

**Instituto Tecnológico de Chetumal**

**Nombre asignatura:** Programación Lógica y Funcional

**Maestra:** Velazco Teh Martha Noemi

**Carrera:** Ingeniería en Sistemas Computacionales

**Semestre:** Octavo Semestre

**Tema:** Entrega Final Clojure

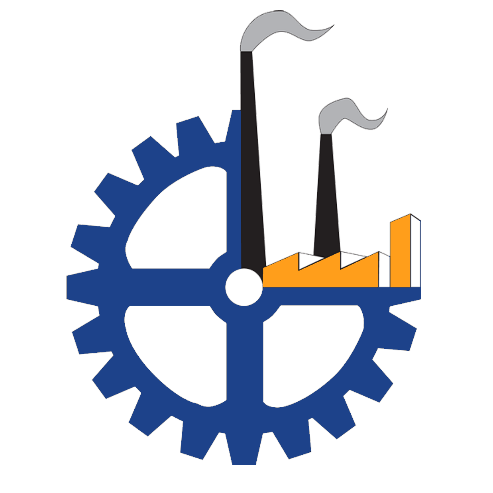
**Alumno:** Alessandro Luizao Reyes Varguez

**Fecha:** 10/06/2024

**Profesor:** Isaías May Canche

**Grupo:** I8U

**INSTITUTO TECNOLÓGICO DE CHETUMAL DEPARTAMENTO DE SISTEMAS Y COMPUTACIÓN**



**Contenido**

[Manual de Usuario 2](#_Toc168897255)

[Ingreso Al Sistema 2](#_Toc168897256)

[Página principal 2](#_Toc168897257)

[Registrar Ceda 3](#_Toc168897258)

[Registrar Entrega 4](#_Toc168897259)

[Cedas 6](#_Toc168897260)

[Entregas 7](#_Toc168897261)

[Manual Tecnico 9](#_Toc168897262)

[Queries 9](#_Toc168897263)

[Backend 12](#_Toc168897264)

[**Frontend** 14](#_Toc168897265)

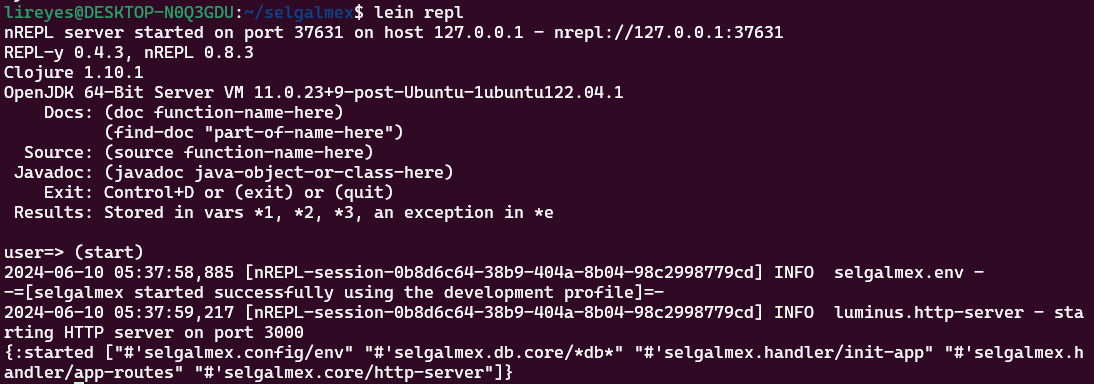
[**home.html** 14](#_Toc168897266)

[**entregas.html** 14](#_Toc168897267)

