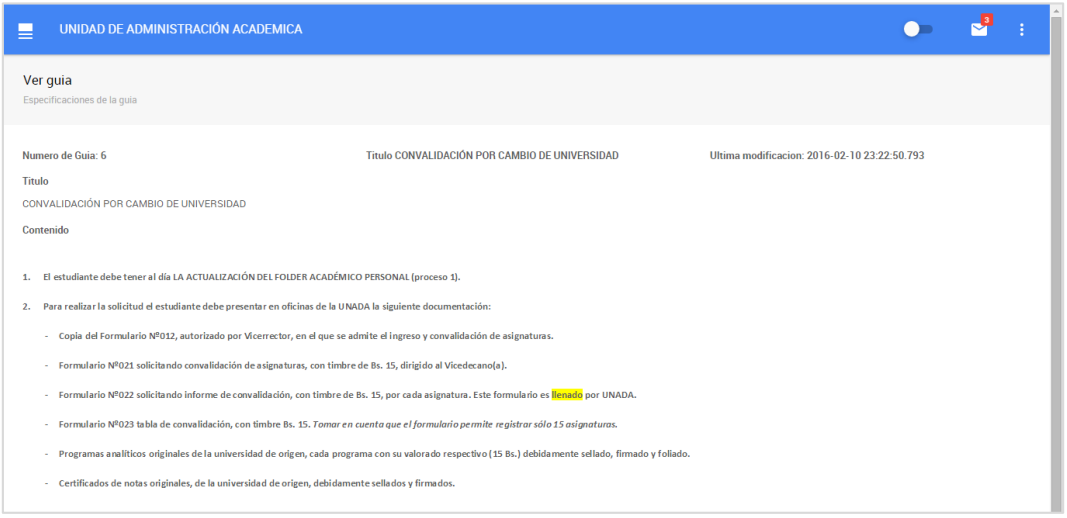
La interfaz se muestra si se ingresa en el enlace de la interfaz 10.2 la cual tiene el icono de modificar la guía, al hacer esto se carga la información actual, para poder cambiarla en caso de ser necesario.

29. Interfaz 10.2.2 (Eliminar guía)



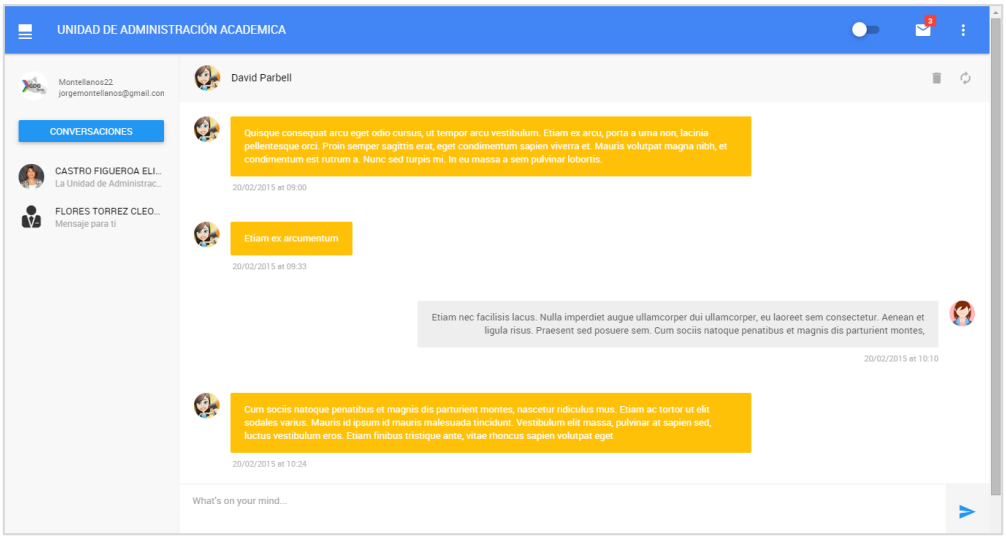
Interfaz donde se pide la confirmación de eliminación de guía por seguridad

30. Interfaz 10.2.3 (Ver guía)



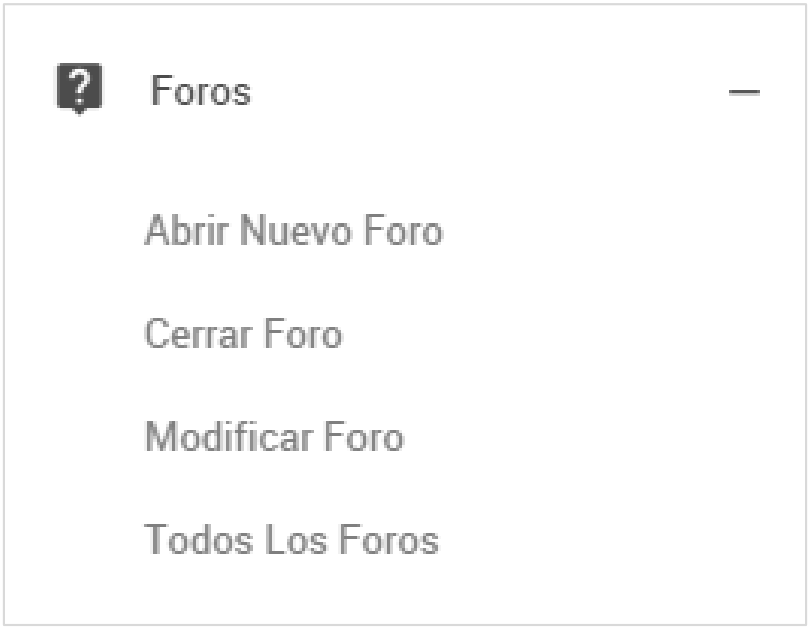
Se muestra la información completa que tiene la guía del tramite

31. Interfaz 11 (Consultas personalizadas)



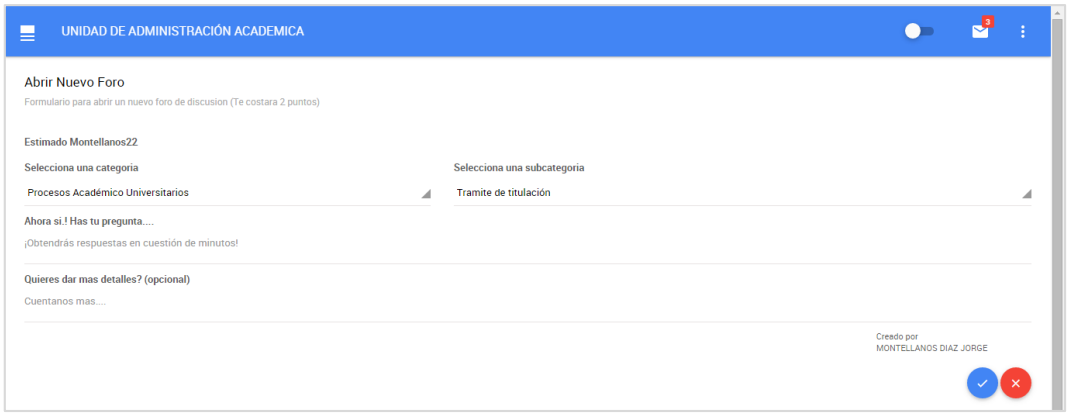
Interfaz con todo lo relativo a las consultas personalizadas, se puede crear nuevas, responderlas, etc.

