

UNIVERSIDADE DE SANTIAGO DE  
COMPOSTELA



ESCOLA TÉCNICA SUPERIOR DE ENXEÑARÍA

## Integración dunha plataforma de xestión de laboratorios co equipamento dos laboratorios de Televés Ladetel

*Autora:*

**Rosa Elena Veiga Otero**

*Directores:*

**José Ángel Taboada González  
Miguel Duro Liñares**

**Grao en Enxeñaría Informática**

**Setembro 2019**

Traballo de Fin de Grao presentado na Escola Técnica Superior de Enxeñaría  
da Universidade de Santiago de Compostela para a obtención do Grao en  
Enxeñaría Informática





**D. José Ángel Taboada González**, Profesor do Departamento de Electrónica e Computación da Universidade de Santiago de Compostela, e **D. Miguel Duro Liñares**, Enxeñeiro de Telecomunicacións en Televés,

INFORMAN:

Que a presente memoria, titulada *Integración dunha plataforma de xestión de laboratorios co equipamento dos laboratorios de Televés Ladetel*, presentada por **Dna. Rosa Elena Veiga Otero** para superar os créditos correspondentes ao Traballo de Fin de Grao da titulación de Grao en Enxeñaría Informática, realizouse baixo nosa dirección no Departamento de Electrónica e Computación da Universidade de Santiago de Compostela.

E para que así conste aos efectos oportunos, expiden o presente informe en Santiago de Compostela, a (Data):

O director,

O codirector,

O alumno,

José Ángel Taboada González   Miguel Duro Liñares   Rosa Elena Veiga Otero



# Agradecementos

A José Ángel Taboada, por guiarme e axudarme no desenvolvemento e documentación do proxecto.

A Miguel Duro, por toda a axuda técnica e a formación proporcionada, especialmente no relativo á implicación do usuario. Aos meus compañeiros de Ladetel, por crear un ambiente tan ameno e produtivo, no que se pode disfrutar cada hora da xornada.

Á miña familia, por apoiarme e animarme sempre nos estudos, a pesar de ter que escoitarme falar en informático.

A Pedro, polo apoio emocional nos momentos máis duros do proxecto. A todos os meus amigos e compañeiros de clase; fixestes estes anos absolutamente inolvidables. A enxeñaría informática só me trouxo cousas boas.



# Índice xeral

<b>1. Introducción</b>	<b>1</b>
1.1. Necesidade e contexto normativo . . . . .	1
1.2. Necesidade e contexto no laboratorio . . . . .	2
1.3. Funcionamento do laboratorio . . . . .	5
1.4. Glosario . . . . .	5
1.4.1. Acrónimos . . . . .	5
1.4.2. Conceptos . . . . .	6
1.5. Obxectivos . . . . .	7
1.6. Descrición do alcance . . . . .	7
1.6.1. Exclusiones do proxecto . . . . .	8
1.7. Organización do documento . . . . .	8
<b>2. Planificación e presupostos</b>	<b>11</b>
2.1. Xestión da configuración . . . . .	11
2.1.1. Infraestrutura . . . . .	11
2.1.2. Control de cambios . . . . .	13
2.2. Xestión de riscos . . . . .	14
2.2.1. Fontes de risco e categorización dos mesmos . . . . .	14
2.2.2. Matrices de probabilidade e impacto . . . . .	14
2.2.3. Identificación e análise de riscos . . . . .	16
2.3. Xestión do tempo . . . . .	26
2.3.1. Metodoloxía do desenvolvemento . . . . .	26
2.3.2. Cronograma . . . . .	26
2.3.3. Variacións no cronograma . . . . .	28
2.4. Xestión de custos . . . . .	28
2.4.1. Unidades e precisión . . . . .	28
2.4.2. Custos de equipamento . . . . .	30
2.4.3. Custos de recursos humanos . . . . .	31
2.4.4. Custos indirectos . . . . .	31
2.4.5. Custos totais . . . . .	31
2.5. Xestión de comunicacións e interesados . . . . .	32
2.5.1. Interesados . . . . .	32
2.5.2. Comunicacións . . . . .	34

<b>3. Especificación de requisitos</b>	<b>37</b>
3.1. Relación cos requisitos do LIMS . . . . .	37
3.1.1. Requisitos funcionais do LIMS con impacto no LAS . . . . .	38
3.1.2. Requisitos non funcionais do LIMS con impacto no LAS . . . . .	38
3.2. Actores . . . . .	42
3.3. Casos de uso . . . . .	42
3.4. Requisitos non funcionais . . . . .	51
3.5. Requisitos de información . . . . .	52
<b>4. Arquitectura e deseño preliminar</b>	<b>55</b>
4.1. Deseño de XesLab . . . . .	55
4.2. Estrutura do sistema de automatización . . . . .	57
4.2.1. Módulos de comunicación en XesLab . . . . .	58
4.2.2. Software da pasarela . . . . .	60
4.3. Tecnoloxías a empregar . . . . .	61
4.3.1. Ferramentas de desenvolvemento . . . . .	61
4.3.2. Ferramentas de documentación . . . . .	62
4.3.3. Librarías externas . . . . .	62
4.3.4. Equipamento hardware . . . . .	63
<b>5. Deseño e implementación en XesLab</b>	<b>65</b>
5.1. Detalles de deseño e implementación da aplicación XesLab . . . . .	65
5.2. Módulo de procesamento de comandos . . . . .	67
5.2.1. Estado dos equipos . . . . .	67
5.2.2. Bloqueos . . . . .	68
5.2.3. Enviar comando . . . . .	70
5.3. Módulo de preparación de comandos . . . . .	71
5.4. Módulo de sockets . . . . .	72
5.5. Diagramas de secuencia . . . . .	72
5.6. Interfaces gráficas . . . . .	76
<b>6. Deseño e implementación na pasarela</b>	<b>79</b>
6.1. Paquete de rede . . . . .	79
6.2. Paquete controlador . . . . .	82
6.2.1. A xerarquía de fíos . . . . .	82
6.2.2. O controlador da aplicación . . . . .	84
6.2.3. O controlador de fío . . . . .	86
6.2.4. Outros elementos . . . . .	88
6.3. Paquete de mensaxes . . . . .	88
6.3.1. A xerarquía de mensaxes . . . . .	88
6.3.2. O xestor de mensaxes . . . . .	89
6.4. Paquete de módulos . . . . .	90
6.4.1. Os módulos . . . . .	90



6.4.2. O xestor de módulos . . . . .	93
6.5. Tipos de mensaxes e as súas respostas . . . . .	93
6.6. Diagramas de secuencia . . . . .	96
<b>7. Verificación e validación</b>	<b>105</b>
7.1. Plan de probas . . . . .	105
7.1.1. Restricións e enfoque . . . . .	105
7.1.2. Supostos e exclusións . . . . .	106
7.1.3. Criterios de paso ou fallo . . . . .	106
7.2. Deseño de probas . . . . .	107
7.2.1. Estratexia de construción . . . . .	107
7.2.2. Probas xeradas . . . . .	107
7.3. Casos de proba . . . . .	116
7.3.1. Casos de proba de XesLab . . . . .	118
7.3.2. Casos de proba da pasarela . . . . .	139
7.4. Implementación e execución das probas . . . . .	144
7.4.1. Implementación . . . . .	144
7.4.2. Incidencias . . . . .	145
7.5. Informe de execución . . . . .	147
7.6. Validación . . . . .	147
7.6.1. Envío de comandos a equipos de xeito individual . . . . .	147
7.6.2. Execución de medidas . . . . .	153
<b>8. Conclusións e posibles ampliacións</b>	<b>157</b>
8.1. Conclusións . . . . .	157
8.2. Posibles ampliacións . . . . .	158
<b>A. Manuais técnicos</b>	<b>161</b>
A.1. Manual de instalación . . . . .	161
A.1.1. Instalación dos drivers de GPIB . . . . .	161
A.1.2. Instalación das librarías de Boost . . . . .	163
A.1.3. Compilación do código da pasarela . . . . .	164
A.2. Engadir un novo módulo . . . . .	165
<b>B. Manuais de usuario</b>	<b>167</b>
B.1. Envío de comandos de forma individual . . . . .	167
B.2. Execución de medidas . . . . .	170
<b>Bibliografía</b>	<b>175</b>



# Índice de figuras

1.1. Captura de pantalla de GestLAB . . . . .	4
1.2. Captura de pantalla de ControLAB . . . . .	4
1.3. Diagrama de conceptos de alto nivel de XesLab . . . . .	5
2.1. Sección de desenvolvemento en XesLab . . . . .	12
2.2. Estrutura do repositorio para o código da pasarela . . . . .	12
2.3. Estrutura de descomposición de riscos . . . . .	15
2.4. Cronograma . . . . .	27
2.5. Cronograma de seguimento . . . . .	29
3.1. Implementacións do LIMS e o LAS . . . . .	37
3.2. Diagrama de casos de uso . . . . .	42
4.1. Diagrama de negocio de XesLab . . . . .	56
4.2. Diagrama de despregamento . . . . .	57
4.3. Equipos Keysight MXG e R&S FSV7 empregados no desenvolvemento . . . . .	64
5.1. Diagrama de clases de XesLab . . . . .	66
5.2. Diagrama de estados de bloqueo . . . . .	69
5.3. Enviar comando sen resposta a un equipo conectado por Ethernet . . . . .	73
5.4. Enviar comando con resposta a un equipo conectado por Ethernet . . . . .	74
5.5. Enviar comando sen resposta a un equipo conectado por GPIB . . . . .	75
5.6. Pantalla do equipo 522 en XesLab . . . . .	77
5.7. Panel de comunicacións do equipo 522 en XesLab . . . . .	77
5.8. Pantalla dun procedemento, co panel da medida técnica . . . . .	78
6.1. Diagrama de clases . . . . .	80
6.2. Diagrama de clases do paquete de rede . . . . .	81
6.3. Diagrama de clases do paquete controlador . . . . .	83
6.4. Diagrama de clases do paquete de mensaxes . . . . .	89
6.5. Diagrama de clases do paquete de módulos . . . . .	91
6.6. Enviar comando sen resposta a un equipo conectado por GPIB . . . . .	96
6.7. Enviar comando con resposta a un equipo conectado por GPIB . . . . .	97
6.8. Mensaxe baleira . . . . .	98

6.9. *IDN? en texto plano . . . . .	99
6.10. Mensaxe “alive” dirixida a un dispositivo GPIB . . . . .	100
6.11. Erro de rede, notificado ao servidor . . . . .	101
6.12. Erro de rede que non se pode notificar ao servidor; é ignorado . . . . .	102
6.13. Erro de <i>parse</i> . . . . .	103
6.14. Erro GPIB . . . . .	104
7.1. Resultados de comunicacións co equipo 522 . . . . .	152
7.2. Pantalla dunha medida técnica . . . . .	153
7.3. Panel de execución da medida . . . . .	154
7.4. Pantalla do equipo MXG . . . . .	154
7.5. Pantalla do equipo ESCI . . . . .	155
7.6. Resultados da medida, obtidos en tempo real . . . . .	155
B.1. Acceso á pantalla de equipos . . . . .	167
B.2. Lista de equipos e resultados do filtrado . . . . .	168
B.3. Datos do equipo 522 . . . . .	168
B.4. Panel de comunicacións do equipo 522 . . . . .	169
B.5. Resultados do envío de comandos . . . . .	169
B.6. Erro nos módulos de comunicacións en XesLab . . . . .	170
B.7. Erro na pasarela . . . . .	170
B.8. Acceso á lista de procedementos a través do menú contextual . . . . .	171
B.9. Lista de procedementos . . . . .	172
B.10. Menú contextual do procedemento . . . . .	172
B.11. Pantalla do procedemento . . . . .	173
B.12. Panel da medida técnica . . . . .	174
B.13. Resultados da medida, obtidos en tempo real . . . . .	174

# Índice de cadros

1.1. Cadro de fases expostas na actividade de implantación ISO 17025	2
2.1. Valoración do impacto	15
2.2. Valoración da probabilidade	15
2.3. Nivel de exposición ao risco	16
2.4. Características dos riscos do proxecto	25
2.5. Clasificación dos riscos por exposición	26
2.6. Cadro de identificación de interesados	26
2.7. Cadro de custos de amortización	30
2.8. Cadro de custos de recursos humanos	31
2.9. Cadro de custos totais	32
2.10. Cadro de identificación de interesados	33
2.11. Detalles dos interesados	34
3.1. Requisitos funcionais do LIMS con impacto no LAS	38
3.2. Requisitos non funcionais do LIMS con impacto no LAS	41
3.3. Requisitos de información	53
4.1. Cadro de licenzas do software de terceiros	62
5.1. Restricións relativas aos bloqueos	69
6.1. Proceso de empaquetamento e desempaquetamento	89
6.2. Tipos de mensaxes e posibles erros	93
6.3. Exemplos, tratamento, e exemplos de respostas para as distintas mensaxes que poden chegar á pasarela	95
7.1. Probas xeradas para os módulos de comunicación de XesLab	114
7.2. Cadro de clasificación de respostas aos mensaxes enviadas á pasarela	116
7.3. Proba xerada para a pasarela	117
7.4. Resultados das probas	151



# Capítulo 1

## Introdución

### 1.1. Necesidade e contexto normativo

**Ladetel** é un laboratorio pertencente á Corporación Televés, cuxa actividade consiste fundamentalmente na realización de ensaios para o mercado CE de aparatos electrónicos, tanto de clientes internos como externos.

Como parte do “Programa Industrias do Futuro 4.0” da Xunta de Galicia, Televés está a desenvolver o seu proxecto Corp4Future, contexto no que o seu laboratorio de ensaios Ladetel enmarca unha actividade de implantación dun sistema de calidade conforme á **norma ISO 17025** [1].

A norma ISO 17025 contén os requisitos que teñen que cumprir os laboratorios de ensaio e calibración, no caso de que desexen demostrar que posúen un sistema de xestión, son técnicamente competentes e son capaces de xerar resultados tecnicamente válidos [2]. O recoñecemento formal que recibe un laboratorio cun sistema de xestión acorde á norma ISO 17025 é a **acreditación**. En Europa a acreditación está regulada polo Regulamento (CE) nº 765/2009, no que se fixa un modelo de acreditación baseado na existencia dun único Organismo Nacional de Acreditación en cada Estado membro, formalmente designado e con potestade pública para levar a cabo a súa función. Deste xeito, un laboratorio acreditado garante que a calidade dos seus ensaios e servizos é avaliada por un organismo competente e cualificado. En España, este organismo é a Entidade Nacional de Acreditación (ENAC).

A norma ISO 17025 establece unha serie de requisitos de xestión similares a outras normas de calidade tipo ISO 9001, pero a maiores esixe outra serie de requisitos técnicos, totalmente específicos e orientados aos laboratorios de ensaio e calibración. Ademais disto, a ENAC establece os seus propios requisitos.

Entre os contidos da norma non se especifica expresamente a necesidade de contar cun sistema informático para a satisfacción dos requisitos, pero chégase de xeito natural a esta solución. O sistema plantexado divídese en dúas compoñentes:

- **Laboratory Information Management System (LIMS)**: sistema de

xestión da información do laboratorio. Será o sistema empregado polos técnicos do laboratorio para realizar as tarefas de xestión.

Permitirá o cumprimento dos requisitos da norma ISO 17025 e aliviará a carga de traballo que supoñen as tarefas de xestión e mantemento, mellorando a calidade e produtividade do traballo realizado no laboratorio.

Optouse por un LIMS de desenvolvemento propio debido á flexibilidade que iso aporta. O sistema denomínase XesLab.

- **Laboratory Automation System (LAS):** estará integrado co LIMS, e permitirá a interacción directa cos equipos do laboratorio dende a plataforma, sen necesidade de cambiar de ferramenta ou desprazarse fisicamente ata o equipo. O paso de parámetros para as medidas e a recuperación de resultados serán comunicados automaticamente da plataforma aos equipos e viceversa, de xeito que non é preciso realizar accións manuais. Isto diminúe o erro humano e aumenta a fiabilidade.

## 1.2. Necesidade e contexto no laboratorio

Actualmente o laboratorio posúe un sistema informático de xestión (GestLAB; figura 1.1) que non responde exhaustivamente aos requerimentos da norma ISO 17025 [2], e un sistema de automatización de medidas (ControLAB; figura 1.2) independente do anterior.

O obxectivo para o laboratorio, tras a finalización da actividade de implantación ISO 17025 descrita no apartado anterior, é posuír un sistema informático de xestión e automatización de medidas totalmente integrado, e que satisfaga todos os requerimentos da norma ISO 17025. Para a consecución deste obxectivo é fundamental o correcto desempeño de tódalas fases expostas da actividade, entre elas a FASE 4, obxectivo directo deste traballo de fin de grao.

Descrición	
FASE 0	Planificación
FASE 1	Reparto de responsabilidades
FASE 2	Elaboración da documentación base
FASE 3	Implantación de requisitos mediante LIMS
<b>Fase 4</b>	<b>Automatización de sistemas mediante LAS</b>
FASE 5	Integración LIMS + LAS
FASE 6	Auditorías internas
FASE 7	Accións correctoras

Cadro 1.1: Cadro de fases expostas na actividade de implantación ISO 17025

No laboratorio existen unha variedade de equipos de medida; analizadores de espectro, osciloscopios e analizadores de redes, entre outros. As interfaces software



dos equipos son distintas: cada equipo é operado con comandos específicos. Estes comandos, entre outros detalles da máquina, están dispoñibles nunha base de datos interna do laboratorio.

Ademais disto, os equipos presentan diversas interfaces hardware.

- Algúns deles contan cunha conexión GPIB (General Purpose Instrument Bus): un bus paralelo de 8 bits e 24 pins, creado por HP para conectar dispositivos de medición. Foi estandarizado como IEEE-488; no estándar IEEE-488.1 [3] defínense os parámetros básicos, e en IEEE-488.2 [4] unha serie de comandos específicos independentes do dispositivo empregados para a comunicación.
- Outros reciben a información a través dunha conexión de tipo Ethernet, cableada ou por conexión Wi-Fi.

Os equipos GPIB son especialmente difíciles de automatizar. A tecnoloxía GPIB ten límites estritos con respecto á lonxitude máxima do cable e o número de dispositivos conectados; un cable GPIB pode medir como máximo 20 metros. Trátase dun bus serie, polo que os dispositivos se poden seriar, pero imponse para todos os dispositivos que:

- A separación máxima entre dispositivos é de 4 metros se se quere conseguir a máxima taxa de transferencia. Existen cables GPIB de maior lonxitude, pero as transmisións serán máis lentas e o sistema non cumprirá as especificacións da IEEE-488.
- O número máximo de dispositivos conectados a un bus é de 15, sendo un deles necesariamente a placa controladora. Polo menos dous tercios deles deben estar acendidos.

Isto fai que sexa inviable conectar os equipos GPIB directamente aos ordenadores da oficina, sexa instalando tarxetas GPIB nestes ou a través de adaptadores; na oficina hai 10 ordenadores e o laboratorio conta con máis de 5 equipos. Ademais, a oficina está separada do laboratorio por un pasillo de máis de 20 metros.

Agora mesmo non existe ningunha automatización de equipos IP, polo que as medidas nestes dispositivos precisan ser realizadas a man. No caso dos equipos GPIB, que son a maioría, existen dous programas (un propio, ControlLAB, e un comercial, EMC32) que permiten desenvolver algunhas das medicións. Non obstante, son programas autónomos; hai que executar as medicións, exportalas no formato que permite cada programa (respectivamente, CSV e PDF) e, posteriormente, ingresar e procesar os datos de xeito manual en GestLAB.

Esta disposición redonda en dificultades para a automatización e o control das medicións, xa que a interacción con cada equipo é tremendamente individualizada e require intervención humana. Isto fai que aumente o tempo empregado en cada ensaio.

The screenshot shows the 'ENSAYOS ASIGNADOS' section of the GestLAB application. It features a table with columns for 'Copiar', 'Excel', 'CSV', 'PDF', 'Prioridad', 'Id', 'Fecha', 'Estado', 'PCE', 'Nombre', and 'Conclusiones'. The table contains several rows of test data, including IDs like 102, 106, 111, 78, 79, 80, and 2150, with dates ranging from 19-06-2019 to 06-06-2019. A search bar is visible on the right side of the table.

Copiar	Excel	CSV	PDF	Prioridad	Id	Fecha	Estado	PCE	Nombre	Conclusiones
					102	2162	19-06-2019	Activo		
					106	2164	21-06-2019	Activo		
					111	2169	27-06-2019	Activo		
					78	2111	27-03-2019	Activo		
					79	2128	23-04-2019	Activo		
					80	2129	26-04-2019	Activo		
					-	2151	07-06-2019	Activo		
					-	2150	06-06-2019	Activo		

Mostrando del registro 1 al 9 de 9 en total

Figura 1.1: Captura de pantalla de GestLAB

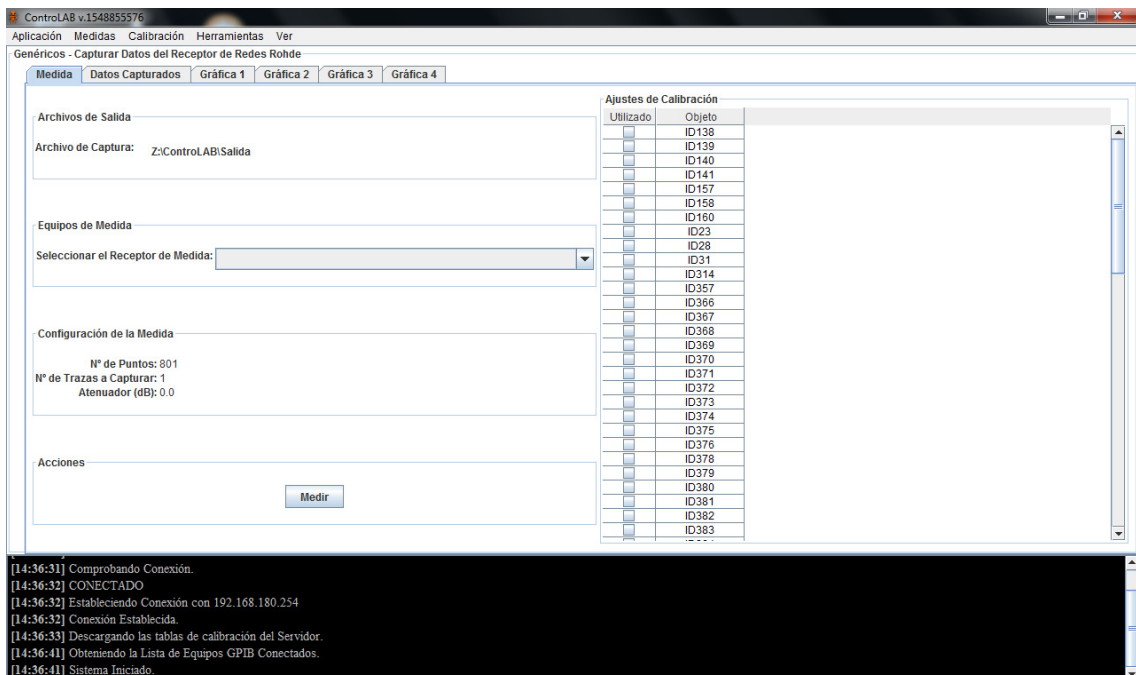


Figura 1.2: Captura de pantalla de ControlLAB

## 1.3. Funcionamento do laboratorio

O traballo realizado no laboratorio consiste en realizar probas sobre as características físicas de dispositivos. Un modelo de dispositivo que se envía a Ladetel para ser medido denomínase DUT ou dispositivo baixo ensaio; dun DUT poden chegar varios dispositivos físicos, que son equivalentes a efectos das medidas. Cada dispositivo físico é unha mostra do DUT.

Un conxunto de probas sobre un ou varios DUTs é un ensaio, e cada unha das probas é unha medida. Durante a execución da medida, os DUTs teñen un papel pasivo; colócanse nun entorno, ao que se lle aplican unha serie de cambios. A medida consiste en averiguar como o DUT responde a estes cambios. As ferramentas que provocan os cambios ou comprobán a resposta do equipo son os equipos, que se poden controlar de forma remota mediante o envío de comandos. Polo tanto, unha medida implica que unha serie de equipos executen uns comandos concretos.

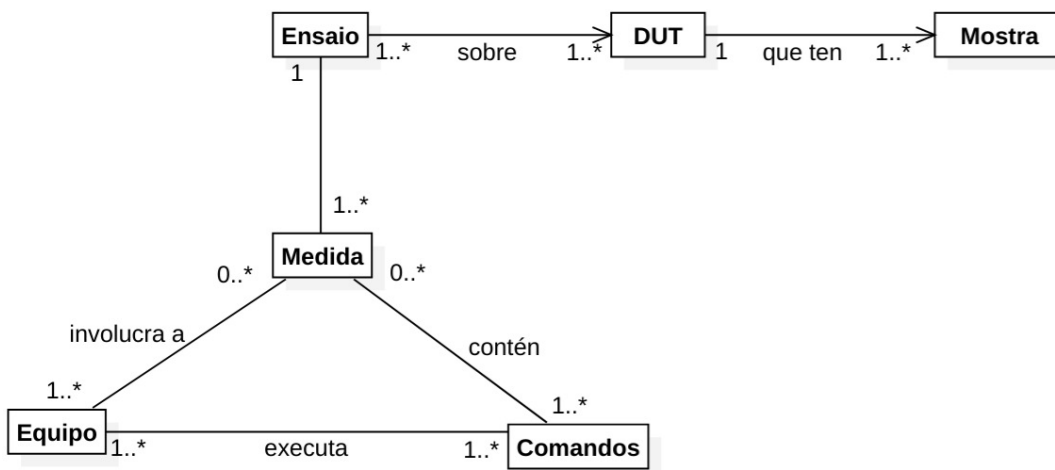


Figura 1.3: Diagrama de conceptos de alto nivel de XesLab

## 1.4. Glosario

### 1.4.1. Acrónimos

- **DUT:** *Device Under Test*
- **LAS:** *Laboratory Automation System*
- **LIMS:** *Laboratory Information Management System*

- **PEE:** Procedemento Específico de Ensaio
- **PEQ:** Procedemento Específico de Equipo

### 1.4.2. Conceptos

- **Característica técnica:** magnitude física. Por exemplo, frecuencia.
- **Comandos:** empréganse para programar os equipos; unha medida implica a execución dunha secuencia deles.
  - **Comandos de equipo:** comando en formato interpretable por un equipo concreto; por exemplo, `FREQ 2GHz`.
  - **Comandos de medida:** expresión xenérica dun comando, lexible por humanos pero non interpretable por ningún equipo concreto, nin asociada a ningún. Por exemplo, “xerador->frecuencia 2 GHz”.
- **Contabilidade:** define que procedementos específicos de ensaio se aplican ao traballo. O custo dos PEEs inflúe directamente no presuposto do traballo.
- **ControlAB:** software desenvolvido por Televés para configurar e operar os dispositivos.
- **Device under test (DUT):** ou dispositivo baixo experimento. O obxecto dun traballo é realizar medicións sobre eles. Cada DUT é un modelo concreto do dispositivo, ou un conxunto de modelos cuxas características sexan equivalentes a efectos da medición.
- **EMC32:** software de terceiros para configurar e operar os dispositivos.
- **Ensaio:** conxunto de medidas dun tipo concreto a realizar sobre un DUT. Se non foi parametrizado completamente en etapas anteriores, establece a parametrización para as medidas.
- **Equipo:** un instrumento concreto do laboratorio. Mide características físicas.
- **GestLAB:** sistema de xestión de laboratorios, actualmente en uso polo persoal de Ladetel. Non permite automatización de medicións.
- **Implementación dun PEE:** asociación de normas a un PEE, que establecen parametrización e criterios de avaliación dos resultados.
- **Medida:** avaliación do comportamento dun DUT nunhas condicións específicas e con respecto a unhas características técnicas concretas.

- **Medida técnica:** resultado, en implementación, dunha medida, tras pasar todo o proceso de parametrización (a través dun PEE, implementación e ensaio).
- **Mostra:** cada un dos dispositivos físicos, correspondentes a un DUT, que se envían ao laboratorio para ser medidos.
- **Procedemento específico de ensaio (PEE):** establece un conxunto de ensaios a realizar. Pode indicar parametrización e criterios de avaliación dos resultados.
- **Procedemento específico de equipo (PEQ):** conxunto de equipos equivalentes, no aspecto de que miden as mesmas características físicas (por exemplo, xeneradores).
- **Traballo:** servizo que o laboratorio realiza para un cliente, cunhas entradas (os produtos a medir) e unha saída (os resultados das medicións).
- **XesLab:** sistema de xestión de laboratorios que está actualmente en desenvolvemento, no que se enmarca este proxecto, e ao que proporcionará capacidades de automatización de medicións.

## 1.5. Obxectivos

O obxectivo do proxecto é a automatización do laboratorio, e está enmarcado dentro das tarefas necesarias para a obtención da certificación ISO 17025. Minimizando a intervención humana conséguese as seguintes melloras.

- Diminuirá a posibilidade de erro, aumentando a fiabilidade.
- Liberará aos técnicos dunha parte do traballo, podendo dedicarse estes a outras tarefas. Isto repercute nun aumento da calidade e da produtividade.

O obxectivo final é asegurar a competencia técnica do laboratorio de ensaios, como medio para demostrar a capacidade que o laboratorio ten para producir resultados tecnicamente válidos.

## 1.6. Descrición do alcance

O alcance deste proxecto é o desenvolvemento do sistema de automatización do laboratorio. Resolverá os problemas mencionados nas seccións anteriores, permitindo a comunicación directa entre LIMS e equipos, sen intervención humana. O alcance deste proxecto comprende o desenvolvemento do LAS, e require a integración de dous subsistemas distintos:

- Un conxunto de módulos de XesLab que proporcionarán unha capa de abstracción para as comunicacións, de forma que o resto de módulos poidan interactuar cos dispositivos de forma sinxela e uniformizada, independentemente da interface hardware ou software que teña cada equipo.
- Un software pasarela que se executará nunha plataforma física diferente ao servidor de XesLab e que realizará unha tradución física e un enrutado dos paquetes ao dispositivo correcto. Este nodo actuará como intermediario entre XesLab e os equipos.