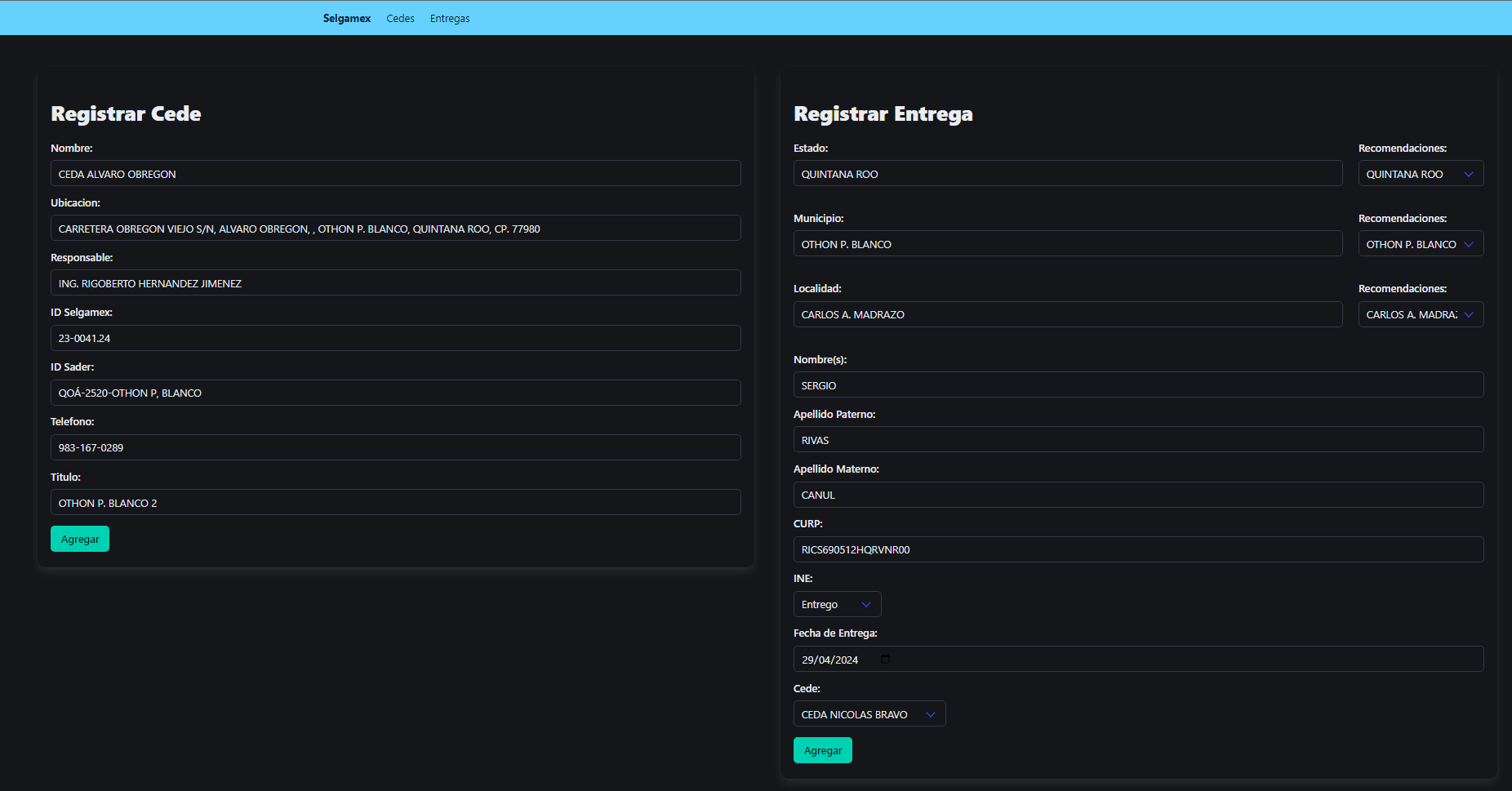
[**cedas.html** 15](#_Toc168897268)