32. Interfaz 12 (Foros)



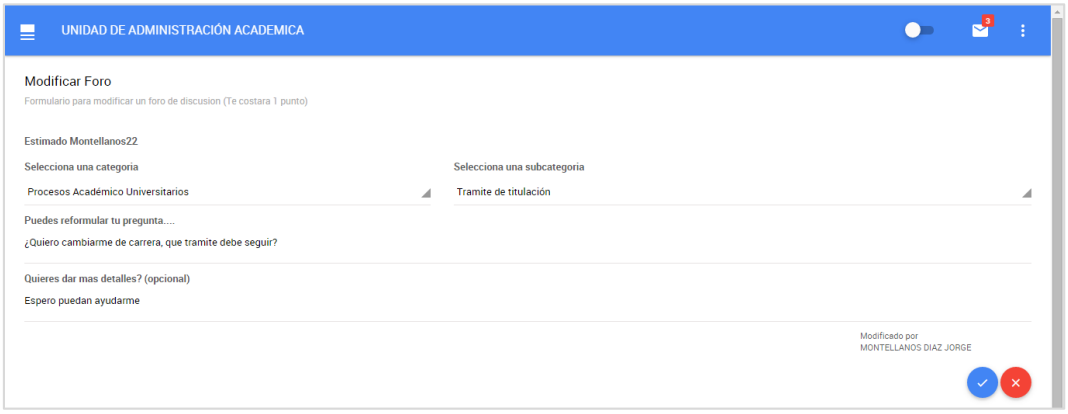
La presente interfaz muestra el menú de opciones para manejar los foros

33. Interfaz 12.1 (Abrir nuevo foro)



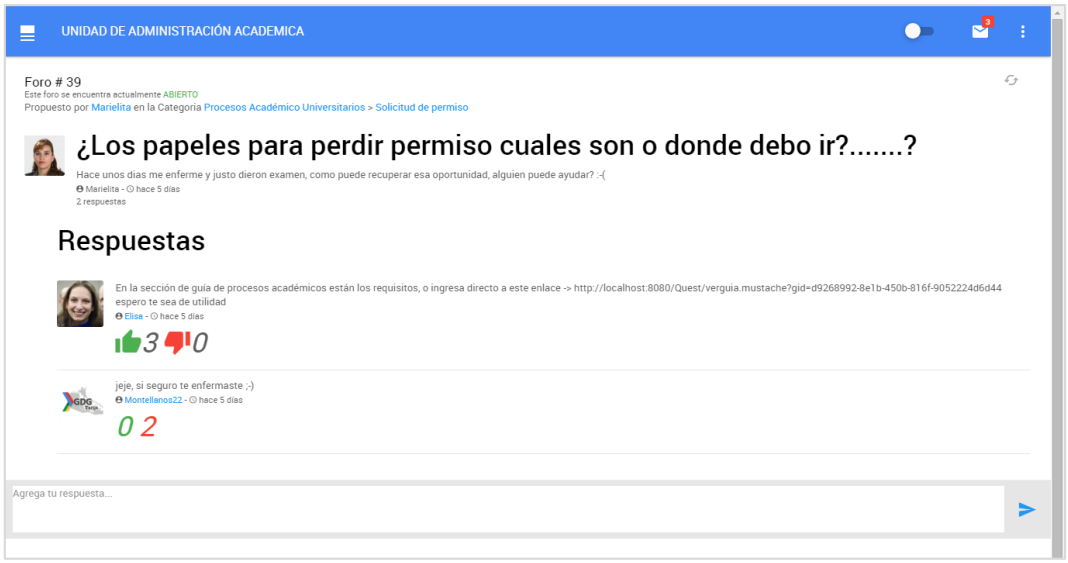
Interfaz que presenta el formulario para la apertura de un nuevo foro, la misma se contrala para tener un mínimo de contenido, tanto en pregunta como en descripción.

34. Interfaz 12.2 (Modificar foro)



Interfaz que se muestra con los datos actuales del foro, solo puede accederse a ella si el foro no tiene respuestas

35. Interfaz 12.3 (Ver foro)



La presente interfaz es la que más opciones tiene, en esta misma se puede agregar respuestas, también valorar las misma.

36. Interfaz 12.4 (Cerrar foro)



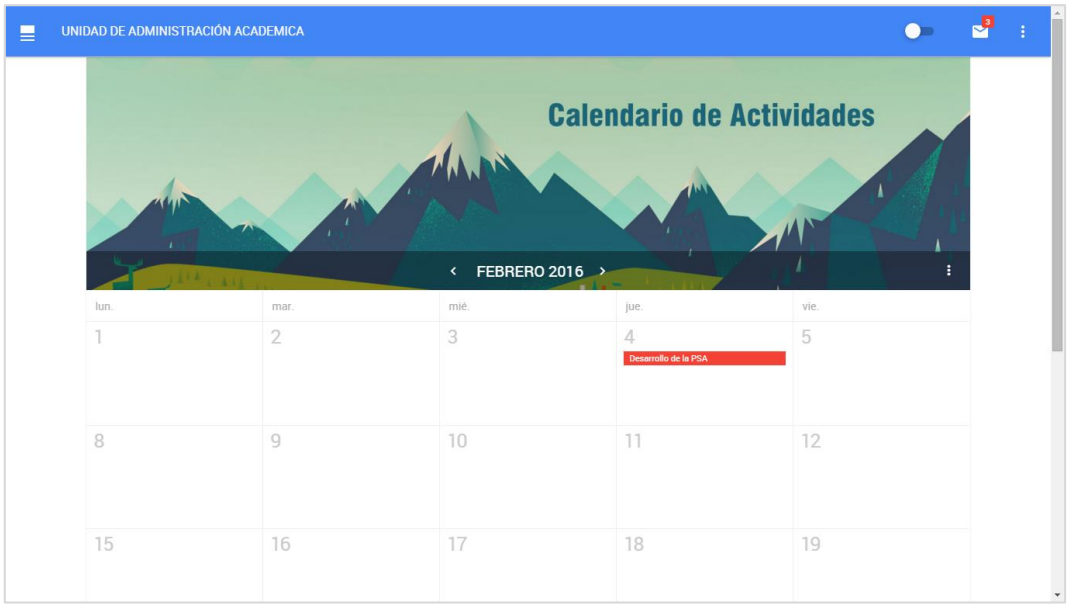
A esta interfaz solo accede el moderador del foro, es decir el usuario que la creo, la puede cerrar con el botón que se muestra, al cerrar el foro nadie más puede comentar en ella.

37. Interfaz 13 (Calendario Académico)



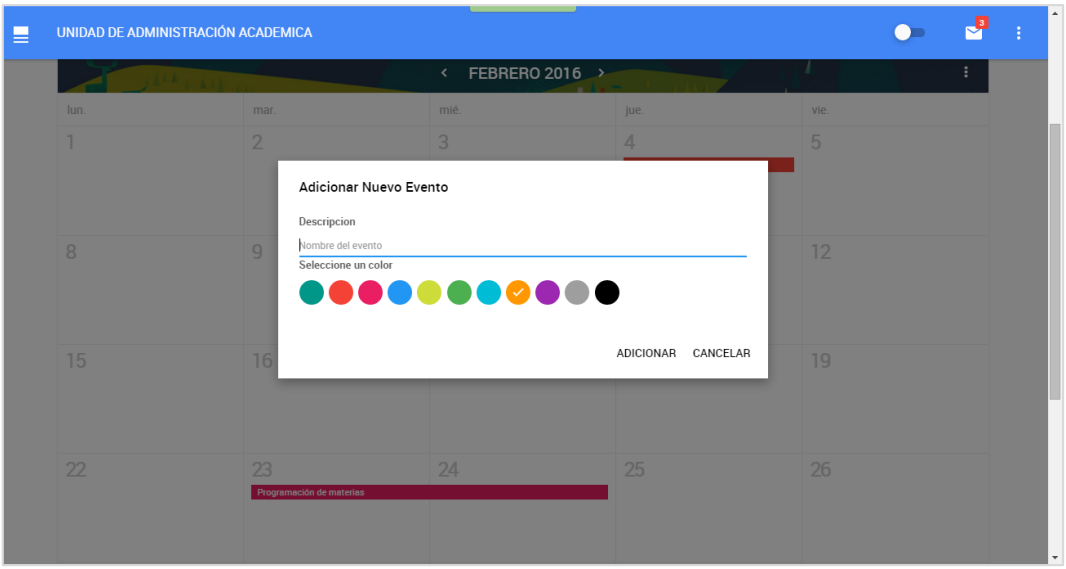
Esta interfaz tiene el menú de opciones para el calendario académico que controla el Jefe de UNADA

