

Sistema para el registro y control de visitas y estancias académicas

Información del reporte:

Licencia Creative Commons



El contenido de los textos es responsabilidad de los autores y no refleja forzosamente el punto de vista de los dictaminadores, o de los miembros del Comité Editorial, o la postura del editor y la editorial de la publicación.

Para citar este reporte técnico:

Aguilar Sierra, A. (2024) Sistema para el registro y control de visitas y estancias académicas. *Cuadernos Técnicos Universitarios de la DGTIC*, 2 (1), páginas (111 - 122).

<https://doi.org/10.22201/dgtic.ctud.2024.2.1.36>

Alejandro Aguilar Sierra

Instituto de Ciencias de la Atmósfera
y Cambio Climático

Universidad Nacional Autónoma de México

asierra@unam.mx

ORCID: 0000-0003-1018-8521

Resumen:

El Sistema de Estudiantes y Estancias Académicas se desarrolló para registrar el acceso de estudiantes y visitantes académicos al Instituto de Ciencias de la Atmósfera y Cambio Climático. Agiliza su entrada al no tener que hacer fila para anotarse en la libreta de ingreso y permite mantener un control sobre las asistencias al instituto. Al entrar al instituto, el visitante muestra su credencial digital al sensor óptico y si es reconocida, aparece su ficha correspondiente en la pantalla del vigilante en turno. El sistema fue desarrollado en la plataforma *Ruby on Rails* y opera desde una máquina virtual y en la estación de vigilancia.

Palabras clave:

Visitantes externos, control de acceso, credencial digital, seguridad.

1. INTRODUCCIÓN

En el Instituto de Ciencias de la Atmósfera y Cambio Climático (ICAyCC) se imparten cursos de posgrado, proyectos de investigación y numerosos eventos académicos que requieren el constante flujo de personas ajenas al instituto. Cuando hay un evento con un horario específico, como un seminario o un curso, se producen aglomeraciones en la entrada, pues los asistentes deben registrar su ingreso a mano en la libreta de ingreso. Un sistema de registro de ingreso ágil, con el uso de tecnología moderna, puede agilizar significativamente este proceso. Además, para el instituto es muy conveniente mantener una lista de visitantes autorizados, controlar su vigencia y su asistencia.

En enero de 2023 se inició el desarrollo del Sistema de Estudiantes y Estancias Académicas (SEEA). Después de un proceso de pruebas y correcciones, se liberó el 2 de octubre de 2023 y se puso en funcionamiento en un servidor virtual perteneciente al instituto y en la estación de vigilancia en la entrada del mismo.

2. OBJETIVOS

El objetivo principal de este proyecto es contar con un sistema semiautomático para el control del acceso a las instalaciones del ICAyCC de personal externo, ya sean estudiantes, investigadores o profesionales que participen en alguna actividad académica. Los objetivos particulares son:

- Simplificar actividades administrativas como el registro de fecha y hora de entrada.
- Reducir el tiempo de registro en la estación de vigilancia en la entrada y evitar aglomeraciones cuando acuden muchas personas al mismo tiempo.
- Contar con un control de acceso de personal académico externo al ICAyCC cuyas actividades tienen una vigencia determinada.

Los alcances del sistema son:

- Puede usarlo cualquier persona que deba realizar alguna actividad académica en el instituto en un periodo determinado.
- Se genera una credencial digital solamente válida para el sistema, presentable por medio de la pantalla del teléfono celular individual, con opción a generar una física a petición del visitante. Dicha credencial tiene un código de barras a ser leído por un lector instalado en la estación de vigilancia en la entrada del instituto.
- Las actividades académicas consideradas serán: clases, servicio social, desarrollo de tesis, servicios profesionales, estancias posdoctorales o colaboración en algún proyecto de investigación.
- Todas estas actividades deberán tener asociado a un académico responsable adscrito al instituto, así como una vigencia.
- Debe ser responsivo, fácil de usar tanto en pantalla normal como desde un teléfono celular.