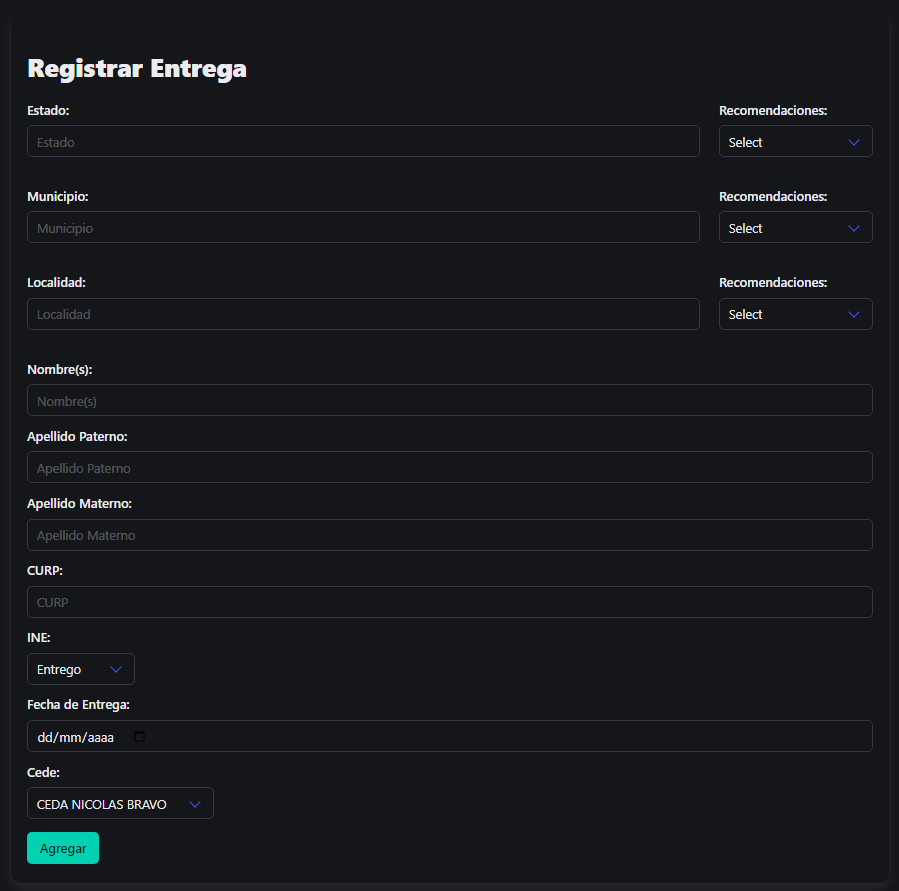
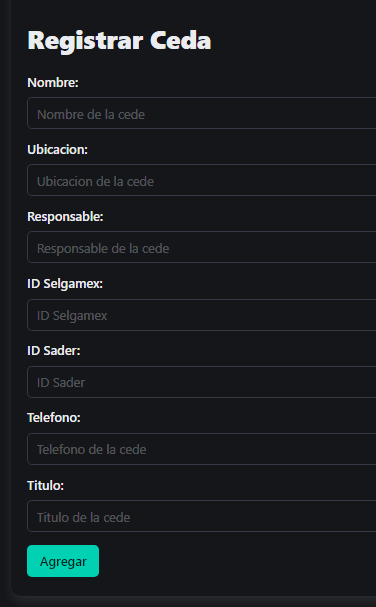
# Manual de Usuario

## Ingreso Al Sistema

Para ejecutar correctamente el sistema, se deberá hacer a través de Ubuntu, ejecutando primer lein repl, para iniciar el proyecto y posteriormente poner “(start)” para iniciar el servidor y en el navegador ingresar con localhost:3000

## Página principal

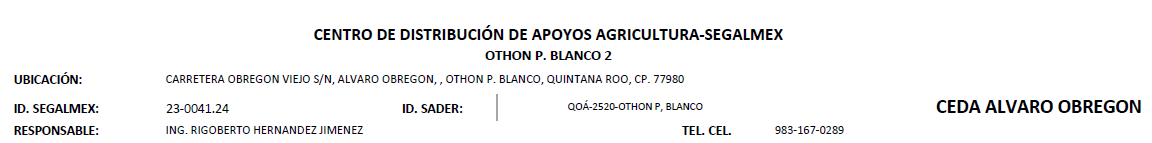


En la página principal, encontraremos 2 formularios, en el del apartado izquierdo, encontraremos el formulario para registrar cedes y del lado derecho uno para el registro de entregas.

### Registrar Ceda

Para el registro de cedes contaremos con el siguiente formulario, donde toda la informacion es requerida.

