# Students Representation

DEL5 es una aplicación destinada al uso con fines educativos. Las organizaciones de representación estudiantil tienen muchos asuntos que gestionar en cortos periodos de tiempo. Estas actividades son, por ejemplo: quejas sobre el profesorado, problemas con infraestructuras, manejo de las diferentes asociaciones de la escuela, control de inventario de salas comunes, permutas de alumnos de un grupo a otro, entre otros.

# C – Level Requirements

## Information requirements

1. The actors of the system are administrators, members, and students. For every actor, the system must store a name, an optional middle name, a surname, an optional photo, an email, an optional phone number, an optional address, and an arbitrary number of social profiles. The system must store the following data regarding such profiles: a nick, the name of the social network, a link to a profile in that social network.
2. Actors can exchange messages. For every message, the system must keep track of the send-er, the recipient, the moment when it was sent, the subject, the body, its priority, and some optional tags. Priorities are HIGH, NEUTRAL, or LOW, other are not expected. Every actor has the following message boxes: in box, out box, trash box, notification box, and spam box. When an actor receives a message, it gets to the in box unless the system flags it as spam, in which case it gets to the spam box. When he or she sends a message to another user, it’s saved to the out box. When an actor removes a message from a box other than trash box, it is moved to the trash box; when he or she re-moves it from the trash box, then it is actually removed from his folders; when a message is not in any folder, it’s actually removed from the system. The previous boxes are pre-defined and the actors must not be allowed to delete them, to change their names, or to move them. Actors are allowed to create new boxes that they can manage arbitrarily; managing boxes includes nesting a folder within another folder. Note that a message may be stored in several boxes, but the system must keep a unique copy.
3. Phone numbers should adhere to the following patterns: “+CC (AC) PN”, "+CC PN", or "PN": “+CC” denotes a country code in range “+1” up to “+999”, “(AC)” denotes an area code in range “(1)” up to “(999)”, and “PN” denotes a number that must have at least four digits. Phone numbers with pattern “PN” must be added automatically a default country, which is a parameter that can be changed by administrators. Note that phone numbers should adhere to the previous patterns, but they are not required to. Whenever a phone number that does not match this pattern is entered, the system must ask for confirmation; if the user confirms the number, it then must be stored.
4. Un estudiante puede registrar una queja. Para cada queja, deberá almacenar un ticker, un título, una descripción, la fecha en que se crea y algunos documentos adjuntos. Cada vez que un estudiante emita una queja, deberá proveer carnet de estudiante. Cada carnet de estudiante deberá llevar un número de identificación, el centro donde estudia y los últimos cuatro dígitos de la parte superior trasera. La queja puede guardarse en modo borrador. Si la queja se guarda en modo final, esta no puede editarse ni eliminarse y podrá ser atendida por cualquier miembro y su estado pasa a pendiente.
5. Cada miembro puede auto asignarse una queja que no este atendida en el sistema. Sin perjuicio de poder asignarse esta a otros miembros a la vez.
6. Las quejas serán atendidas por un miembro. Para cada queja deberá determinar si es ACEPTADA o RECHAZADA. En caso de ser aceptada, deberá adjuntar una ley. Para cada ley, deberá escribir un título, un texto del articulo al que se refiere y algunos adjuntos opcionales. Si es rechazada, dar un texto que justifique porque no procede. Las quejas aceptadas deberán mostrarse en verde, las rechazadas en rojo, y las pendientes en naranja.
7. Las quejas podrán ser de diferentes tipos. Quejas de profesores, infraestructura, departamentos, organización docente o incluso por acoso. A su vez, estas pueden tener otra serie de ítems.
8. Cada miembro tiene un buscador para encontrar quejas que no se haya asignado. El buscador variara según el siguiente criterio: fecha de registro, ticker, descripción o tipo de queja. Si el criterio especificado no se encuentra, deberá devolverse toda la lista de quejas.
9. Si la queja es aceptada, tanto el miembro como el estudiante puede realizar diferentes comentarios para clarificar toda la información relacionada con la queja.

## Functional requirements

1. Un actor que no esta registrado
   * Registrarse en el sistema como miembro o estudiante.
2. Un actor que esta registrado
   * Hacer lo mismo que uno que no esta registrado.
   * Editar sus datos personales
3. Un actor que esta registrado como miembro
   * Crear, Editar, Eliminar y Listas las quejas. Además de realizar comentarios a ellas. Los comentarios podrán publicarse siempre y cuando estos estén en modo final.
   * Asignarse una queja no asignada por otros miembros del sistema.
   * Buscar las diferentes quejas no asignadas en el sistema.
4. Un actor que esta registrado como Estudiante
   * Publicar las quejas que considere, así como comentar aquellas que no estén cerradas. Los miembros pueden cerrar aquellas quejas que piensen que no se puede hacer nada mas.
5. Un actor que está registrado como Administrador
   * Crear nuevos administradores.
   * Crear, Editar, Eliminar y Listar los tipos de quejas. Un tipo de queja se puede eliminar siempre y cuando mas de dos quejas no tengan dicho tipo.
   * Crear, editar, eliminar y listar los tipos de leyes.
   * Ban and Unban suspicious actors.
   * Ver un dashboard con los siguientes valores:
     + Media, Desviacion Tipica, Minimo y Maximo de las quejas publicadas por los estudiantes.
     + Media, Desviacion Tipica, Minimo y Maximo de las quejas atendidas por los miembros.
     + Visualizar un histograma de los tipos de quejas publicadas.
     + Media, Desviación Tipica, Minimo y Maximo del número de resultados por finders.