3. DESARROLLO TÉCNICO

3.1 ANTECEDENTES

A principios de 2018 hacía falta un registro semiautomático de estudiantes que acudían al entonces Centro de Ciencias de la Atmósfera y facilitar así el censo de la población estudiantil. Se recibió el código fuente del sistema llamado *students*, usado por varios años en el vecino Instituto de Física, con la misión de adaptarlo para el uso de nuestro centro. Se invirtió un esfuerzo considerable para implementarlo, pero no fue posible pues se requerían versiones de bibliotecas que ya eran obsoletas. Se desarrolló una versión nueva que se denominó Sistema de Control de Acceso de Estudiantes, SICA, que estuvo en operación de 2018 a principios de 2020, cuando se dispuso el encierro debido a la pandemia de COVID-19.

Como el SICA fue probado exitosamente durante dos años y tenía prestaciones similares a las nuevas necesidades, era un buen precedente a considerar. Dado que los objetivos eran diferentes y algunos de sus catálogos ya eran obsoletos, se decidió desarrollar un sistema nuevo pero tomando en cuenta la experiencia obtenida con el SICA.

3.2 ROLES DE USUARIO

El SEEA tiene 3 roles de usuario, el visitante, el vigilante y el administrador, con las siguientes responsabilidades:

1. Visitante

- Su primera interacción con el sistema es un formulario en línea que debe llenar y subir una fotografía reciente de su rostro, que puede tomar en ese mismo momento, con la cámara de su teléfono celular. Debe elegir una actividad y un académico responsable. Recibirá a vuelta de correo electrónico su solicitud en PDF que deberá firmar el académico responsable y será verificada por la Secretaría Académica. En caso aprobatorio, su registro será activado y recibirá por correo electrónico el enlace a su credencial digital. En caso de que no desee usar su teléfono celular, podrá pedir una credencial física.
- Al ingresar al instituto, mostrará dicha credencial al lector de código de barras en la entrada. Si es reconocido por el sistema y su cuenta es vigente, aparecerá su ficha de identificación con su fotografía, su nombre, su institución de procedencia, su actividad y el nombre del académico responsable. De otro modo, debe registrarse manualmente en la lista de la manera convencional.

2. Vigilante

- En la estación de vigilancia habrá un lector de código de barras conectado a una computadora personal dedicada, incrustada en la parte trasera de la pantalla. No contará con teclado.
- Cuando ingrese un visitante y muestre su credencial digital, el vigilante verá su ficha de identificación. El vigilante podrá verificar la identidad del visitante por su fotografía en la ficha.
- Si al pasar su credencial, el sistema no muestra una ficha de identificación, el vigilante deberá solicitar al estudiante que se registre manualmente.

3.Administrador

- Podrá dar de alta, consultar o eliminar actividades y académicos.
- Activar o desactivar visitantes. Un visitante no activado existirá en la base de datos pero su ficha de identidad no aparecerá en la estación de vigilancia.
- Consultar la fecha y hora de las entradas de los estudiantes y recuperar esta información para su uso posterior.
- Generar reportes en formato CSV de ingresos de visitantes y de la lista de visitantes registrados.

Para cumplir con las políticas de privacidad (UNAM 2019), se minimizó la información personal tanto de los visitantes como de los académicos responsables, en la base de datos.

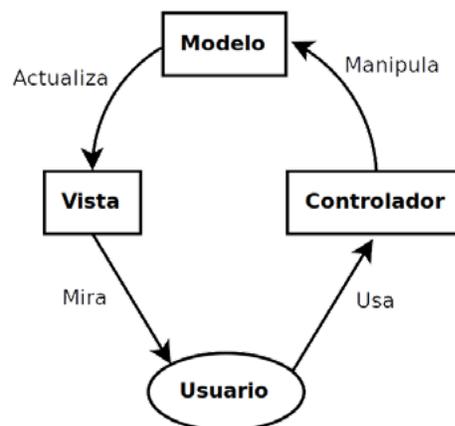
3.3 IMPLEMENTACIÓN

Dados los antecedentes, se consideró conveniente desarrollar el SEEA con el mismo marco de trabajo que la aplicación anterior SICA, *Ruby on Rails* (RoR), (Hansson, 2024, Ruby y Thomas, 2023), que aplica un patrón moderno de arquitectura de software llamado Model View Controller (MVC) en el lenguaje de programación orientada a objetos *Ruby* (Matsumoto, 2024). Permite gestionar separadamente el flujo de trabajo (controlador), la interacción con el usuario (vista), y la gestión de la base de datos (modelo). En una *aplicación web*, el modelo y el controlador actúan en el servidor remoto y la vista en el cliente local, un navegador web.