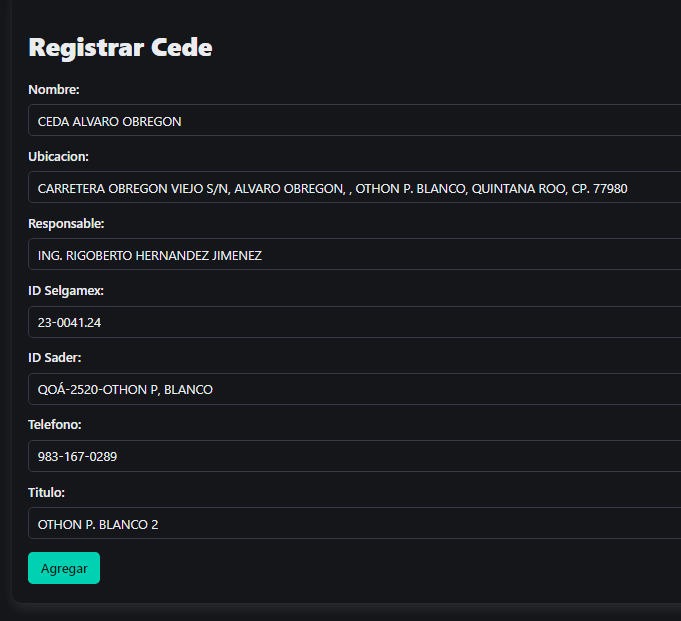
Los campos requeridos son los siguientes:



Titulo

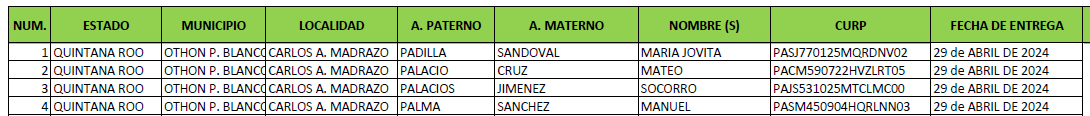
Nombre

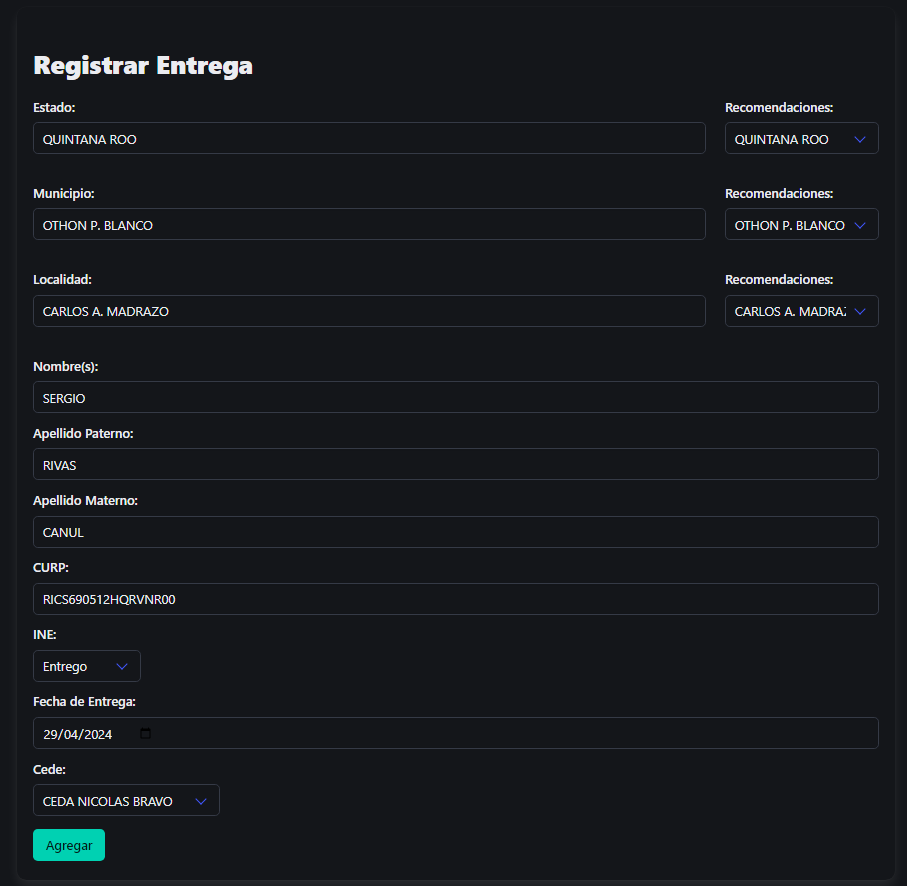
Al terminar de llenar el formulario, se podra dar boton en Agregar y se agregara la ceda.