### 1.6.1. Exclusiones do proxecto

Non forma parte do alcance o desenvolvemento do LIMS, que xa se atopa en proceso no momento do comezo deste proxecto, nin a integración entre LIMS e LAS.

## 1.7. Organización do documento

O documento estrutúrase nas seccións descritas a continuación.

- **Capítulo 1: introdución.** Breve introdución ao proxecto, informando ao lector do contexto normativo e a necesidade do produto, así como os conceptos necesarios para comprender o traballo no laboratorio e que determinan gran parte do catálogo de funcionalidades de XesLab. Derívanse de esta información os obxectivos do proxecto e o alcance do mesmo.
- **Capítulo 2: planificación e presuposto.** Descrición e datos resultantes de todas as fases de xestión do proxecto. Contémpnanse a xestión da configuración, riscos, tempo, custos, e comunicacións e interesados.
- **Capítulo 3: especificación de requisitos.** Indícanse cales son os requisitos para o LIMS que se trasladan ao LAS, e especifícanse os requisitos funcionais, non funcionais e de información para o LAS, derivados dos anteriores.
- **Capítulo 4: arquitectura e deseño preliminar.** Móstrase unha visión de alto nivel da arquitectura do sistema, e de cada un dos seus subsistemas. Do mesmo xeito, menciónanse as tecnoloxías empregadas para o desenvolvemento do proxecto.
- **Capítulo 5: deseño e implementación en XesLab.** Profundízase na estrutura dos módulos de comunicación de XesLab, e elabórase en cuestións de implementación.

- **Capítulo 6: deseño e implementación na pasarela.** Profundízase na estrutura do software da pasarela, e elabórase en cuestións de implementación.
- **Capítulo 7: verificación e validación.** Descríbese todo o proceso de proba do software. Inclúe o plan de probas, o informe de execución das probas, e os aspectos relativos á validación do software por parte dos usuarios.
- **Capítulo 8: conclusións e posibles ampliacións.** Conclusións do proxecto, e potenciais melloras ou novas funcionalidades que se poden implementar no futuro.
- **Anexos.** Inclúe o manual de instalación do software, para XesLab e para a pasarela.





# Capítulo 2

## Planificación e presupostos

### 2.1. Xestión da configuración

#### 2.1.1. Infraestrutura

##### Código de XesLab

XesLab foi, ata o momento, un proxecto interno no que traballou un só desenvolvedor, polo que non foi preciso un sistema de xestión da configuración complexo; existe unha soa rama de desenvolvemento. As funcionalidades implementáronse conforme se foron necesitando, e o versionado realizouse cada vez que se completaba un módulo ou sucedía unha refactorización importante.

Debido a que a partir de agora haberá máis dun desenvolvedor, será preciso implementar un sistema de xestión de versións apropiado.

Si existen dous servidores diferentes, identificados como `xeslab.ladete1` (servidor de produción, cos cambios validados e verificados polos técnicos do laboratorio) e `labman.ladete1` (servidor de desenvolvemento). Ademais disto, na aplicación existe unha sección de desenvolvemento para probar informalmente e depurar o código, sen interferir co resto de módulos.

##### Outros elementos do proxecto

Non estaba xustifico dentro do alcance deste proxecto e as responsabilidades da desenvolvedora implementar un sistema de control de versións para a aplicación de XesLab completa; porén, si se considerou conveniente incluír nun repositorio privado na plataforma GitHub os demais elementos que compoñen o proxecto e cuxo editor non aporta control de versións, tales como o código da pasarela e os diagramas do software.

O repositorio ten a seguinte estrutura de directorios.

- **Diagramas:** representacións gráficas incluídas na memoria como apoio á documentación de deseño; inclúe tanto o ficheiro de edición dos diagramas

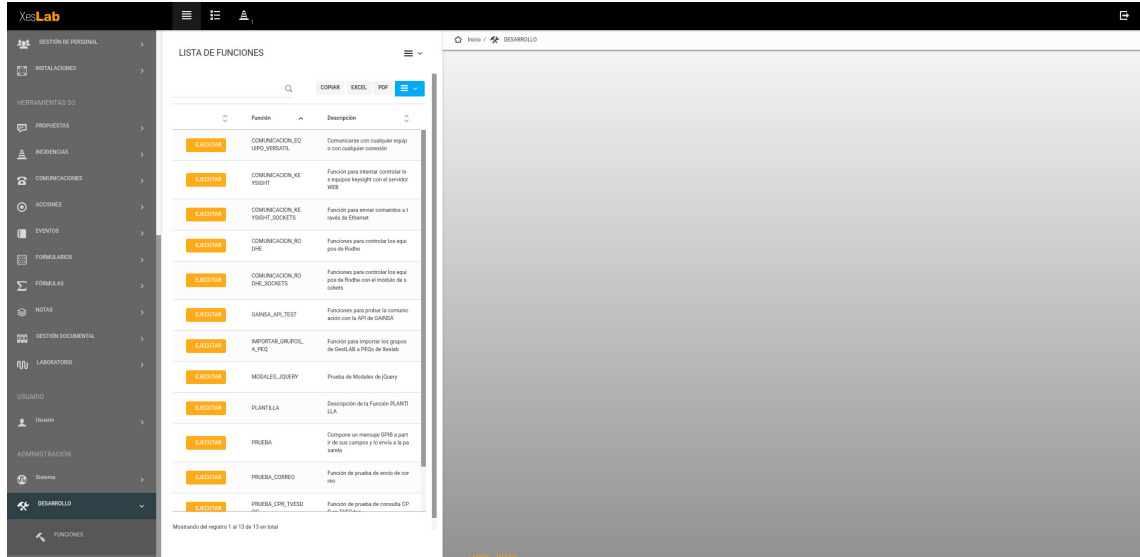


Figura 2.1: Sección de desenvolvemento en XesLab

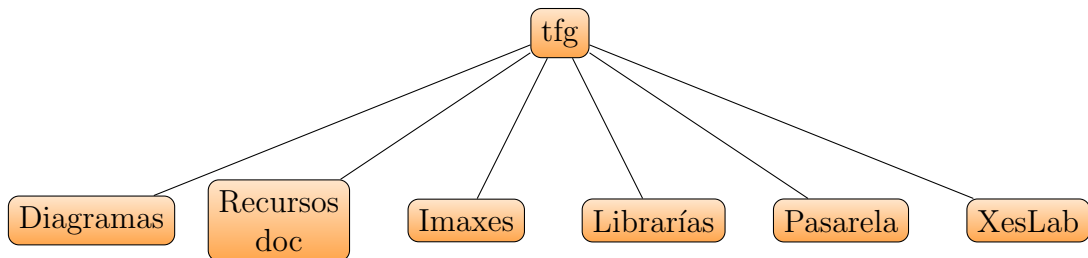


Figura 2.2: Estrutura do repositorio para o código da pasarela

como as renderizacións destes como imaxes.

- **Recursos doc:** documentos útiles para a elaboración da documentación do proxecto.
- **Imaxes:** fotografías ou capturas de pantalla incluídas na memoria.
- **Librarías:** ficheiros asociados a librarías de terceiros empregadas no proxecto. Consérvanse elementos útiles, como a documentación ou os exemplos de código.
- **Pasarela:** código C/C++ do software da pasarela.
- **XesLab:** tense acceso a todo o código da aplicación de XesLab, pero só se controla desta forma o versionado de aqueles ficheiros que foi necesario modificar para o desenvolvemento do proxecto; isto é, o código dos módulos de comunicación e certos ficheiros do módulo de equipos. O resto de código tense ignorado.

## Documentación

A maior parte da documentación realizouse en ferramentas que aportan control de versións e por iso non precisou ser incluída no repositorio.

- **Overleaf V2** empregouse para a memoria.
- **Microsoft Word** e **Microsoft Powerpoint** empregáronse, respectivamente, para a edición do anteprojecto e da presentación mostrada durante a defensa. Ambos contan co soporte de Microsoft Office 365 para control de versións.

### 2.1.2. Control de cambios

No relativo á documentación, os cambios foron solicitados polo director do traballo a través da función de comentarios de Overleaf. No momento en que se implementa un cambio, o comentario é resolto (aínda accesible a través da sección de "Comentarios resoltos" do proxecto en Overleaf).

No momento en que os cambios foron resoltos, ou que unha nova sección está dispoñible para revisión, notifícase ao director por correo electrónico.

Considérase que unha versión da documentación foi completada cando se resolveron todos os cambios solicitados e está lista para ser revisada. No tocante á versión inicial, é criterio da autora cando é o suficientemente completa como para enviar a revisión.

Con respecto ao código, decidiuse realizar un *commit* cada vez que o código está en condicións de facer probas informais (compila e executa) e se acaba de rematar:

- unha funcionalidade,
- unha parte importante dunha funcionalidade (por exemplo, a decodificación dos mensaxes dentro da funcionalidade de desempaquetamento),
- ou unha refactorización importante,

e, ademais disto, as probas informais non mostran erros aparentes.

O código considérase completamente finalizado cando é verificado e validado con éxito polos usuarios finais.

## 2.2. Xestión de riscos

Considérase un risco un evento ou condición incerta que, no caso de ocorrer, ten un efecto positivo ou negativo sobre os obxectivos dun proxecto; un risco ten unha causa, e, se ocorre, un efecto. Habitualmente xestiónanse os riscos negativos, para evitar unha consecuencia non desexable sobre o proxecto; neste caso, ao consistir este proxecto nun traballo de fin de grao, cobra maior sentido aínda xestionar únicamente os riscos negativos.

### 2.2.1. Fontes de risco e categorización dos mesmos

Para a recompilación de riscos, empregouse un *check-list* cuxos ítems son os índices do documento *Gestión de riesgos en proyectos software*[5], no que se lista un conxunto exhaustivo de riscos xenéricos para proxectos software. Analizouse a lista para determinar cales dos elementos son aplicables ao contexto, e particularizouse o risco ao caso específico xerado neste proxecto. Anque a maiores se intentou identificar riscos específicos deste proxecto, non se atopou ningún que non fora xerado xa a través desta lista nas primeiras categorías. En consecuencia, considerouse que estudar os riscos procedentes do proxecto por separado sería redundante, e que era preferente indicar en cada un dos ítems seleccionados da *check-list* como este risco se proxectaba sobre o noso proxecto.

Na figura 2.3 móstrase unha descomposición dos riscos en categorías que organizan os riscos pola súa fonte.

### 2.2.2. Matrices de probabilidade e impacto

Para modelar a gravidade dunha situación na que se materializara un risco, utilizaranse unha serie de índices.

- **Probabilidade:** indica a expectativa de materialización dun risco.
- **Impacto:** representa o efecto que a ocorrencia do risco tería sobre o proxecto.

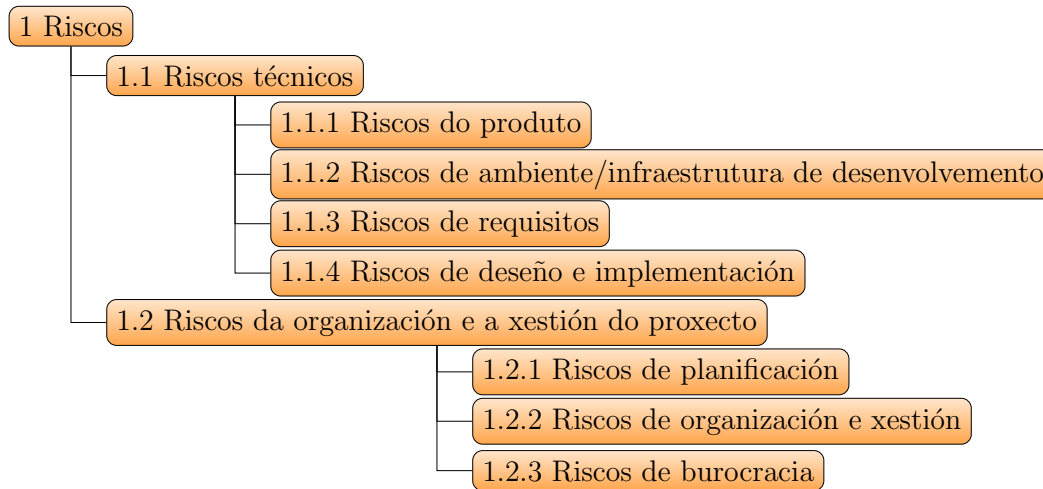


Figura 2.3: Estrutura de descomposición de riscos

- **Exposición:** é o produto dos dous índices anteriores. Canto maior sexa, tratarase dun risco máis probable ou que causará un efecto maior sobre o proxecto, sendo necesario prestarlle unha maior atención.

Nos cadros 2.1, 2.2 e 2.3 indícanse os umbrais seleccionados para a categorización dos riscos.

Valoración do impacto	
Repercusión en prazo	Impacto
Atraso asumible; solucionable a partir dun sobreesforzo de como máximo dúas semanas.	Baixo
Atraso serio; ademais dun sobreesforzo durante un período de tempo prolongado, impactará na calidade do proxecto e na cantidade de ciclos de revisión posibles.	Medio
Atraso catastrófico; o atraso afecta á data de entrega planificada.	Alto

Cadro 2.1: Valoración do impacto

Valoración da probabilidade	
Ocorrencia do risco	Probabilidade
$\geq 70\%$ (case segura)	Alta
Entre 30 % e 70 % (bastante probable)	Media
$\leq 30\%$ (pouco probable)	Baixa

Cadro 2.2: Valoración da probabilidade

Nivel de exposición ao risco				
		Probabilidade		
		Alta	Media	Baixa
Impacto	Alto	Alto	Alto	Medio
	Medio	Alto	Medio	Baixo
	Baixo	Medio	Baixo	Baixo

Cadro 2.3: Nivel de exposición ao risco

### 2.2.3. Identificación e análise de riscos

O proceso de identificación e análise de riscos implica definir e rexistrar os riscos que pode ter o proxecto, realizar un análise sobre eles para coñecer as súas características e, especialmente, o seu nivel de exposición, e deseñar respostas aos riscos para evitar consecuencias pouco desexables. Neste documento preséntase o resultado final da xestión: os riscos identificados, as accións realizadas sobre eles, os riscos que se xestionaron e os que se manifestaron, entre outros. Documentábase toda a información obtida nos cadros que compoñen o rexistro de riscos, para mellorar a lexibilidade.

#### Resposta aos riscos

É preciso definir un conxunto de estratexias de prevención para os riscos con exposición alta, para poder xestionalos adecuadamente e impedir a súa manifestación. As estratexias atópanse contempladas nas seguintes categorías:

- **Evitar:** prever a aparición do risco.
- **Mitigar:** reducir as consecuencias negativas do risco.
  - **Escalar:** transferir o problema a un superior. Trátase dun caso particular da transferencia de riscos, pero resulta interesante destacalo porque se utilizou explícitamente no proxecto.
- **Transferir:** trasladar a responsabilidade do problema a un terceiro.

As consecuencias dun risco tamén poden ser aceptadas, se a súa xestión constitúe un esforzo que consume unha cantidade maior de recursos dos que se gañan por non manifestarse o risco. Aceptar un risco podería describirse como unha “non acción”, que neste proxecto só se aplicará a riscos de baixa exposición. Concretamente, aplicarase a riscos con baixo impacto e alta ou media probabilidade, polas graves consecuencias que pode ter non xestionar un risco de alto impacto.

### Rexistro de riscos

Nesta sección inclúense os riscos identificados, ordenados pola súa exposición de acordo ao análise realizado.

Os riscos descríbense usando unha plantilla en forma de táboa, cos seguintes campos.

- Bloque de identificación
  - Número identificador do risco e nome corto
  - Número identificador do risco na *check-list* de referencia
  - Categoría
  - Descrición amigable do risco
- Bloque de accións executadas de tratamento do risco
  - Accións preventivas executadas
  - Accións correctivas executadas
    - Acción
    - Tempo que empregou a execución da acción
- Bloque de análise actual do risco (cálculo da exposición)
  - Probabilidade
  - Impacto
  - Exposición
- Bloque de planificación do risco
  - Estratexia de prevención (só se a exposición é alta ou o impacto é medio; se non, acéptase)
    - Indicador de risco
    - Acción
  - Acción de continxencia (só se a exposición é alta ou o impacto é medio; se non, acéptase)
- Manifestación do risco, se aplica
- Comentarios, se aplica

<b>RSC-1</b>		<b>As áreas descoñecidas requiren máis tempo do esperado</b>
<b>Descrición</b>		A curva de aprendizaxe dalgunha ferramenta é máis elevada do esperado, ou ben algunha tarefa toma máis tempo do que se planificou.
<b>Categoría</b>	<b>Identificador</b>	A.12, C.7
	<b>Descrición</b>	Planificación, ambiente e infraestrutura de desenvolvemento
<b>Accións executadas de tratamento do risco</b>		
<b>Preventivas</b>		Ningunha
<b>Correctivas</b>	<b>Acción</b>	Solicitouse axuda ao co-titor sobre o problema concreto (difícil instalación dos <i>drivers</i> GPIB).
	<b>Tempo</b>	42 horas
<b>Análise actual do risco</b>		
<b>Probabilidade</b>		Alta
<b>Impacto</b>		Alto
<b>Exposición</b>		Alta
<b>Planificación do risco</b>		
<b>Estratexia de prevención</b>	<b>Indicador</b>	Non aplica
	<b>Acción</b>	Para solucionar o problema concreto relacionado cos <i>drivers</i> GPIB en próximas instalacións, elaborouse un <i>script</i> coa secuencia de comandos adecuada.
<b>Estratexia de continxencia</b>		Solicitar axuda ao co-titor acerca da área problemática. Dedicar tempo á tarefa de forma extraordinaria se non se logra obter apoio técnico adecuado.
<b>Manifestación e consecuencias</b>		Este risco manifestouse; a información sobre a instalación dos <i>drivers</i> para a interacción con dispositivos GPIB era escasa e obsoleta, e tardouse en atopar a secuencia de pasos correcta.
<b>Comentarios</b>		Ningún

<b>RSC-2</b>		<b>Abandono dos plans do proxecto</b>
<b>Descrición</b>		Desenvólvese sen ter en conta o plan.
<b>Categoría</b>	<b>Identificador</b>	B.11
	<b>Descrición</b>	Organización e xestión
<b>Accións executadas de tratamento do risco</b>		
<b>Preventivas</b>		Ningunha
<b>Correctivas</b>	<b>Acción</b>	Ningunha
	<b>Tempo</b>	Non aplica
<b>Análise actual do risco</b>		
<b>Probabilidade</b>		Alta
<b>Impacto</b>		Alto
<b>Exposición</b>		Alta
<b>Planificación do risco</b>		
<b>Estratexia de prevención</b>	<b>Indicador</b>	Retrásanse ou non se realizan tarefas clave.
	<b>Acción</b>	Realizar comprobacións periódicas da planificación e intentar corrixir os desvíos, replantexando a planificación ou realizando traballo en remoto.
<b>Estratexia de continxencia</b>		Elaborar unha nova planificación de acordo ás circunstancias actuais e adecuarse a esa.
<b>Manifestación e consecuencias</b>		Este risco non se manifestou.
<b>Comentarios</b>		A planificación cambiou ao longo do proxecto por consecuencia da manifestación doutro risco, polo que non se considera que este se materializou neste caso.



RSC-3		Retrasos por procesos burocráticos
Descrición		Debido a que se trata dun TFG realizado no contexto dunha empresa, os tempos de firma de convenio e incorporación da desenvolvedora poden supoñer retrasos na planificación do proxecto.
Categoría	Identificador	L.1
	Descrición	Burocracia
Accións executadas de tratamento do risco		
Preventivas		Finalización do anteprojecto cunha marxe razoable para asegurar que a incorporación se realiza con tempo para rematar o proxecto.
Correctivas	Acción	Realizouse traballo en remoto durante algunhas tardes, días festivos e fins de semana. Ademais, replanificouse a data de entrega.
	Tempo	92 horas
Análise actual do risco		
Probabilidade		Media
Impacto		Alto
Exposición		Alta
Planificación do risco		
Estratexia de prevención	Indicador	Non aplica
	Acción	Ningunha
Estratexia de continxencia		Foméntase o traballo en remoto, en fins de semana e días festivos, especialmente aquel relativo á realización da memoria.
Manifestación e consecuencias		Este risco manifestouse; o anteprojecto entregouse a finais de febreiro e a desenvolvedora incorporouse en abril.
Comentarios		Ningún

RSC-4		Planificación incompleta
Descrición		A planificación non contempla tarefas clave, polo que é inadecuada.
Categoría	Identificador	A.3
	Descrición	Planificación
Accións executadas de tratamento do risco		
Preventivas		Revisase a lista de tarefas co titor para asegurar que sexa completa.
Correctivas	Acción	Ningunha
	Tempo	Non aplica
Análise actual do risco		
Probabilidade		Alta
Impacto		Medio
Exposición		Alta
Planificación do risco		
Estratexia de prevención	Indicador	Resulta preciso dedicar tempo a algo que non se asocia a ningunha tarefa do cronograma. O cronograma non comprende todas as fases necesarias para o desenvolvemento.
	Acción	Realizar unha replanificación do proxecto.
Estratexia de continxencia		Recorrer ao traballo en remoto para aumentar o número de horas dedicadas ao proxecto. Replanificar a data de entrega se é necesario.
Manifestación e consecuencias		Este risco non se manifestou.
Comentarios		Ningún

RSC-5		Planificación optimista (en tempo)
Descrición		Estímense para as tarefas planificadas tempos demasiado optimistas (mellor caso) en lugar de realistas (caso esperado).
Categoría	Identificador	A.2
	Descrición	Planificación
Accións executadas de tratamento do risco		
Preventivas		Revisanse as estimacións das tarefas co titor para asegurar que sexan coherentes.
Correctivas	Acción	Replanifícase o proxecto cambiando a data de finalización.
	Tempo	184 horas
Análise actual do risco		
Probabilidade		Baixa
Impacto		Alto
Exposición		Media
Planificación do risco		
Estratexia de prevención	Indicador	A execución do proxecto desvíase da planificación a pesar de cumprir ou incluso exceder os horarios.
	Acción	Realizar unha replanificación do proxecto.
Estratexia de continxencia		Recorrer ao traballo en remoto para aumentar o número de horas dedicadas ao proxecto. Replanificar a data de entrega se é necesario.
Manifestación e consecuencias		Este risco manifestouse no aspecto de que o alcance do proxecto foi excesivo para poder realizalo en tres meses.
Comentarios		Ningún

RSC-6		Retrasos no ciclo de revisión ou colaboración cos directores)
Descrición		O ciclo de revisión cos titores e implementación de cambios supón retrasos na entrega do proxecto.
Categoría	Identificador	B.6
	Descrición	Organización e xestión
Accións executadas de tratamento do risco		
Preventivas		Abordouse a revisión como un proceso por incrementos, no que as revisións se realizaban de forma paralela á redacción dos novos capítulos da documentación.
Correctivas	Acción	Replanifícase o proxecto cambiando a data de finalización. Realízase traballo en remoto.
	Tempo	184 horas
Análise actual do risco		
Probabilidade		Baixa
Impacto		Alto
Exposición		Media
Planificación do risco		
Estratexia de prevención	Indicador	Tárdase en enviar o traballo para revisión.
	Acción	Abordar as sucesivas revisións como un proceso por incrementos.
Estratexia de continxencia		Realizar traballo en remoto.
Manifestación e consecuencias		Este risco manifestouse; o director do proxecto tivo outros compromisos e a autora non puido dedicar horas de oficina á realización do proxecto a partir dunha certa data.
Comentarios		Ningún

<b>RSC-7</b>		<b>Inicio lento</b>
<b>Descrición</b>		As tarefas preliminares de análise e deseño toman demasiado tempo.
<b>Categoría</b>	<b>Identificador</b>	B.2
	<b>Descrición</b>	Organización e xestión
<b>Accións executadas de tratamento do risco</b>		
<b>Preventivas</b>		Enfócanse as sesións de deseño como procesos de <i>brainstorming</i> co co-titor que non toman máis dun día.
<b>Correctivas</b>	<b>Acción</b>	Ningunha
	<b>Tempo</b>	Non aplica
<b>Análise actual do risco</b>		
<b>Probabilidade</b>		Baixa
<b>Impacto</b>		Medio
<b>Exposición</b>		Baixa
<b>Planificación do risco</b>		
<b>Estratexia de prevención</b>	<b>Indicador</b>	Non aplica
	<b>Acción</b>	Ningunha
<b>Estratexia de continxencia</b>		Comézanse a implementar os módulos xa definidos e posponse o resto do deseño.
<b>Manifestación e consecuencias</b>		Este risco non se manifestou.
<b>Comentarios</b>		Ningún

<b>RSC-8</b>		<b>Deseño inadecuado</b>
<b>Descrición</b>		O deseño non respecta os requisitos non funcionais de modularidade e facilidade para engadir soporte para unha nova interface hardware.
<b>Categoría</b>	<b>Identificador</b>	K.1, K.2, K.3
	<b>Descrición</b>	Deseño e implementación
<b>Accións executadas de tratamento do risco</b>		
<b>Preventivas</b>		Procurárase que a responsabilidade de cada módulo sexa específica, e que ningún deles xestione cousas demasiado variadas.
<b>Correctivas</b>	<b>Acción</b>	Ningunha
	<b>Tempo</b>	Non aplica
<b>Análise actual do risco</b>		
<b>Probabilidade</b>		Baixa
<b>Impacto</b>		Medio
<b>Exposición</b>		Baixa
<b>Planificación do risco</b>		
<b>Estratexia de prevención</b>	<b>Indicador</b>	Non aplica
	<b>Acción</b>	Ningunha
<b>Estratexia de continxencia</b>		Refactorízase na medida en que sexa sinxelo e posible; se non, acéptase o risco.
<b>Manifestación e consecuencias</b>		Este risco non se manifestou.
<b>Comentarios</b>		Ningún

RSC-9		Incompatibilidades entre módulos
Descrición		Os compoñentes deseñados e implementados son incompatibles entre eles
Categoría	Identificador	K.9
	Descrición	Deseño e implementación
Accións executadas de tratamento do risco		
Preventivas		Nas sesións de deseño co co-titor do traballo, establécese unha interface común para os compoñentes, que asegure a correcta interpretación das mensaxes en todos os subsistemas involucrados.
Correctivas	Acción	Ningunha
	Tempo	Non aplica
Análise actual do risco		
Probabilidade		Baixa
Impacto		Medio
Exposición		Baixa
Planificación do risco		
Estratexia de prevención	Indicador	Non aplica
	Acción	Ningunha
Estratexia de continxencia		Redefinición da interface común e reimplementación dos módulos de xestión ou preparación de mensaxes.
Manifestación e consecuencias		Este risco non se manifestou.
Comentarios		Ningún

RSC-10		Código de baixa calidade
Descrición		O código escrito é difícil de comprender ou ten gran cantidade de erros.
Categoría	Identificador	K.7
	Descrición	Deseño e implementación
Accións executadas de tratamento do risco		
Preventivas		Formación na forma correcta de programar nas linguaxes empregadas e uso de boas prácticas (por exemplo, uso de punteiros intelixentes en C++).
Correctivas	Acción	Ningunha
	Tempo	Non aplica
Análise actual do risco		
Probabilidade		Baixa
Impacto		Medio
Exposición		Baixa
Planificación do risco		
Estratexia de prevención	Indicador	Non aplica
	Acción	Ningunha
Estratexia de continxencia		Refactorización do código na medida en que sexa posible, con accións sinxelas pero con impacto significativo; por exemplo, refactorización das variables para que teñan nomes ilustrativos. Se non é posible, acéptase o risco.
Manifestación e consecuencias		Este risco non se manifestou.
Comentarios		Ningún

<b>RSC-11</b>		<b>Rigor inadecuado na xestión</b>
<b>Descrición</b>		Dedícase moito tempo aos detalles de xestión, ou moi pouco, resultando en retrasos ou un proxecto mal xestionado.
<b>Categoría</b>	<b>Identificador</b>	L.5, L.6
	<b>Descrición</b>	Burocracia
<b>Accións executadas de tratamento do risco</b>		
<b>Preventivas</b>		Pregúntase aos titores se se deben incluír os detalles que se consideran demasiado específicos.
<b>Correctivas</b>	<b>Acción</b>	Ningunha
	<b>Tempo</b>	Non aplica
<b>Análise actual do risco</b>		
<b>Probabilidade</b>		Media
<b>Impacto</b>		Baixo
<b>Exposición</b>		Baixa
<b>Planificación do risco</b>		
<b>Estratexia de prevención</b>	<b>Indicador</b>	Non aplica
	<b>Acción</b>	Ningunha
<b>Estratexia de continxencia</b>		Acéptase o risco.
<b>Manifestación e consecuencias</b>		Este risco non se manifestou.
<b>Comentarios</b>		Ningún

<b>RSC-12</b>		<b>Non dispoñibilidade dos equipos hardware</b>
<b>Descrición</b>		Non se ten acceso aos equipos hardware necesarios para desenvolver.
<b>Categoría</b>	<b>Identificador</b>	C.1
	<b>Descrición</b>	Ambiente e infraestrutura de desenvolvemento
<b>Accións executadas de tratamento do risco</b>		
<b>Preventivas</b>		Seleccionar equipos pouco utilizados para o desenvolvemento; contar con varios equipos equivalentes para o desenvolvemento e as probas.
<b>Correctivas</b>	<b>Acción</b>	Substituíuse a pasarela de Televés por unha Raspberry Pi, que a efectos de desenvolvemento e interfaces hardware necesarias ten as mesmas características.
	<b>Tempo</b>	0 horas
<b>Análise actual do risco</b>		
<b>Probabilidade</b>		Media
<b>Impacto</b>		Baixo
<b>Exposición</b>		Baixa
<b>Planificación do risco</b>		
<b>Estratexia de prevención</b>	<b>Indicador</b>	Non aplica
	<b>Acción</b>	Ningunha
<b>Estratexia de continxencia</b>		Substituír o equipo por outro equivalente que estea dispoñible.
<b>Manifestación e consecuencias</b>		Este risco manifestouse: non se tivo acceso ás pasarelas de Televés para o desenvolvemento do software.
<b>Comentarios</b>		Ningún

RSC-13		Requisitos mal definidos
Descrición		Os requisitos non se definen claramente, son ambiguos ou xenéricos.
Categoría	Identificador	G.1, G.2, G.3
	Descrición	Requisitos
Accións executadas de tratamento do risco		
Preventivas		Extraer os requisitos de fontes pre-existentes sobre Xes-Lab e contrastalos con frecuencia cos usuarios e o co-titor, xa que ao estar fisicamente no mesmo espazo resulta cómodo.
Correctivas	Acción	Ningunha
	Tempo	Non aplica
Análise actual do risco		
Probabilidade		Baixa
Impacto		Baixo
Exposición		Baixa
Planificación do risco		
Estratexia de prevención	Indicador	Non aplica
	Acción	Ningunha
Estratexia de continxencia		Acéptase o risco.
Manifestación e consecuencias		Este risco non se manifestou.
Comentarios		Ningún

RSC-14		Desenvolvemento de funcións software erróneas
Descrición		Impleméntase funcionalidades que non son as que os usuarios requirían.
Categoría	Identificador	H.4
	Descrición	Produto
Accións executadas de tratamento do risco		
Preventivas		Contrástanse os requisitos co co-titor antes de comezar a implementación dos módulos, e durante esta.
Correctivas	Acción	Ningunha
	Tempo	Non aplica
Análise actual do risco		
Probabilidade		Baixa
Impacto		Baixo
Exposición		Baixa
Planificación do risco		
Estratexia de prevención	Indicador	Non aplica
	Acción	Ningunha
Estratexia de continxencia		Acéptase o risco.
Manifestación e consecuencias		Este risco non se manifestou.
Comentarios		Ningún

<b>RSC-15</b>		<b>Traballo extra debido a requisitos de compatibilidade con XesLab ou outras interfaces</b>
<b>Descrición</b>		É preciso realizar modificacións non previstas por compatibilidade con outras compoñentes.
<b>Categoría</b>	<b>Identificador</b>	H.8, H.9
	<b>Descrición</b>	Produto
<b>Accións executadas de tratamento do risco</b>		
<b>Preventivas</b>		Estúdanse previamente os requisitos de compatibilidade dos elementos indicados.
<b>Correctivas</b>	<b>Acción</b>	Adáptanse as interfaces existentes da forma máis sinxela posible.
	<b>Tempo</b>	2 horas
<b>Análise actual do risco</b>		
<b>Probabilidade</b>		Baixa
<b>Impacto</b>		Baixo
<b>Exposición</b>		Baixa
<b>Planificación do risco</b>		
<b>Estratexia de prevención</b>	<b>Indicador</b>	Non aplica
	<b>Acción</b>	Ningunha
<b>Estratexia de continxencia</b>		Adáptanse as interfaces existentes da forma máis sinxela posible.
<b>Manifestación e consecuencias</b>		Este risco manifestouse; certas funcionalidades do módulo de rede de XesLab enviaban mensaxes inesperadas ao software da pasarela.
<b>Comentarios</b>		Ningún

Cadro 2.4: Características dos riscos do proxecto

## Clasificación dos riscos por exposición

Nivel de exposición ao risco				
		Probabilidade		
		Alta	Media	Baixa
Impacto	Alto	RSC-1, RSC-2	RSC-4	RSC-5, RSC-6
	Medio	RSC-3	-	RSC-7, RSC-8, RSC-9, RSC-10
	Baixo	-	RSC-11, RSC-12	RSC-13, RSC-14, RSC-15

Cadro 2.5: Clasificación dos riscos por exposición

## 2.3. Xestión do tempo

### 2.3.1. Metodoloxía do desenvolvemento

Decidiuse planificar o proxecto de acordo cun **ciclo de vida por incrementos**, que se adecuaba particularmente ben á estrutura en módulos deste proxecto.