38. Interfaz 13.1 (Calendario de eventos)



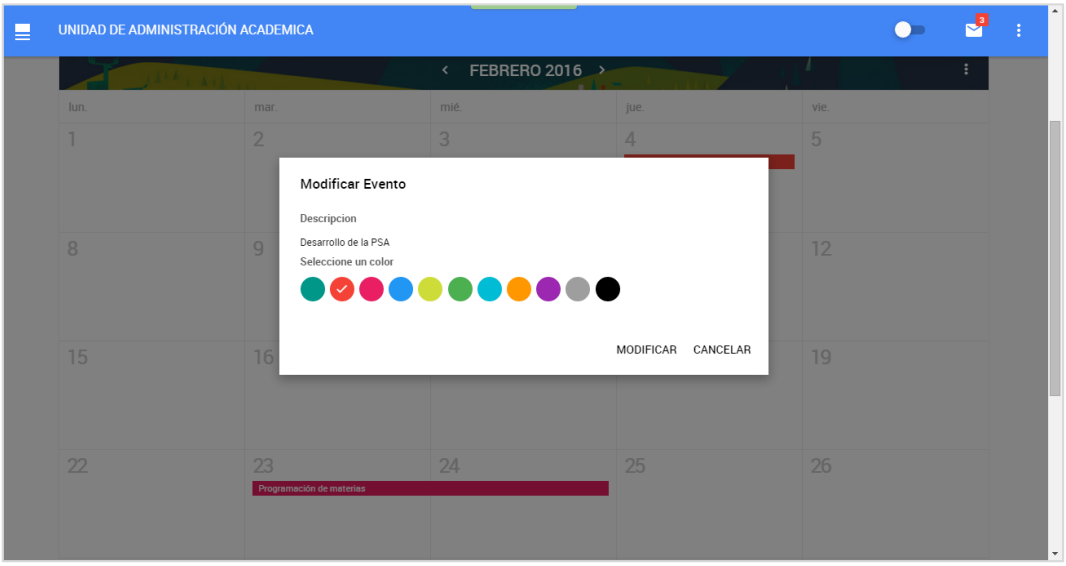
Interfaz que presenta el espacio de trabajo amigable para la creación y modificación de eventos según se requiera

39. Interfaz 13.1.1 (Nuevo evento)



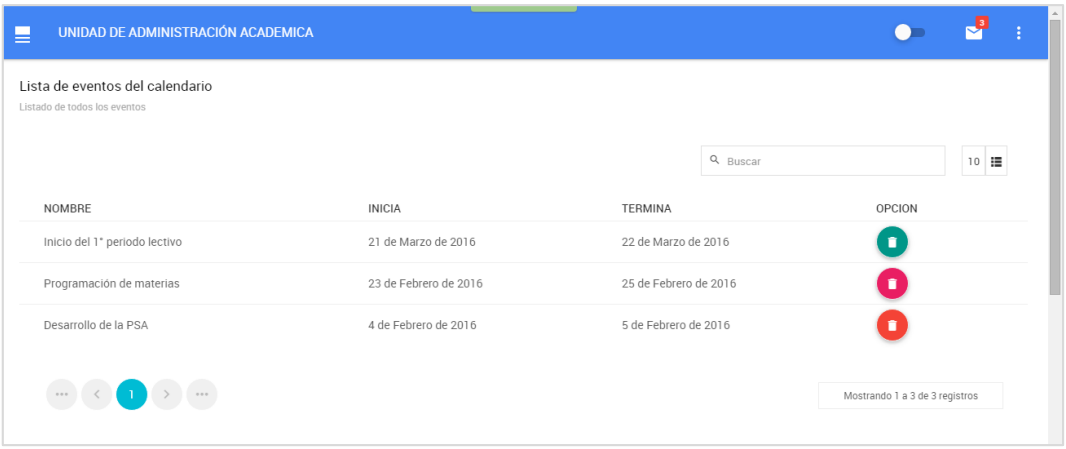
La interfaz se despliega al seleccionar un día o varios del calendario para agregar el evento, por seguridad las fechas se obtiene con un calendario valido y en fechas de jornada laboral

40. Interfaz 13.1.2 (Modificar evento)



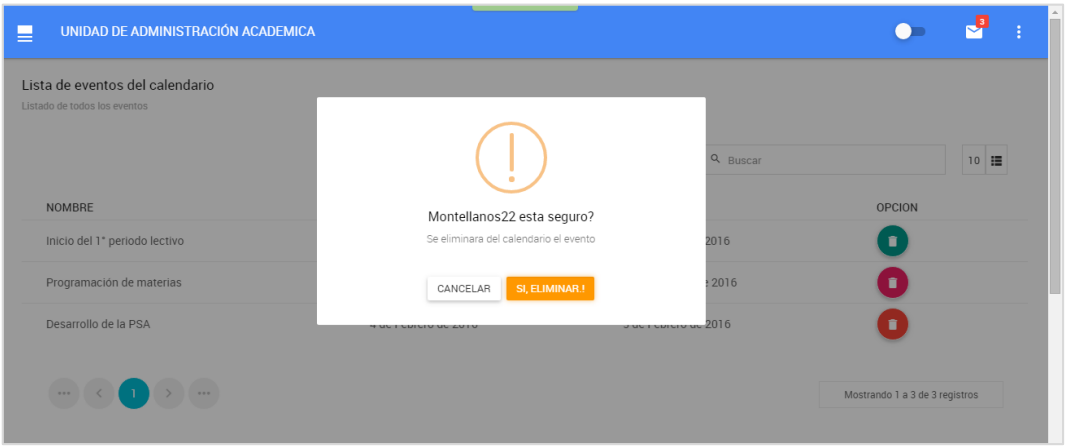
Se muestra la información actual del evento, el cual puede cambiar en color del evento y el nombre, en cuanto a fecha de inicio y fin se lo hace arrastrando el evento.

41. Interfaz 13.2 (Listar eventos)



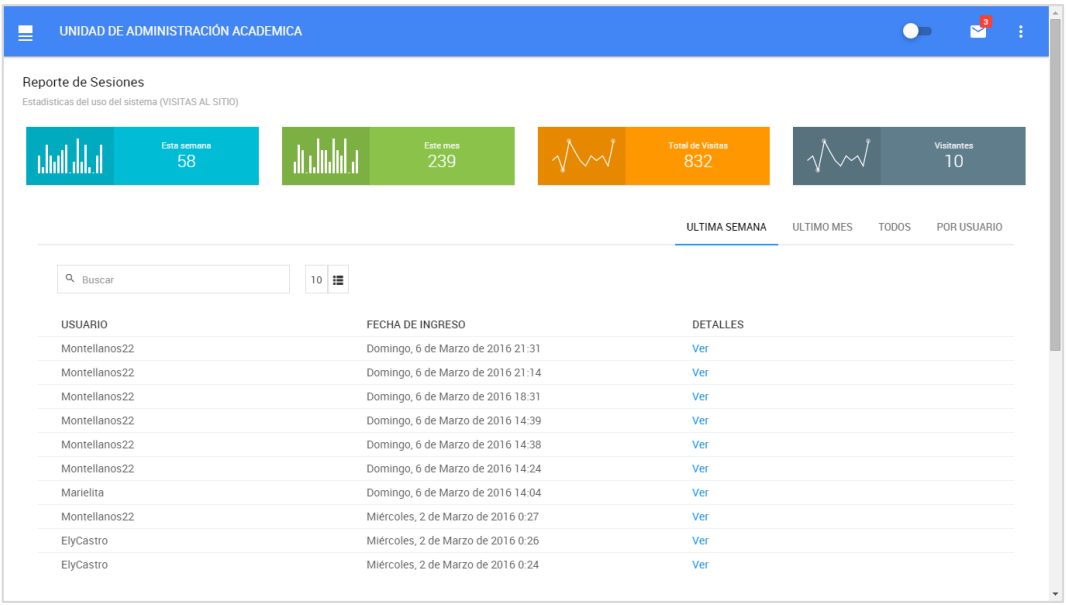
Listado de los eventos en orden cronológico, para su fácil compresión, también viene a la opción de eliminar evento, la cual habrá la interfaz que describimos a continuación

42. Interfaz 13.2.1 (Eliminar evento)



Interfaz de confirmación antes de eliminar de la base de datos puesto que debido a lo fácil que es crear un evento se hace borrado físico de los mismos

43. Interfaz 14 (Reporte de sesiones)



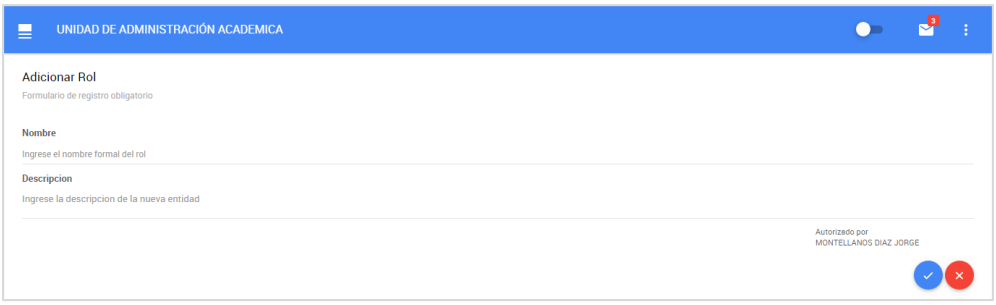
Reporte completo de los accesos al sistema, que se ven reflejados en las sesiones que describe la interfaz