### Registrar Entrega

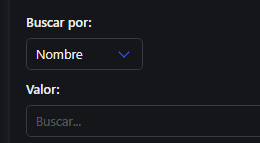
Para poder registrar una entrega todos los campos son requeridos, se cuenta con recomendaciones para Estado, Municipio y Localidad, en caso de que ya se encuentre registrada alguna de esas.

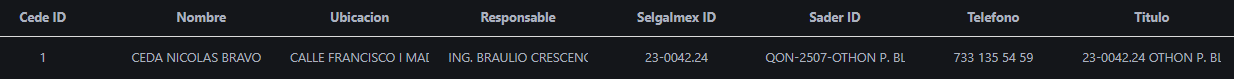


Una vez llenado el formulario simplemente agregamos el registro y listo.

## Cedas

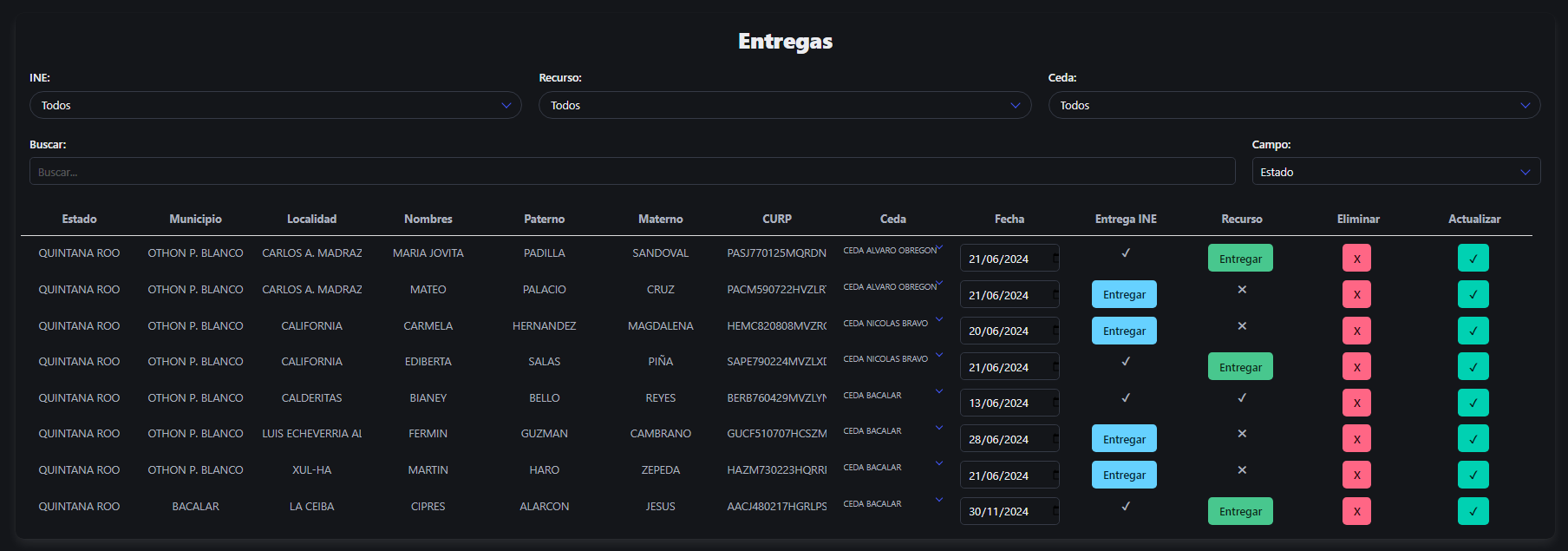
En el apartado de cedes, podremos visualizar las Cedas registradas.

Se cuenta con un buscador para filtrar por alguno de los campos.

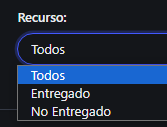
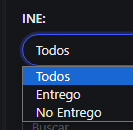
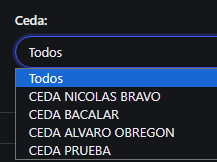
Por cada campo se tendran la siguientes opciones:

Para poder actualizar informacion de alguna de las cedas, se podra hacer directamente en los campos por cada fila y dando clic en el boton de actualizar se actualizara la informacion de esa ceda en especifico y para visualizar las entregas de esa ceda en especifico o eliminar la ceda se podra hacer en sus opciones especificas.