### 2.3.2. Cronograma

Na figura 2.3.2 móstrase o cronograma do proxecto.

#### Fitos do proxecto

Establécese un fito no fin de cada incremento, dando así lugar a 5 fitos, cuxos detalles se mostran no cadro 2.6.

Id	Nome	Entregable	Data
H-1	Fin do incremento de tarefas preliminares	Documentación preliminar do proxecto: capítulos 1, 2, 3 e 4 da memoria do proxecto	19/04/2019
H-2	Módulos de comunicación en XesLab funcionais	Código dos módulos de comunicación de XesLab e documentación relativa (capítulo 5 da memoria)	22/05/2019
H-3	Fin do incremento de XesLab	Código e documentación das probas dos módulos de comunicación de XesLab	27/05/2019
H-3	Software da pasarela funcional e fin do incremento	Código e documentación (capítulos 4 e 6 da memoria) da pasarela	17/07/2019
H-4	Fin do proxecto	Código do proxecto e documentación completos	28/06/2019

Cadro 2.6: Cadro de identificación de interesados



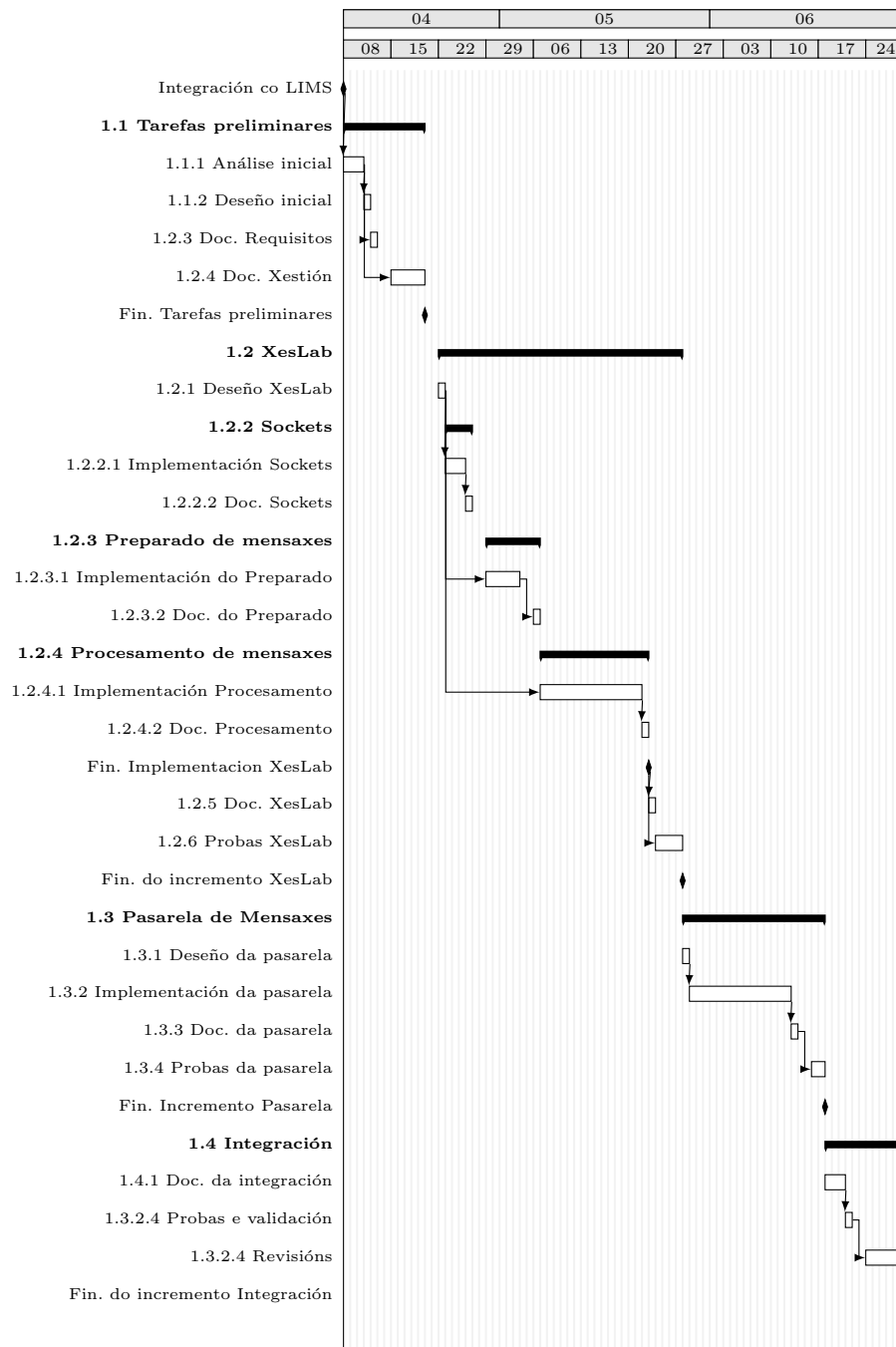


Figura 2.4: Cronograma

### 2.3.3. Variacións no cronograma

Durante a realización do proxecto déronse certas condicións que fixeron que fora necesario realizar unha replanificación.

Realizouse unha replanificación para o proxecto pai deste, polo que a implementación do módulo de medidas, do que se consideraba que dependían os módulos de planificación correspondentes a este proxecto, foi trasladada a un incremento posterior [6]. Debido a isto, no momento en que se iniciou o desenvolvemento dos módulos de comunicación, existía unha incertidume acerca da implementación para certos conceptos que afectaban aos módulos. Considerouse que sería intelixente interromper o desenvolvemento do incremento correspondente á creación dos módulos de comunicación. No seu lugar, realizouse a implementación do software da pasarela e proseguíuse máis tarde ca tarefa interrompida. Ademais disto, trasladouse a elaboración da documentación técnica dos módulos de comunicación a un punto no que a implementación destes estivera completamente rematada, pola comodidade que supón, ao tratarse de tarefas de redacción. Tamén se trasladaron todas as probas ao seu propio incremento, que depende da realización de todos os incrementos relacionados co deseño detallado e a documentación.

Por último, decidiuse retrasar a data de fin do proxecto a finais de xullo. A duración habitual dos proxectos realizados como traballos de fin de grao é de catro meses; porén, as características particulares deste proxecto fixeron que se iniciara un mes máis tarde do habitual, de forma relativa ao curso escolar. Por isto, considerouse que tres meses non era un período o suficientemente longo como para lograr rematar un proxecto deste alcance coa calidade desexada.

Na figura 2.3.3 represéntase o diagrama de Gantt de seguimento do proxecto.

## 2.4. Xestión de custos

Nesta sección estímense os custos do proxecto, aos que contribúen tanto o persoal que participou neste proxecto como o equipamento empregado. Habitualmente, nos traballos de fin de grao, trátase dun exercicio teórico; neste caso cobra unha maior importancia debido á necesidade de uso do instrumental do laboratorio, e a que a desenvolvedora supón un custo real para a empresa.

### 2.4.1. Unidades e precisión

A moeda empregada é o euro. Os custos asociados aos recursos humanos poderán expresarse tanto anualmente (€/ano) como a nivel de hora (€/hora), indicándose a forma e unidade en cada caso. Esta cantidade non consiste no salario do recurso, senón que se trata do custo que o recurso supón para a empresa.

No caso dos custos de equipamento, exprésanse os valores de compra e residual en euros, e o custo de amortización en euros por hora, xa que a unidade máis

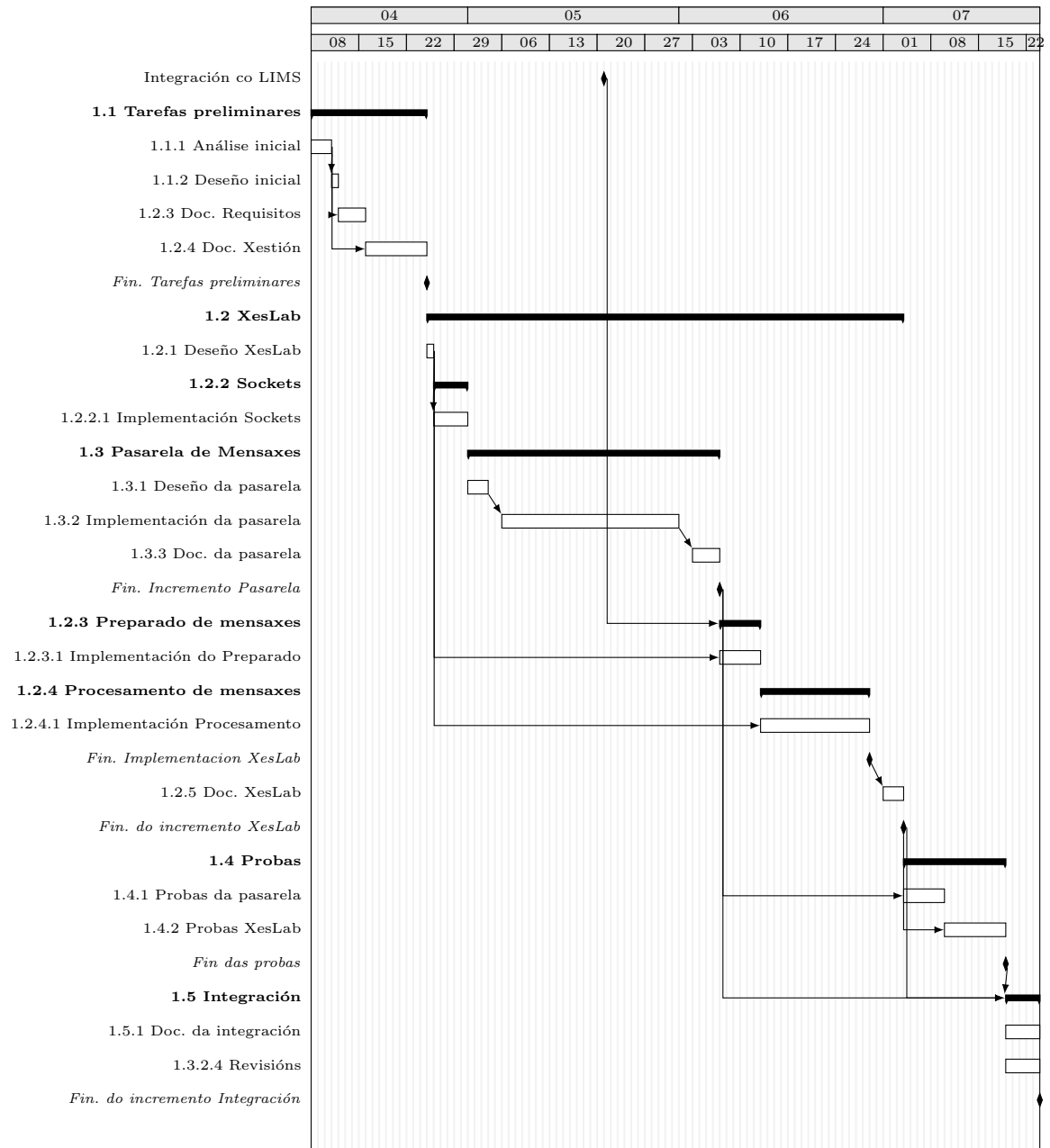


Figura 2.5: Cronograma de seguimento

adecuada para expresar o tempo de uso de cada equipo é a hora.

Admítese un 5 % de precisión nas estimacións. Valores con precisión decimal superior ás dúas cifras serán redondeados ás dúas cifras, podendo así representar ata céntimos de euro. Utilizarase o método estándar de redondeo.

Ningunha das cantidades inclúe a porcentaxe correspondente ao imposto do valor engadido, que se considera aparte.

## 2.4.2. Custos de equipamento

A maioría dos recursos materiais non foron adquiridos expresamente para o proxecto, senón que foron recursos pertencentes a Televés ou á desenvolvedora que se empregaron para o desenvolvemento. Polo tanto, para a xestión de custos, considérase o seu custo de amortización, que representa o custo que supón o uso do recurso durante o proxecto, debido á súa consecuenta perda de valor. A ecuación 2.1 indica o cálculo do custo de amortización, e o cadro 2.7 os custos por cada activo material do proxecto.

$$\text{custo de amortización por hora} = \frac{(\text{prezo de compra} - \text{valor residual})}{\text{vida técnica} * 365 * 24} \quad (2.1)$$

Estímase o valor residual dos equipos informáticos como o 15 % do seu valor inicial, e o dos instrumentos de medida como o 20 %. A vida técnica de cada equipo estímase individualmente.

Activo	Prezo de compra	Valor residual	Vida técnica	Amortización/hora
Portátil persoal <sup>1</sup>	700€	105€	5 anos	0,01€/h
Sobremesa de oficina con periféricos	809€	121,35€	5 anos	0,01€/h
NI-USB-HS	747€	149,40€	15 anos	0,004€/h
MXG Vector Signal Generator	59.495,55€	11.899,11€	20 anos	0,27€/h
Rodhe & Schwarz Signal and Spectrum Analyzer FSV7	25.856,60€	3878,49€	20 anos	0,12€/h
Switch e accesorios	60€	9€	10 anos	0,001€/h
Servidor de Ladetel	Trátase dun servidor virtual, compartindo unha máquina real con outros servidores virtuais de Televés; contéplase como custo indirecto.			
Raspberry Pi 3B+ e accesorios	79,99€	Este activo comprouse para este proxecto, polo que o seu prezo de compra repercute directamente nos custos do proxecto.		

Cadro 2.7: Cadro de custos de amortización

<sup>1</sup>Refléxanse as horas de uso dun dispositivo persoal porque foi preciso realizar traballo en remoto para cumprir as restricións temporais; porén, isto non foi un custo para a empresa. Decidiuse contemplalo igualmente nos custos totais do proxecto como se fose un equipo da empresa, xa que esta presta equipos aos empregados que deban traballar en remoto, e non se fixo así neste caso porque engadía complicacións burocráticas innecesarias.

### 2.4.3. Custos de recursos humanos

Para coñecer os custos anuais nos que a USC e Televés incorren, respectivamente, a consecuencia de empregar ao titor e ao cotitor, utilizáronse as tablas salariais do Servizo de Xestión de Persoal da USC [7] e o convenio colectivo de empresa de Televés [8]. Suponse que o custo para a empresa é o salario bruto do traballador máis un 30 % deste, e obtense o custo por hora asumindo unha media de 1750 horas traballadas ao ano.

No caso da desenvolvedora, é preciso optar por un sistema distinto, xa que só supón un custo durante a duración do proxecto, que non chega ao ano. Neste caso, calculouse o custo por hora baseándose no custo por día, que si é coñecido. Sabendo que a desenvolvedora obtén unha bolsa por un valor de 300€ ao mes, e que a duración da bolsa comprende dende a segunda semana de abril de 2019 ata finais de xuño de 2019 (60 días, tendo en conta festivos pagados), o custo por día é de 15 euros.

Recurso	Custo anual	Custo por hora
Profesor titular de universidade	44.452,12€ /ano	25,40€ /h
Enxeñeiro de telecomunicacións	52.340,18€ /ano	29,90€ /h
Desenvolvedora C/C++ en prácticas	-	1,88€ /h

Cadro 2.8: Cadro de custos de recursos humanos

### 2.4.4. Custos indirectos

Son aqueles que non se poden trazar de forma directa a activos ou recursos do proxecto, senón que proceden da contorna na que se desenvolve; facturas de electricidade, auga, infraestruturas, mobiliario... A USC especifica para proxectos TIC uns custos indirectos do 21 % do total de custos directos do proxecto.

### 2.4.5. Custos totais

No cadro 2.9 móstrase a análise de custos realizada a partir do custo por hora de cada recurso, e das horas durante as que foi empregado.

Recurso	Horas de uso	Custo por hora	Custo do recurso
<b>Recursos materiais</b>			
Portátil persoal	72 horas	0,01€/h	72,00€
Sobremesa de oficina con periféricos	400 horas	0,01€/h	4,00€
NI-USB-HS	272 horas	0,004€/h	1,09€
MXG Vector Signal Generator	272 horas	0,27€/h	73,44€
Rodhe & Schwarz Signal and Spectrum Analyzer FSV7	272 horas	0,12€/h	32,64€
Switch e accesorios	272 horas	0,01€	2,72€
Raspberry Pi 3B+ e accesorios			79,99€
<b>Total de recursos materiais</b>			<b>265,88€</b>
<b>Recursos humanos</b>			
Profesor titular de universidade	10 horas	25,40€/h	0€ <sup>2</sup>
Enxeñeiro de telecomunicacións	150 horas	29,90€/h	3.450,00€
Desenvolvedora C/C++ en prácticas	472 horas	1,88€/h	887,36€
<b>Total de recursos humanos</b>			<b>4.337,36€</b>
<b>Custos directos</b>			<b>4603,24€</b>
<b>Custos indirectos</b>			<b>966,68€</b>
<b>Custos totais</b>			<b>5569,92€</b>

Cadro 2.9: Cadro de custos totais

## 2.5. Xestión de comunicacións e interesados

### 2.5.1. Interesados

#### Identificación de interesados

Os interesados do proxecto son, nun nivel de maior implicación, a alumna desenvolvedora, o titor e o cotitor; e nun rol máis pasivo, como usuarios do sistema, o persoal de Ladetel. No cadro 2.10 identifícanse aos interesados, e no cadro 2.11 establécese a guía de control da implicación de cada un.

<sup>2</sup>Refléxanse as horas nas que foi necesaria a implicación do profesor titular de universidade, como director dun proxecto académico e de cara á revisión do traballo. Porén, o seu custo é 0€ xa que non supón un custo para a empresa.

<b>Id</b>	<b>Nome</b>	<b>Rol</b>	<b>Correo electrónico</b>
<b>INT-1</b>	Rosa Elena Veiga Otero	Desenvolvedora	rosaelena.veiga@rai.usc.es
<b>INT-2</b>	Miguel Duro Liñares	Cotitor	migdur@televes.es
<b>INT-3</b>	José Ángel Taboada González	Titor	joseangel.taboada@usc.es
<b>INT-4</b>	Empregados de Ladetel	Usuarios	-

Cadro 2.10: Cadro de identificación de interesados

Os detalles de cada interesado descríbense usando unha plantilla en forma de táboa, cos seguintes campos.

- Número identificador do interesado dentro da lista
- Nome completo
- Empresa ou grupo ao que pertence o interesado
- Que se require dese interesado
- Que espera ese interesado do proxecto
- Nivel de influencia potencial
- Fase na que o interese do interesado é maior

<b>INT-1</b>	<b>Rosa Elena Veiga Otero</b>
<b>Empresa ou grupo</b>	Televés, USC
<b>Rol no proxecto</b>	Alumna desenvolvedora
<b>Requerimentos primordiais</b>	Desenvolvemento do proxecto de acordo ao plan e ás necesidades dos usuarios, cumprimento dos obxectivos e das restricións do proxecto
<b>Expectativas principais</b>	Logro dos obxectivos do proxecto, defensa do proxecto na Universidade como Traballo de Fin de Grao
<b>Influencia potencial</b>	Alta
<b>Fases de maior interese</b>	Todo o proxecto

<b>INT-2</b>	<b>Miguel Duro Liñares</b>
<b>Empresa ou grupo</b>	Televés
<b>Rol no proxecto</b>	Co-titor
<b>Requerimentos primordiais</b>	Desenvolvemento dos módulos de XesLab que dependen dos módulos desenvolvidos neste proxecto, e das utilidades do sistema empregadas no módulo de comunicación. Apoio á desenvolvedora; proporcionar a esta material de formación na arquitectura de XesLab, e información sobre os requisitos para o sistema
<b>Expectativas principais</b>	Logro dos obxectivos do proxecto no prazo e recursos especificados, finalización dun subsistema funcional que solucione as necesidades de comunicación automática dos equipos
<b>Influencia potencial</b>	Alta
<b>Fases de maior interese</b>	Deseño, implementación e probas do software

<b>INT-3</b>	<b>José Ángel Taboada González</b>
<b>Empresa ou grupo</b>	USC
<b>Rol no proxecto</b>	Titor
<b>Requerimentos primordiais</b>	Apoio á desenvolvedora en materias relacionadas coa xestión do proxecto, redacción da documentación, e cuestións burocráticas ou procedimentais do Traballo de Fin de Grao
<b>Expectativas principais</b>	Logro dos obxectivos do proxecto no prazo e recursos especificados
<b>Influencia potencial</b>	Media
<b>Fases de maior interese</b>	Redacción do anteprojecto, redacción da documentación, presentación de solicitudes

<b>INT-4</b>	<b>Empregados de Ladetel</b>
<b>Empresa ou grupo</b>	Televés
<b>Rol no proxecto</b>	Usuarios
<b>Requerimentos primordiais</b>	Proporcionar información para a desenvolvedora sobre os requisitos do sistema, o contexto do proxecto e o funcionamento do laboratorio, de cara á recollida de requisitos, a redacción da documentación e a validación
<b>Expectativas principais</b>	Consecución dos obxectivos, obtención dun LAS funcional
<b>Influencia potencial</b>	Media
<b>Fases de maior interese</b>	Recollida de requisitos, redacción da documentación, validación

Cadro 2.11: Detalles dos interesados

## Control de interesados

Debido a que este proxecto non ten un conxunto de interesados demasiado grande, as relacións entre estes son sinxelas e todos os usuarios apoian o proxecto, non ten sentido crear unha matriz describindo o nivel de interese dos diferentes interesados no proxecto.

- Os interesados INT-1 e INT-2 (a desenvolvedora e o cotitor) teñen unha posición líder no proxecto.
- Os interesados INT-3 e INT-4 (o titor e o persoal de Ladetel) teñen unha posición de apoio ao proxecto.

Este nivel de interese mantívose constante nestes valores durante todo o proxecto, xa que a desenvolvedora e o cotitor son os desenvolvedores do proxecto e está nos seus intereses que o proxecto saia adiante, ademais de ser a súa responsabilidade; mentres que os usuarios e o titor apoian o proxecto xa que é de beneficio para eles, pero non é a súa responsabilidade sacalo adiante.

### 2.5.2. Comunicacións

Ao ser este un proxecto realizado no marco dunha empresa, as comunicacións co cotitor foron triviais xa que ambos o cotitor e a desenvolvedora están físicamente na mesma oficina. As comunicacións realizadas previamente á incorporación



da desenvolvedora ao laboratorio tiveron lugar a través de correo electrónico e reunións puntuais.

As comunicacións co titor mantivéronse principalmente a través de correo electrónico, ademais de reunións ocasionais. Non se considerou necesario realizar reunións periódicas xa que estas tratarían temas esencialmente académicos e de xestión; o coñecemento relativo ao proxecto e o seu contexto, relevante para a implementación do sistema, está no laboratorio. Porén, si tiveron lugar reunións puntuais durante o proceso de elaboración da documentación.

É importante destacar que neste proxecto existe unha división marcada do coñecemento. O titor non coñece os detalles de implementación de XesLab nin o funcionamento do laboratorio; mentres que os empregados de Ladetel non coñecen os detalles e as necesidades que un traballo de fin de grao impón sobre o proxecto, nin cal foi a formación de xestión de proxectos que se recibiu en materias da titulación, información que si ten o titor.

Ademais disto, non aplica a fórmula tradicional de  $\frac{n(n-1)}{2}$  canles de comunicación para  $n$  interesados, xa que non existe comunicación directa entre o titor e o persoal de Ladetel; nin debería ser necesaria, xa que a desenvolvedora debe contar cos suficientes coñecementos de detalles de implementación e contexto como para informar ao titor do que precise saber, e viceversa.

Por isto, é preciso poñer especial coidado nas comunicacións entre interesados. Seguiranse as seguintes pautas:

- Comunicarase só a información mínima necesaria para lograr os obxectivos da comunicación; deste xeito, evítaranse malentendidos por pretender expor conceptos moi complexos, posiblemente a través dunha canle inadecuada.
- Empregaranse representacións gráficas do software sempre que sexa posible.
- Preferirase o medio da reunión en lugar do correo electrónico se é preciso discutir conceptos de deseño ou información de requisitos.



# Capítulo 3

## Especificación de requisitos

### 3.1. Relación cos requisitos do LIMS

De forma previa ao comezo a este proxecto, realizouse en Ladetel o deseño das implementacións concretas para o sistema de xestión da información (LIMS) e o sistema de automatización do laboratorio (LAS). Isto está directamente relacionado coa arquitectura de XesLab.

A información recollida nesta sección sinala aspectos da especificación de requisitos para XesLab [2].

- XesLab implementa a totalidade do sistema de xestión, e parte do sistema de automatización. Como xa foi mencionado, contén módulos para xestionar cada un dos elementos que forman parte do traballo do laboratorio: rexistro de dispositivos, xestión de incidencias, realización de compras, etc. Todos estes módulos formarían parte do LIMS.

Por outra parte, conta cuns módulos de comunicación que conteñen as funcionalidades que permiten intercambiar información cos equipos. Estes módulos forman parte do sistema de automatización.

- As pasarelas mais o seu software completan o sistema de automatización.



Figura 3.1: Implementacións do LIMS e o LAS

As funcionalidades que aporta o LAS só son accesibles a través do LIMS. Por isto, certos requisitos para o LIMS tradúcense directamente en requisitos para o LAS, xa que o primeiro está, en parte, limitado polo segundo. Nos cadros 3.1 e 3.2 móstranse aqueles requisitos de XesLab que son relevantes para o desenvolvemento deste software; só se comprenden os requisitos de máis baixo nivel, isto é, requisitos de información, de conduta do sistema, e requisitos non funcionais.