44. Interfaz 15 (Roles)



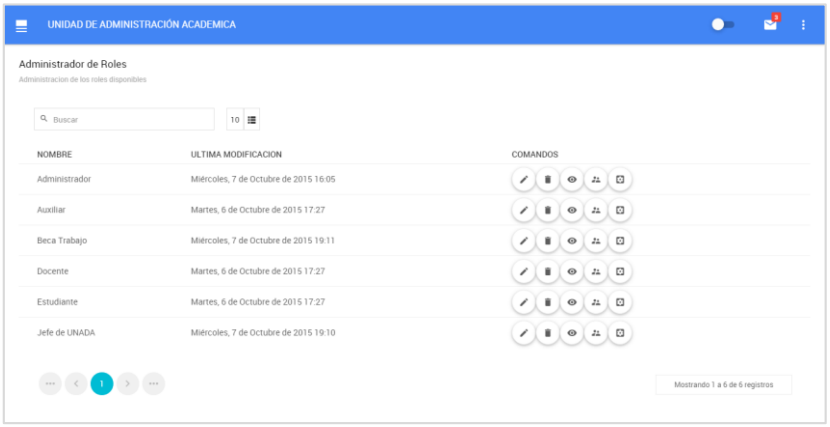
Menú de opciones que ofrece todo lo que contiene a los roles del sistema para los usuarios

45. Interfaz15.1 (Agregar rol)



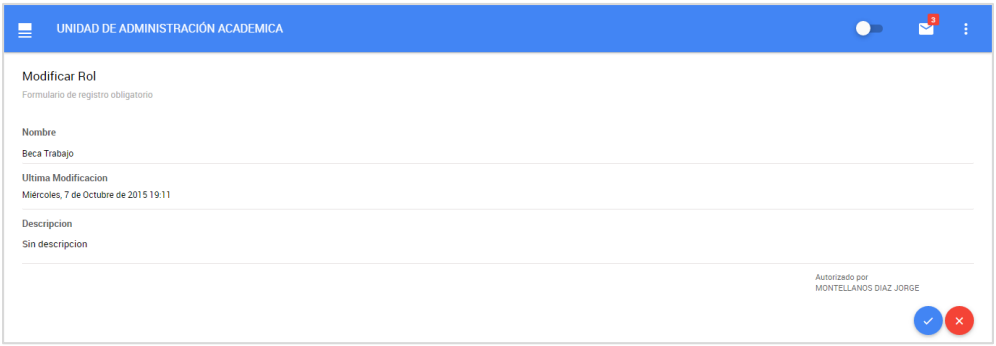
Formulario de llenado obligatorio para registrar un nuevo rol en el sistema

46. Interfaz 15.2 (Administración de roles)



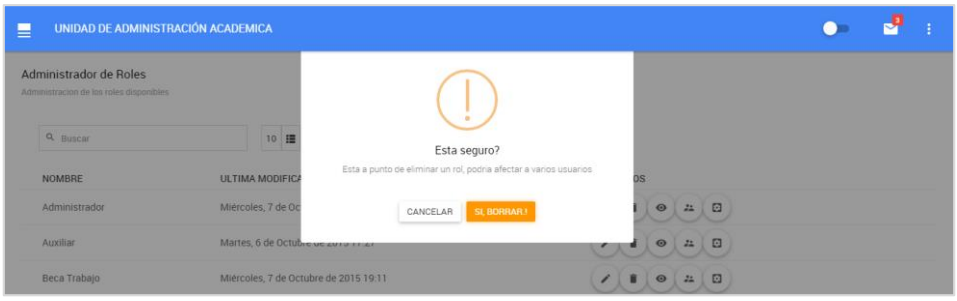
Panel de administración de los roles activos en el sistema, la tarea de los iconos de la columna comandos son: modificar, deshabilitar ver, usuarios por rol y asignar procesos rol respectivamente

47. Interfaz 15.2.1 (Modificar rol)



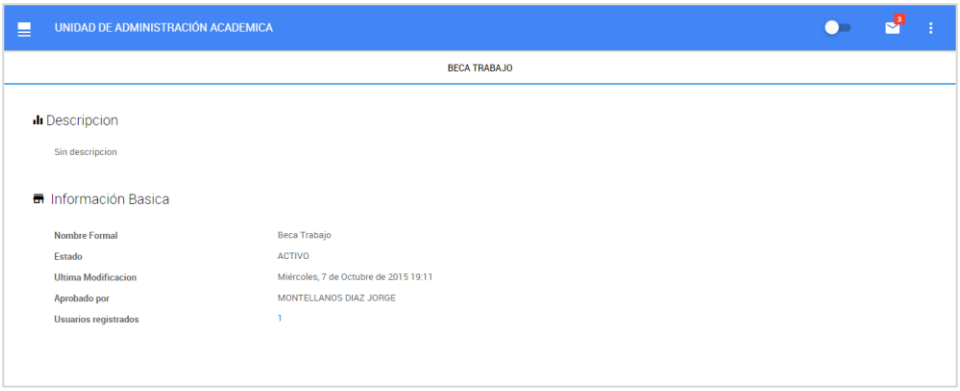
Formulario con la información actual del rol, nombre y descripción pueden ser modificados

48. Interfaz 15.2.2 (Deshabilitar rol)



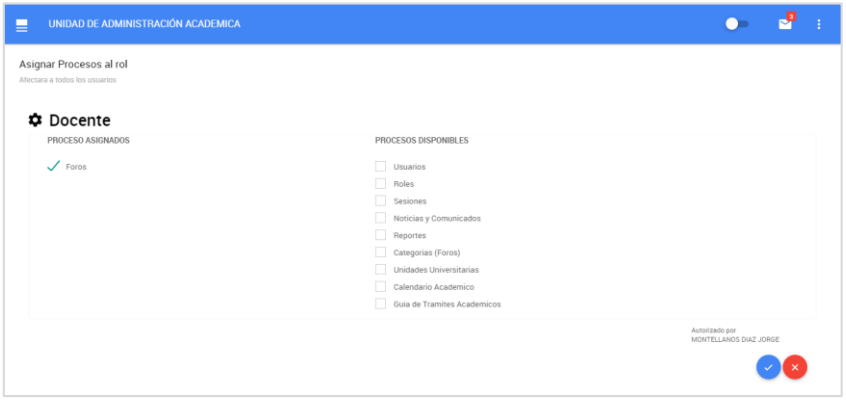
Mensaje de confirmación de la baja del rol, se puede cancelar en cualquier momento

49. Interfaz 15.2.3 (Ver rol)



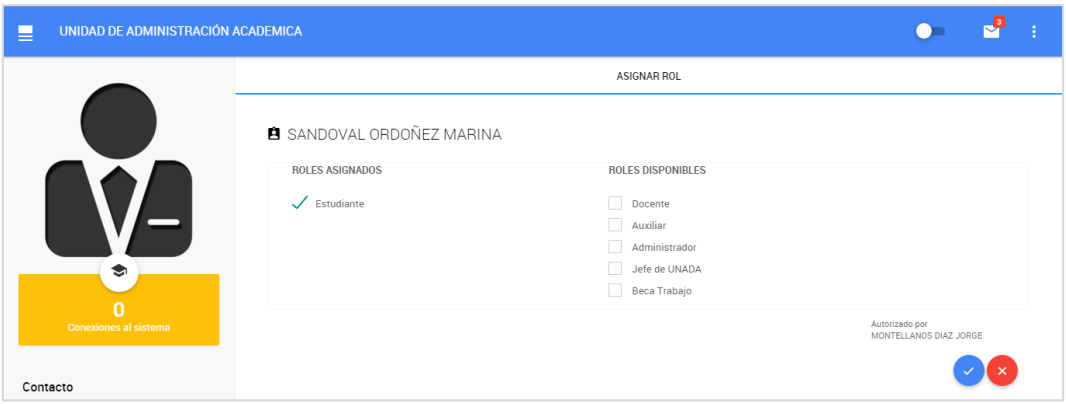
Descripción detallada del rol que se selecciona para ver

50. Interfaz 15.2.4 (Asignar procesos a rol)



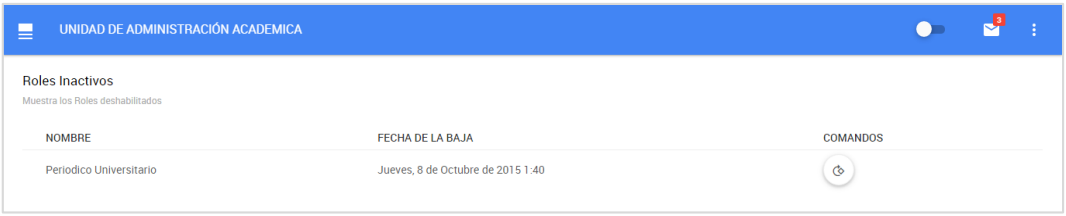
Interfaz donde se listan los procesos que tiene asignados un rol (lado izquierdo), y los que se le pueden asignar (lado derecho), hay que tomar en cuenta que usuarios tendrán ese rol