## Entregas

En el apartado de entregas podremos visualizar las Entregas registradas.

Se cuenta con diversos filtros:

* Recursos, se podran filtrar por recursos entregados, no entregados o todos.
* INE, para este se podra filtrar por los que entregaron ine, no entregaron o todos.
* Ceda, se podra filtrar las entregas por ceda.

Tambien se cuenta con las siguientes opciones que son por si al inicio no entrego el ine, pueda entregarla despues y si ya se entrego el recurso, si esta disponible para entregar o entregarlo. Ademas de Eliminar una entrega o actualizar su informacion

# Manual Tecnico

## Queries

Para el funcionamiento del sistema, se tuvo que realizar diversas queries

* get-cede: Esta querie es para obtener informacion de una cede en especifico por la id de la ceda

-- :name get-cede :? :1

-- :doc retrieves 1 cede

**SELECT** **\*** **FROM** cedes

**WHERE** id **=** **:**id**;**

* get-cedes: Obtener informacion de todas las cedas

-- :name get-cedes :? :\*

-- :doc retrieves all cedes

**SELECT** **\*** **FROM** cedes**;**

* create-cede: Esta consulta es utilizada para crear un nuevo registro de cede en la base de datos. Inserta una nueva fila en la tabla cedes con los valores proporcionados para cada columna.

-- :name create-cede! :! :n

-- :doc creates a new cede

**INSERT** **INTO** cedes

**(**nombre**,** ubicacion**,** responsable**,** selgalmexid**,** saderid**,** telefono**,** titulo**)**

**VALUES** **(:**nombre**,** **:**ubicacion**,** **:**responsable**,** **:**selgalmexid**,** **:**saderid**,** **:**telefono**,** **:**titulo**);**

* update-cede: Esta consulta actualiza un registro existente de ceda en la base de datos. Modifica los valores de las columnas correspondientes en la fila de la tabla cedes que coincida con el ID proporcionado.

-- :name update-cede! :! :n

-- :doc updates an existing cede record

**UPDATE** cedes

**SET** nombre **=** **:**nombre**,**

ubicacion **=** **:**ubicacion**,**

responsable **=** **:**responsable**,**

selgalmexID **=** **:**selgalmexid**,**

saderID **=** **:**saderid**,**

telefono **=** **:**telefono**,**

titulo **=** **:**titulo

**WHERE** id **=** **:**id**;**

* delete-cede: Esta consulta elimina un registro de cede de la base de datos según su ID.

-- :name delete-cede! :! :n

-- :doc deletes a cede record given the id

**DELETE** **FROM** cedes **WHERE** id **=** **:**id**;**

* get-entregas: Esta consulta busca obtener información de todas las entregas en la base de datos. Retorna todas las entregas presentes en la tabla entregas.

-- :name get-entregas :? :\*

-- :doc retrieves all entregas

**SELECT** **\*** **FROM** entregas**;**

* get-entregas-id: Esta consulta busca obtener información de todas las entregas asociadas a una cede específica, identificada por su ID.

-- :name get-entregas-id :? :\*

-- :doc retrieves all entregas

**SELECT** **\*** **FROM** entregas **WHERE** cedeid **=** **:**cedeid**;**

* create-entrega!: Esta consulta crea un nuevo registro de entrega en la base de datos.

-- :name create-entrega! :! :n

-- :doc creates a new entrega

**INSERT** **INTO** entregas

**(**estado**,** municipio**,** localidad**,** nombres**,** apaterno**,** amaterno**,** curp**,** ine**,** curpfile**,** inefile**,** fecha**,** estatusentrega**,** cedeid**)**

**VALUES** **(:**estado**,** **:**municipio**,** **:**localidad**,** **:**nombres**,** **:**apaterno**,** **:**amaterno**,** **:**curp**,** **:**ine**,** 'Si'**,** 'Si'**,** **:**fecha**,** **:**estatusentrega**,** **:**cedeid**);**

* delete-entrega!: Esta consulta elimina un registro de entrega de la base de datos según su ID.