En la figura 1 se muestra el flujo de trabajo del usuario con el sistema. Se envían órdenes al sistema por medio de elementos web como botones y cajas de texto, que son recibidas por el controlador, quien a su vez manipula los datos y los pasa al modelo, cuyas actualizaciones pueden ser observadas de nuevo en la vista.

Figura 1

Arquitectura de interacciones con el patrón MVC



RoR proporciona una serie de comandos básicos para crear una aplicación web completa. Creada en 2005, RoR es una plataforma madura aún muy apreciada tanto por desarrolladores como por empresas y se usa en GitHub, Netflix, Airbnb, entre otras (Reddy 2020, Patel 2023, Derevianko 2024).

3.3.1 MODELOS

Cada entidad abstracta se implementa con un modelo. El comando *rails generate scaffold* crea el armazón de código correspondiente a la vista, al modelo y al controlador, a partir del nombre de la entidad y sus componentes, indicados por un nombre y un tipo de datos. Por ejemplo, para crear el código correspondiente al visitante, se usa el siguiente comando:

```
rails generate scaffold Visitor name:string \  
  surname_paternal:string surname_maternal:string account:string \  
  email:string institution:string nationality:references \  
  status:integer \  
  academic:references activity:references \  
  begin:date end:date
```

Contiene las tres partes del nombre del visitante, su correo electrónico personal, su estatus, un número de cuenta, una referencia a la tabla de académicos y otra a la de actividades. Las fechas *begin* y *end* corresponden a la vigencia. Si la fecha actual no entra dentro de la vigencia, el sistema automáticamente desactivará esa cuenta.

Para el rol de administrador se desarrolló el acceso a sesión con contraseña, usando los recursos que proporciona RoR, sin necesidad de instalar ninguna gema¹ externa:

```
rails generate scaffold User name:string password:digest  
rails generate controller Sessions new create  
rails generate controller Admin index
```

Una vez creadas las entidades con sus tablas correspondientes en la base de datos y los armazones de los modelos, vistas y controladores, se procede a codificar propiamente la aplicación.

Como la captura de información del visitante es fundamental, se implementaron varios filtros para prevenir el ingreso de datos incorrectos. En una aplicación web hay dos oportunidades para filtrar datos capturados por el usuario, en el navegador antes de ser enviados al servidor (usualmente con Javascript) y en el servidor antes de que los datos ingresen a la base de datos. En este segundo caso, los filtros se pueden programar en el código del modelo. Donde dice *validates_presence_of* se verifica que el dato correspondiente haya sido enviado. Donde dice *uniqueness* se verifica que el dato no exista previamente en la base de datos.

```
1 validates_presence_of :name  
2 validates_presence_of :surname_paternal  
3 validates_presence_of :email, uniqueness: true  
5 validates_presence_of :activity
```

1 En el lenguaje Ruby, a una biblioteca creada por terceros se le llama “gema”.

```
6 validates_presence_of :academic
8 validates_presence_of :begin_date
9 validates :end_date, comparison: { greater_than: :begin_date, message:
'debe ser posterior a la fecha inicial' }
10 validates :aviso_privacidad, acceptance: { message: 'debe ser leído y
aceptado' }, on: :create
```

En este segmento de código se validan todos los datos obligatorios, que el *email* sea único, que las fechas sean coherentes y que se haya leído y aceptado el aviso de privacidad.

Es responsabilidad del visitante ingresar un formato correcto de fotografía, con una resolución mínima de 200x200 píxeles y rostro claramente visible y bien iluminado. En la actualidad es muy práctico usar la cámara del teléfono celular para subir la foto en el mismo momento en que se llena el formulario. Un teléfono celular produce imágenes de gran tamaño, típicamente 2000x3000 píxeles, o el usuario podría subir su fotografía en un formato sin comprimir. El recurso *Active Storage* de RoR (Rails Community 2023a) gestiona el manejo de imágenes en el disco sin necesidad de guardarlas en la base de datos. Asociada a este recurso está la biblioteca de procesamiento de imágenes VIPS que permite ajustar el tamaño de la imagen de la fotografía personal a 200x200 píxeles, que es suficiente para el uso del sistema, y guardarla en formato JPEG. Estos recursos no existían en versiones anteriores de RoR y son muy convenientes pues de otro modo se habría tenido que escribir un código más extenso para las operaciones con imágenes.