### 3.1.1. Requisitos funcionais do LIMS con impacto no LAS

ID	Descrición e <i>especificacións funcionais</i> , se aplica
27	As instalacións e as condicións ambientais deben ser adecuadas para as actividades do laboratorio e non deben afectar adversamente á validez dos resultados. <i>XesLab estará comunicado cos dispositivos encargados de controlar as condicións ambientais do laboratorio (sistema de automatización LAS).</i>
29	O laboratorio debe realizar o seguimento, controlar e rexistrar as condicións ambientais de acordo coas especificacións, os métodos ou procedementos pertinentes, ou cando inflúen na validez dos resultados. <i>XesLab estará comunicado cos dispositivos encargados de controlar as condicións ambientais do laboratorio (sistema de automatización LAS).</i>
35	O equipo utilizado para medición debe ser capaz de lograr a exactitude da medición e/ou a incertidume de medición requeridas para proporcionar un resultado válido. <i>En XesLab estará registrada a incertidume asociada a cada equipo, como unha das fontes de incertidume asociadas á medición.</i>

Cadro 3.1: Requisitos funcionais do LIMS con impacto no LAS

### 3.1.2. Requisitos non funcionais do LIMS con impacto no LAS

Característica	Eficiencia		
	ID	Sub-característica	Descrición
	01	Comportamento temporal	Tempos de resposta e procesamento
	02	Utilización de recursos	Cantidades e tipos de recursos empregados
	03	Capacidade	Grao en que os límites máximos do sistema cumpren cos requisitos

<b>Característica</b>	<b>Manteñibilidade</b>	
<b>ID</b>	<b>Sub-característica</b>	<b>Descrición</b>
20	Modularidade	Capacidade dun sistema que permite que un cambio nunha compoñente tiña un impacto mínimo nas demais
21	Reusabilidade	Capacidade dun activo que permite que sexa utilizado en máis dun sistema software ou na construción doutros activos
22	Analizabilidade	Capacidade ca que se pode avaliar o impacto dun determinado cambio sobre o resto do software, diagnosticar as deficiencias ou causas de fallos no software, ou identificar as partes a modificar
23	Capacidade de ser modificado	Capacidade do produto que permite que sexa modificado de forma efectiva e eficiente sen introducir defectos ou degradar o desempeño
24	Capacidade de ser probado	Facilidade coa que se poden establecer criterios de proba para o sistema e coa que se poden levar a cabo as probas para determinar se se cumpren ditos criterios

Característica	Seguridade	
ID	Sub-característica	Descrición
15	Confidencialidade	Capacidade de protección contra o acceso de datos e información non autorizados, xa sexa accidental ou deliberadamente
16	Integridade	Capacidade do sistema para previr accesos ou modificacións non autorizados a datos ou programas.
17	Non repudio	Capacidade de demostrar as accións ou eventos que tiveron lugar, de xeito que esas accións ou eventos non poidan ser repudiados posteriormente
18	Autenticidade	Capacidade de demostrar a identidade dun suxeito ou recurso
19	Responsabilidade	Capacidade de rastrexar de forma inequívoca as accións dunha entidade

Característica	Portabilidade	
ID	Sub-característica	Descrición
25	Adaptabilidade	Capacidade do produto que lle permite ser adaptado de forma efectiva e eficiente a diferentes entornos determinados de hardware, software, operacionais ou de uso
26	Facilidade de instalación	Facilidade coa que o produto se pode instalar e/ou desinstalar de forma exitosa nunha determinada contorna
27	Capacidade de reemprazo	Capacidade do produto para ser utilizado en lugar doutro produto software determinado co mesmo propósito e no mesmo entorno

Característica		Fiabilidade
ID	Sub-característica	Descrición
11	Madurez	Capacidade do sistema para satisfacer as necesidades de fiabilidade en condicións normais
12	Dispoñibilidade	Capacidade do sistema ou compoñente de estar operativo e accesible para o seu uso cando se require
13	Tolerancia a fallos	Capacidade do sistema ou compoñente para operar según o previsto en presenza de fallos hardware ou software
14	Capacidade de recuperación	Capacidade do produto software para recuperar os datos directamente afectados e restablecer o estado desexado do sistema en caso de interrupción ou fallo.

Cadro 3.2: Requisitos non funcionais do LIMS con impacto no LAS

### 3.2. Actores

<b>Identificador</b>	<b>ACT-1</b>
<b>Nome</b>	LIMS
<b>Descrición</b>	Xestión dos datos relacionados cunha medida; recupera os comandos e os equipos asociados a ela, e emprega as funcionalidades proporcionadas polo módulo de procesamento de comandos para a comunicación. É o punto de integración do LIMS e o LAS.

<b>Identificador</b>	<b>ACT-2</b>
<b>Nome</b>	Equipo
<b>Descrición</b>	Un instrumento de medición con capacidades de manexo en remoto, conectado a través dunha interface hardware contemplada no sistema.

### 3.3. Casos de uso

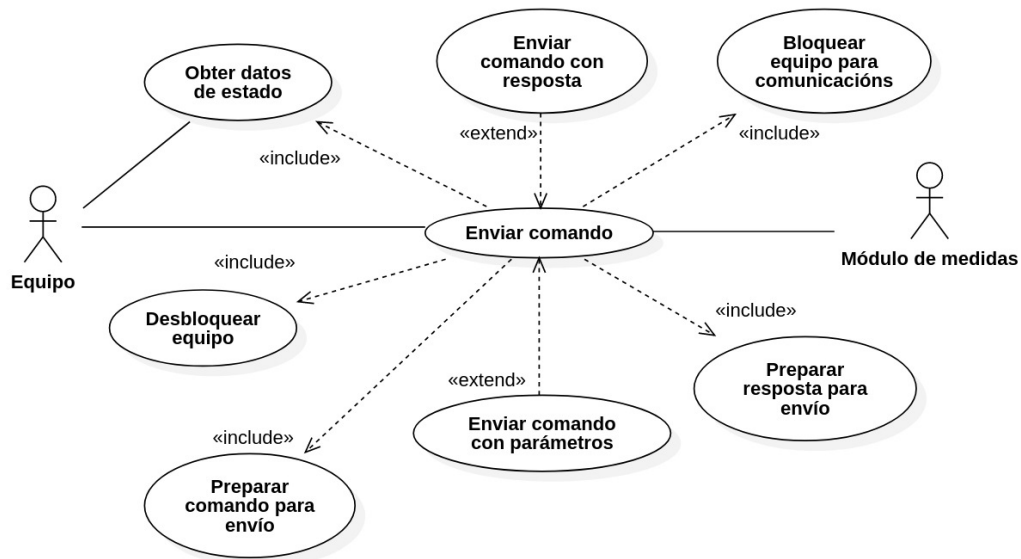


Figura 3.2: Diagrama de casos de uso

A continuación descríbense os casos de uso, empregando para os detalles de cada un deles unha plantilla en forma de táboa cos seguintes campos.



- Número identificador do caso de uso
- Actores involucrados nese caso de uso
- Precondicións
- Poscondicións
- Camiño habitual do caso de uso
- Camiños excepcionais
- Comentarios

Non se inclúen os campos habituais de prioridade e importancia, xa que ao ser poucos casos todos son importantes e de alta prioridade; nin o campo de estado, xa que só hai un desenvolvedor que pode cambialo.

<b>Identificador</b>	<b>CU-1</b>
<b>Nome</b>	Enviar comando
<b>Actores</b>	ACT-1 (LIMS), ACT-2 (equipo)
<b>Precondicións</b>	Ningunha
<b>Postcondicións</b>	Ningunha
<b>Proceso</b>	<ol style="list-style-type: none"> <li>1. O módulo de medidas solicita ao módulo de procesamento de comandos que se envíe un comando a un equipo.</li> <li>2. O módulo de procesamento de comandos comproba se o equipo está online invocando o caso de uso CU-4.</li> <li>3. O equipo está online.</li> <li>4. O módulo de procesamento de comandos comproba se o equipo foi bloqueado por algunha entidade que non sexa a invocadora.</li> <li>5. O equipo foi bloqueado pola entidade invocadora.</li> <li>6. Conéctase a través de TCP co equipo.</li> <li>7. Executa o código que combina o comando cos seus parámetros. Neste caso non hai parámetros, así que o código non fai nada.</li> <li>8. Invoca o caso de uso CU-7 para preparar o comando para o seu envío.</li> <li>9. Envía o comando a través da conexión TCP.</li> <li>10. Desconéctase do socket.</li> </ol>

<b>Excepcións</b>	1. O comando utiliza parámetros. Execútase o caso de uso CU-2, que estende a este.
	1. O comando ten resposta. Execútase o caso de uso CU-3, que estende a este.
	3. O equipo non está online. 4. O módulo de procesamento de comandos devolve un erro ao invocador.
	5. O equipo foi bloqueado por outra entidade. 6. O módulo de procesamento de comandos devolve un erro ao invocador.
	5. O equipo non está bloqueado por ningún elemento. 6. O módulo de procesado de comandos bloquea o equipo consigo mesmo.Desbloquea o equipo.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CU-2</b>
<b>Nome</b>	Enviar comando con parámetros
<b>Actores</b>	ACT-1 (LIMS), ACT-2 (equipo)
<b>Precondicións</b>	Ningunha
<b>Postcondicións</b>	Ningunha
<b>Proceso</b>	<ol style="list-style-type: none"> <li>1. O módulo de medidas solicita ao módulo de procesamento de comandos que se envíe un comando a un equipo, e <b>indícalle os seus parámetros</b>.</li> <li>2. O módulo de procesamento de comandos comproba se o equipo está online invocando o caso de uso CU-4.</li> <li>3. O equipo está online.</li> <li>4. O módulo de procesamento de comandos comproba se o equipo foi bloqueado por algunha entidade que non sexa a invocadora.</li> <li>5. O equipo foi bloqueado pola entidade invocadora.</li> <li>6. Conéctase a través de TCP co equipo.</li> <li>7. <b>Recupéranse todos os parámetros do comando e os seus valores.</b></li> <li>8. <b>Executa o código que combina o comando cos seus parámetros.</b></li> <li>9. Invoca o caso de uso CU-7 para preparar o comando para o seu envío.</li> <li>10. Envía o comando a través da conexión TCP.</li> <li>11. Desconéctase do socket.</li> </ol>
<b>Excepcións</b>	<ol style="list-style-type: none"> <li>8. <b>O código que combina os comandos cos parámetros non é código PHP válido.</b></li> <li>9. <b>O módulo de procesamento de comandos devolve un erro ao invocador.</b></li> </ol>
<b>Comentarios</b>	Este caso de uso estende ao caso de uso CU-1. Non se contemplan as excepcións que teñen que ver co bloqueo e desbloqueo dos equipos nin co seu estado, xa que esas están cubertas en CU-1, e trataríase da mesma situación.

<b>Identificador</b>	<b>CU-3</b>
<b>Nome</b>	Enviar comando con resposta
<b>Actores</b>	ACT-1 (LIMS), ACT-2 (equipo)
<b>Precondicións</b>	Ningunha
<b>Postcondicións</b>	Ningunha
<b>Proceso</b>	<ol style="list-style-type: none"> <li>1. O módulo de medidas solicita ao módulo de procesamento de comandos que se envíe un comando a un equipo.</li> <li>2. O módulo de procesamento de comandos comproba se o equipo está online invocando o caso de uso CU-4.</li> <li>3. O equipo está online.</li> <li>4. O módulo de procesamento de comandos comproba se o equipo foi bloqueado por algunha entidade que non sexa a invocadora.</li> <li>5. O equipo foi bloqueado pola entidade invocadora.</li> <li>6. Conéctase a través de TCP co equipo.</li> <li>7. Executa o código que combina o comando cos seus parámetros. Neste caso non hai parámetros, así que o código non fai nada.</li> <li>8. Invoca o caso de uso CU-7 para preparar o comando para o seu envío.</li> <li>9. Envía o comando e <b>recibe a resposta</b> a través da conexión TCP.</li> <li>10. <b>Invoca o caso de uso CU-8 para preparar a resposta para ser interpretada.</b></li> <li>11. <b>Executa o código que procesa a resposta. É código PHP válido.</b></li> <li>12. Desconéctase do socket.</li> </ol>
<b>Excepcións</b>	<ol style="list-style-type: none"> <li>11. <b>O código que procesa a resposta non é código PHP válido.</b></li> <li>12. <b>O módulo de procesamento de comandos devolve un erro ao invocador.</b></li> </ol>
<b>Comentarios</b>	Este caso de uso estende ao caso de uso CU-1. Non se contemplan as excepcións que teñen que ver co bloqueo e desbloqueo dos equipos nin co seu estado, xa que esas están cubertas en CU-1, e trataríase da mesma situación.

<b>Identificador</b>	<b>CU-4</b>
<b>Nome</b>	Obter datos de estado
<b>Actores</b>	ACT-2 (equipo)
<b>Precondicións</b>	Ningunha
<b>Postcondicións</b>	Ningunha
<b>Proceso</b>	<ol style="list-style-type: none"> <li>1. O módulo de procesamento de comandos carga os datos de rede do equipo.</li> <li>2. Comproba se o equipo está conectado a través de Ethernet.</li> <li>3. O equipo está conectado a través de Ethernet</li> <li>4. Envía un comando de identificación ao equipo para saber se é unha pasarela.</li> <li>5. Non é unha pasarela.</li> <li>6. Indica ao invocador que o equipo efectivamente está online.</li> </ol>
<b>Excepcións</b>	<ol style="list-style-type: none"> <li>3. Non está conectado.</li> <li>4. Indica ao invocador que o equipo non está online.</li> </ol>
	<ol style="list-style-type: none"> <li>5. É unha pasarela.</li> <li>6. Recorre os tipos de conexión posibles e pregunta á pasarela se o equipo está conectado a través dun deles. <ol style="list-style-type: none"> <li>a) Se está conectado, indica ao invocador que o equipo efectivamente está online e a través de que interface.</li> <li>b) Se non está conectado, indica ao invocador que non está online.</li> </ol> </li> </ol>
<b>Comentarios</b>	Este caso de uso é incluído polo caso de uso CU-1.

<b>Identificador</b>	CU-5
<b>Nome</b>	Bloquear equipo
<b>Actores</b>	Ningún
<b>Precondicións</b>	O equipo está libre.
<b>Postcondicións</b>	O equipo está bloqueado pola entidade que solicitou o bloqueo.
<b>Proceso</b>	<ol style="list-style-type: none"> <li>1. O módulo de procesamento de comandos comproba se o equipo está bloqueado.</li> <li>2. O equipo non está bloqueado.</li> <li>3. Inserta na base de datos os datos de bloqueo e indica á entidade que solicitou o bloqueo que este se realizou con éxito.</li> </ol>
<b>Excepcións</b>	<ol style="list-style-type: none"> <li>2. Está bloqueado pola mesma entidade que solicitou o bloqueo.</li> <li>3. Non fai nada e indica á entidade que o bloqueo se realizou con éxito.</li> </ol>
	<ol style="list-style-type: none"> <li>3. Está bloqueado por outra entidade.</li> <li>4. Devolve un erro á entidade que solicitou o bloqueo.</li> </ol>
<b>Comentarios</b>	Este caso de uso é incluído polo caso de uso CU-1.

<b>Identificador</b>	<b>CU-6</b>
<b>Nome</b>	Desbloquear equipo
<b>Actores</b>	Ningún
<b>Precondicións</b>	O equipo está bloqueado pola entidade que solicita o desbloqueo.
<b>Postcondicións</b>	O equipo non está bloqueado pola entidade que solicita o desbloqueo.
<b>Proceso</b>	<ol style="list-style-type: none"> <li>1. O módulo de procesamento de comandos comproba se o equipo está bloqueado pola entidade que solicitou o desbloqueo.</li> <li>2. O equipo está bloqueado pola entidade que solicitou o desbloqueo.</li> <li>3. Elimina de base de datos os datos de bloqueo e indica á entidade que solicitou o desbloqueo que este se realizou con éxito.</li> </ol>
<b>Excepcións</b>	<ol style="list-style-type: none"> <li>2. Non está bloqueado por ese elemento.</li> <li>3. Devolve un erro á entidade que solicitou o bloqueo.</li> </ol>
<b>Comentarios</b>	Este caso de uso é incluído polo caso de uso CU-1.

<b>Identificador</b>	<b>CU-7</b>
<b>Nome</b>	Preparar comando para envío
<b>Actores</b>	Ningún
<b>Precondicións</b>	O equipo debe estar online, directamente conectado á rede local.
<b>Postcondicións</b>	Ningunha
<b>Proceso</b>	<ol style="list-style-type: none"> <li>1. O módulo de preparación de comandos comproba o tipo de interface a través da que está conectado o equipo.</li> <li>2. O equipo está conectado por Ethernet.</li> <li>3. Non se realizan cambios no comando.</li> </ol>
<b>Excepcións</b>	<ol style="list-style-type: none"> <li>2. Está conectado por unha interface distinta de Ethernet.</li> <li>3. Recompílanse todos os datos necesarios para a comunicación con esa interface e empaquétanse para o envío.</li> </ol>
<b>Comentarios</b>	Este caso de uso é incluído polo caso de uso CU-1.



<b>Identificador</b>	<b>CU-8</b>
<b>Nome</b>	Preparar resposta para interpretación
<b>Actores</b>	Ningún
<b>Precondicións</b>	O equipo debe estar online, directamente conectado á rede local.
<b>Postcondicións</b>	Ningunha
<b>Proceso</b>	<ol style="list-style-type: none"> <li>1. O módulo de preparación de comandos comproba o tipo de interface a través da que está conectado o equipo.</li> <li>2. O equipo está conectado por Ethernet.</li> <li>3. Non se realizan cambios na resposta.</li> </ol>
<b>Excepcións</b>	<ol style="list-style-type: none"> <li>2. Está conectado por unha interface distinta de Ethernet.</li> <li>3. Desempaquetase a mensaxe para obter a resposta real do equipo.</li> </ol>
<b>Comentarios</b>	Este caso de uso é incluído polo caso de uso CU-1.

### 3.4. Requisitos non funcionais

<b>Identificador</b>	<b>RNF-1</b>
<b>Nome</b>	Modularidade e extensibilidade
<b>Descrición</b>	Debe ser posible engadir soporte software para novos tipos de interfaces hardware de xeito sinxelo, reducindo ao mínimo a cantidade de código xa existente modificado.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>RNF-2</b>
<b>Nome</b>	Tempo máximo de procesamento
<b>Descrición</b>	O tempo de execución dos comandos pode variar; sen embargo, o tempo transcurrido durante as etapas de procesamento anteriores (nos módulos do servidor e na pasarela) non pode ser superior a 5000 milisegundos.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>RNF-3</b>
<b>Nome</b>	Transparencia no relativo ás interfaces físicas dos equipos
<b>Descrición</b>	A efectos do usuario ou do módulo de medidas, debe ser indiferente a conexión hardware a través da cal o equipo estea comunicado co sistema; todos os detalles relacionados con isto deben xestionarse en módulos subxacentes, proporcionando unha interface común.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>RNF-4</b>
<b>Nome</b>	Concorrenza no software da pasarela
<b>Descrición</b>	O software da pasarela debe soportar concurrencia, e poder procesar varias peticións do servidor simultaneamente.
<b>Comentarios</b>	XesLab xa soporta concurrencia no momento de realización deste proxecto, polo que non hai que implementar nada a maiores nese subsistema.

### 3.5. Requisitos de información

<b>Identificador</b>	<b>RI-1</b>
<b>Nome</b>	Información dos bloqueos
<b>Datos</b>	<ul style="list-style-type: none"> <li>▪ Identificador do equipo bloqueado</li> <li>▪ Tipo de entidade que bloquea o equipo</li> <li>▪ Identificador da entidade que bloquea o equipo</li> <li>▪ Tempo máximo de bloqueo, para evitar reservas indefinidas de equipos.</li> </ul>

<b>Identificador</b>	<b>RI-2</b>
<b>Nome</b>	Información sobre os equipos do laboratorio
<b>Datos</b>	<ul style="list-style-type: none"> <li>▪ Identificador único do equipo</li> <li>▪ Nome do equipo</li> <li>▪ Datos de conexión á rede local <ul style="list-style-type: none"> <li>• Enderezo IP</li> <li>• Porto</li> </ul> </li> <li>▪ Datos de conexión a través das interfaces secundarias, se aplica; por exemplo, o enderezo GPIB</li> <li>▪ Comandos que executa</li> </ul>

<b>Identificador</b>	<b>RI-3</b>
<b>Nome</b>	Información sobre os comandos que son executados polos equipos
<b>Datos</b>	<ul style="list-style-type: none"> <li>▪ Identificador único do comando</li> <li>▪ Nome amigable do comando</li> <li>▪ Código do comando</li> <li>▪ Variables aceptadas polo comando</li> <li>▪ Código PHP que combina o comando cos parámetros provistos, se se aplica</li> <li>▪ Código PHP que adapta ou interpreta a resposta, se se aplica</li> <li>▪ Tipo de resposta que o equipo devolve tras executar o comando</li> </ul>

Cadro 3.3: Requisitos de información



# Capítulo 4

## Arquitectura e deseño preliminar

### 4.1. Deseño de XesLab

Descríbense a continuación unha serie de conceptos relacionados co funcionamento e uso de XesLab e o traballo do laboratorio, debido ao forte impacto que teñen na comprensión das mensaxes que recibe o sistema obxecto deste proxecto. Esta descrición finaliza indicando como o resto de XesLab se comunica co sistema obxecto do proxecto, por que o fai, e que información lle comunica.

No contexto de XesLab, o servizo que Ladetel proporciona para un cliente denomínase traballo. Os dispositivos que o cliente solicita medir denomínanse DUTs (*Device Under Test*); un DUT consiste nun modelo concreto de dispositivo, ou varios modelos que se consideren equivalentes a efectos da medida. Cada DUT pode ter varias mostras, que serían os dispositivos físicos.

A cada traballo está asociada unha contabilidade, na que se especifica que procedementos específicos de ensaio (PEEs) se deben aplicar; debido a que cada PEE ten un custo, que pode depender das condicións das medidas (por exemplo, o tempo durante o que o DUT esté sendo medido). O proceso de definir a contabilidade e seleccionar os PEEs é manual.

Un procedemento específico de ensaio establece que características técnicas do DUT hai que medir. Os equipos miden estas características técnicas, e son controlables mediante comandos. Entón, coñecendo un PEE, é posible inferir que equipos serán necesarios para a medida, e que comandos haberá que executar. Un PEE tamén pode indicar unha parametración para as medidas, pero non é preciso. Os PEEs teñen implementacións, que asocian ao procedemento as normas aplicables. Estas normas poden establecer parámetros.

Un procedemento específico de equipo, ou PEQ, é un conxunto de equipos que miden as mesmas características técnicas. Un exemplo de PEQ son os xeradores; esencialmente é unha forma de agrupar os equipos nun formato xenérico, lexible para o usuario.

Todo isto serve para determinar que ensaios se realizan sobre os DUTs (por

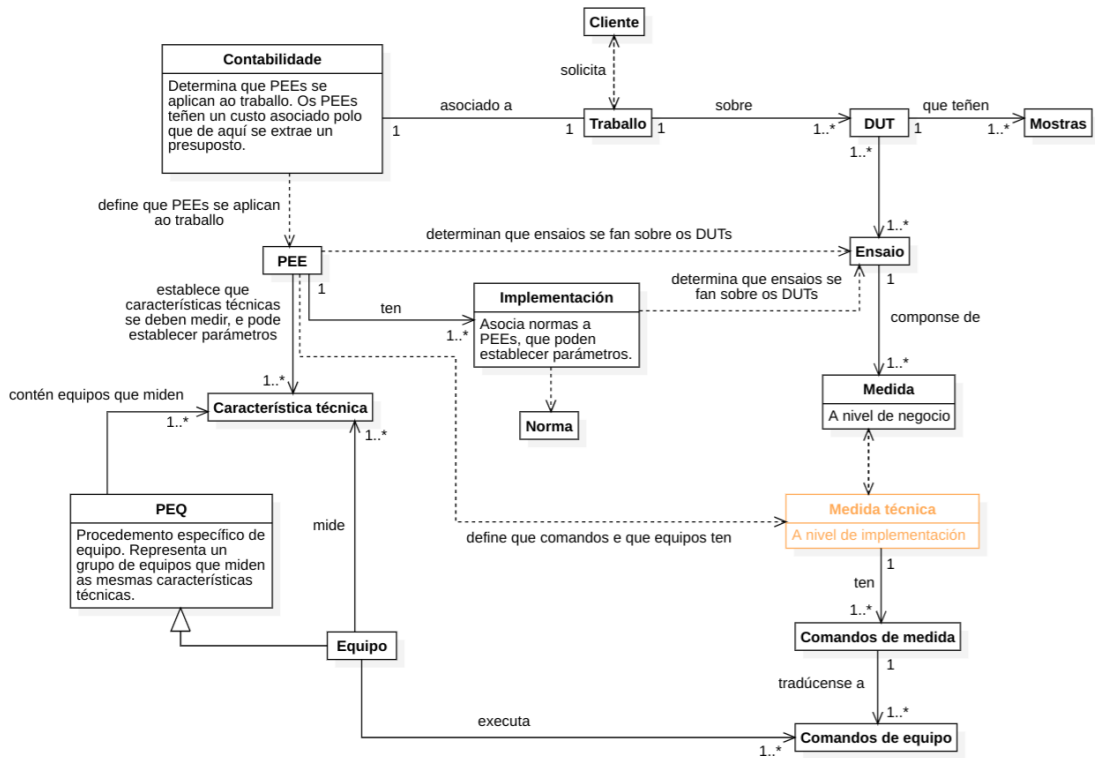


Figura 4.1: Diagrama de negocio de XesLab

ejemplo, cámara climática). Un ensaio pode involucrar a máis dun DUT, mentres que cada DUT pode participar en varios ensaios. Este é o punto onde se complementa a parametrización se anteriormente non se especificaron todos os parámetros necesarios.

Un ensaio compoñese de medidas. Neste punto do entramado é preciso diferenciar unha medida, a nivel de negocio, dunha medida técnica, a nivel de implementación. A medida almacena o conxunto de PEQs necesarios, e o conxunto de comandos que hai que executar. Estes comandos almacénanse nun formato xenérico e lexible polo usuario, que non é interpretable por ningún equipo concreto (por exemplo, “xerador->frecuencia 2 GHz”). Posteriormente, no proceso de executar unha medida, especifícanse os equipos concretos; crúzanse os comandos xenéricos cos comandos de que o equipo concreto pode executar, e obtéñense comandos concretos, interpretables polo equipo (por exemplo, FREQ 2GHz).

Os pares de equipo e comando pasan entón ao sistema de automatización para a súa execución. O deseño e desenvolvemento do sistema de automatización son o obxecto deste proxecto, polo que nos seguintes apartados se contemplará en máis detalle a súa arquitectura.

## 4.2. Estrutura do sistema de automatización

Como xa foi mencionado, é preciso implementar dúas compoñentes software distintas; a parte correspondente ao grupo de módulos de comunicación en XesLab, e o software tradutor na pasarela. O servidor está conectado coas pasarelas e os instrumentos con capacidades de comunicación en rede a través da rede local da oficina de Ladetel.

Na figura 4.2 móstrase o diagrama de despregamento do sistema.

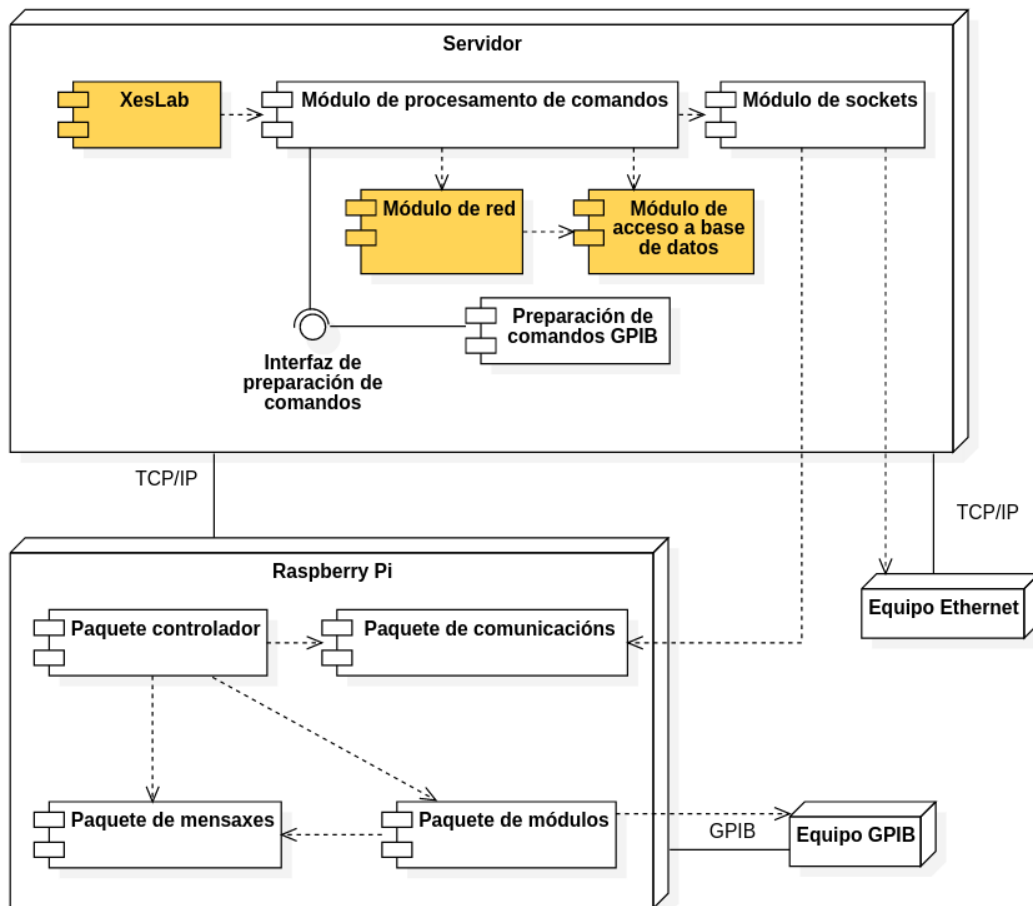


Figura 4.2: Diagrama de despregamento

1

<sup>1</sup>Os módulos en laranxa non foron desenvolvidos como parte do proxecto, senón que as súas funcionalidades foron usadas polos módulos que si se desenvolveron durante o proxecto.

### 4.2.1. Módulos de comunicación en XesLab

Este proxecto enmárcase dentro doutro de maior envergadura, que consiste na programación do LIMS; para resolver a necesidade dunha ferramenta de xestión para o laboratorio, e de forma anterior á realización deste proxecto, decidiuse desenvolver a aplicación XesLab.