51. Interfaz 15.3 (Asignar rol a usuario)



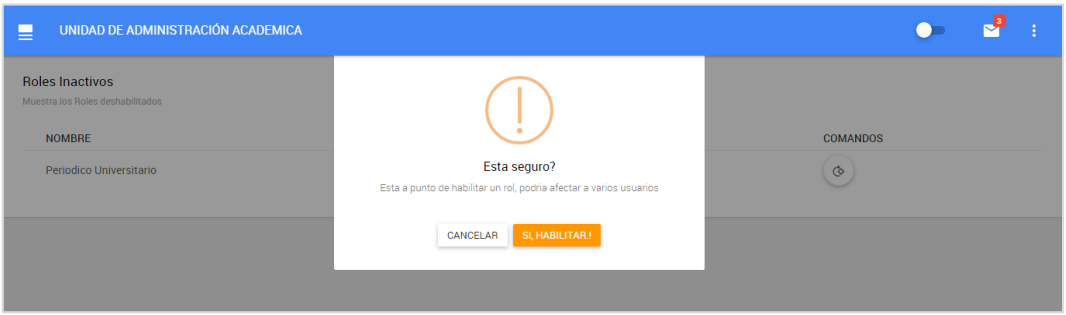
Interfaz donde se listan los roles que tiene asignados un usuario (lado izquierdo), y los que se le pueden asignar (lado derecho)

52. Interfaz 15.4 (Roles deshabilitados)



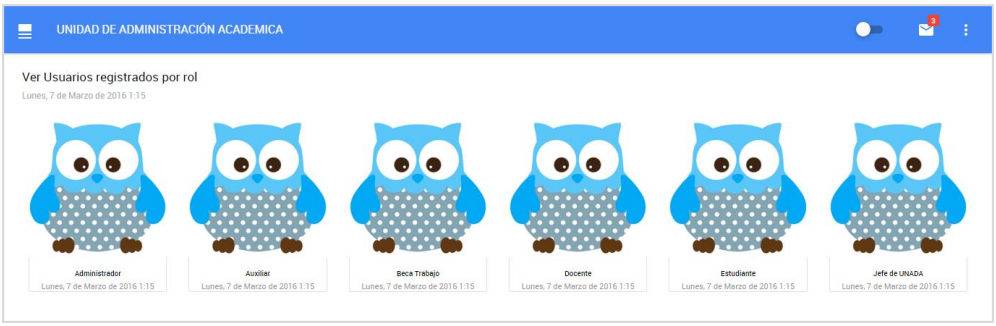
Listado de los roles deshabilitados con la opción de volverse a habilitar

53. Interfaz 15.4.1 (Habilitar rol)



Mensaje de confirmación para dar de alta el rol, se puede habilitar o cancelar

54. Interfaz 15.5 (Usuarios por rol)



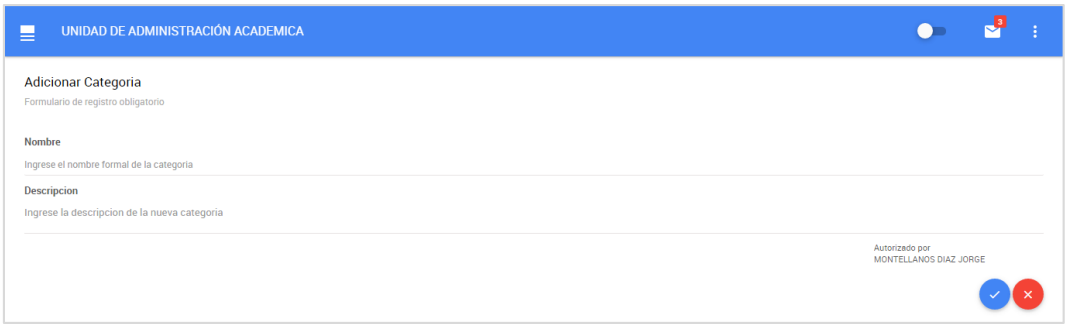
Clasificación de los usuarios por roles, se muestran todos los que estén activos en el sistema

55. Interfaz 16 (Categorías)



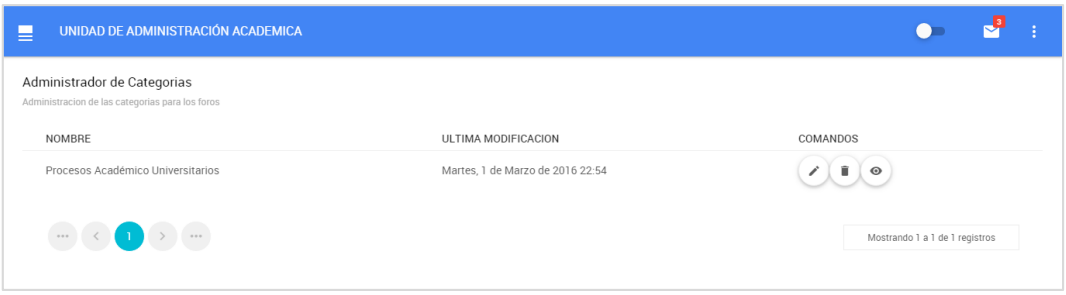
Listado de las opciones que tienen las categorías de los foros

56. Interfaz 16.1 (Agregar categoría)



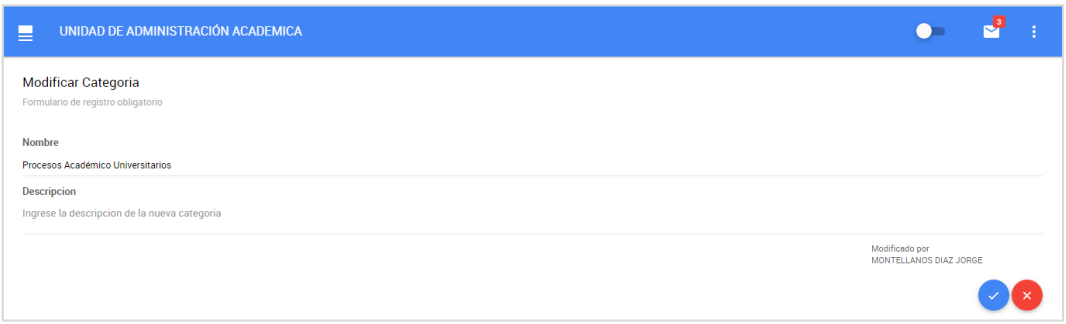
Formulario de agregación para una nueva categoría, controlada con la base de datos para asegurar la consistencia de datos

57. Interfaz 16.2 (Administración de categorías)



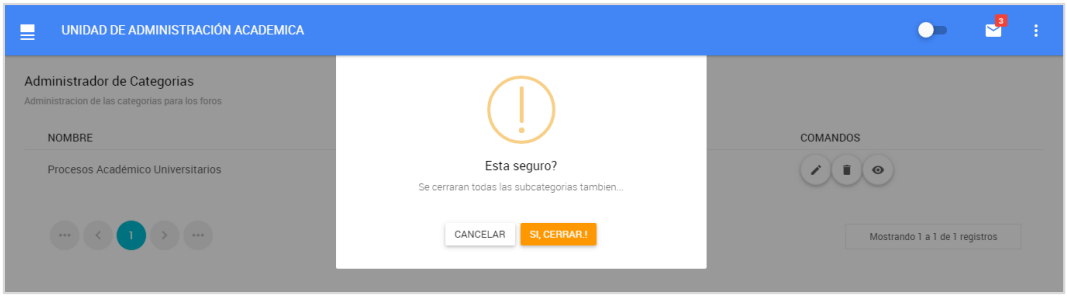
Panel de administración de categorías, las opciones de la columna comandos realizan la acción de modificar, deshabilitar y ver respectivamente