3.4 CONTROLADORES

Cada entidad tiene su propio controlador cuyos métodos corresponden a una interacción del usuario en el navegador web. Por ejemplo, una consulta a una página web genera la acción *index*, que muestra una lista del contenido de la entidad correspondiente.

```
1 class VisitorsController < ApplicationController
2   before_action :set_visitor, only: %i[ show edit update destroy ]
3   skip_before_action :authorize, only: [:new, :create, :edit, :show, :card]
5   def index
6     if params[:search]
7       @pagy, @visitors = pagy(Visitor.search(params[:search]), items: 10)
8     else
9       @pagy, @visitors = pagy(Visitor.all, items: 10)
10    end
11    ...
20 end
```

Las líneas 2 y 3 de este listado limitan las operaciones que pueden hacer usuarios no autenticados. En el método *index* en las líneas 7 y 9 se filtra la lista de visitantes, que puede ser muy larga, a 10 visitantes por página y en caso de que se haya efectuado una búsqueda, mostrará solamente los resultados de la misma.

El administrador puede ver las listas de todas las entidades, ver y editar los detalles específicos de cada entidad o eliminarla por completo del sistema, lo cual es una operación irreversible, por lo que siempre aparece una ventana de alerta que solicita al usuario confirmar la operación.

La consola del vigilante sólo le permite hacer dos operaciones: 1) autenticarse con una contraseña especial que introduce por medio de su credencial con código de barras al inicio del día, y 2) registrar la entrada de un visitante por el mismo medio, una lectura de credencial digital con código de barras, que corresponde a la acción *paso*. La consola del vigilante no tiene teclado y no puede hacer ninguna otra operación. El registro de entrada, *visitor_entry*, asocia la identidad del visitante con la fecha y hora actual.

```
1 def paso
2   unless params[:search].nil?
3     @account = params[:search][0..Rails.configuration.account_size]
4     @visitor = Visitor.find_by_account(@account)
5     if @visitor and @visitor.activo and @visitor.vigente
6       visitor_entry = VisitorEntry.new(:visitor => @visitor, :entry =>
                                         Time.now.localtime)
7       visitor_entry.save
8     else
9       @visitor = nil
10    end
11  end
12 end
```

En este fragmento, el parámetro *search* contiene el resultado de la lectura del código de barras de la credencial. En la línea 3 se asigna a la variable *account* un número predefinido de cifras (la variable *account_size*) y de encontrar al visitante correspondiente a la cuenta en la línea 4, se verifica en la línea 5 que su cuenta esté activa y que sea vigente. De serlo, se almacena la entrada, que como ya mencionamos, asocia una referencia del visitante con la fecha y hora de la visita.

3.5 VISTAS

A cada una de las acciones del controlador corresponde una vista en el navegador web del usuario. Las vistas usan una hoja de estilo CSS con los ajustes necesarios para hacerlas responsivas, de modo que puedan ser usadas cómodamente con cualquier dispositivo (Leal, 2023). El visitante solo puede ver el formulario que debe llenar para generar su solicitud (figura 2). Al llenarla recibirá a vuelta de correo electrónico un PDF con la solicitud a ser firmada por el académico responsable. Esta solicitud es recibida por la Secretaría Académica, que activa la cuenta en caso de ser aprobada. Cuando es activada, el visitante recibe su credencial digital (figura 3) por correo electrónico con un número de cuenta único y exclusivo, generado aleatoriamente por el sistema.

Figura 2

Pantalla única sin autenticación con el formulario de solicitud de ingreso al sistema



The screenshot shows a web browser window with the URL <https://seea.atmosfera.unam.mx>. The page header includes the logos of the Universidad Nacional Autónoma de México (UNAM) and the Instituto de Ciencias de la Atmósfera y Cambio Climático (ICAYCC), along with the text "SISTEMA DE ESTUDIANTES Y ESTANCIAS ACADÉMICAS (SEEA)".