XesLab é unha aplicación web escrita en PHP e estruturada en módulos, onde cada módulo agrupa as funcionalidades de xestión relacionadas con entidades concretas (traballos, contactos, incidencias...), ademais de módulos dedicados a utilidades comúns (por exemplo, integración coa base de datos). Por cuestións de simplicidade e consistencia, traballárase neste proxecto co mesmo enfoque.

Corresponde a este proxecto o deseño e construción do conxunto de módulos que xestionan as comunicacións cos equipos. Permitirán enviar mensaxes a equipos e recibir as súas respostas (se as hai), independentemente do tipo de conexión.

- O módulo de procesamento de comandos actúa como punto de entrada único para a comunicación cos equipos; aporta todas as funcións que calquera outra entidade de XesLab poida necesitar para enviar comandos a un equipo. Invoca funcionalidades dos módulos de preparación de comandos e de sockets para lograr este obxectivo.
- O módulo de preparación de comandos traduce un comando ao formato correcto para ser enviado ao seu destino, sexa este equipo ou pasarela.
- O módulo de sockets encárgase de establecer conexións cos destinatarios das mensaxes e realizar o intercambio de datos propiamente dito; encapsula á librería de sockets do sistema.

#### Módulo de procesado de comandos

Exerce como punto de acceso único ás comunicacións cos dispositivos; recibe instrucións do módulo de medidas, nas que se lle comunican o comando e o equipo concretos, e pon en marcha o proceso de envío de datos e recepción da resposta.

Para coñecer o estado online ou offline dos equipos, fai uso das utilidades do sistema de acceso á rede e á base de datos; é preciso manter o estado do sistema en base de datos, xa que as ferramentas empregadas manteñen o estado de memoria para cada usuario. Na base de datos relaciónase cada MAC dun equipo cunha IP, e o módulo de rede permite efectuar un escaneo ARP sobre a rede local para coñecer o estado de cada equipo ou pasarela.

Existe o problema de que este escaneo notificaría como online a un equipo asociado á IP dunha pasarela, se a pasarela está online pero o equipo non está realmente conectado a esta. Polo tanto, o escaneo realízase en varios pasos: un escaneo ARP para comprobar o estado da IP, e unha comunicación de proba



co instrumento, que indicará se é unha conexión directa ou a través dunha pasarela e se realmente está activo e en condicións de comunicarse. Esta comunicación de proba tamén indicará o tipo de conexión que se mantén co equipo (GPIB, Bluetooth, ou calquera outra que estea implementada) se é a través dunha pasarela. Para a comunicación de proba, emprégase o comando de identificación, que está definido no estándar IEEE-488.2 e é común a case todos os instrumentos do laboratorio.

Este módulo tamén ofrece funcionalidades para bloquear as comunicacións cun equipo, de xeito que só o elemento responsable do bloqueo poida comunicarse con el. Os elementos que poden bloquear comunicacións son medidas, calibracións, probas, os propios equipos, etc. Os bloqueos xestiónanse a través dunha táboa na base de datos, que almacena os equipos bloqueados e o seu elemento responsable.

Depende do módulo de preparación de comandos e do módulo de sockets para o envío de comandos.

Debido a que no momento de realización do traballo o módulo de medidas, que se encargaría de recuperar os comandos e equipos asociados á medida que se quere executar, non está implementado, créase no módulo de equipos unha función que permite enviar un comando a un equipo, invocando ao módulo de procesamento de comandos. Porén, o módulo de equipos corresponde máis ao LIMS que ao LAS, e polo tanto a súa implementación non forma parte do alcance do proxecto.

### **Módulo de preparación de comandos**

Encárgase de converter unha mensaxe para un equipo ao formato adecuado para que se poida transmitir. Isto é trivial no caso dos equipos conectados por IP; os comandos están almacenados na base de datos en texto plano, de forma interpretable polos equipos. Porén, no caso dos equipos conectados a través dunha pasarela, é preciso indicar tamén datos como o tipo de conexión e o enderezo do equipo para ese tipo de conexión, para que a pasarela poida realizar o enrutado.

Por isto, este módulo é o responsable de transformar a mensaxe ao formato correcto para o seu procesamento polo dispositivo de destino ou a pasarela, baseándose no tipo de conexión do dispositivo e de forma transparente para o módulo de procesamento de comandos.

No caso de que sexa unha mensaxe dirixida a unha pasarela, utilízase JSON para a serialización da mensaxe e Base64 para a súa codificación, para lograr unha transmisión segura.

### **Módulo de sockets**

Establece conexións cos destinatarios das mensaxes, e envía e recibe datos. É completamente independente do tipo de destinatario. Emprégase o protocolo TCP, e establécese unha conexión por cada mensaxe enviada; debido á estrutura habitual das medidas, non é frecuente enviar varias mensaxes ao mesmo equipo

de forma secuencial, polo que non existe unha diferenza notable con executar todos os comandos entre conexión e desconexión.

### 4.2.2. Software da pasarela

Provee unha interface Ethernet para os equipos que non a teñen, permitindo que estes manteñan unha comunicación co servidor. Recibe as mensaxes do servidor e envíaas ao equipo correcto, devolvendo a resposta deste (se aplica), unha confirmación de que todo se desenvolveu correctamente, ou calquera mensaxe de erro que puidera xerarse durante o proceso.

A funcionalidade distribúese nunha serie de paquetes. Cada un deles invoca a métodos dos demais para facer uso das funcións que ofrece.

- O paquete controlador xestiona as peticións do servidor e cede a cada unha os recursos que necesita para a súa execución.
- O paquete de comunicacións controla o envío e recepción de mensaxes co servidor.
- O paquete de módulos distribúe as mensaxes ao equipo correcto.
- O paquete de mensaxes realiza as traducións necesarias.

#### Paquete controlador

Xestiona as peticións entrantes por parte do servidor de XesLab e adquire os recursos necesarios para procesar cada unha delas; inicia fíos de execución e instancia os obxectos necesarios.

Cada fío executa un ciclo de recepción dunha mensaxe, procesamento, e envío da resposta; toda mensaxe chegada da aplicación é respondida, sexa coa resposta do equipo, cunha confirmación de recepción ou cunha notificación de erro. A única excepción que existe a isto sucede no caso do escaneo ARP, que causa o envío de mensaxes baleiras ao proceso. Estas mensaxes ignóranse.

#### Paquete de comunicacións

Comprende tanto ao xestor de conexións como aos obxectos que representan cada conexión individual co servidor. Utiliza a librería nativa de sockets de C, e xestiona todo o relativo ao establecemento de conexións coa aplicación de XesLab e o envío e recepción de datos; é o análogo ao módulo de sockets do servidor. Existe unha conexión por mensaxe recibida.

### Paquete de módulos

Cada tipo de interface hardware é xestionado por un módulo, e o xestor de módulos é o responsable de dirixir as mensaxes ao módulo correcto. Ademais disto, ocúpase de xestionar os casos especiais de mensaxes; á pasarela poden chegar mensaxes baleiras ou comprobacións de conexión.

Cada módulo debe proporcionar procedementos para comprobar a existencia dun equipo e para enviar unha mensaxe a un equipo. A efectos da realización deste proxecto, só se implementará o módulo de comunicación a través da interface GPIB, que emprega a librería `linux-gpib`.

### Paquete de mensaxes

É probable que unha pasarela teña conectados distintos equipos, xa que non é economicamente eficiente ter unha pasarela por dispositivo. Para enrutar os paquetes cara o equipo correcto, a pasarela precisa coñecer máis información que o comando, como por exemplo os enderezos GPIB. Porén, os equipos son incapaces de interpretar calquera cousa que non sexa un comando válido en texto plano. Deste problema xorde a necesidade de empregar distintos tipos de paquetes, e, polo tanto, dunha tradución.

O paquete de mensaxes encárgase de realizar esta tradución. Consta dos tipos de mensaxes específicas para cada módulo, e un xestor de mensaxes que transforma decodifica e deserializa o texto e instancia o obxecto correspondente á mensaxe adecuada (e viceversa).

Ademais de existir un tipo por cada interface hardware, existen mensaxes de control, que se empregan para devolver erros a XesLab. O xestor de mensaxes proporciona un método para crear mensaxes de control de forma rápida e sinxela.

Emprega as librerías externas `libb64` para a codificación e decodificación en Base64, e `nlohmann-json` para a serialización e deserialización.

## 4.3. Tecnoloxías a empregar

### 4.3.1. Ferramentas de desenvolvemento

Contémplanse neste apartado as tecnoloxías e ferramentas empregadas para realizar o proceso de desenvolvemento, tanto nas etapas de análise e deseño preliminares a cada incremento como na implementación do software.

- **PHP:** linguaxe de programación na que se escriben os módulos que se executarán no servidor.
- **Eclipse:** contorna de desenvolvemento empregado para a implementación do proxecto.

- **C/C++:** linguaxe de programación para o software da pasarela.
- **CLion:** contorna de desenvolvemento para o software da pasarela.
- **Apache:** contorna de aplicacións no que se executa XesLab.

### 4.3.2. Ferramentas de documentación

Lístanse a continuación os programas utilizados na documentación do proxecto.

- **Overleaf V2:** aplicación web de edición de textos en LaTeX na que se editou a memoria do proxecto.
- **Microsoft Word:** software de procesamento de textos no que se editou o anteprojecto.
- **Microsoft Excel:** software de follas de cálculo empregado na xestión de custos.
- **Microsoft Powerpoint:** programa de edición de presentacións empregado para construír a presentación da defensa.
- **Microsoft Project:** software para xestión de proxectos no que se realizaron diversas tarefas de planificación e xestión de custos.
- **StarUML:** aplicación de edición de diagramas.

### 4.3.3. Librarías externas

Nome da librería	Licenza	Orixe
<i>libb64</i>	Creative Commons	<a href="https://github.com/cburstedde/libsc/">https://github.com/cburstedde/libsc/</a>
<i>nlohmann json</i>	MIT	<a href="https://github.com/nlohmann/json">https://github.com/nlohmann/json</a>
<i>linux-gpib</i>	GPL	<a href="https://linux-gpib.sourceforge.io/">https://linux-gpib.sourceforge.io/</a>
<i>Boost.Test</i>	Boost (similar a MIT)	<a href="https://boost.org/">https://boost.org/</a>
<i>PHPUnit</i>	Creative Commons	<a href="https://phpunit.de/">https://phpunit.de/</a>

Cadro 4.1: Cadro de licenzas do software de terceiros

No cadro 4.1 podemos ver que a licenza máis restritiva é GPL. Esta permítenos utilizar o código de forma interna na empresa pero obrigaría a utilizar a liberar o código en caso de querer distribuíla.

#### 4.3.4. Equipamento hardware

Este proxecto conta cun compoñente de hardware moi importante, debido ás necesidades de interoperabilidade cos equipos do laboratorio.

- **Equipos de desenvolvemento** (portátil persoal da alumna e sobremesa da oficina).
- **Pasarela de Televés:** dispositivo portátil cunha variedade de interfaces hardware de comunicación e con sistema operativo Linux. Debido ás dificultades para obter unha pasarela para o desenvolvemento, empregouse unha **Raspberry Pi** no seu lugar, xa que é un dispositivo de características similares.
- **Adaptador USB a GPIB:** empregouse o dispositivo de control GPIB-USB-HS da marca National Instruments.
- **Instrumentos de medición:** para depuración e probas, empregáronse os seguintes equipos pola súa facilidade de transporte e variedade de comandos, ademais de que ambos contan con interfaces GPIB e Ethernet.
  - Keysight MXG Vector Signal Generator N5182B
  - Ronde & Schwartz FSV7 Signal and Spectrum Analyzer
- **Servidor de Ladetel**

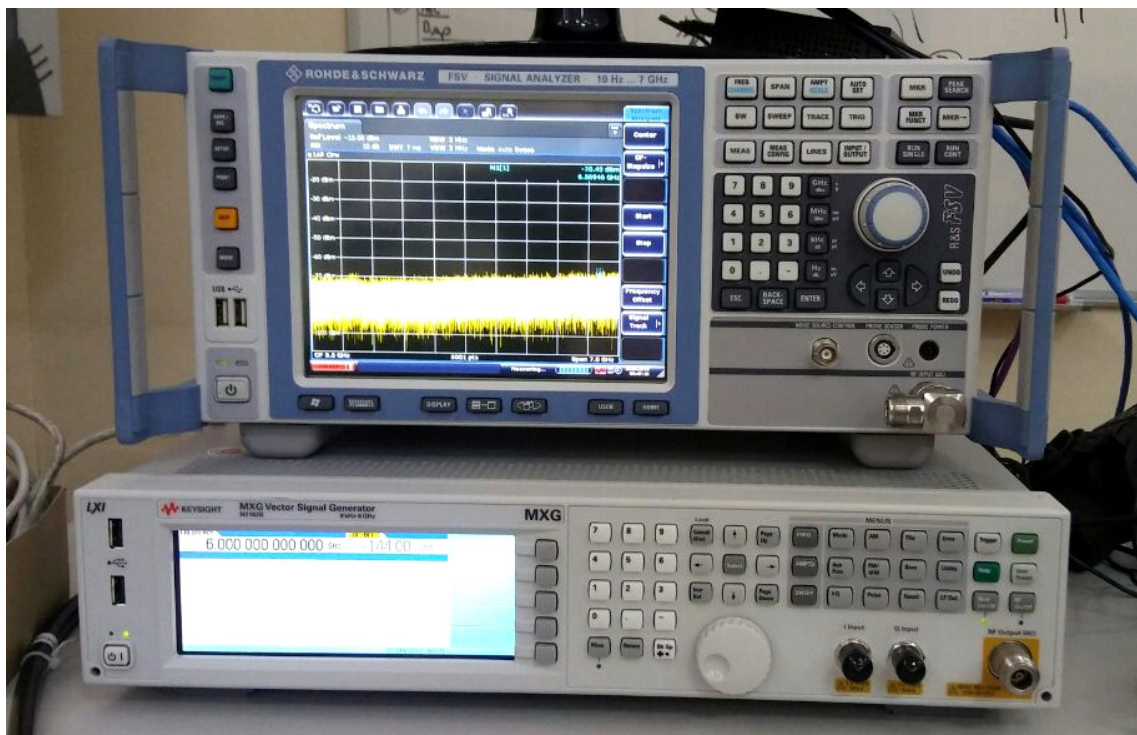


Figura 4.3: Equipos Keysight MXG e R&S FSV7 empregados no desenvolvimento

# Capítulo 5

## Deseño e implementación en XesLab

### 5.1. Detalles de deseño e implementación da aplicación XesLab

Como xa foi mencionado, XesLab está escrito en PHP e estruturado en módulos, considerándose un módulo unha estrutura de clases e ficheiros que xestionan un concepto ou conxunto de datos concreto. Existen módulos que tratan conceptos de negocio, como poden ser as incidencias, e módulos puramente técnicos, como o módulo de *logging*. Neste proxecto só se modifica o módulo de equipos, que contén a lóxica relacionada cos equipos, as súas características técnicas e os seus comandos.

En xeral, ao longo de toda a aplicación úsase unha combinación de programación procedimental e programación orientada a obxectos. As partes procedimentais impleméntanse como clases estáticas, que almacenan constantes e métodos estáticos; PHP permite programar por procedementos sen facer esta adaptación, pero no momento do deseño preliminar de XesLab decidiuse facer isto pola facilidade de organización que aportan as clases. Polo tanto, nos módulos de comunicación mantívose este tipo de estrutura por consistencia co resto do software. Tamén se intentou manter a nomenclatura, cos nomes das variables en maiúsculas, atributos en *camel case*, etc. e utilizáronse nomes en español, a pesar de que este documento está sendo escrito en galego.

Na figura 5.1 móstrase o diagrama de clases. Co obxectivo de mellorar a lexibilidade do diagrama, fixéronse as seguintes modificacións con respecto do código.

- Elimináronse os símbolos \$ que preceden ás variables en PHP.
- Incluíronse caracteres que no código non están permitidos (por exemplo, o acento agudo e a letra Ñ).

- Escribíronse todos os nomes en minúsculas.
- Non se indicou que os atributos e operacións son abstractos, xa que a notación involucra subliñar o nome e isto xera unha sobrecarga visual se se aplica para todos os membros.

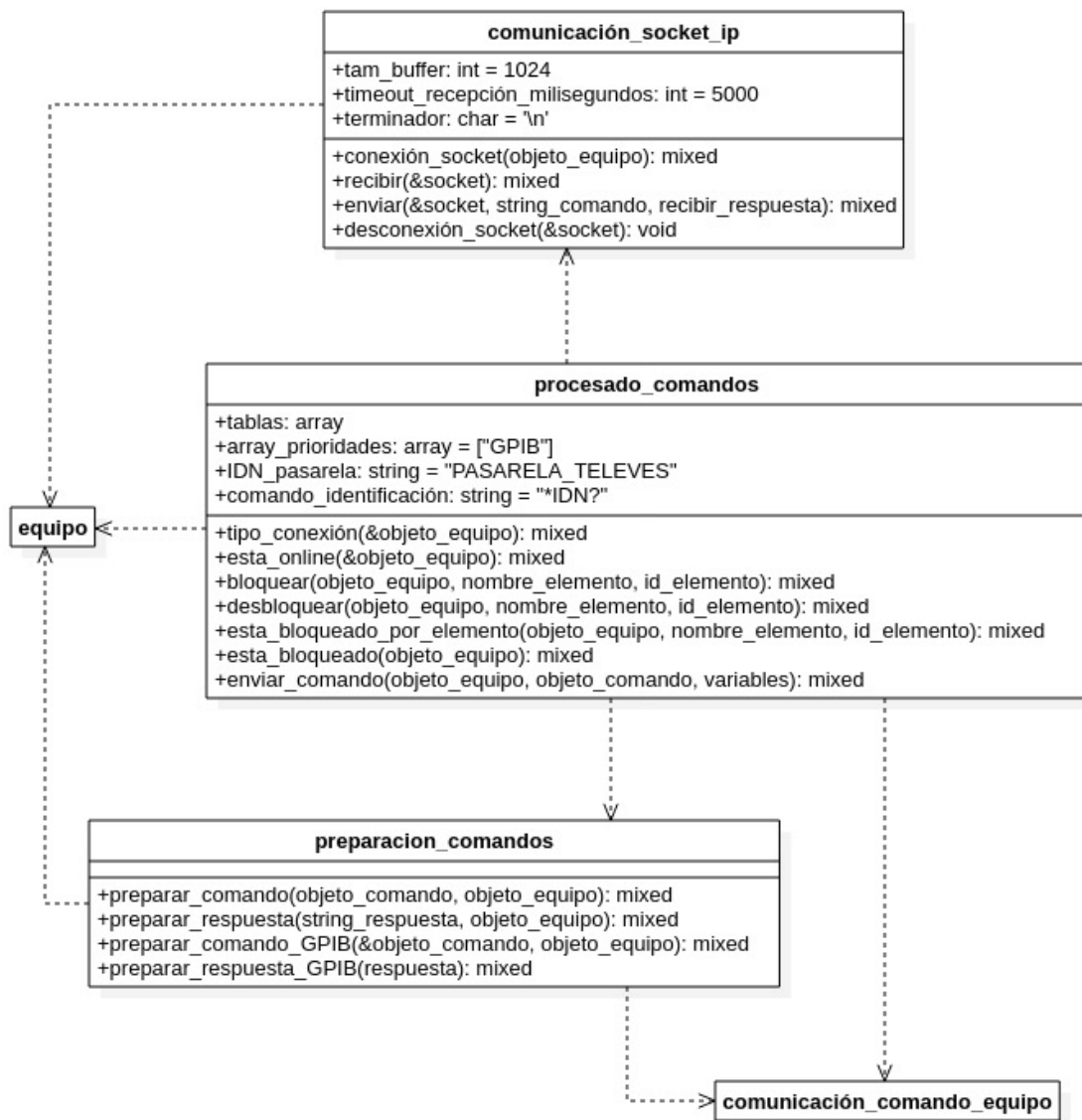


Figura 5.1: Diagrama de clases de XesLab



## 5.2. Módulo de procesamiento de comandos

É o punto único de entrada ao módulo de comunicacións cos equipos. Proporciona funcionalidades para comprobar o estado dos equipos, bloquealos para comunicación e enviarlles comandos.

### 5.2.1. Estado dos equipos

Para valorar o estado online ou offline dos equipos cóntase con dous métodos: TIPO\_CONEXION e ESTA\_ONLINE. A segunda fai uso da primeira.

#### Obter o tipo de conexión

Entre as utilidades de rede do sistema, existe un método que realiza un escaneo ARP sobre a rede local, e almacena en base de datos a información sobre que equipos están conectados á rede. Por outra parte, un método da clase EQUIPO permite recuperar estes valores e almacenalos nunha propiedade de tipo array. Porén, isto ten un problema; se un equipo está asociado á IP da pasarela en base de datos, pero non está efectivamente conectado á pasarela (ou está conectado fisicamente, pero non funciona ou non está en condicións de comunicarse), o sistema reconece o equipo como online anque isto non sexa real. Para comprobar que está verdadeiramente online, é preciso intentar comunicarse con el, e que responda activamente.

Esta é a razón de ser do método TIPO\_CONEXION. Só contempla as conexións secundarias, xa que non ten sentido invocalo para a interface Ethernet; aplicalo sobre un equipo soamente conectado por Ethernet daría a entender que o equipo non está online, porque non está conectado a través de ningunha conexión secundaria.

Debido a que un equipo pode ter varias interfaces, existe unha prioridade; os nomes das interfaces almacénanse por orde de prioridade na variable \$ARRAY\_PRIORIDADES. TIPO\_CONEXION recorre este *array*, recuperando o enderezo do equipo de base de datos, e, se este existe, intentando comunicarse co equipo a través desa conexión. Selecciona a primeira conexión (de maior prioridade) para a que obtén unha resposta.

Para saber se o equipo está operativo ou non, envíase un comando especial, almacenado en \$COMANDO\_ALIVE e con contido \$XESLAB\_ALIVE. Ao recibir esta mensaxe, a pasarela realiza a comunicación de proba co equipo presente no enderezo indicado, e devolve **true** se logra unha resposta e **false** en caso contrario.

TIPO\_CONEXION devolve null se o equipo non está conectado a través de ningunha interfaz distinta de Ethernet, e o nome da interfaz con máis prioridade a través da que está conectado en caso contrario.

### Obter o estado online ou offline

Fai uso das utilidades de rede para saber se \$OBJETO\_EQUIPO está conectado por IP, e envía unha mensaxe de proba para comprobar se se trata de un instrumento ou unha pasarela. Se é unha pasarela, invoca a TIPO\_CONEXION para facer as comprobacións anteriormente mencionadas.

É posible saber se un equipo é pasarela ou non enviándolle un comando \*IDN?. Trátase dun comando definido no estándar IEE-488.2, común a case todos os equipos do laboratorio; provoca que o equipo responda cun *string* de identificación de si mesmo, con información como o nome ou o modelo. Porén, as pasarelas están programadas para responder aos comandos de identificación en texto plano co *string* PASARELA\_TELEVES, polo que é sinxelo identificalas.

Devolve **true** se o equipo &\$OBJETO\_EQUIPO está conectado por Ethernet e as utilidades de rede indican que está online, ou se está conectado a través dunha pasarela e TIPO\_CONEXION non devolve null.

### 5.2.2. Bloqueos

A información relativa aos bloqueos almacénase en base de datos, na táboa BLOQUEOS, cuxos campos están almacenados no atributo TABLAS deste módulo. Ademais de certos campos de implementación, como o identificador e a data de inserción, cada bloqueo identifícase por tres elementos: o identificador do equipo bloqueado, o tipo de elemento que o desexa bloquear, e o identificador do elemento bloqueador.

A idea principal é que bloquear un equipo serve para reservar o dereito ás comunicacións con el, de xeito que só o elemento que o bloqueou pode acceder ao punto de acceso único de envío de mensaxes. Isto é importante porque así garántizase que unha medida que involucra a distintos comandos pode executarse por completo antes de que outros elementos se comuniquen cos equipos implicados; no caso contrario, poderíase interferir coa medida e os resultados non serían válidos.

Consecuentemente, existen as seguintes restricións no relativo a quen pode bloquear que equipos, dependendo do estado destes.

		Estado do equipo		
		Libre	Bloqueado por esa entidade	Bloqueado por outra entidade
Acción	Bloquear	Éxito	Éxito (non provoca cambios)	Erro
	Desbloquear	Erro	Éxito	Erro
	Enviar comando	Bloquea o equipo consigo mesmo	Éxito	Erro

Cadro 5.1: Restricións relativas aos bloqueos

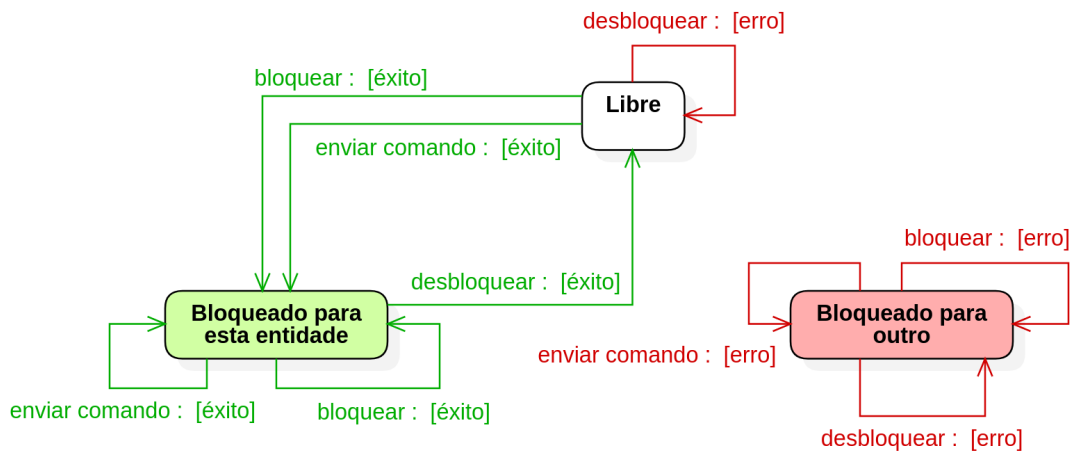


Figura 5.2: Diagrama de estados de bloqueo

Esencialmente, un elemento que bloquea un equipo pode realizar calquera acción sobre el, e posúe o dereito sobre as comunicacións de forma exclusiva, ata que o libera.

Existe un caso especial, que é aquel no que o envío dun comando bloquea o equipo consigo mesmo se non foi bloqueado antes por ningún elemento. Isto é para asegurar a integridade das medidas incluso no caso en que un comando se envíe sen formar parte dunha medida; por exemplo, cando se fan probas de comunicación dende o panel de información do equipo. Obriga ao envío do comando a seguir as semánticas detalladas no cadro 5.1, que son as que garanten a integridade.

Existen catro métodos para traballar con bloqueos:

- BLOQUEAR(\$OBJETO\_EQUIPO, \$NOMBRE\_ELEMENTO, \$ID\_ELEMENTO)
- DESBLOQUEAR(\$OBJETO\_EQUIPO, \$NOMBRE\_ELEMENTO, \$ID\_ELEMENTO)

- `ESTA_BLOQUEADO_POR_ELEMENTO($OBJETO_EQUIPO, $NOMBRE_ELEMENTO, $ID_ELEMENTO)`
- `ESTA_BLOQUEADO($OBJETO_EQUIPO)`

A variable `$TIMEOUT_GENERICO_BLOQUEO` almacena un tempo límite no que un equipo pode estar bloqueado, para evitar erros en casos nos que unha entidade reserve un equipo e non o libere nunca. Almacénase na base de datos cando se bloquea un equipo; porén, decidiu non implementarse neste punto do proxecto.

### 5.2.3. Enviar comando

Esta funcionalidade consta dun único método. Recibe o obxecto que identifica o equipo ao que se lle envía o comando, e o obxecto que identifica ao comando en si. Tamén recibe unha serie de variables que almacenan a parametrización para ese comando, se se aplica.

Ten a seguinte secuencia de execución:

- Comproba se o equipo está online. Se non, lanza unha excepción.
- Comproba se o equipo está bloqueado pola entidade que desexa comunicarse, e se non, bloquea ao equipo consigo mesmo. Se non consegue bloquearse, lanza unha excepción.
- Invoca a `COMUNICACION_SOCKET_IP::CONEXION_SOCKET` para que cree unha conexión co equipo.
- Se o argumento `$VARIABLES` é un *array* de pares (nome da variable, valor), instancia todas estas variables na contorna actual, usando as capacidades de reflexión de PHP.

```

if (is_array($VARIABLES)) {
    foreach ($VARIABLES as $VARIABLE => $VALOR) {
        $$VARIABLE = $VALOR;
    }
}

```

- Executa, utilizando a función de PHP `eval`, o código almacenado en `$OBJETO_COMANDO->Comando_PHP`, que une o comando e a súa parametrización nun só *string* e almacena este en `$COMANDO_ENVIAR`. Se esta propiedade non contén código PHP válido, lánzase unha excepción.
- Modifica `$OBJETO_COMANDO` para que almacene o comando coa parametrización, e invoca a `PREPARACION_COMANDOS::PREPARAR_COMANDO` para transformar o comando ao formato correcto para ser interpretado polo seu destino (instrumento ou pasarela).

- Invoca a `COMUNICACION_SOCKET_IP::ENVIAR`, que envía o comando ao equipo.
- Se o comando ten resposta, invoca a `PREPARACION_COMANDOS::PREPARAR_RESPUESTA` para que obteña a resposta en texto plano, e execútase o código PHP almacenado en `$OBJETO_COMANDO->Respuesta_Codigo`, que procesa ese texto plano segundo requira o comando concreto. Se non é código PHP válido, lanza unha excepción.
- Desconéctase do equipo con `COMUNICACION_SOCKET_IP::DESCONEXION_SOCKET`.
- Desbloquea o equipo, se se aplica.