-- :name delete-entrega! :! :n

-- :doc deletes a entrega record given the id

**DELETE** **FROM** entregas

**WHERE** id **=** **:**id**;**

* update-entrega!: Esta consulta actualiza un registro existente de entrega en la base de datos.

-- :name update-entrega! :! :n

-- :doc updates an existing entrega record

**UPDATE** entregas

**SET** estado **=** **:**estado**,**

municipio **=** **:**municipio**,**

localidad **=** **:**localidad**,**

nombres **=** **:**nombres**,**

apaterno **=** **:**apaterno**,**

amaterno **=** **:**amaterno**,**

curp **=** **:**curp**,**

ine **=** **:**ine**,**

curpfile **=** 'Si'**,**

inefile **=** 'Si'**,**

fecha **=** **:**fecha**,**

estatusentrega **=** **:**estatusentrega**,**

cedeid **=** **:**cedeid

**WHERE** id **=** **:**id**;**

* update-entrega-estatus!: Esta consulta actualiza el estatus de una entrega específica en la base de datos.

-- :name update-entrega-estatus! :! :n

-- :doc updates an existing entrega record

**UPDATE** entregas

**SET** estatusentrega **=** **:**flagestatusentrega

**WHERE** id **=** **:**id**;**

* update-entrega-ine!: Esta consulta actualiza el archivo INE de una entrega específica en la base de datos.

-- :name update-entrega-ine! :! :n

-- :doc updates an existing entrega record

**UPDATE** entregas

**SET** ine **=** **:**flagine

**WHERE** id **=** **:**id**;**

* get-estados: Esta consulta busca obtener una lista de estados únicos presentes en la tabla de entregas. Retorna una lista de estados sin duplicados.

-- :name get-estados :? :\*

-- :doc retrieves estados

**SELECT** **DISTINCT** estado **FROM** entregas**;**

* get-municipios: Esta consulta busca obtener una lista de municipios únicos presentes en la tabla de entregas. Retorna una lista de municipios sin duplicados.

-- :name get-municipios :? :\*

-- :doc retrieves municipios

**SELECT** **DISTINCT** municipio **FROM** entregas**;**

* get-localidades: Esta consulta busca obtener una lista de localidades únicas presentes en la tabla de entregas. Retorna una lista de localidades sin duplicados.

-- :name get-localidades :? :\*

-- :doc retrieves localidades

**SELECT** **DISTINCT** localidad **FROM** entregas**;**

## Backend

* home-page: Renderiza la página de inicio.

Parámetros:

- request: Mapa que representa la solicitud HTTP recibida.

Retorna:

- Renderiza la página 'home.html' con los siguientes datos:

\* entregas: Lista de entregas obtenida de la base de datos.

\* cedes: Lista de cedes obtenida de la base de datos.

\* estados: Lista de estados obtenida de la base de datos.

\* municipios: Lista de municipios obtenida de la base de datos.

\* localidades: Lista de localidades obtenida de la base de datos.

* cedes-page: Renderiza la página de cedes.

Parámetros:

- request: Mapa que representa la solicitud HTTP recibida.

Retorna:

- Renderiza la página 'cedes.html' con la lista de cedes obtenida de la base de datos.

* entregas-page: Renderiza la página de entregas.

Parámetros:

- request: Mapa que representa la solicitud HTTP recibida.

Retorna:

- Renderiza la página 'entregas.html' con la lista de entregas y cedes obtenidas de la base de datos.

* create-cede!: Crea un nuevo registro de CEDE en la base de datos.

Parámetros:

- request: Mapa que representa la solicitud HTTP recibida.

Retorna:

- Redirige a la lista de cedes en caso de éxito.

* update-cede!: Actualiza un registro de la tabla CEDE en la base de datos.

Parámetros:

- request: Mapa que representa la solicitud HTTP recibida.

Retorna:

- Redirige a la lista de cedes en caso de éxito.

- Redirige a una página de error en caso de fallo o parámetros inválidos.

* delete-cede!: Elimina un registro de CEDE de la base de datos.

Parámetros:

- request: Mapa que representa la solicitud HTTP recibida.

Retorna:

- Redirige a la lista de cedes en caso de éxito.

* create-entrega!: Crea un nuevo registro de ENTREGA en la base de datos.