Sistema de registro y acceso

El Instituto de Ciencias de la Atmósfera y Cambio Climático (ICAYCC) cuenta con diferentes actividades docentes y de formación de recursos humanos. Es requisito indispensable el registro de los estudiantes, estancias académicas, servicios profesionales y otros usuarios temporales que asisten de manera regular a nuestras instalaciones por periodos definidos. Para realizar este trámite es necesario registrarse en línea y entregar este formato con firma autógrafa del académico responsable en la Secretaría Académica del ICAYCC.

Form fields:

- Nombre:
- Apellido paterno:
- Apellido materno:
- Correo electrónico:
- Institución de procedencia:
- Actividad:
- Período de estancia: a
- Académico/a responsable: No file selected.
- Foto (formato JPEG):
- Nacionalidad:

Lei y estoy de acuerdo con el [Aviso de privacidad del ICAYC.](#)

Figura 3

Ejemplo de credencial digital generada por el sistema



En la estación de vigilancia se ve la pantalla con la entrada de texto para ingresar el número de cuenta, véase figura 4.

Figura 4

Pantalla base de la estación del vigilante



The screenshot shows the base station screen for the vigilante. It features the logos of UNAM and ICAYCC, and the text "SISTEMA DE ESTUDIANTES Y ESTANCIAS ACADÉMICAS (SEEA)". Below this is a search bar with a "Buscar" button.

Cuando el visitante expone su credencial al lector de código de barra, si es reconocida y vigente, se mostrará la ficha correspondiente en la pantalla del vigilante, véase figura 5.

Figura 5

Ejemplo de ficha de visitante con credencial reconocida



El administrador puede ver la lista de todos los visitantes registrados, sus datos, su foto, su actividad, su académico responsable y su vigencia. Puede activar o desactivar su cuenta con un botón en la columna de estatus, véase figura 6.

Figura 6

Pantalla con la lista de visitantes. Los nombres en la versión de prueba son simulados



Puede consultar la lista de entradas, en las que aparecerá el nombre, el número de cuenta y la fecha y hora de ingreso al instituto, véase figura 7.

Figura 7

Lista de accesos. Se ocultan los nombres y números de cuenta reales



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

 INSTITUTO DE CIENCIAS DE LA

ATMÓSFERA

 Y CAMBIO CLIMÁTICO

SISTEMA DE ESTUDIANTES Y ESTANCIAS ACADÉMICAS (SEEA)

Consulta de accesos de visitantes

Fechas del 01 / 11 / 2023 al 01 / 12 / 2023

Visitante	ID Generado	Día y hora de ingreso
Juan Diego Rodríguez Rodríguez	100000000000	2023-12-01 10:53:40 -0600
María Isabella Miranda Sánchez	100000000000	2023-12-01 10:02:13 -0600
Thomas Elizabeth Arredondo Pineda	100000000000	2023-12-01 09:42:17 -0600
Diego Antonio Reyes	100000000000	2023-12-01 08:58:51 -0600
Alfonso José García Aguilera	100000000000	2023-12-01 08:58:47 -0600
Carla Gabriela Velasco Jiménez	100000000000	2023-11-30 16:46:39 -0600
Carla Gabriela Velasco Jiménez	100000000000	2023-11-30 16:46:39 -0600
Diego Sebastián Ortega Landín	100000000000	2023-11-30 16:41:00 -0600
Andrés Sebastián Velasco Jiménez	100000000000	2023-11-30 16:23:00 -0600
Ignacio José Torres	100000000000	2023-11-29 10:12:22 -0600

· Prev 1 2 3 4 5 ... 10 Next ·

3.6 SEGURIDAD

Después de las pruebas de usuario, la Coordinación de Seguridad de la DGTIC identificó algunos problemas de seguridad, que fueron corregidos y validados por la DGTIC; entre ellas se encontraron los siguientes hallazgos:

- Todas las acciones administrativas requieren autenticación mediante usuario y contraseña.
- Los números de cuenta en las credenciales tienen 12 cifras, lo que las hace prácticamente una contraseña segura.
- El sistema sólo responde al protocolo seguro HTTPS.
- La solicitud en PDF y la credencial digital se envían únicamente al interesado, a su correo electrónico personal.
- Para verificar que el solicitante es humano, se utiliza la gema *Humanizer*, muy recomendada en la comunidad RoR y no depende de una empresa externa como el *recaptcha* de Google.
- Se instalaron las versiones más recientes del software y las configuraciones seguras de Apache2 (Apache Software Foundation, 2023) y Passenger, la interfaz de RoR para Apache (Phusion, 2023).
- En el código, toda la información sensible está encriptada como recomienda la guía de seguridad de RoR (Rails Community, 2023b).
- La PC dedicada del vigilante contiene una versión básica del sistema operativo Linux que sólo ejecuta el navegador web *Surf* (Rozet, S., 2009) con la URL preestablecida del SEEA.