### 5.3. Módulo de preparación de comandos

Este elemento é esencial para poder traballar indistintamente cun dispositivo ou cunha pasarela. As funcións `PREPARAR_COMANDO` e `PREPARAR_RESPUESTA` reciben os obxectos correspondentes ao equipo e ao comando, e, examinando o tipo de conexión do primeiro, transforman o comando no formato adecuado para ser interpretable polo destino.

```
$TIPO = $OBJETO_EQUIPO->RED["TIPO_CONEXION"];

if ($TIPO === "ETHERNET") {
    return $OBJETO_COMANDO->Comando;
} else {
    $METODO_PREPARACION = "PREPARAR_COMANDO_" . $TIPO;
    return self::$METODO_PREPARACION($OBJETO_COMANDO,
        $OBJETO_EQUIPO);
}
```

O uso de reflexión fai que non sexa necesario modificar estas funcións para engadir interfaces novas.

Se o tipo de conexión é Ethernet, o comando non se modifica e devólvese en texto plano. Se non, invócase á función `PREPARAR_COMANDO_<tipo conexion>`, que encapsula a mensaxe nun *array* JSON que contén os campos necesarios para que a pasarela interprete e enrute a mensaxe correctamente.

```
$TIPO_COMANDO = 'GPIB';
$DIR_GPIB = $OBJETO_EQUIPO->GPIB_Direccion;

$RECIBIR_RESPUESTA = $OBJETO_COMANDO->Respuesta_Tipo ===
    "SIN_RESPUESTA" ? false : true;
$OBJETO_COMANDO->Respuesta_Tipo = "STRING";

$ARRAY_JSON = array(
    'tipo' => $TIPO_COMANDO,
    'direccion_equipo' => $DIR_GPIB,
    'contenido' => $OBJETO_COMANDO->Comando,
    'tipo_respuesta' => $RECIBIR_RESPUESTA
```

```

);

$SERIALIZADO = json_encode($ARRAY_JSON);
$CODIFICADO = base64_encode($SERIALIZADO);

return $CODIFICADO;

```

A propiedade relacionada co tipo de resposta do comando modifícase para que sempre se espere unha resposta se a mensaxe vai atravesar unha pasarela. Isto é para obrigar ao servidor a recibir posibles mensaxes de erro procedentes da pasarela e asociadas a ese mensaxe (por exemplo, un erro de *parse*, un erro de *time-out* de lectura do dispositivo...), que se obviarían se non se esperara por unha mensaxe.

## 5.4. Módulo de sockets

Emprégase o protocolo TCP/IP para comunicarse co dispositivo remoto. O módulo de sockets é entón un *wrapper* da librería de sockets de PHP; contén funcións para establecer unha conexión, enviar unha mensaxe, recibir unha mensaxe e cerrar a conexión.

A función de recepción é significativamente máis complexa que o resto; certos equipos envían as respostas aos comandos en bloques, e é preciso realizar varias lecturas para asegurar que se leron todos os datos. Isto impón a necesidade de empregar un socket non bloqueante e implementar a saída por *time-out* manualmente, que sucede se transcorren máis de \$TIMEOUT\_RECEPCION\_MILISEGUNDOS milisegundos sen recibir ningún byte.

## 5.5. Diagramas de secuencia

Nas figuras 5.3, 5.4, e 5.5 móstranse os diagramas de secuencia resultantes da execución de, respectivamente, o envío dun comando sen resposta a un equipo Ethernet, dun comando con resposta a un equipo Ethernet e dun comando sen resposta a un equipo de GPIB.

As invocacións de métodos sinálanse en cor negra, e as mensaxes enviadas pola rede ou por GPIB, en laranxa.

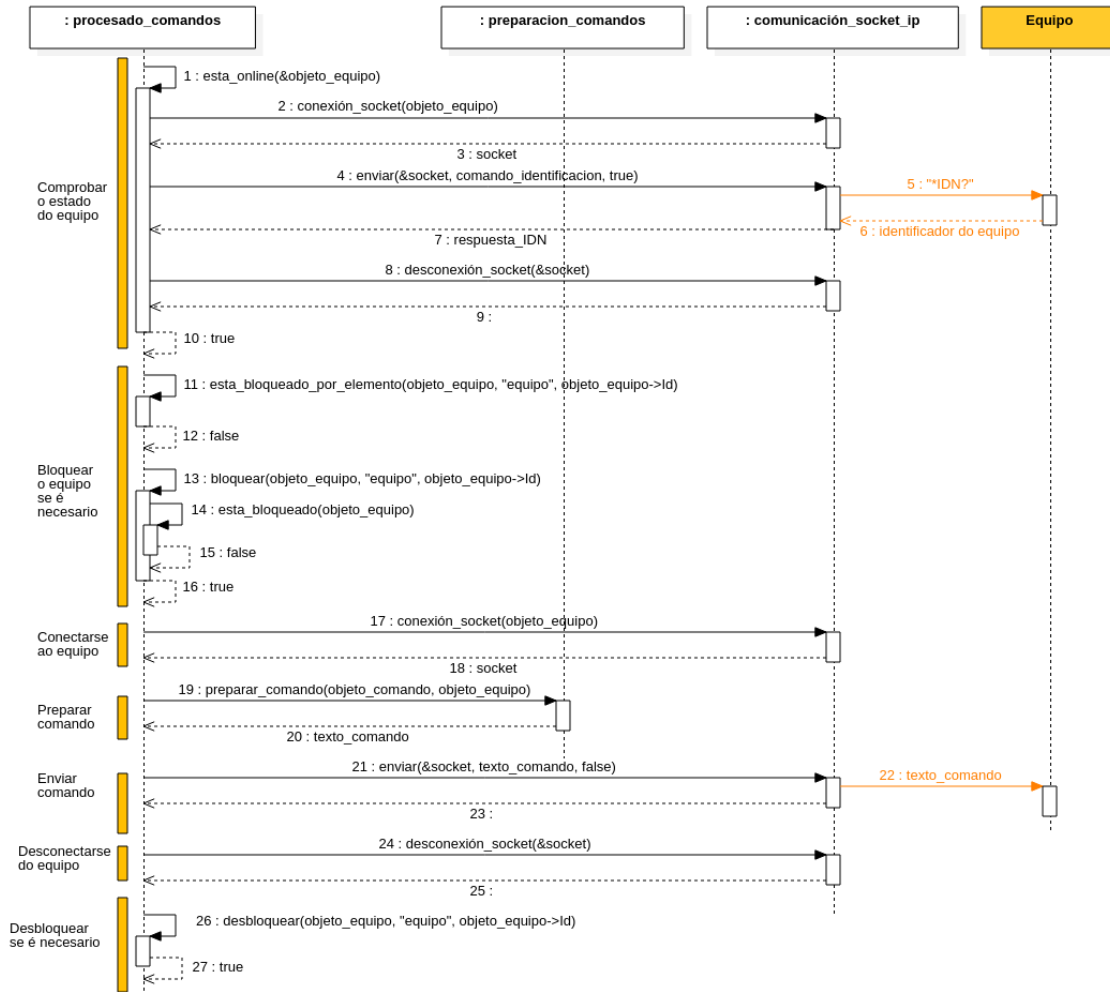


Figura 5.3: Enviar comando sin respuesta a un equipo conectado por Ethernet

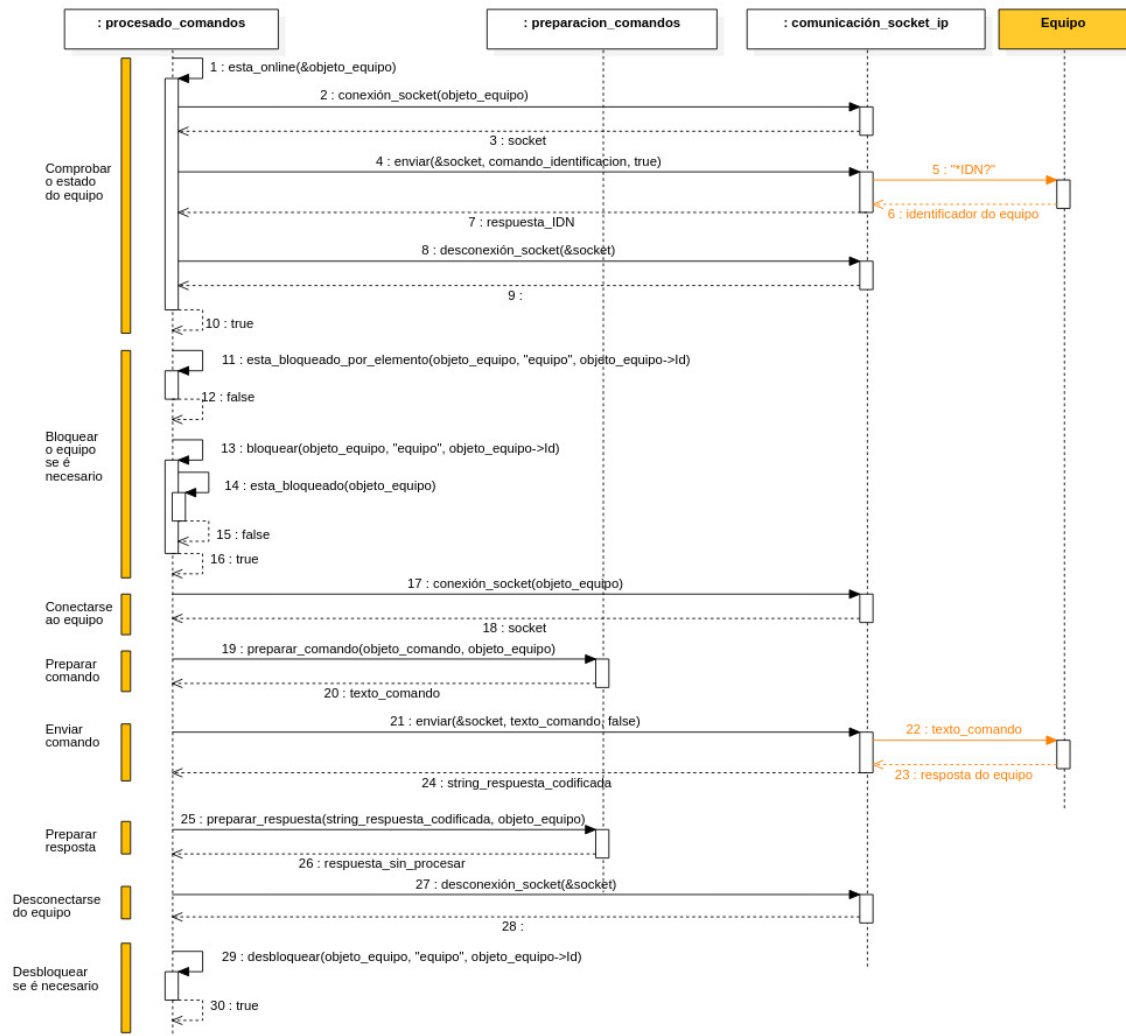


Figura 5.4: Enviar comando con respuesta a un equipo conectado por Ethernet



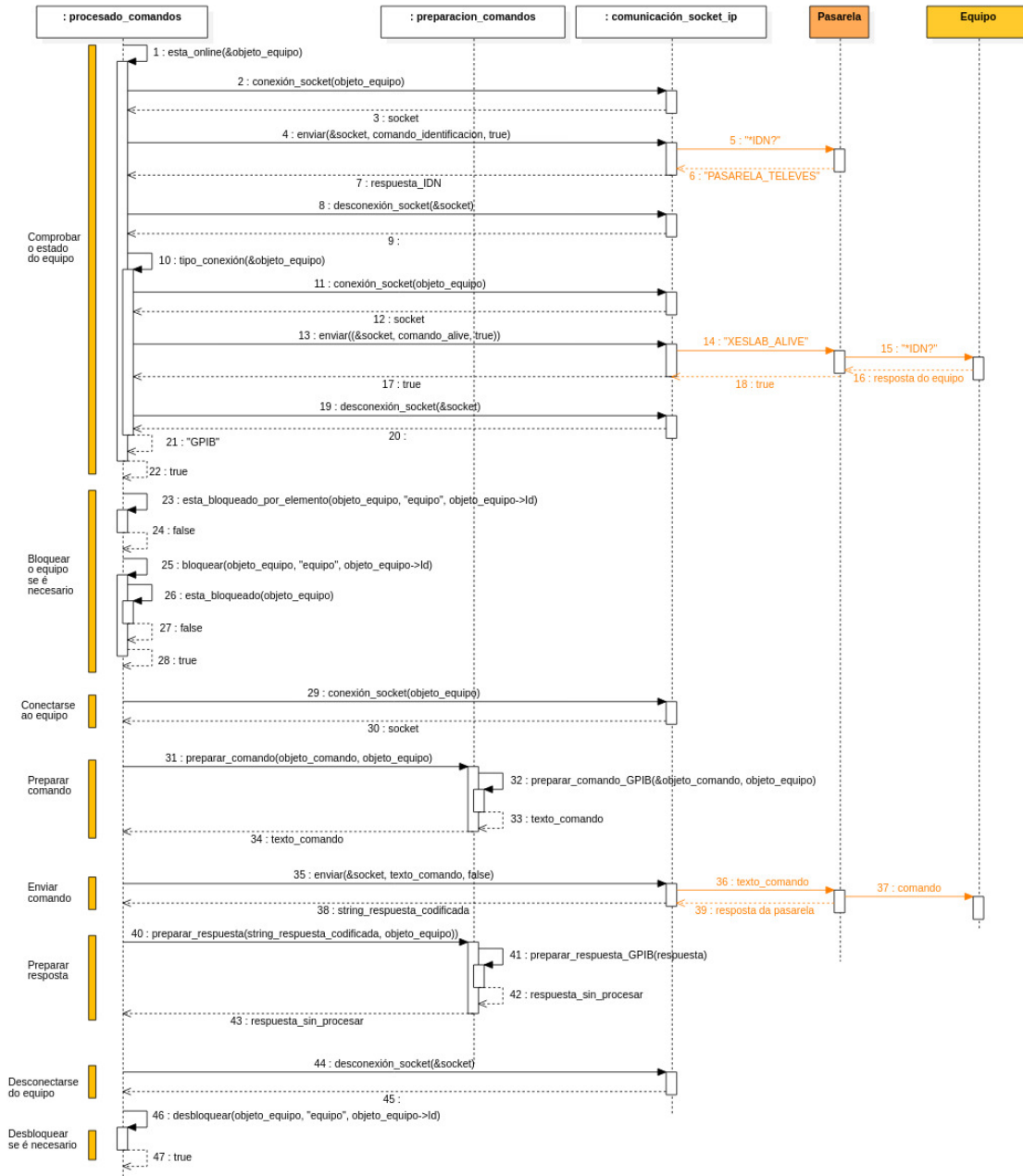


Figura 5.5: Enviar comando sen resposta a un equipo conectado por GPIB

## 5.6. Interfaces gráficas

Non foron contempladas no alcance deste TFG, pero resulta interesante ver a interface a través da cal os usuarios invocan as utilidades de comunicación.

Pódese iniciar a comunicación cun equipo dende dúas ubicacións distintas. Unha delas é o panel de comandos da páxina dun equipo: mostra todos os comandos que pode executar ese equipo, e permite enviarllos individualmente. Esta opción está destinada a usarse para realizar probas do sistema ou do equipo, ou ben calibracións e configuracións.

A outra opción será a máis habitual: poderase executar unha medida dende o panel da medida técnica na pantalla dun procedemento, e o sistema determinará que comandos se deben enviar aos equipos. Este panel carga os parámetros establecidos polos procedementos, normas e outras entidades involucradas, e permite editalos para personalizar a medida. Neste panel tamén se poden ver os resultados da medida recollidos en tempo real.

Pódense ver estas interfaces nas figuras 5.6, 5.7 e 5.8.

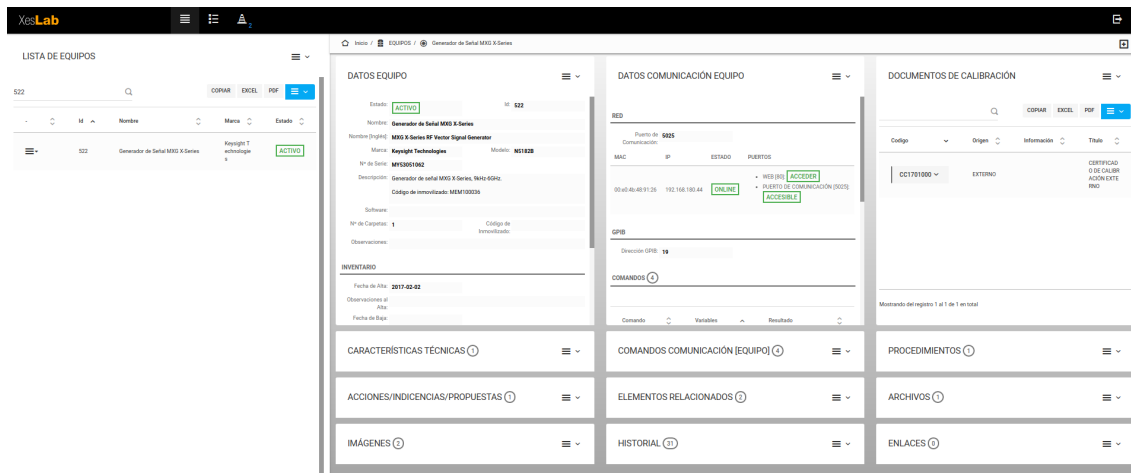


Figura 5.6: Pantalla do equipo 522 en XesLab

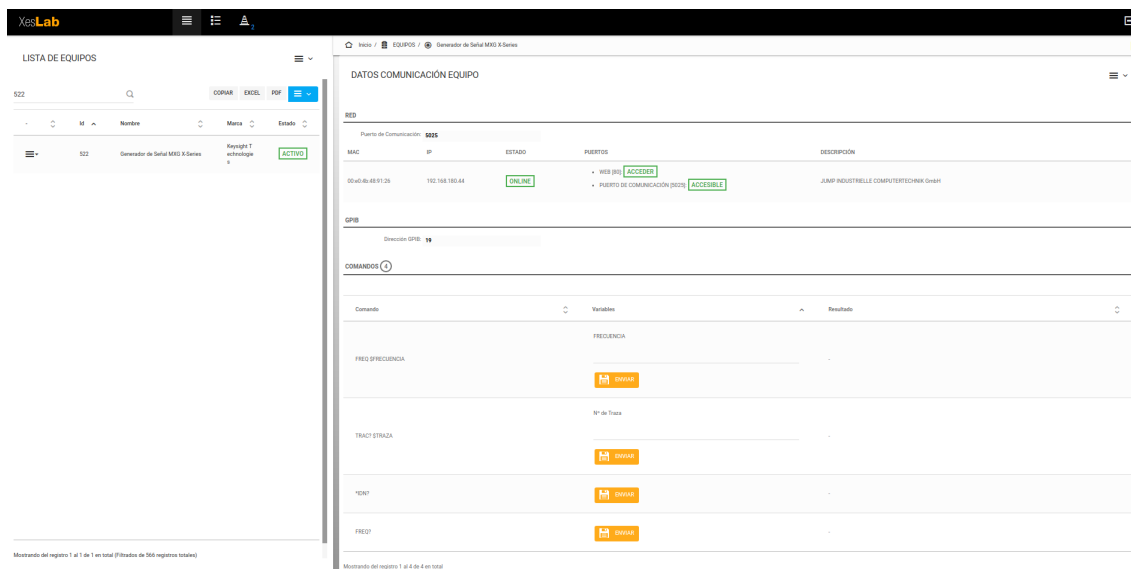


Figura 5.7: Panel de comunicaciones do equipo 522 en XesLab

The screenshot displays the XESLAB software interface for configuring a procedure. The top navigation bar shows the path: Inicio / DOCUMENTACIÓN DEL SISTEMA / PEE-EMP-001. The main interface is divided into several sections:

- DATOS PROCEDIMIENTO:** A form containing:
  - Tipo: PROCEDIMIENTO ESPECÍFICO DE ENSAYO
  - Área: Electromagnéticos
  - SubÁrea: Prestaciones EM
  - Título: Atenuación de apantallamiento en cables. Método triaxial.
  - DOCUMENTO: PEE-EMP-001
  - Descripción: Atenuación de apantallamiento en cables. Método triaxial.
- CONTABILIDAD: CONFIGURACIÓN DE LÍNEAS**
- ZONAS/PEQS**
- REQUISITOS DE CARACTERÍSTICAS TÉCNICAS**
- REQUISITOS DE CARACTERÍSTICAS DE INSTALACIÓN**
- ELEMENTOS DEL SISTEMA DE GESTIÓN**
- ELEMENTOS RELACIONADOS**
- ARCHIVOS**
- HISTORIAL**

Below these sections, there are two main panels:

- IMPLEMENTACIONES DEL PEE**
- MEDIDA TÉCNICA (COMPROBACIÓN PEE)**

The **MEDIDA TÉCNICA (COMPROBACIÓN PEE)** panel includes a table of variables and a selection menu for the receiver.

Longitud del Tubo (m)	Frecuencia [INICIO] (Hz)	Frecuencia [FIN] (Hz)	Nº de Puntos	Impedancia del cable	Generador
2	30000000.0	2150000000.0	801	75	Generador de Señal MXG X-Series [Id: 522]

Receptor: Receptor de medida R&S ESCI (9KHz-3GHz) [Id: 75]

ENVIAR

© 2019. LAOETEL - TELEVES

Figura 5.8: Pantalla dun procedemento, co panel da medida técnica

# Capítulo 6

## Deseño e implementación na pasarela

O software da pasarela está escrito en C/C++; en C temos acceso á interface proporcionada pola librería `linux-gpib` para interacción con dispositivos GPIB, e C++ proporciona útiles capacidades de abstracción e de orientación a obxectos. Esencialmente, a pasarela ofrece unha funcionalidade de enrutado das mensaxes cara a interface hardware correcta.

Con respecto á súa estrutura, hai que facer unha distinción con respecto ao código escrito para o servidor. En XesLab, as unidades de código que xestionan un concepto determinado denomínanse **módulos**; na pasarela, emprégase a palabra módulo para designar a clase que xestiona unha interface hardware determinada (por exemplo, módulo GPIB). Para evitar confusión, chamaráselle **paquete** ao conxunto de clases cuxas funcionalidades se deseñen arredor dunha mesma idea (por exemplo, paquete de rede ou paquete controlador).

Ao longo de todo o programa, úsanse extensivamente ferramentas esenciais de C++; herencia e polimorfismo con métodos virtuais, punteiros intelixentes, etc. O principio RAII (*Resource Acquisition Is Initialization*) causa que todos os elementos da xerarquía se instancien e se destrúan de forma recursiva, xa que todos descenden dun obxecto controlador.

Na figura 6.1 móstrase o diagrama de clases global; os paquetes están diferenciados por cores. Decidiron obviarse métodos comúns como os construtores, destrutores, *getters* e *setters*, xa que xeran interferencias visuais e dificultan a interpretación do diagrama.

### 6.1. Paquete de rede

Implementa funcionalidades de comunicación IP; abstrae ao resto do software da comunicación co servidor de XesLab. Componse das seguintes clases.

- `GestorConexiones` é un servidor TCP sinxelo, construído arredor da librería

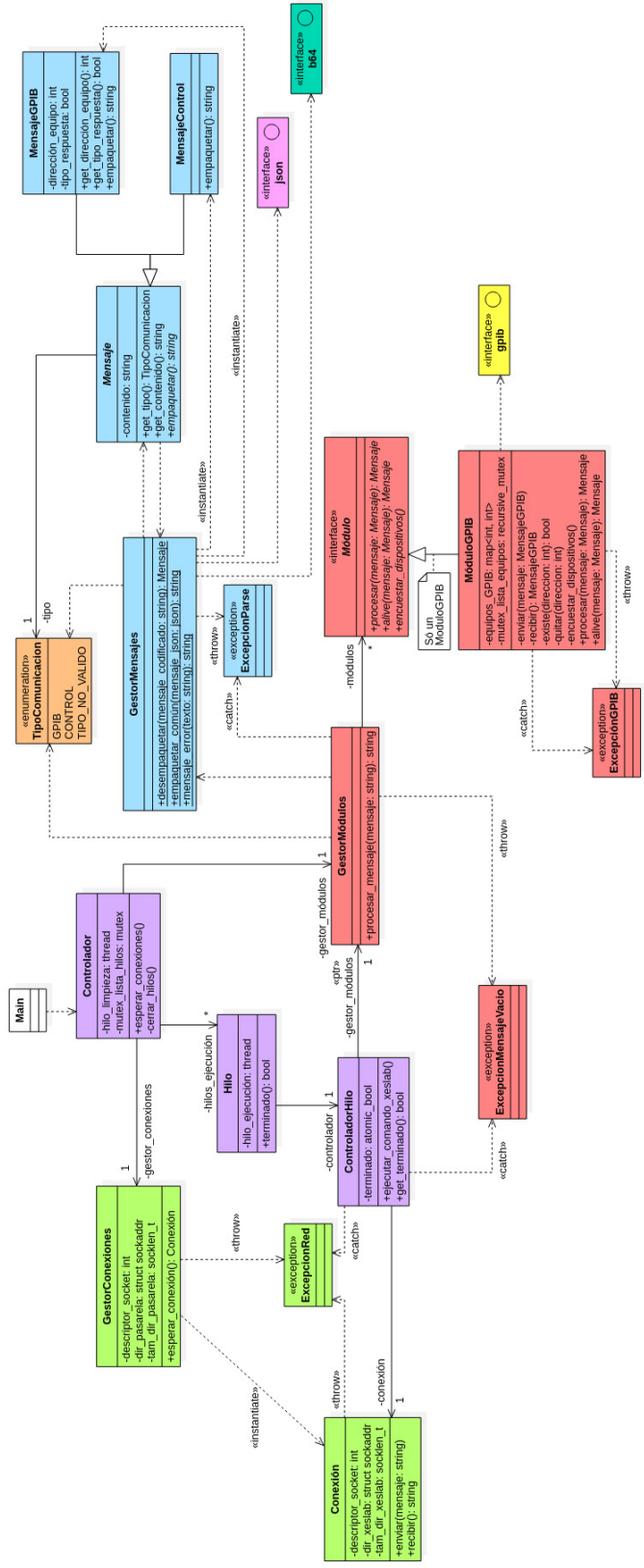


Figura 6.1: Diagrama de clases

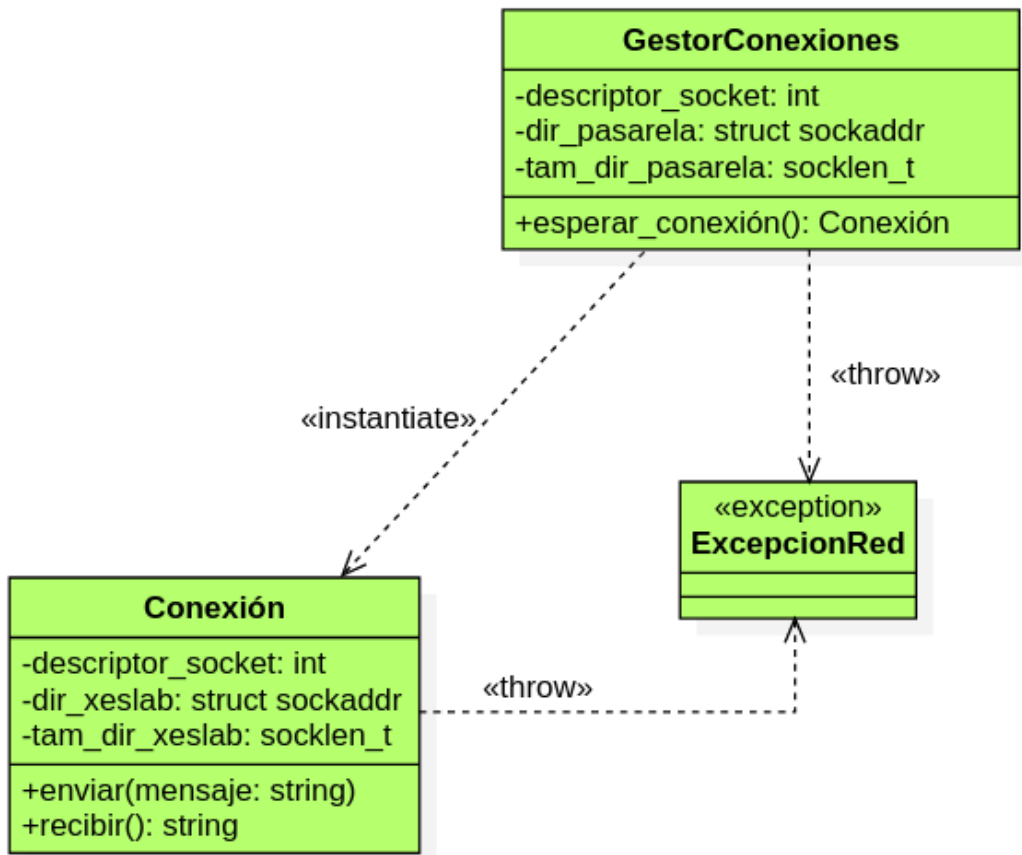


Figura 6.2: Diagrama de clases do paquete de rede

estándar de sockets de C.

- **Conexion** encapsula o socket resultante dunha conexión entrante, e proporciona métodos para o envío e recepción de mensaxes.
- **ExcepcionRed** é unha especialización de `runtime_error`, que se lanza se ocorre un erro nas operacións de comunicación IP.

Considerouse empregar librerías de terceiros para a comunicación IP, como, por exemplo, `Boost.Asio`; sen embargo, decidiuse que era preferible minimizar o número de compoñentes externas, para evitar incompatibilidades.

O método máis importante de **GestorConexiones** é `esperar_conexion`; espera por unha conexión entrante de `XesLab` no porto indicado na macro `PUERTO`, e instancia unha `Conexion` que encapsula o socket resultante da conexión. Devólvese ao invocador un punteiro único (de tipo `std::unique_ptr<Conexion>` a este obxecto, de xeito que só unha referencia ao obxecto será válida ao mesmo tempo; é necesario

devolver un punteiro para evitar que se invoque o destrutor sobre o obxecto local, xa que o destrutor cerra o socket, e este punteiro é único porque só unha entidade precisa acceder ao fío.