Parámetros:

- request: Mapa que representa la solicitud HTTP recibida.

Retorna:

- Redirige a la lista de entregas en caso de éxito.

* delete-entrega!: Elimina un registro de ENTREGA de la base de datos.

Parámetros:

- request: Mapa que representa la solicitud HTTP recibida.

Retorna:

- Redirige a la lista de cedes en caso de éxito.

* update-entrega!:Actualiza un registro de la tabla ENTREGA en la base de datos.

Parámetros:

- request: Mapa que representa la solicitud HTTP recibida.

Retorna:

- Redirige a la lista de entregas en caso de éxito.

- Redirige a una página de error en caso de fallo o parámetros inválidos.

* entregas-id-page: Renderiza la página de entregas para un cede específico.

Parámetros:

- request: Mapa que representa la solicitud HTTP recibida.

Retorna:

- Renderiza la página 'entregas.html' con las entregas asociadas a la ceda especificado e informacion de las cedas.

* update-entrega-estatus!: Actualiza el estatus de una entrega en la base de datos.

Parámetros:

- request: Mapa que representa la solicitud HTTP recibida.

Retorna:

- Redirige a la lista de entregas en caso de éxito.

- Redirige a una página de error en caso de fallo o parámetros inválidos.

* update-entrega-ine!:

Actualiza el archivo INE de una entrega en la base de datos.

Parámetros:

- request: Mapa que representa la solicitud HTTP recibida.

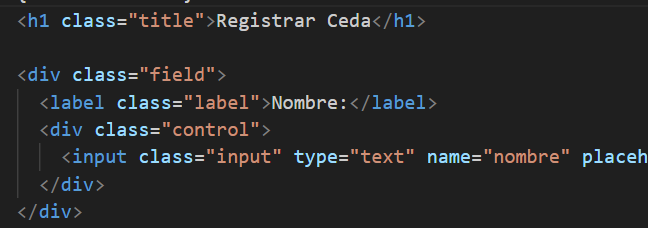
Retorna:

- Redirige a la lista de entregas en caso de éxito.

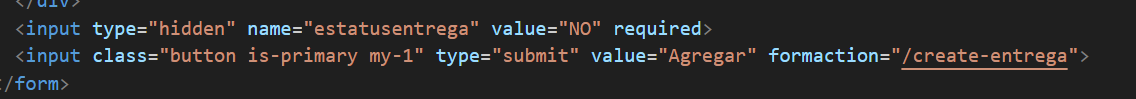
- Redirige a una página de error en caso de fallo o parámetros inválidos.

* home-routes: Define las rutas para la aplicación.

**Frontend**

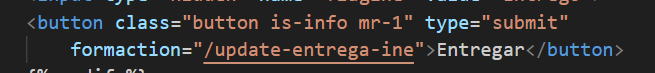
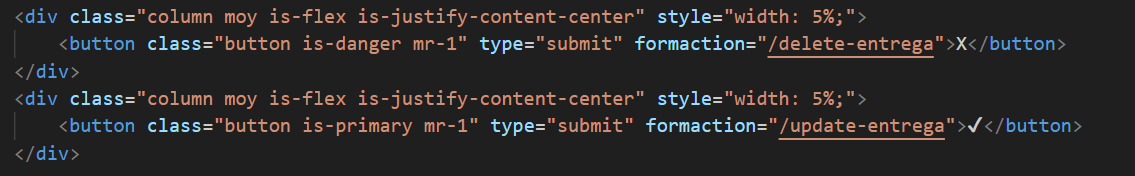
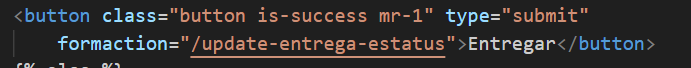
home.html, entregas.html y cedes.html cuentan con diversos inputs, cada uno con un name especifico para cada campo y único, el name asignado es dependiendo de la querie

**home.html**

Los botones, cada uno esta dirigido a un endpoint de clojure que esta configurado en el backend, para agregar cede “/create-cede” y para agregar entrega “create-entrega”

**entregas.html**

Para las entregas, llaman a sus respectivos endpoints del backend



**cedas.html**

Para las cedas, llaman a sus respectivos endpoints del backend

