

# Documento de Ingresos (Registrar Equipamiento (1)) - [store\_equipament]

## ? Descripción

Registra o actualiza las tallas de **equipamiento de EPP** (botas, polo, pantalón) para un postulante, validando previamente:

- Que el documento exista en el sistema.
- Que las tallas enviadas sean válidas según los rangos permitidos.
- Que el postulante exista.
- Que ya tenga o no un registro previo en **Triage de Postulante 2**, para decidir entre **crear (POST)** o **actualizar (PUT)**.

Es un servicio que integra:

- Validaciones locales.
- Consulta de postulantes.
- Inserción o actualización en el ERP vía **ServiceErp()**.

---

## ? Endpoint

POST `/store-equipament`

---

## ? Seguridad

Requiere autenticación interna contra el ERP mediante:

- `ServiceErp()`
- `searchPostulanteByDocument()`

No hace validaciones de token en el controlador; se maneja internamente por los métodos usados.

---

# ? Flujo del Servicio (explicación real)

## 1. Lee los parámetros enviados en la petición:

- documento
- botas
- polo
- pantalon

## 2. Valida la estructura y tallas permitidas:

| Elemento | Tallas permitidas |
|----------|-------------------|
| botas    | 36-45             |
| polo     | S, M, L, XL       |
| pantalón | S, M, L, XL       |

## 3. Busca al postulante por documento:

Llama a:

```
searchPostulanteByDocument($documento)
```

Si no existe → retorna error.

## 4. Busca si ya existe un registro de triaje activo en el ERP:

GET

```
/resource/Triaje de Postulante 2
```

con filtros:

```
documento = $documento
```

```
docstatus != 2
```

## 5. Dependiendo del resultado:

- **No existe registro → crea uno (POST)**
- **Sí existe → actualiza el existente (PUT)**

## 6. Realiza el POST o PUT correspondiente hacia el ERP.

## 7. Si ERP devuelve error, intenta decodificar `_server_messages` para obtener error real del ERP.

## 8. Retorna respuesta final indicando éxito o error.

---

# ? Request Body

Ejemplo:

```
{  
  "documento": "12345678",  
  "botas": 42,  
  "pOLO": "M",  
  "pantalon": "L"  
}
```

## ? Response 200 – Ejemplos

?? Registro/Actualización exitosa

```
{  
  "valor": true,  
  "msn": "Equipamiento registrado con éxito"  
}
```

? Documento no enviado

```
{  
  "valor": false,  
  "msn": "Falta enviar el documento"  
}
```

? Talla inválida

```
{  
  "valor": false,  
  "msn": "Ingrese una talla de botas válida"  
}
```

? Postulante no encontrado

```
{  
  "valor": false,  
  "msn": "No existe postulante con ese documento"  
}
```

## ? Error del ERP

```
{
  "valor": false,
  "msn": "Ocurrió un error al registrar el triaje del postulante",
  "data": [
    "PUT",
    { ...body... },
    "URL del recurso usado"
  ]
}
```

## ? Posibles Errores del Servicio

| Tipo                             | Ejemplo  |
|----------------------------------|--|
| Documento no enviado             | "Falta enviar el documento"  |
| Talla no válida                  | "Ingrese una talla de botas válida"                                    |
| Postulante no encontrado         | respuesta directa del servicio <code>searchPostulanteByDocument</code> |
| Error al crear/actualizar en ERP | Mensaje del <code>_server_messages</code> o mensaje genérico           |
| Error inesperado                 | Devuelve el error capturado del ERP                                    |

## ? ERP: Documentos involucrados

### Triaje de Postulante 2 (GET / POST / PUT)

Campos usados:

| Campo     | Tipo   |
|-----------|--------|
| documento | string |
| botas     | int    |
| polo      | string |
| pantalon  | string |

| Campo | Tipo                    |
|-------|-------------------------|
| name  | string (clave para PUT) |

# ? Lógica en Pseudocódigo

```
read document, botas, polo, pantalon

validate document
validate botas in [36..45]
validate polo in [S,M,L,XL]
validate pantalon in [S,M,L,XL]

postulante = searchPostulanteByDocument(document)

if postulante.invalid:
    return error

triaje = GET TriajeDePostulante2 where documento=document and docstatus != 2

if triaje exists:
    method = PUT
    url = recurso + "/" + triaje.name
else:
    method = POST
    url = recurso

body = {
    documento, botas, polo, pantalon
}

result = CALL ERP with method, body, url

if result.error:
    return readable ERP error

return success
```

Revisión #1

Creado 2025-11-26 11:50:16 -05 por Elian

Actualizado 2025-11-26 11:53:34 -05 por Elian