Como xa foi mencionado, `Conexion` encapsula un socket de comunicacións, así como o enderezo IP de XesLab, anque este último dato non é importante para a execución. Os seus métodos `enviar` e `recibir` son fachadas para as funcións de C `send` e `recv`; preferiuse unificar toda esta lóxica nunha mesma clase para non sobrecargar outros elementos con responsabilidades de rede que non lles corresponden.

## 6.2. Paquete controlador

Xestiona os fíos de execución do programa; a súa creación, destrución, e limpeza cando se dá a orde de terminar o proceso. En todo momento, hai un mínimo de dous fíos executándose, pero é probable que existan máis, creados durante o proceso de execución normal da pasarela.

- O fío principal espera por conexións do servidor (facendo uso das funcionalidades do paquete de rede) e, para cada conexión entrante, instancia un fío que procesará as súas mensaxes, asignándolle os recursos necesarios.
- Do parágrafo anterior, infírese que existe un fío por cada conexión activa co servidor de XesLab.
- O fío de limpeza recorre a lista de fíos activos e destrúe aqueles que xa remataron todas as súas tarefas, pero aínda non foron recollidos polo fío principal (están en estado "zombi"). Isto evita que se acaparen recursos do sistema de forma inútil.

Cada un dos fíos de procesamento ten un obxecto que actúa como controlador do fío (unha instancia de `ControladorFío`), e que se considera un axudante para o controlador global da aplicación (a única instancia de `Controlador`). Este é o dono dos recursos do fío.

### 6.2.1. A xerarquía de fíos

A inclusión de soporte para *multithreading* deu lugar ao uso dunha serie de clases que poden levar á confusión.

- `std::thread` é a clase da librería estándar de C++ que se emprega para manexar e identificar un fío de execución "real". No momento en que se instancia, se se lle asigna unha función, **entra automaticamente en estado "listo para executarse"**.

Cando na clase `Controlador` se fala de fíos de execución, referímonos a este obxecto ou ao fío "real", executable que identifica.



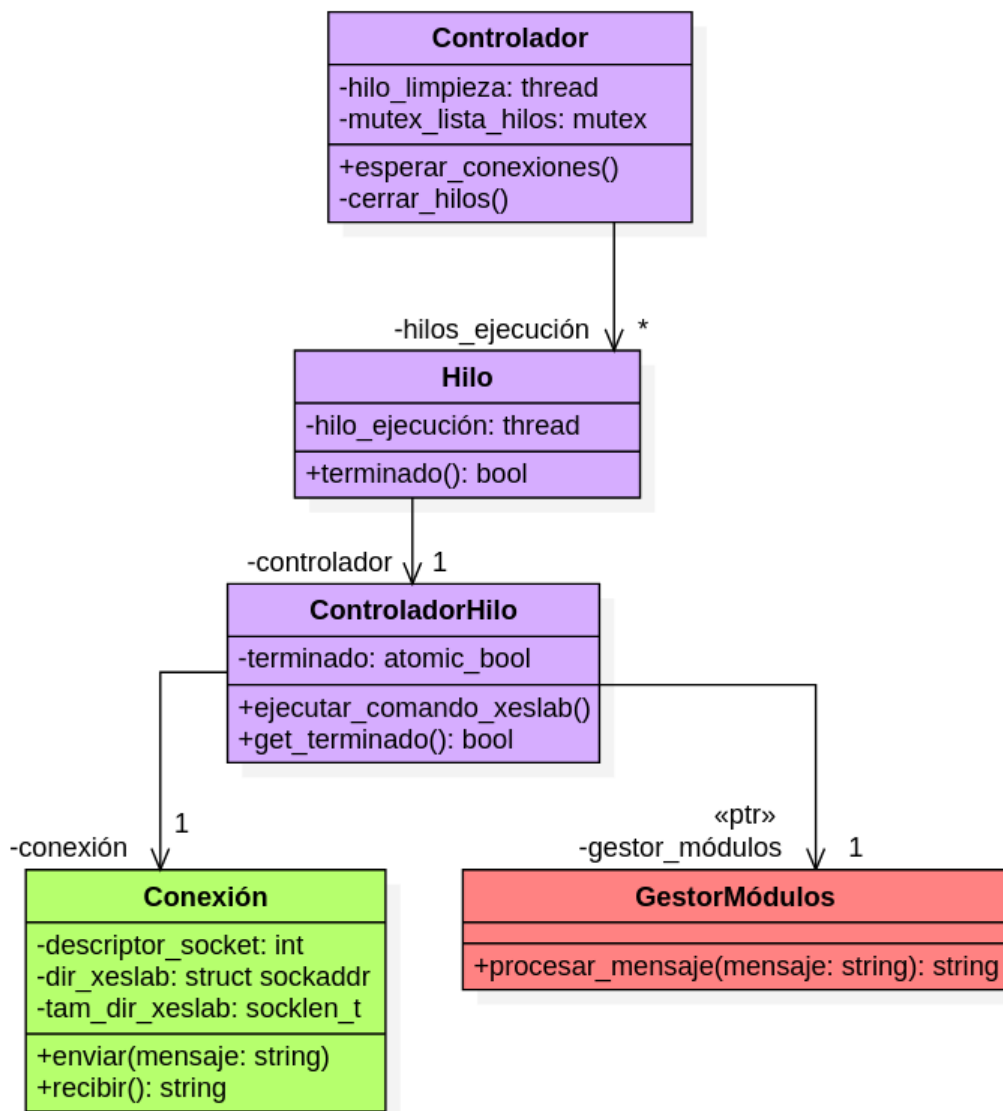


Figura 6.3: Diagrama de clases do paquete controlador

- `ControladorHilo` é a clase que contén os obxectos locais ao fío (por exemplo, a conexión). Cada `std::thread` creado por consecuencia dunha conexión entrante está asociado a un `ControladorHilo`.
- `Hilo`. Debido a que os obxectos de tipo `std::thread` non son copiables, e por consecuencias de RAII e outros detalles de implementación de C++, foi preciso crear unha clase personalizada para asociar o `ControladorHilo` e o `std::thread`. Non aporta máis funcionalidade; o único relevante é que invoca `join` sobre o seu `std::thread` na destrución.

Cando na clase `Controlador` se fala de fíos, este é o obxecto ao que nos referimos.

### 6.2.2. O controlador da aplicación

A clase `Controlador` é aquela da que colga toda a xerarquía de paquetes. Contén os seguintes elementos.

- A lista que almacena os fíos correspondentes ás conexións activas: un `std::vector` de `std::shared_ptr`. Utilízanse punteiros para evitar a copia, e son punteiros compartidos para permitir a iteración sobre o vector.
- Un obxecto de tipo `mutex` que controla o acceso á lista de fíos. Evita carreiras críticas entre o fío principal, que inserta fíos na lista, e o fío de limpeza, que os elimina.
- A única instancia de `GestorConexiones`.
- A única instancia de `GestorModulos`; o comportamento desta clase detallarase posteriormente, pero é esencialmente o obxecto que multiplexa as mensaxes á interface correcta. É importante que exista unha soa instancia desta clase xa que dela medra a xerarquía de clases que coñecen e manteñen o estado dos equipos conectados á pasarela, e copiar esta clase para cada fío implicaría a copia de toda a árbore, de xeito que o estado xa non sería global. Cada fío é o dono dun punteiro a ela.
- O obxecto `std::thread` que representa o fío de limpeza.

#### O fío principal

Executa o método `Controlador::esperar_conexiones`.

```
while (ejecucion) {
    try {
        std::unique_ptr<Conexion> nueva_conexion =
            gestor_conexiones.esperar_conexion();
```

```

std::shared_ptr<Hilo> hilo =
    std::make_shared<Hilo>(&gestor_modulos,
        std::move(nueva_conexion));

mutex_lista_hilos.lock();
hilos_equipos.push_back(std::move(hilo));
mutex_lista_hilos.unlock();

} catch (ExcepcionRed &e) {
;
}

std::this_thread::yield();
}

```

Mentres non se solicitara a saída do programa (ejecucion é **true**), espérase por unha conexión entrante de XesLab. No momento en que se recibe unha, instánciase un fío; o principio RAII (*Resource Acquisition Is Initialization*) de C++ fai que se inicie a execución do seu correspondente fío inmediatamente.

Insírese o fío na lista, bloqueando e desbloqueando o acceso nos momentos adecuados.

Se se captura unha excepción de rede durante este proceso, pódese ignorar de forma segura; a consecuencia límitase a que a pasarela non acepte esa conexión, o que non afecta ao resto de procesos en marcha.

Tras todo isto cédese o control ao novo fío ou ao de limpeza con `std::thread::yield`.

### O fío de limpeza

Executa a función `Controlador::cerrar_hilos`. Esta, mentres non se solicitara a saída do programa, elimina todos os fíos acabados, bloqueando e desbloqueando o acceso á lista de forma correcta. Cede o control aos outros fíos tras rematar de percorrer a lista.

Para eliminar elementos da lista sen invalidar os iteradores emprégase o modismo *remove-erase*.

- `std::remove_if` coloca ao final da lista todos os elementos para os que `terminado` devolve **true**, e devolve un iterador que apunta ao “novo fin” (o primeiro elemento dos que foron trasladados ao final).
- `erase` elimina todos os elementos que foron trasladados ao final.

```

auto nuevo_fin = std::remove_if(hilos_equipos.begin(),
    hilos_equipos.end(), terminado);
hilos_equipos.erase(nuevo_fin, hilos_equipos.end());

```

`terminado` é unha función que recibe un fío e devolve o seu campo `terminado`, un valor que o propio fío de execución pon a **true** cando remata a súa tarefa.

É preciso facer esta adaptación (non se pode converter `terminado` nun atributo público e simplemente acceder a este) polas restricións que `remove_if` impón sobre os predicados que acepta.

### Saída do programa

Na instanciación do controlador, ademais da instanciación dos atributos (que é automática), establécese o método de saída do programa. Debido a que se trata dun programa *multithread*, é necesario que todos os fíos de execución que estean executándose rematen a súa tarefa e `std::thread::join` sexa invocado sobre eles para poder iniciar a execución da xerarquía de destrutores, para poder liberar todos os recursos de forma segura.

Deséxase que o programa frene os fíos e libere os recursos cando reciba unha sinal de terminación (`SIGTERM`) ou de interrupción (`SIGINT`). Establécese para estes dous sinais un manexador que pon a **false** a variable global `ejecucion`. Esta é de tipo `std::atomic`, o que impide que existan carreiras críticas sobre o seu acceso. Para isto emprégase a función de C `sigaction`; é importante que non se active o *flag* `SA_RESTART` para que a chegada dos sinais poida interromper as funcións `accept` e `recv`, que bloquean ao fío invocador.

```

struct sigaction sig_salir;
sig_salir.sa_handler = manexador;
sigemptyset(&sig_salir.sa_mask);
sig_salir.sa_flags = 0;

sigaction(SIGTERM, &sig_salir, nullptr);
sigaction(SIGINT, &sig_salir, nullptr);

```

Se `ejecucion` non é **true**, nin o fío que espera conexións entrantes nin o de limpeza inician unha nova iteración, polo que se sae do alcance de `main` e se inicia a secuencia de destrutores. Nesta secuencia invócase `std::thread::join` sobre todos os fíos de execución que non son o principal, xa que un fío de execución non pode rematarse a si mesmo, e na xerarquía hai obxectos de tipo `std::thread` que non se poden destruír sen rematar primeiro aos fíos de execución aos que representan.

### 6.2.3. O controlador de fío

Contén os obxectos locais ao fío, os que este invoca para procesar as mensaxes.

- Un punteiro ao xestor de módulos; trátase da mesma instancia para todos os fíos.
- A conexión co servidor.
- A variable `terminado` de tipo `std::atomic`, que o fío pon a **true** cando remata a súa tarefa.

A función executada polo fío é `ControladorHilo::ejecutar_comando_xeslab`.

```

try {
    std::string comando, respuesta;

    comando = conexion->recibir();
    respuesta = gestor_modulos->procesar_mensaje(comando);
    conexion->enviar(respuesta);

} catch (ExcepcionMensajeVacio &e) {
    ;

} catch (ExcepcionRed &e) {
    try {
        conexion->enviar(GestorMensajes::mensaje_error(e.what()));

    } catch (ExcepcionRed &e_error) {
        ;
    }
}

```

Decidiuse que cada conexión só durará un comando, por sinxeleza e pola forma que teñen habitualmente as medicións; non é común enviar moitas mensaxes en pouco tempo ao mesmo equipo, polo que abrir e pechar a conexión de cada vez non supón un *overhead* notable.

Toda mensaxe que chega do servidor é respostada, independentemente de que o comando motivo da mensaxe implique unha resposta ou non. Isto só sucede na pasarela, porque poden existir erros no procesamento das mensaxes que é importante trasladar ao servidor; se non se fixera isto, o servidor daría un erro de *time-out* en calquera situación de erro da pasarela, facendo que sexa moito máis difícil interpretar a causa correctamente. Deste xeito, un comando pode ter as seguintes respostas:

- A resposta do equipo (ou da pasarela, en casos coma o do comando \*IDN?)
- Unha mensaxe de erro; se en calquera punto do proceso de procesamento se produce un erro, descártase a resposta xerada polo equipo e devólvese no seu lugar a de erro. Os elementos que están por debaixo na pila de invocación non son conscientes de que se trata dun erro, e devólvena como se fose unha resposta normal.
- Unha confirmación de que o comando se executou sen erros

A única excepción que hai a isto é no caso das mensaxes vacías, que poden chegar á pasarela durante os escaneos ARP, interferindo co proceso. Non ten sentido respostar estas mensaxes, polo que se descartan.

Se sucede unha excepción de rede durante o proceso, inténtase notificar ao servidor. Se non é posible, ignórase; o servidor interpretarao como un *time-out*.

### 6.2.4. Outros elementos

No paquete controlador tamén se inclúen certos elementos dos que depende todo o sistema.

- `TipoComunicacion` é un **enum** que indica o tipo de interface hardware dun módulo ou mensaxe. Ademais dos nomes das interfaces, un `TipoComunicacion` pode ser algún dos pseudo-tipos `CONTROL` (para as mensaxes de erro ou de estado) ou `TIPO_NO_VALIDO`.

Proporciona funcións para transformar de `std::string` a `TipoComunicacion` e viceversa.

- No ficheiro `Utilidades.cpp` hai funcións empregadas durante o proceso de desenvolvemento para facilitar a depuración.

## 6.3. Paquete de mensaxes

Comprende todos os tipos de mensaxes, así como unha clase estática que facilita as conversións e as serializacións. As mensaxes sempre se acceden a través dun punteiro, encapsulado nun `std::unique_ptr`; realmente, estes obxectos poderían estar na pila, pero con frecuencia dase o caso de que se instancian dentro dun bloque condicional, e se necesita dispor deles unha vez fóra deste bloque, polo que é necesario usar punteiros.

### 6.3.1. A xerarquía de mensaxes

`Mensaje` é unha clase abstracta da que herdan os tipos de mensaxes específicos. Esta clase proporciona un tipo e un contido (un *string* que almacena o comando), así como *getters* para estes tipos.

Intentouse implementar unha interface a través da que se puidera requerir que as clases fillas tiveran implementado un construtor a partir dun *string* e un método que as empaquetara nun *string* para o seu envío. Porén, C++ non permite crear construtores virtuais, polo que se optou por implementar un método único de desempaqetamento no xestor de mensaxes, e extraer tamén a este a parte común do empaquetamento, por consistencia.

As clases fillas deben implementar o método `empaquetar`; a funcionalidade deste será a de crear un obxecto JSON con todos os campos da clase (os específicos do tipo de mensaxe concreto, máis tipo e contido), e invocar ao método de `GestorMensajes` que implementa a parte común.

No momento da redacción deste documento, só existen dous tipos de mensaxes específicas: mensaxes GPIB, que complementan os campos comúns co enderezo GPIB e o tipo de resposta; e mensaxes de control, que non engaden información nova, só se empregan pola categoría.

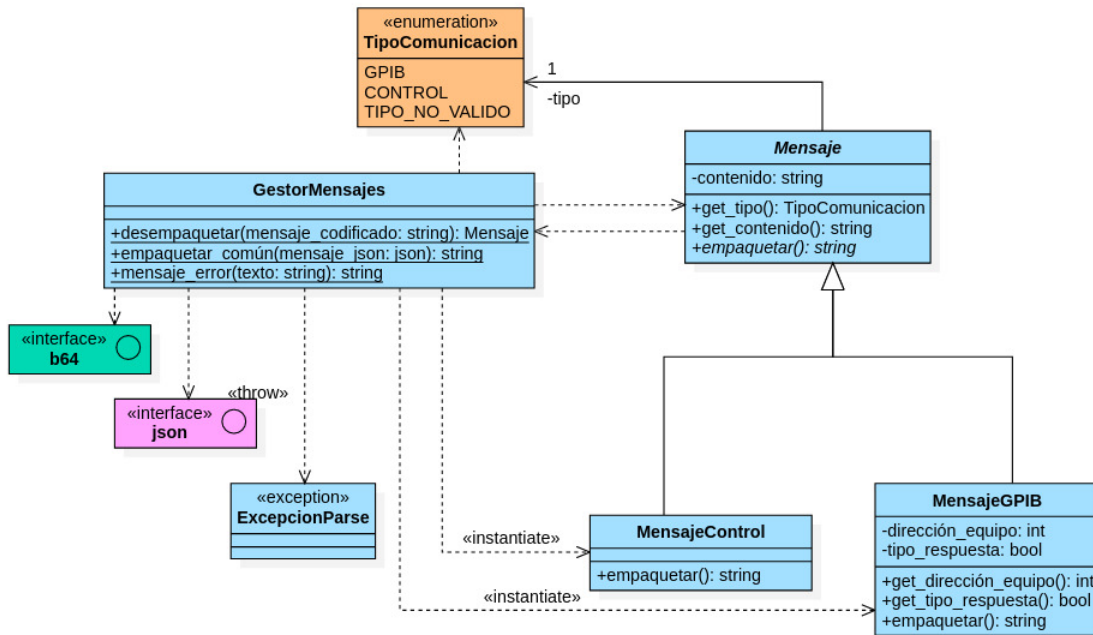


Figura 6.4: Diagrama de clases do paquete de mensaxes

### 6.3.2. O xestor de mensaxes

A clase `GestorMensajes` proporciona tres funcionalidades principais: empaquetamento de mensaxes, desempaquetamento de mensaxes, e creación de mensaxes de erro de forma non verbosa.

Para decodificar e codificar en Base64 emprégase a librería `libb64`, que permite o uso con `streams`. Para a serialización, emprégase a librería `nlohmann::json`; proporciona as funcións `json::parse` para deserializar e `json::dump` para serializar.

A librería de Base64 causa que os `strings` decodificados teñan caracteres de *padding* ao final, polo que se deseñou a función `eliminar_padding` para eliminalo.

Empaquetamento	Desempaquetamento
Crear o obxecto JSON cos campos da mensaxe	Decodificar
Serializar	Eliminar <i>padding</i> da mensaxe
Codificar	Deserializar
	Determinar tipo da mensaxe
	Instanciar mensaxe

Cadro 6.1: Proceso de empaquetamento e desempaquetamento

Unha vez obtida a mensaxe en forma de obxecto JSON, examínase o seu tipo

para instanciar a clase correcta.

```

switch (tipo) {
  case GPIB: {
    std::unique_ptr<Mensaje> mensaje_GPIB =
      std::make_unique<MensajeGPIB>(
        mensaje_json.at("direccion_equipo").get<int>(),
        mensaje_json.at("contenido").get<std::string>() +
          TERMINADOR,
        mensaje_json.at("tipo_respuesta").get<bool>());

    return mensaje_GPIB;
  }

  default:
    throw ExcepcionParse("El_tipo_del_mensaje_no_es_valido");
}

```

O xestor de mensaxes intercepta todas as excepcións lanzadas pola librería `nlohmann::json` e lanza a súa propia `ExcepcionParse`, que resulta máis ilustrativa para comprender os erros.

O método `mensaje_error(const std::string &texto)` devolve unha mensaxe de erro con `texto` como contido. Serve para reducir a verbosidade nas zonas en que se xeran mensaxes de erro, e facer o código en xeral máis lexible.

## 6.4. Paquete de módulos

Comprende o xestor de módulos e os módulos. Realiza o auténtico procesamento das mensaxes.

### 6.4.1. Os módulos

Un módulo debe cumprir a interface `Modulo`, que require os seguintes métodos.

- `std::unique_ptr<Mensaje> procesar(std::unique_ptr<Mensaje> mensaje)`: envía mensaje ao equipo correcto, e devolve unha resposta.
- `std::unique_ptr<Mensaje> alive(std::unique_ptr<Mensaje> mensaje)`: comproba se o equipo ao que vai dirixido a mensaxe está dispoñible para a comunicación.
- `encuestar_dispositivos()`: actualiza o rexistro de equipos conectados.

### O módulo GPIB

De momento, o único módulo implementado que existe é o módulo GPIB. Fai uso da librería `linux-gpib`; crea unha capa de abstracción entre o resto do sistema e as cuestións específicas de GPIB, que se atopan contidas neste módulo.



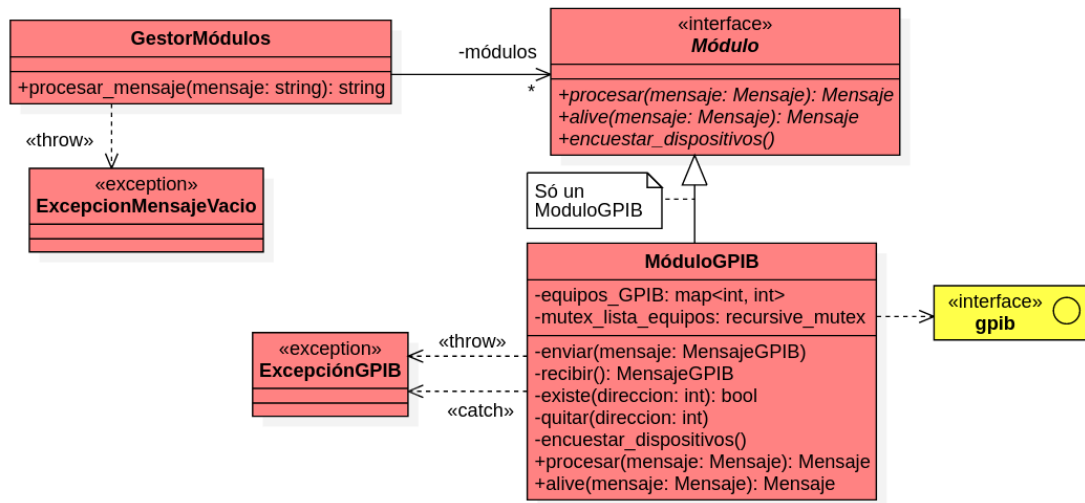


Figura 6.5: Diagrama de clases do paquete de módulos

Para manter o estado dos dispositivos GPIB, conta co campo `equipos_GPIB`, que é un `std::map<int, int>`; o mapa está indexado polo enderezo GPIB dos equipos, e contén os descritores dos equipos abertos. O acceso a el está protexido por un mutex recursivo. Para facer comprobacións e modificacións do mapa de xeito sinxelo, existen as funcións `existe` e `quitar`.

Os enderezos GPIB están identificados por enderezos primarios, que poden ter un valor do 0 ao 30 (incluídos); `ModuloGPIB::encuestar_dispositivos` abre descritores de comunicación para todos estes dispositivos. Como isto se pode facer aínda que non haxa equipos escoitando en todas as direccións, e `linux-gpib` non proporciona unha ferramenta que permita saber se os hai, hai que enviar un comando de proba para comprobar que efectivamente hai un equipo que responde. Se non o hai, cérrese o descriptor, e se o hai, insértase en `equipos_GPIB`. Esta función invócase na instanciación do módulo e cada vez que o servidor solicita comprobar o estado dun dispositivo.

```

for (int direccion = 0; direccion <= 30; direccion++) {
    int descriptor = ibdev(DEFAULT_MINOR, direccion, NO_SAD,
        DEFAULT_TIMEOUT, DEFAULT_ENVIAR_EOI, DEFAULT_MODALITY);

    if (descriptor != -1) {

        if (ibwrt(descriptor, IDN, strlen(IDN)) & ERR) {

            ibonl(descriptor, 0);
            quitar(direccion);
            continue;
        }
    }
}
  
```

```

char *buffer = (char *) malloc(TAM_BUFFER);
bzero(buffer, TAM_BUFFER);

if (ibrd(descriptor, buffer, TAM_BUFFER) & ERR) {
    free(buffer);
    ibonl(descriptor, 0);
    quitar(direccion);
    continue;
}

if (strlen(buffer) > 0) {
    equipos_GPIB[direccion] = descriptor;
    buffer[strlen(buffer) - 1] = 0;
    free(buffer);
} else {
    quitar(direccion);
    ibonl(descriptor, 0);
}
}

```

Deste xeito, o método `ModuloGPIB::alive` só ten que invocar a `encuestar_dispositivos` e comprobar no mapa se o enderezo indicado está asociado a un descriptor.

Os métodos `enviar` e `recibir` envolven as funcións da librería GPIB `ibwrt` e `ibrd`, respectivamente. `procesar` chama a estas funcións e tamén fai as conversións necesarias para as mensaxes.

As mensaxes chegan no tipo `std::unique_ptr<Mensaje>`. Instanciáronse como `MensajeGPIB`, pero é necesario que se traten como o tipo xenérico para poder pasar a través da interface única en `GestorModulos` e `Modulo`. Hai que volver a converter a mensaxe ao seu tipo orixinal cando chega a este punto do procesamento, para poder acceder aos atributos específicos de GPIB.

```

MensajeGPIB mensaje_GPIB = *(dynamic_cast<MensajeGPIB*>
    (mensaje.get()));

```

A resposta que devolve `procesar` debe ter tipo `std::unique_ptr<Mensaje>`; de novo, por causa da interface única.

Dependendo de se o comando ten resposta ou non, espérase pola resposta do equipo ou constrúese unha confirmación de recepción.

```

if (mensaje_GPIB.get_tipo_respuesta()) {
    respuesta = std::make_unique<MensajeGPIB>
        (recibir(mensaje_GPIB.get_direccion_equipo()));
} else {
    respuesta = std::make_unique<MensajeGPIB>
        (mensaje_GPIB.get_direccion_equipo(), "OK", false);
}

```

### 6.4.2. O xestor de módulos

Almacena os módulos correspondentes ás diferentes interfaces hardware no seu atributo `modulos`, de tipo `std::map<TipoComunicacion, Modulo*`. Se ben engade certa complexidade polos requisitos que C++ pon sobre a herdanza, o uso dun mapa fai que sexa moi sinxelo enviar unha mensaxe ao módulo adecuado, sen ter que usar condicionais.

```
respuesta = modulos.at(tipo)->procesar(std::move(mensaje));
```

- As mensaxes vacías, xeradas polo escaneo ARP do servidor, son ignoradas. Lánzase unha `ExcepcionMensajeVacio` para que o fío non espere por ningunha resposta.
- As mensaxes `*IDN?` en texto plano, que envía o servidor para saber se a IP corresponde a unha pasarela ou un equipo, son respondidas co texto `PASARELA_TELEVES`.
- O resto de mensaxes considéranse normais, e non se procesan no xestor de módulos senón nun módulo. Para as mensaxes cuxo contido é o comando `XESLAB_ALIVE`, cuxo propósito é permitir que o servidor coñeza se un equipo concreto coa IP da pasarela está verdadeiramente conectado á pasarela, invócase o método `Modulo::alive`, e para o resto, `Modulo::procesar`.

## 6.5. Tipos de mensaxes e as súas respostas

O cadro 6.2 detalla os tipos de mensaxe que poden chegar á pasarela, e o cadro 6.3 indica que tratamento se lles aplica e que respostas causan.

Tipo		Posibles erros	
Mensaxes especiais	Mensaxe vacía	—	Erro de rede
	*IDN?	—	
Mensaxes normais	Comprobación de estado	Comando non interpretable, tipo incorrecto, tipo é seudo-tipo, enderezo incorrecto, resposta inconsistente co comando	
	Comando		

Cadro 6.2: Tipos de mensaxes e posibles erros

Mensaxe	Exemplo	Causa	Resposta
Baleira		Escaneo ARP do servidor	Ningunha
IDN en texto plano	*IDN?	O servidor quere coñecer se o destinatario é un equipo ou unha pasarela	PASARELA_TELEVES
Comando “alive” para un equipo	{tipo: GPIB, direccion_equipo: 19, contenido: XESLAB_ALIVE, tipo_respuesta: true}	O servidor quere coñecer se un equipo está en condicións de comunicarse	true ou false, dependendo de como estea o equipo.
Comando para un equipo, con resposta	{tipo: GPIB, direccion_equipo: 19, contenido: FREQ?, tipo_respuesta: true}	O usuario quere enviar un comando a un equipo	A resposta do equipo.
Comando para un equipo, sen resposta	{tipo: GPIB, direccion_equipo: 19, contenido: FREQ 2GHz, tipo_respuesta: true}	O usuario quere enviar un comando a un equipo	Unha confirmación de recepción.
Calquera comando	{tipo: GPIB, direccion_equipo: 19, contenido: *IDN?, tipo_respuesta: true}	Non se pode recibir a mensaxe ou enviar a resposta	Excepción de rede: unha mensaxe de control coa mensaxe da excepción como contido.
Comando non construído correctamente	asdfjkl	A mensaxe non está baleira nin é un comando de identificación, e tampouco é unha mensaxe empaquetada en JSON e Base64	Excepción de parse: unha mensaxe de control coa mensaxe da excepción como contido.
Comando cuxo valor tipo non é válido	{tipo: OTRO, direccion_equipo: 19, contenido: +6.000000000000E+09, tipo_respuesta: false}	Non é posible saber a que módulo hai que envíalo	Excepción de parse: unha mensaxe de control coa mensaxe da excepción como contido.
Comando cuxo valor tipo é válido, pero é un pseudo-tipo que non está asociado a un módulo	{tipo: TIPO_NO_VALIDO, direccion_equipo: 19, contenido: +6.000000000000E+09, tipo_respuesta: false}	Non hai un módulo que se encargue dese tipo de mensaxes	Excepción de tipo <i>out of bounds</i> : unha mensaxe de control coa mensaxe da excepción como contido.
Comando cuxo valor contenido non é válido	{tipo: GPIB, direccion_equipo: 19, contenido: OTRO, tipo_respuesta: true}	O equipo non sabe interpretar o comando e polo tanto o módulo xerará un erro	Excepción GPIB: unha mensaxe de control coa mensaxe da excepción como contido.
Comando cuxo valor direccion_equipo non é válido	{tipo: GPIB, direccion_equipo: 31, contenido: OTRO, tipo_respuesta: true}	Intentamos comunicarnos cun equipo que non existe	Excepción GPIB: unha mensaxe de control coa mensaxe da excepción ou erro da librería como contido.