58. Interfaz 16.2.1 (Modificar categoría)



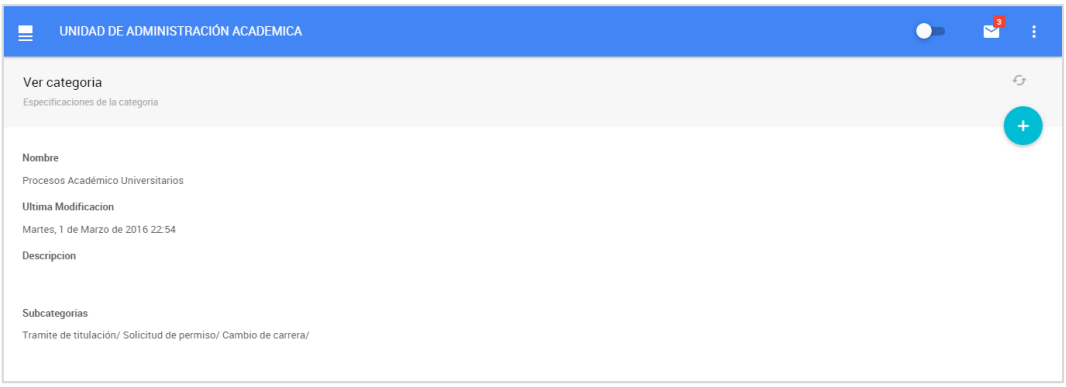
Formulario que carga la información actual de la categoría, se pueden modificar todos los campos

59. Interfaz 16.2.2 (Deshabilitar categoría)



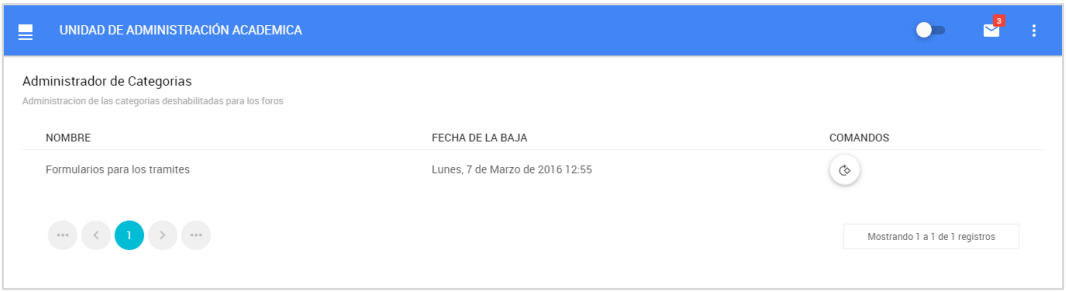
Mensaje de confirmación antes de deshabilitar una categoría, se puede proceder o cancelar en cualquier momento

60. Interfaz 16.2.3 (Ver categoría)



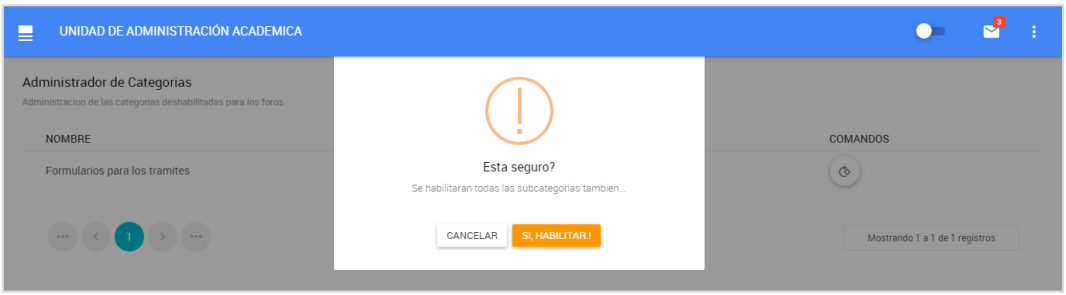
Información completa de la categoría que se selecciona en la interfaz 16.2

61. Interfaz 16.3 (Categorías deshabilitadas)



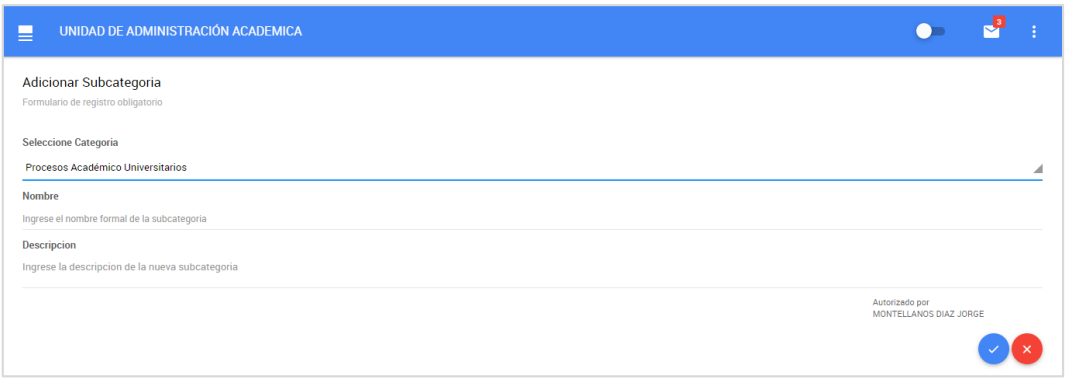
Panel de administración de las categorías deshabilitadas, el icono en la columna comandos es la opción de volver a habilitar la categoría

62. Interfaz 16.3.1 (Habilitar categoría)



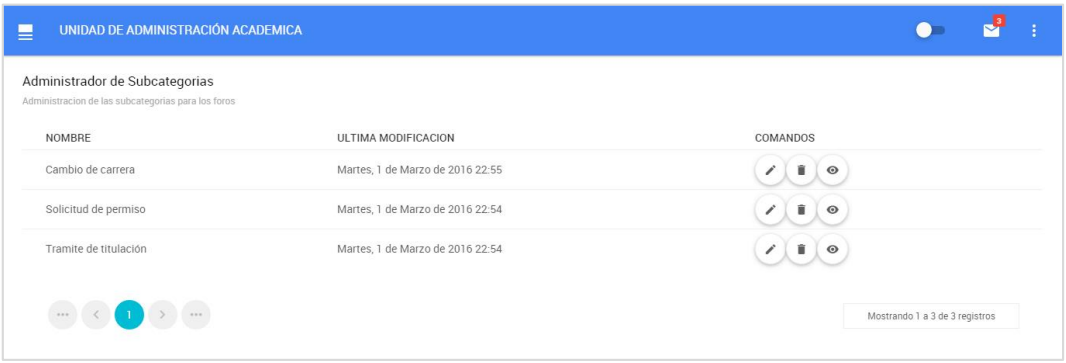
Mensaje de confirmación para habilitar la categoría

63. Interfaz 16.4 (Agregar subcategoría)



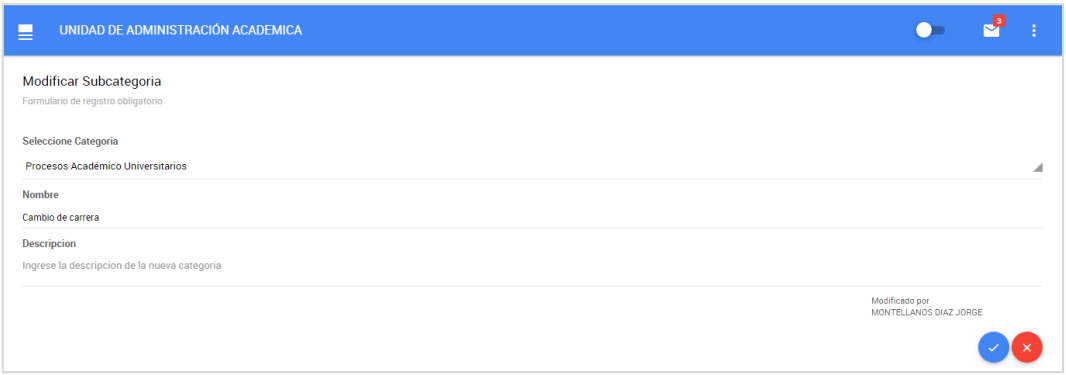
Formulario de agregación de la nueva subcategoría, la cual deberá ser llenado de manera obligatoria para la creación de la subcategoría