4. RESULTADOS

Una vez realizadas las pruebas con usuarios reales y las modificaciones de seguridad, el SEEA se liberó el 2 de octubre de 2023 y se montó en un servidor virtual del instituto y en la estación de vigilancia en la entrada del ICAyCC. Hasta la fecha ha sido utilizado sin contratiempos, se cuenta con ciento veinte visitantes registrados y diariamente se registran del orden de veinte ingresos al instituto, en promedio.

5. CONCLUSIONES

El SEEA permite de manera práctica el acceso controlado de visitantes externos al instituto y sus accesos son registrados en una base de datos que puede ser consultada posteriormente y extender reportes. Este registro de acceso ahorra tiempo a los visitantes que ingresan, simplifica la labor del vigilante y permite a la Secretaría Académica mantener un control sobre el personal académico y estudiantil externo al instituto.

El uso de la plataforma de desarrollo web RoR permitió aprovechar los recursos más actuales para el manejo de imágenes, la seguridad y la responsividad, que resultaron muy convenientes para desarrollar la aplicación de manera ágil, conveniente y segura. El tomar en cuenta las políticas de privacidad y protección de información y las recomendaciones de la DGTIC permitió crear un producto de mayor calidad y conveniencia que sus antecedentes.

REFERENCIAS

- Apache Software Foundation. (2023). *The Apache HTTP Server Project*. Creado en 1997. <https://httpd.apache.org/>.
- Derevianko, L. (2024). *Why Should You [Still] Choose Ruby on Rails for Your Product Development*. <https://mobidev.biz/blog/ruby-on-rails-not-dead-still-good-for-your-product-development>.
- Hansson, D. H. (2023). *Ruby on Rails: A web-app framework that includes everything needed to create database-backed web applications according to the Model-View-Controller*. Creado en el año 2005. <https://rubyonrails.org/>.
- Leal, A. (2023). *The Art of Responsive Web Design: Best Practices and Tips*. <https://buzzvel.com/blog/responsive-web-design-best-practices-tips>.
- Matsumoto, Y. (2023). *Ruby Programming Language*. Creado en el año 2000. <https://www.ruby-lang.org/>.
- Patel, A. (2023). *Why people choose Ruby on Rails?* <https://www.linkedin.com/pulse/why-people-choose-ruby-rails-amit-patel/>
- Phusion B. V. (2023). *Passenger: Enterprise grade web app server for Ruby, Node.js, Python*. Creado en 2004. <https://www.phusionpassenger.com/>.
- Rails Community. (2023a). *Active Storage Overview*. https://guides.rubyonrails.org/active_storage_overview.html

- Rails Community. (2023b). *Securing Rails Applications*. <https://guides.rubyonrails.org/security.html>
- Reddy, S. (2020). *Top 8 Reasons Why Ruby on Rails is The Best Web Development Technology*. <https://medium.com/quick-code/top-8-reasons-why-ruby-on-rails-is-the-best-web-development-technology-3016824e6d34>.
- Rozet, S. (2009). *Surf: A simple web browser*. <https://surf.suckless.org/>.
- Ruby, S., Thomas D. (2023). *Agile Web Development with Rails 7*. The Pragmatic Programmers.
- SQLite Consortium. (2018). *SQLite: SQL Database Engine*. <https://www.sqlite.org/>.
- UNAM. (2019). *Acuerdo por el que se establecen los Lineamientos para la Protección de Datos Personales en posesión de la Universidad Nacional Autónoma de México*, Gaceta UNAM. 25 de febrero de 2019. https://www.red-tic.unam.mx/recursos/2019/2019_Acuerdo_Rectoria_02.pdf
- Vinet, J., Griffin, A., Polyák, L. (2023). *Arch Linux: A simple, lightweight distribution*. <https://www.archlinux.org/>.