

Validar Marcación (1) - [validar]

? Descripción

Este servicio ejecuta **toda la lógica de validación previa** para permitir que un empleado realice una marcación (asistencia).

El servicio centraliza más de **10 validaciones críticas**, verificando:

- Campos obligatorios
- Licencias activas
- Restricciones por puesto
- Validación de firma de documentos obligatorios
- Descarga de contratos y documentos requeridos
- Datos incompletos del empleado
- Tipo de marcación permitido
- Esquema de fiscalización
- Reglas por empresa (Shalom Express vs Shalom Empresarial)

Es un servicio de control que determina si el empleado puede realizar una marcación y qué mensaje debe mostrarse al usuario.

? Endpoint

POST /validar

? Request Body (resumen)

```
{
  "empleado": "EMP-001",
  "type": "ingreso",
  "fecha_asistencia": "2025-02-01",
  "fhingreso": "2023-01-10",
  "status_employee": "Active",
  "designation": "OPERADOR",
  "sucursal": "LIMA-01"
}
```

? Seguridad

- Requiere usuario autenticado.
 - Usa datos precargados del empleado (`$this->employee_detail`).
 - Internamente utiliza múltiples servicios ERP y validadores internos.
-

? Flujo del Servicio (Resumen Real)

1☐ Validar campos obligatorios

Llama a `verifyFields()`.

Si falta algún parámetro → retorna error inmediato.

2☐ Cargar datos principales

- empleado
- tipo de marcación
- fecha de asistencia
- datos del empleado desde sesión previa

3☐ Validar restricciones por puesto

Si el empleado NO es "PreActivo", llama a:

`validateRestrictedDesignation(designation)`

4☐ Validar licencia activa

`verifyLicencia(empleado, fechaIngreso)`

5☐ Validar nuevos ingresos

Solo para empleados activos, ingreso reciente, y NO Express:

`verifySignedNewAdmissionRegistrations()`

6☐ Validar descarga de documentos obligatorios

Obtiene documentos descargados:

`verifyDownloadedDocumentsBd()`

Y valida contrato pendiente:

`verifyContractDownload()`

7☐ Validar documentos de cambio de modalidad

`verifyModalityChangeDocument()`

8 Validar información del empleado (horarios, jornada, etc.)

```
verifyEmployeeData()
```

9 Obtener últimas marcaciones

```
getMarkingsByEmployee()
```

Validación de “No sujeto a fiscalización”

```
verifyEmployeeNotSubjectToInspection()
```

1 1 Validación de refrigerio

Si `type == llegada_refrigerio`:

```
verifyStartOfLunch()
```

1 2 Validar coherencia del tipo de marcación

Ejemplo: que no marque salida sin marcar ingreso.

```
verifyTypeOfMarking()
```

Si todas las validaciones son correctas → retorna éxito.

? Response 200 – Ejemplo (OK)

```
{
  "valor": true,
  "msn": "Marcación permitida",
  "data": {
    "type": "ingreso",
    "valid": true
  }
}
```

? Posibles Errores devueltos por este servicio

1. Campos obligatorios faltantes

```
{
  "valor": false,
  "msn": "Falta el campo empleado"
}
```

2. Puesto restringido

```
{  
  "valor": false,  
  "msn": "Su puesto no permite realizar marcaciones"  
}
```

3. Empleado con licencia activa

```
{  
  "valor": false,  
  "msn": "Tiene licencia activa. No puede marcar"  
}
```

4. Documentos de ingreso no firmados

```
{  
  "valor": false,  
  "msn": "Debe firmar sus documentos de ingreso"  
}
```

5. Contrato pendiente por descargar

```
{  
  "valor": false,  
  "msn": "Debe descargar su contrato para continuar"  
}
```

6. Error en la lógica de marcación (orden incorrecta)

```
{  
  "valor": false,  
  "msn": "No puede marcar salida sin haber marcado ingreso"  
}
```

7. No sujeto a fiscalización

```
{  
  "valor": false,  
  "msn": "Empleado no sujeto a fiscalización"  
}
```

? Funciones internas utilizadas

Nombre	Descripción
verifyFields	Valida campos requeridos del request
validateRestrictedDesignation	Controla puestos con restricciones
verifyLicencia	Evalúa si el empleado tiene licencia activa
verifySignedNewAdmissionRegistrations	Valida firmas de ingreso
verifyDownloadedDocumentsBd	Valida documentos descargados
verifyContractDownload	Revisa si falta descargar contrato
verifyModalityChangeDocument	Valida documentos de cambios de modalidad
verifyEmployeeData	Verifica datos del empleado necesarios para marcar
getMarkingsByEmployee	Obtiene último registro de marcación
verifyEmployeeNotSubjectToInspection	Revisa si está sujeto a fiscalización
verifyStartOfLunch	Valida inicio del refrigerio
verifyTypeOfMarking	Controla coherencia de marcaciones

? Pseudo-código real del servicio

```
if !verifyFields(request): return error
if puesto restringido: return error
if licencia activa: return error
if ingreso reciente y no Shalom Express:
    validar documentos de ingreso
validar documentos descargados
validar contrato
validar cambio de modalidad
validar información del empleado
marcacionesPrevias = getMarkingsByEmployee()
if no sujeto a fiscalización: return error
if marcación es llegada_refrigerio:
    validar inicio refrigerio
validar secuencia de tipo de marcación
return OK
```

Revisión #1

Creado 2025-11-25 15:47:17 -05 por Elian

Actualizado 2025-11-25 15:51:07 -05 por Elian