64. Interfaz 16.5 (Administración de subcategorías)



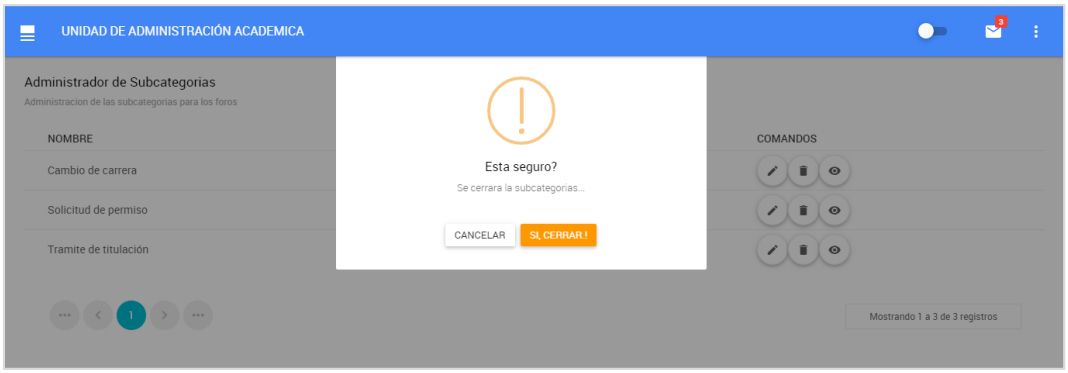
Panel de opciones para administrar las subcategorías, la columna comandos realiza las acciones de modificar, deshabilitar y ver subcategorías respectivamente

65. Interfaz 16.5.1 (Modificar subcategoría)



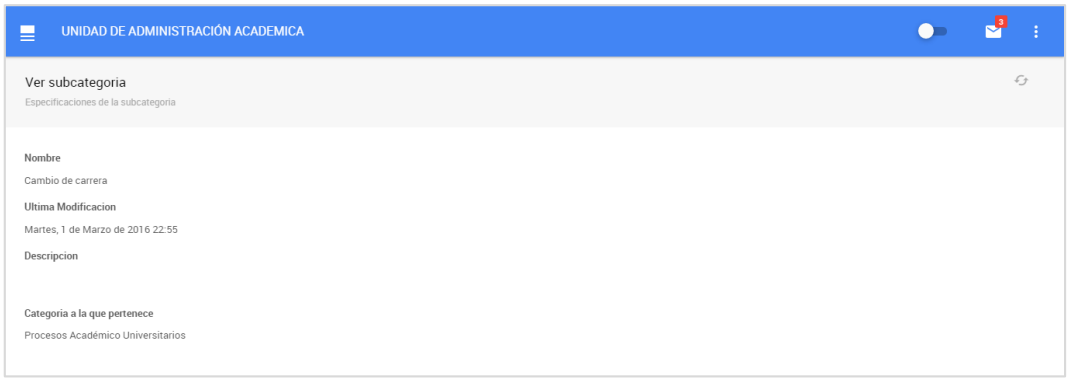
Formulario de modificación de la subcategoría con toda la información actual

66. Interfaz 16.5.2 (Deshabilitar subcategoría)



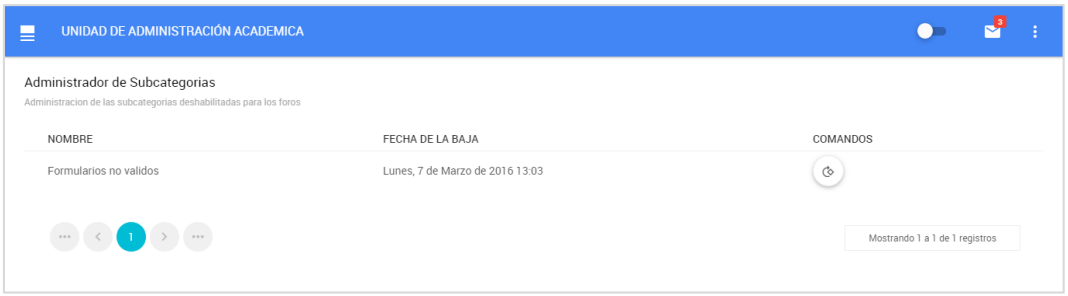
Mensaje de confirmación para deshabilitar la subcategoría

67. Interfaz 16.5.3 (Ver subcategoría)



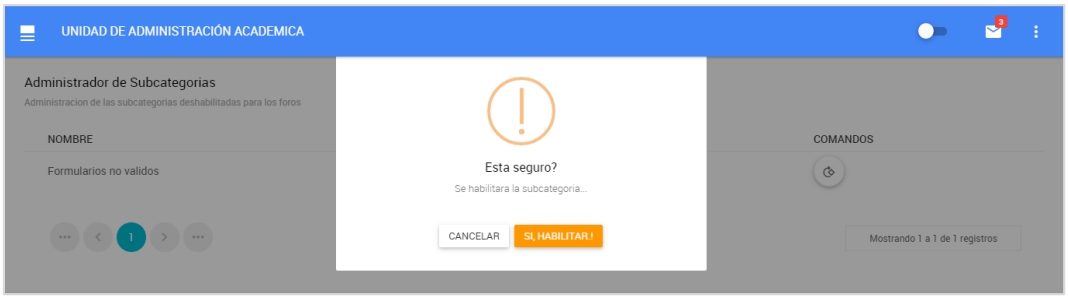
Interfaz que muestra la información vigente de una subcategoría

68. Interfaz 16.6 (Subcategorías deshabilitadas)



Panel de administración de subcategorías deshabilitadas con las opciones de volver a habilitar en la columna comandos

69. Interfaz 16.6.1 (Habilitar subcategoría)



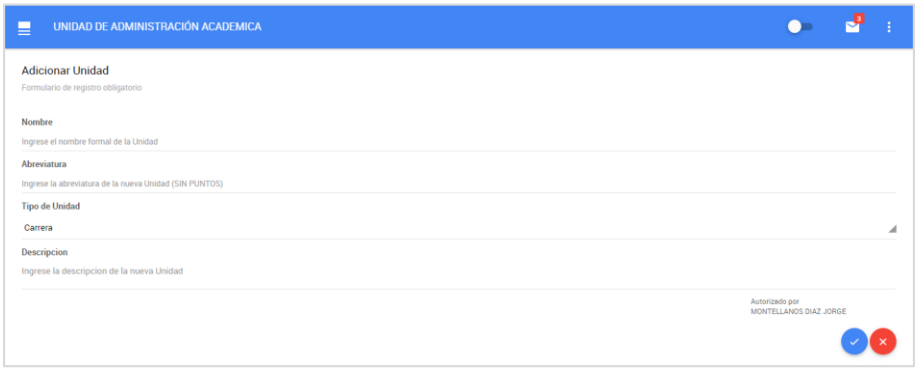
Mensaje de confirmación antes de habilitar una subcategoría

70. Interfaz 17 (Unidades académicas)



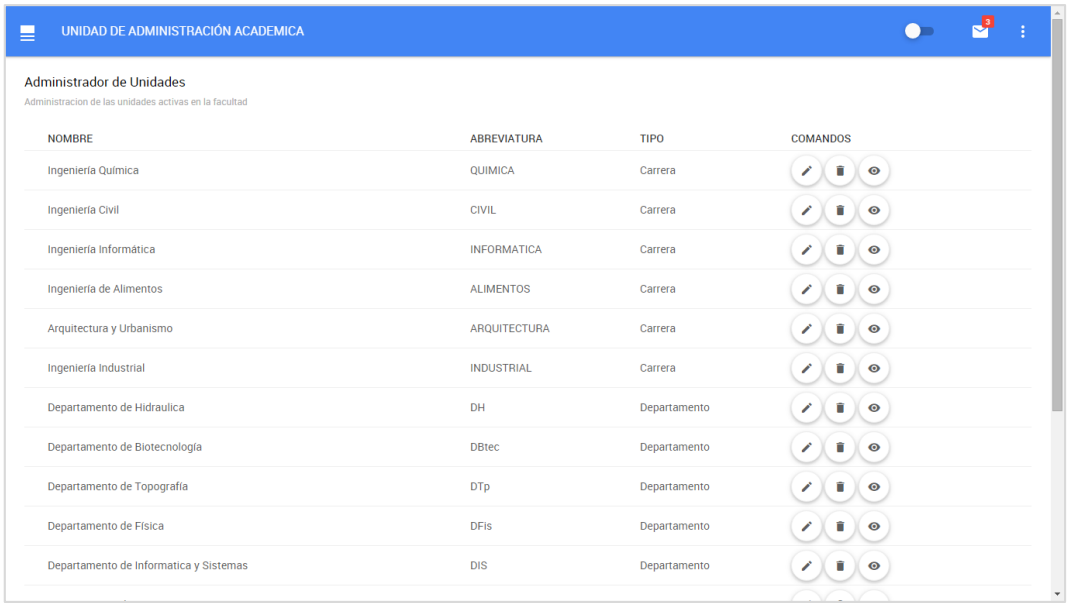
Menú con las opciones para la administración de la unidad del sistema