## Non - Functional requirements

1. The system must be available in English and Spanish. (Unless otherwise stated, the data are not required to be available in several languages, only the messages that the system dis-plays.)
2. The system must be easy to customise at run time. The customisation includes, but is not limited to: the name of the system (it’s “DEL5 – Students Representation” by default); the banner shown at the header (it’s the one available at <banner> by default); the message that is shown on the welcome page (“Welcome to DEL5! The best website for student rights defense” is the default welcome message in English; “¡Bienvenidos a DEL5 – El mejor lugar para la defensa de los derechos de los estudiantes” is the default welcome message in Spanish); a list of spam words (it’s “sex”, “viagra”, “cialis”, “one million”, “you’ve been selected”, “Nigeria”, “sexo”, “un millón”, and “ha sido seleccio-nado” by default).
3. Photos are not required to be stored in the database, but links to external systems like Pin-terest.com or Flickr.com, just to mention a couple of examples.
4. When an actor gets a message that contains a spam word, it must be stored in the spam box instead of the input box.
5. Tickers must adhere to the following pattern: “yymmdd-xxxxxx”, where “yymmdd” refers to the year, month, and day when the corresponding entity is registered, and “xxxxxx” to a random uppercase alpha-numeric string. No two entities may have the same ticker since it’s assumed to be a unique external identifier.
6. The maximum number of results that a finder returns is 10 by default. The administrator should be able to change this parameter in order to adjust the performance of the system. The absolute maximum is 100 results.
7. Attachments are not required to be stored by the system, but their URLs to external storage systems like Drive.com or Dropbox.com, to mention a few examples.
8. The results of a finder are cached for one hour by default. The administrator should be able to configure that period at will in order to adjust the performance of the system. The mini-mum time’s one hour and the maximum time’s 24 hours.

# B – Level Requirements

## Information requirements

1. Existe un nuevo tipo de actor: colaboradores, que ayudan a los distintos miembros perteneciendo a sus diferentes organizaciones y preparando eventos.
2. Los colaboradores pueden manejar sus trayectorias. Una trayectoria se compone de un único informe personal, cero o más informes de trabajos, cero o más informes de estudios universitarios, cero o más informes de estudios obligatorios o grados medios y bachillerato, cero o más informes extras.
3. Para cada informe el sistema debe almacenar un título del informe y una fecha de creación del documento. Para cada informe personal el sistema debe almacenar un nombre del actor, un apellido del actor, número de teléfono y su dirección.

Para cada informe de trabajo, se debe almacenar una fecha de comienzo, una opcional fecha de fin, un nombre de la empresa u organización en la que se haya registrado el trabajo y una descripción opcional del trabajo realizado en la misma.

Para el informe de estudios universitarios se debe almacenar un curso actual en el que se encuentre matriculado el alumnado, una fecha de comienzo de estudios en la universidad y porcentaje de créditos aprobados.

Para el informe de estudios obligatorios o grados medios y bachillerato, se debe almacenar el tipo de estudios realizados: (ESO; Bachillerato, Grado Medio, Grado Superior), fecha de inicio de estudios, fecha de fin y una opcional nota media. Por cada informe extra, se debe añadir una descripción.

1. Los miembros pueden crear organizaciones. Cada organización debe poseer un nombre, una descripción y una fecha de creación. Además, los diferentes colaboradores pueden adherirse a una organización u otra dependiendo de sus intereses.
2. Una vez que un colaborador está dentro de una organización, éste puede crear eventos. Un evento consta de un título, una descripción, una fecha de realización y un estado. Un evento es creado por un colaborador en modo no final y su estado pasa a ser PENDIENTE, y solo pasa a final cuando el miembro creador de la organización en la que este colaborador se encuentra, edita su estado a ACEPTADO o RECHAZADO.
3. Los estudiantes, colaboradores y miembros que asisten al evento pueden crear notas al mismo. Una nota consta de un valor entre 0 y 10 y una descripción opcional sobre la nota elegida.