71. Interfaz 17.1 (Agregar unidad)



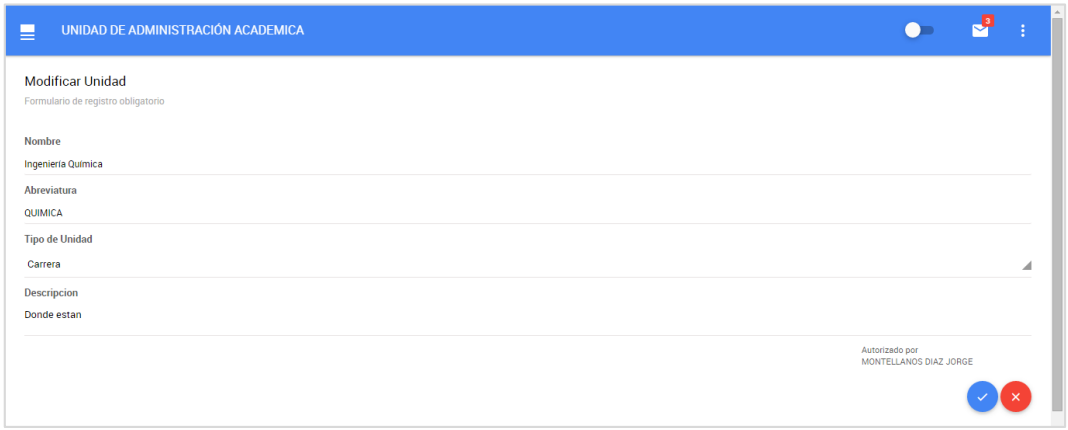
Formulario de llenado obligatorio para dar de alta una nueva unidad académica, se puede cancelar en cualquier momento

72. Interfaz 17.2 (Administración de unidades)



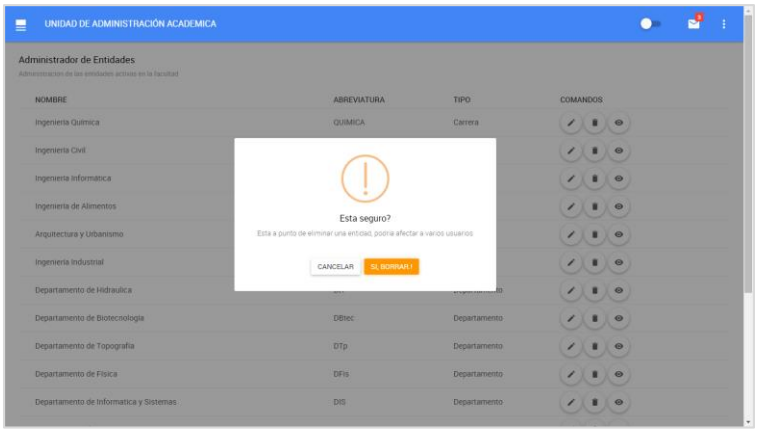
Listado de unidades activas en la facultad, la columna comandos muestra las opciones que son modificar, eliminar y ver respectivamente

73. Interfaz 17.2.1 (Modificar unidad)



Formulario cargado con la información actual de la unidad que se pretende modificar

74. Interfaz 17.2.2 (Deshabilitar unidad)



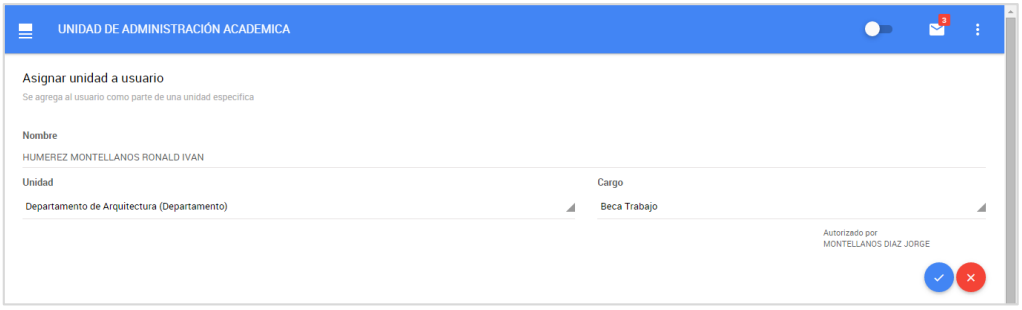
Mensaje de confirmación para dar de baja una unidad académica

75. Interfaz 17.2.3 (Ver unidad)



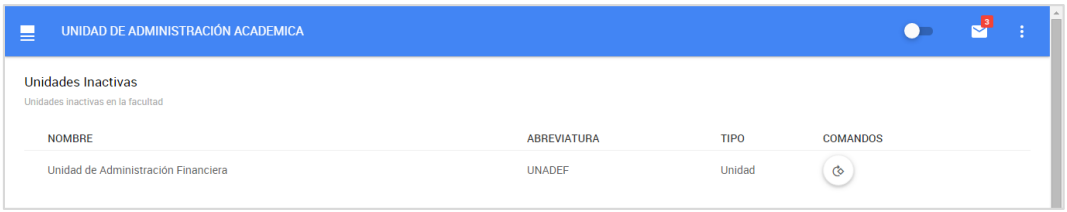
Información detallada de la unidad seleccionada

76. Interfaz 17.3 (Asignar unidad)



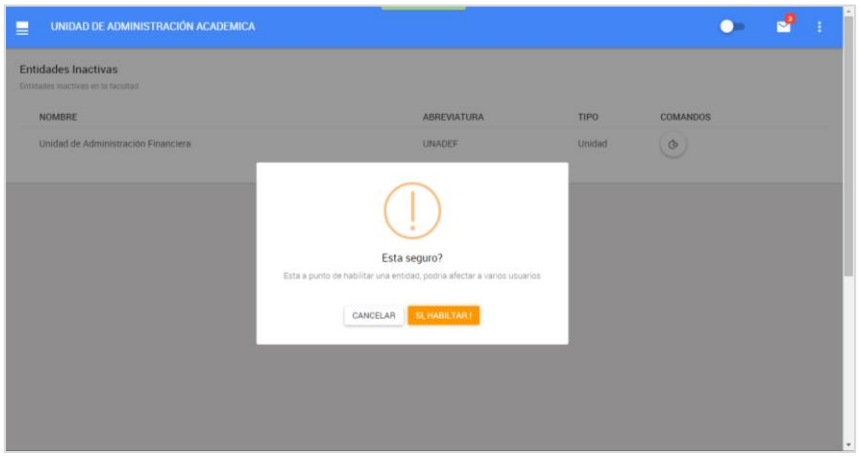
Formulario de asignación de unidad a usuario, se debe llenar con mucho cuidado, pues habilita los roles correspondientes a cada unidad

77. Interfaz 17.4 (Unidades deshabilitadas)



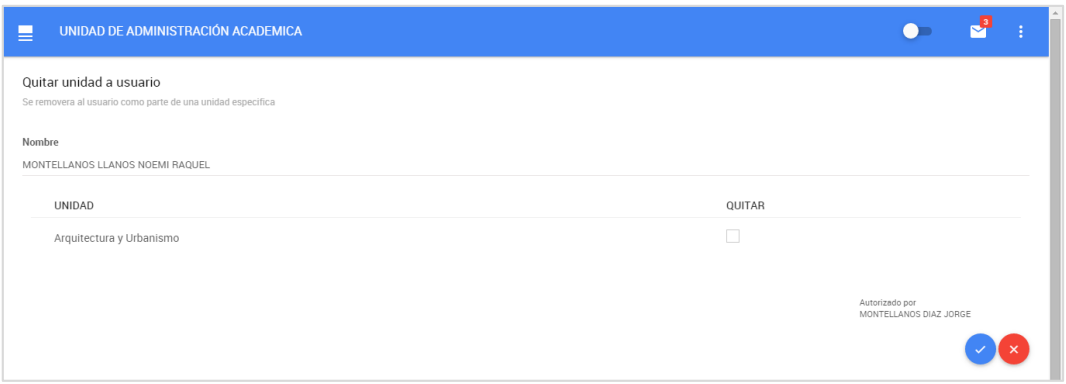
Unidades que fueron dadas de baja, se muestran con la opción de poder habilitarlas de nuevo

78. Interfaz 17.4.1 (Habilitar unidad)



Mensaje de confirmación para dar de alta a una unidad

79. Interfaz 17.5 (Quitar asignación de unidad)



Interfaz que muestra al usuario con las unidades que se les puede quitar

CAPACITACION




FITCH


FITCH