## Functional requirements

1. Un actor que no está autentificado debe ser capaz de:
   * Registrarse como colaborador.
   * Listar y mostrar los diferentes eventos y ver sus notas. Además, el actor debe poder navegar hasta el perfil del colaborador que ha creado el evento, lo que incluye su información personal.
2. Un actor que está autentificado como estudiante debe ser capaz de:
   * Crear una nota referente a uno de los eventos a los que haya asistido.
3. Un actor que está autentificado como colaborador debe ser capaz de:
   * Manejar su trayectoria, lo cual incluye listar, crear, editar y borrar los informes.
   * Listar las diferentes organizaciones existentes y unirse a una de ellas.
   * Manejar sus eventos, lo que incluye crearlos, mostrarlos, editarlos y borrarlos. Cuando un evento es creado, su estado pasa a ser PENDIENTE.
   * Crear una nota referente a uno de los eventos a los que haya asistido.
4. Un actor que está autentificado como miembro debe ser capaz de:
   * Manejar las diferentes organizaciones, lo que incluye listarlas, mostrarlas, crearlas, editarlas y borrarlas. Hasta que una organización no esté en modo final, los colaboradores no podrán unirse a ella.
   * Listar los eventos creados en su organización agrupados por estado y editarlos. Un evento no está en modo final hasta que su estado pase de PENDIENTE a ACEPTADO o RECHAZADO.
   * Crear una nota referente a uno de los eventos a los que haya asistido.
5. Un actor que está autentificado como administrador debe ser capaz de:
   * Display a dashboard with the following information:
     + The minimum, the maximum, the average, and the standard deviation of the number of collaborators per organizations.
     + The minimum, the maximum, the average, and the standard deviation of the number of notes per event.
     + The ratio of members with an organization.
     + The top-three de eventos más cercanos a la fecha actual.

## Non - Functional requirements

1. El porcentaje de créditos es un número entero entre 0 y 100.
2. Los datos pertenecientes al informe personal deben ser coherentes respecto a los datos registrados en el sistema del actor correspondiente.
3. Tenga en cuenta que las notas no pueden ser actualizadas o borradas una vez que están guardadas en base de datos. Por lo que, es requerida una doble confirmación antes de guardar esta entidad.

# A – Level Requirements

## Information requirements

1. Los colaboradores pueden realizar permutas entre ellos. Una solicitud de permuta consta de sender, receiver, organización a la que quiere cambiarse, teléfono de contacto, un estado y una descripción opcional. Un colaborador envía una solicitud de permuta y su estado cambia a PENDIENTE y el colaborador que la recibe, cambia su estado a ACEPTADA o RECHAZADA.
2. There’s a new kind of actors in the system: sponsors, who support some tutorials by means of sponsorships.
3. The system must store the following data regarding sponsorships: a URL to a banner, a link to a target page, and a valid credit card.

## Functional requirements

1. Un actor que no está autentificado debe ser capaz de:
   * Registrarse como sponsor.
2. Un actor que está autentificado como colaborador debe ser capaz de:
   * Listar las diferentes solicitudes recibidas y enviadas, además de una lista con todos los colaboradores de una determinada organización.
   * Crear una solicitud de permuta y enviarla a otro colaborador.
   * Editar una solicitud de permuta recibida, lo que incluye cambiar su estado de PENDIENTE a RECHAZADA o ACEPTADA.
3. Un actor que está autentificado como sponsor debe ser capaz de:
   * Manejar sus diferentes sponsorships, lo que incluye mostrar, listar, crear, editar y borrar estos.
4. Un actor que está autentificado como un administrador debe ser capaz de:
   * Launch a process that computes a score for every event. The score is computed building on the notes that they’ve got. The system must analyse the notes in the events and compute la media de estos. A raíz de esta media, si es superior a 5, se considera como score 1 y si es inferior a 5, el score se considera de –1.

## Non - Functional requirements

1. Una vez que una solicitud de permuta es aceptada, el resto de solicitudes cambian su estado a rechazada.
2. Whenever a event is displayed, a random sponsorship must be selected and its banner shown, if any. Banners must be shown as little intrusively as possible.
3. The system must accept the following credit card makes: VISA, MCARD, AMEX, DINNERS, and FLY. Other makes are expected to be accepted in future. The administrator should be able to easily manage the list of accepted credit card makes.

# Level A+

1. Para facilitar la organización del usuario, se recomienda la implementación de TimeTree. TimeTree es una aplicación que permite al usuario la creación de diferentes calendarios referentes tanto a su vida personal, como a su trabajo o actividades extra. Los requisitos mínimos son los siguientes: permitir al usuario acceder a su cuenta personal, creación de diferentes eventos, a la vez que se registran automáticamente los eventos creados por las diferentes asociaciones.

Es por ello que, se debe producir lo siguiente:

* + La implementación sobre el caso de uso definido.
  + Un informe, de longitud máxima de 1000 palabras, explicando su implementación y como el usuario paso a paso es capaz de trabajar con esta herramienta en nuestro sistema.

# Level A++

1. Hay muchas veces en que la carga de datos es costosa y pesada, bien por la cantidad de usuarios loggeados o bien por la cantidad de datos a cargar. Como buenas prácticas de Spring, se recomienda utilizar un sistema de cache para evitar la llamada repetitiva de datos en múltiples ocasiones.

Es por ello que, se debe producir:

* + La implementación sobre el caso de uso que se desee.
  + Un informe, de longitud máxima de 1000 palabras, explicando su implementación y como Spring trabaja con este sistema.