Mensaxe	Exemplo	Causa	Resposta
Comando cuxo valor <i>respuesta</i> non é consistente co equipo	{tipo: GPIB, direccion_equipo: 19, contenido: FREQ 2GHz, tipo_respuesta: <b>true</b> }	Se o comando devolve resposta e o parámetro é <b>false</b> , entón a pasarela trátalo como caso normal e ignora a resposta do equipo. Se o comando non devolve resposta e o parámetro é <b>true</b> , o módulo esperará por unha resposta que non chegará nunca; devolverá unha excepción específica da interface, cuxa causa será presumiblemente un erro de <i>time-out</i> .	Excepción GPIB: unha mensaxe de control coa mensaxe da excepción ou erro da librería como contido.
Comando cuxo valor <i>respuesta</i> non é consistente co que espera o servidor	Sempre que sexa consistente co que emite o comando, non constitúe erro para a pasarela. Se <i>respuesta</i> é <b>true</b> e o servidor non a espera, simplemente a ignorará, e se é <b>false</b> e o servidor espera resposta do equipo, obterá unha confirmación de resposta en vez dos datos que solicitou.		

Cadro 6.3: Exemplos, tratamento, e exemplos de respostas para as distintas mensaxes que poden chegar á pasarela

## 6.6. Diagramas de secuencia

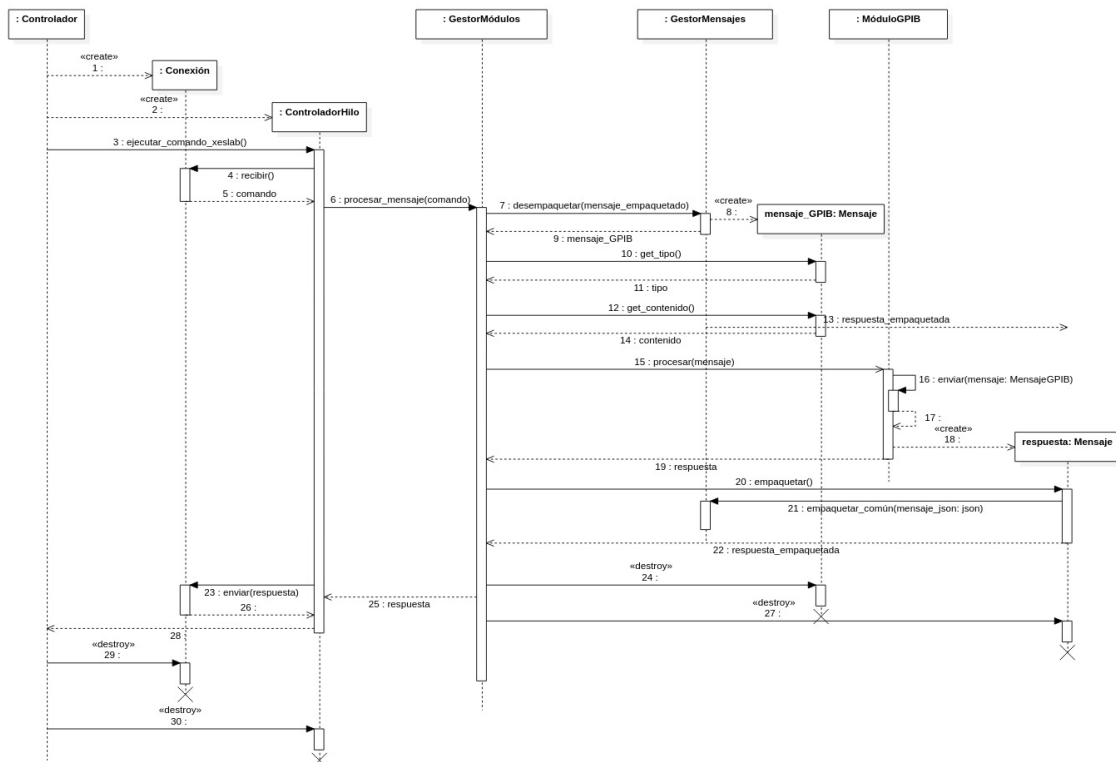


Figura 6.6: Enviar comando sin respuesta a un equipo conectado por GPIB

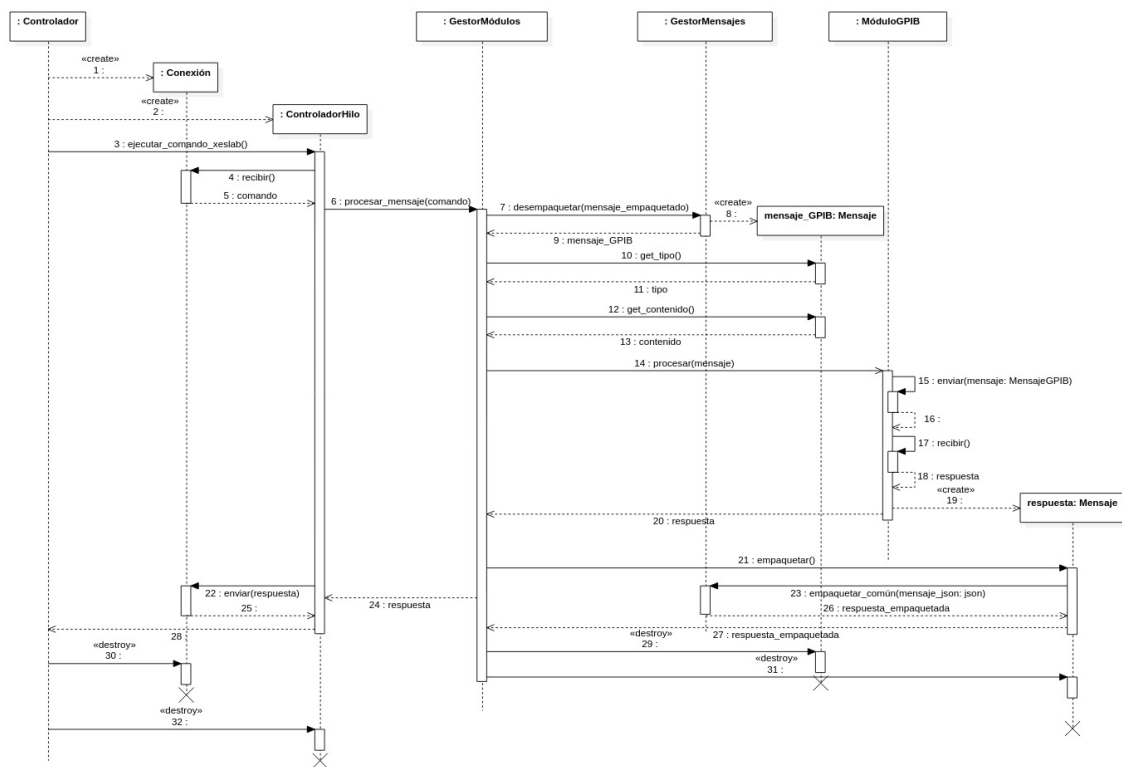


Figura 6.7: Enviar comando con respuesta a un equipo conectado por GPIB

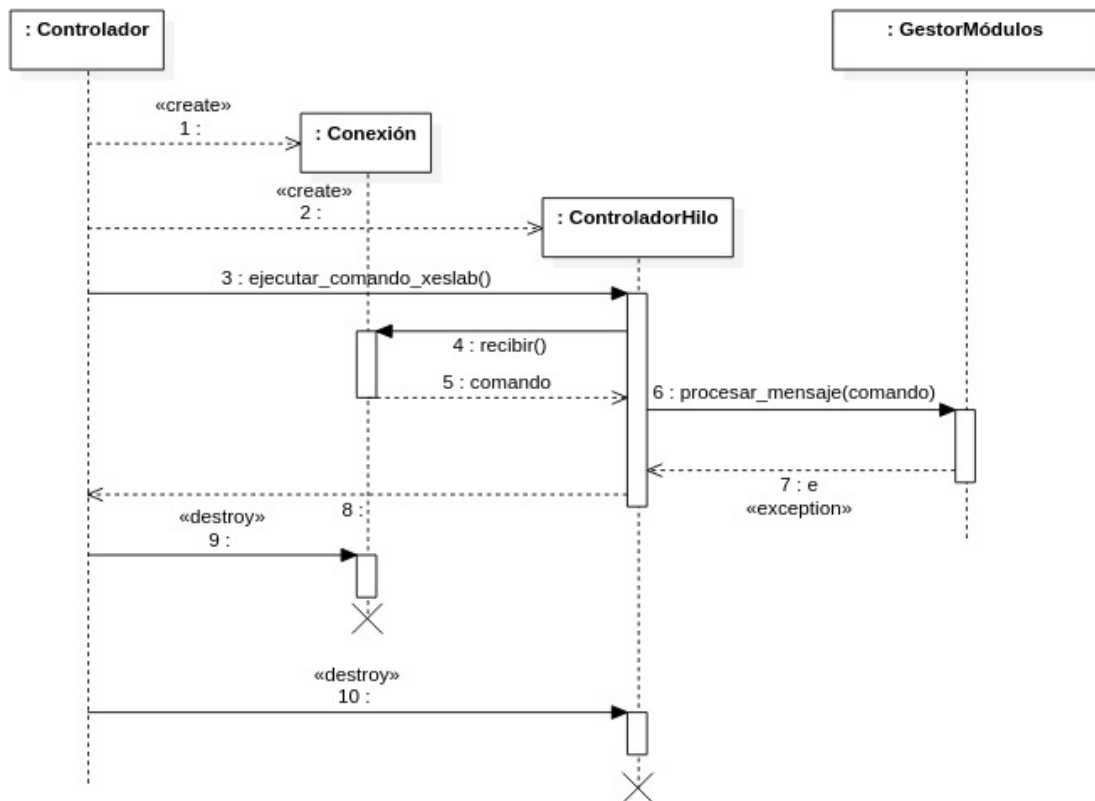


Figura 6.8: Mensaxe baleira



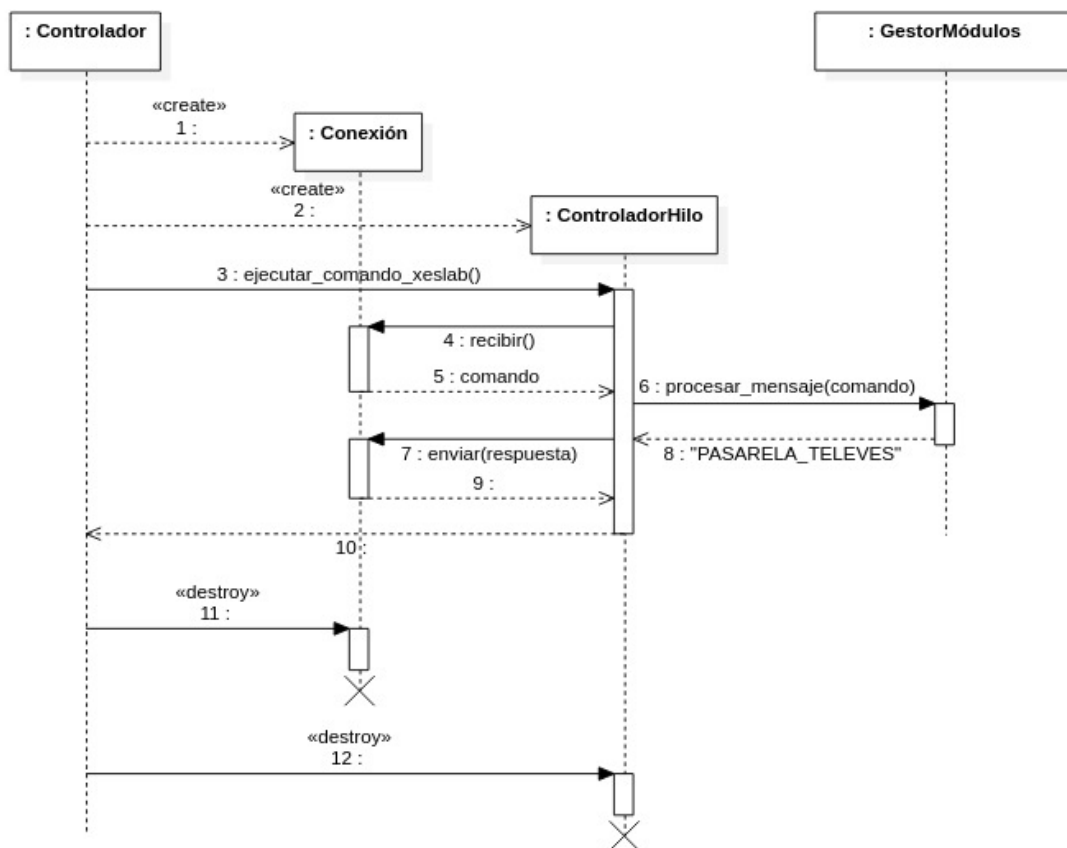


Figura 6.9: \*IDN? en texto plano

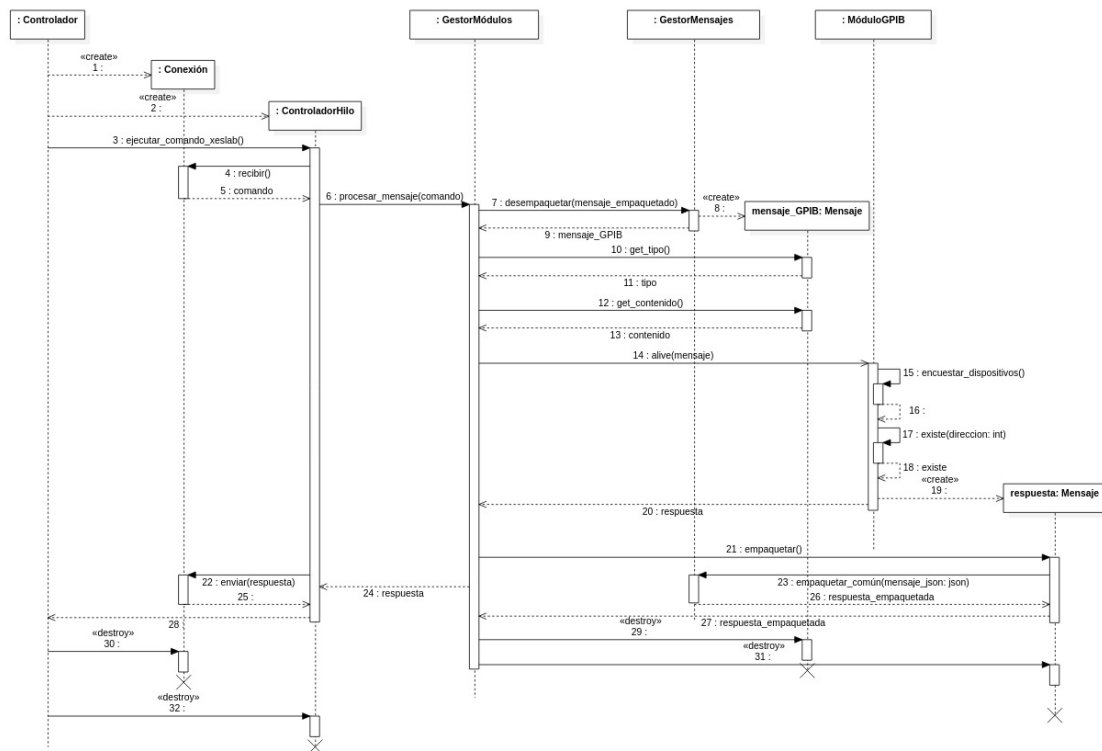


Figura 6.10: Mensaxe “alive” dirixida a un dispositivo GPIB

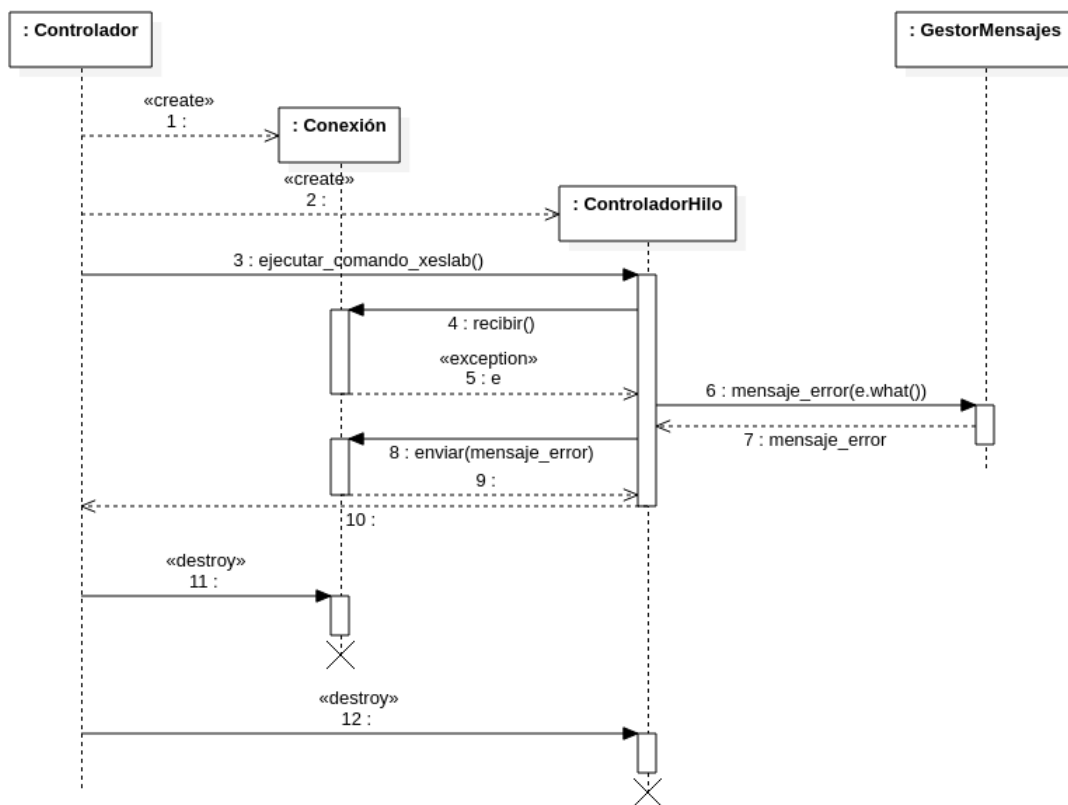


Figura 6.11: Erro de rede, notificado ao servidor

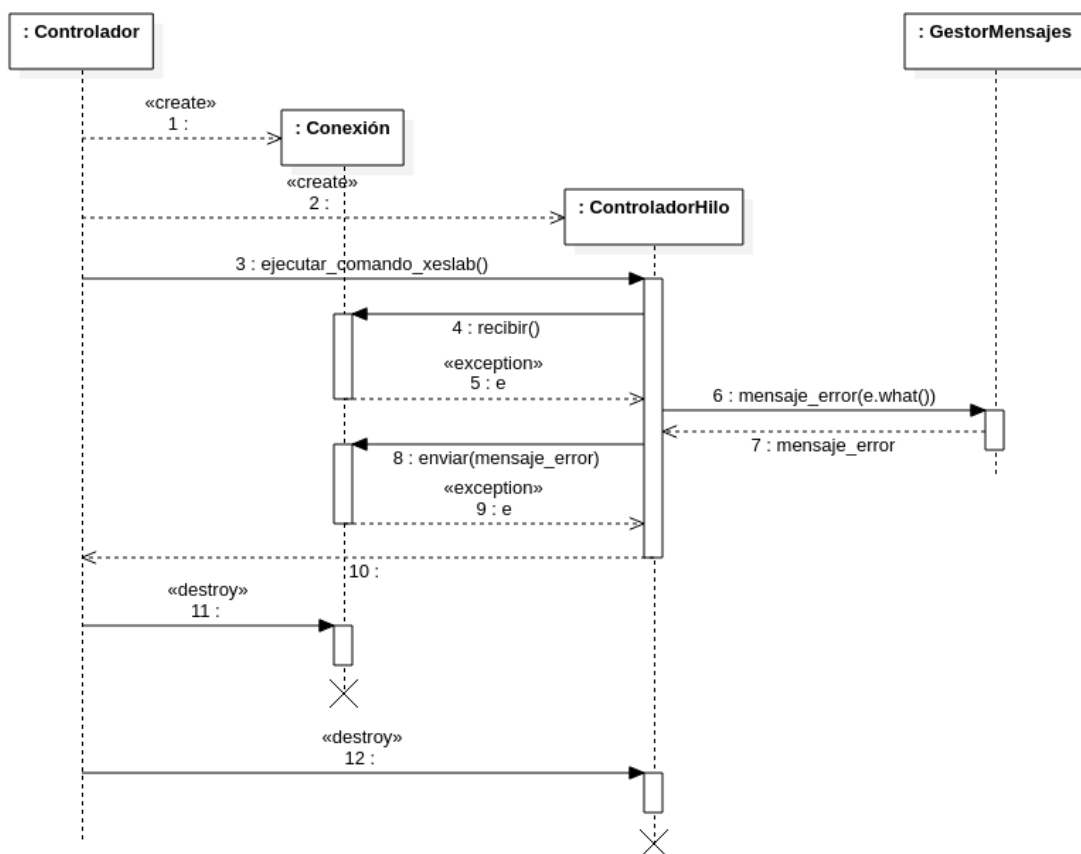
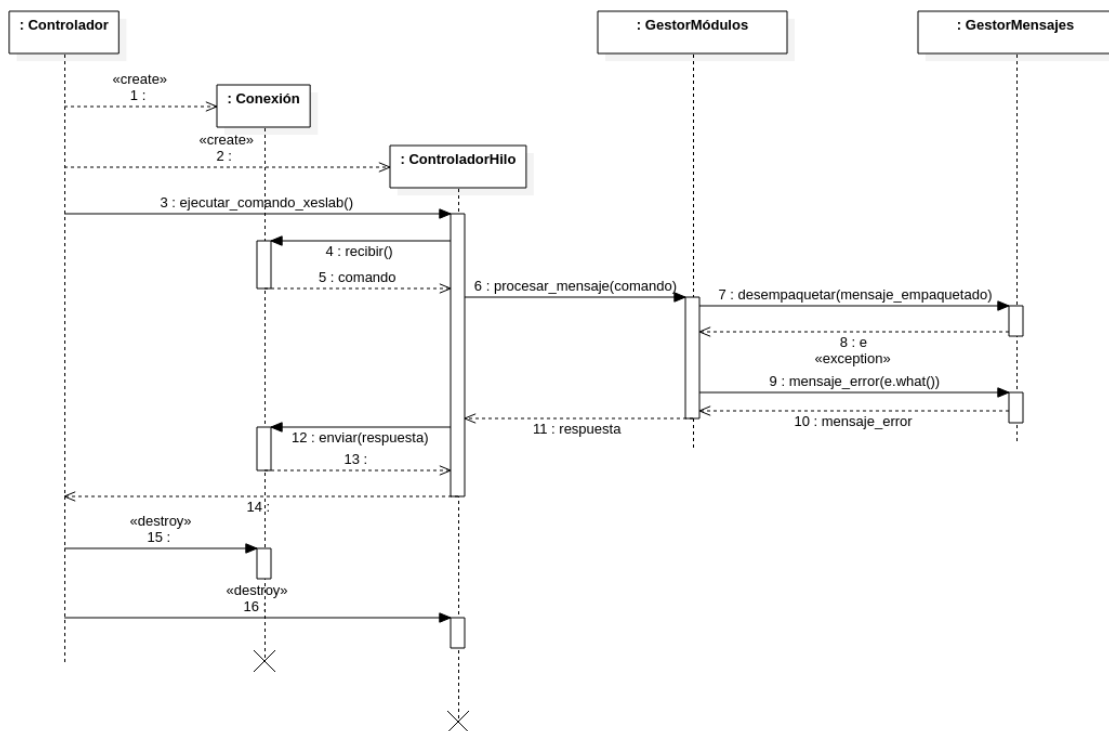


Figura 6.12: Erro de rede que non se pode notificar ao servidor; é ignorado

Figura 6.13: Erro de *parse*

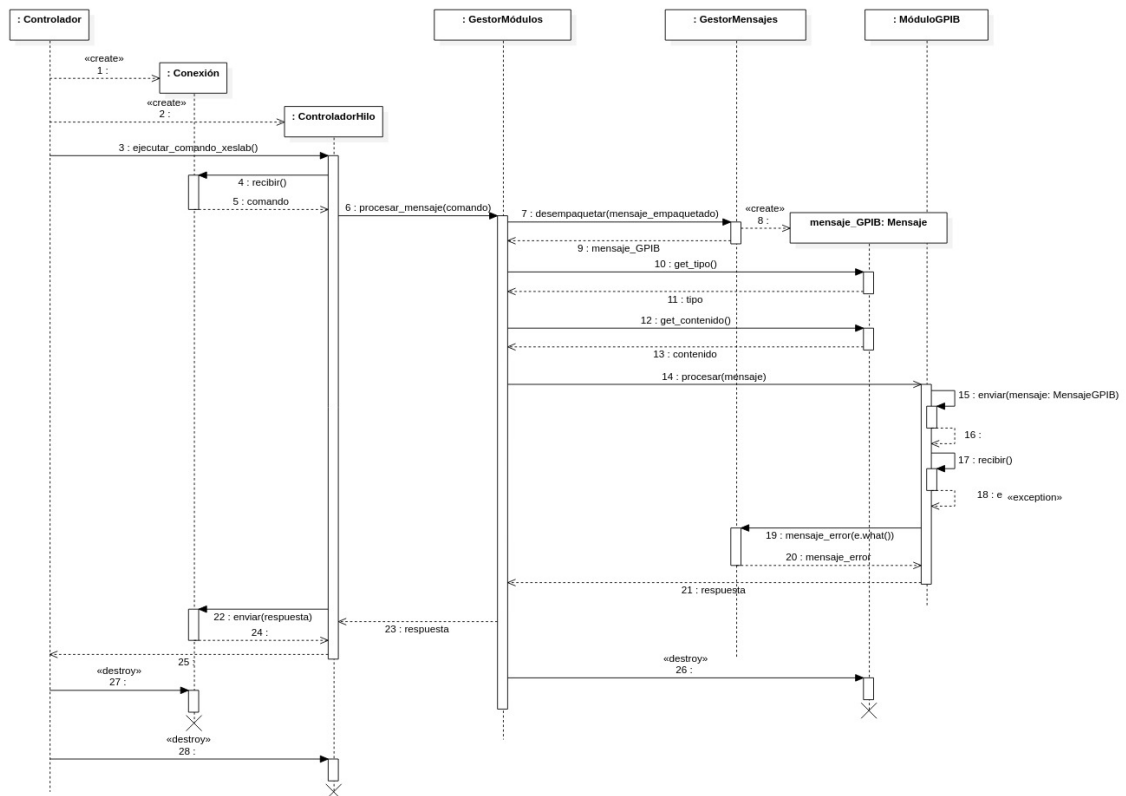


Figura 6.14: Erro GPIB

# Capítulo 7

## Verificación e validación

Neste capítulo cubriranse todos os aspectos relacionados coa verificación e a validación: o plan de probas, onde se establecen os criterios e liñas xerais para a elaboración das probas; o deseño das probas para cada módulo; a especificación de casos de proba concretos; e a documentación relacionada coa execución das probas, tanto os informes de execución como as incidencias atopadas durante o proceso. Tamén se incluírá un apartado onde se relata como se realiza a validación dende o punto de vista do usuario.

### 7.1. Plan de probas

#### 7.1.1. Restricións e enfoque

Dispónse dun límite de tempo de 3 meses para o proxecto, o que supón unha gran restrición temporal para a realización das probas. Debido a isto, é fundamental centrar os esforzos na validación dos elementos de código máis complexos, e obviar os máis triviais, como poden ser os getters ou os setters.

Este proxecto ten características especiais que aportan certa dificultade ás probas. As funcionalidades do software son bastante limitadas, pero o exercicio da funcionalidade máis esencial (enviar un comando a un equipo) require a integración de diversos subsistemas, que é o foco de defectos máis importante. Polo tanto, é lóxico adicar o maior esforzo de validación á integración entre sistemas. As probas de integración cobrarán importancia sobre as probas unitarias.

Deste xeito, plantéxase a división do sistema en tres apartados principais para a súa validación:

- **Módulos de comunicación no servidor:** probaranse todos os métodos do módulo de procesado de comandos e do módulo de preparación de comandos.
- **Comunicacións entre sistemas:** probarase a comunicación en ambos sentidos entre o servidor e a pasarela.

- **Software da pasarela:** crearanse mensaxes representativas de todos os posibles casos, para verificar que o software as xestiona correctamente.

### 7.1.2. Supostos e exclusións

Tendo en conta as limitacións mencionadas anteriormente, non se considera necesario realizar probas formais dos seguintes sistemas.

- Librarías de terceiros empregadas no software da pasarela
- Librarías estándar da linguaxe, tanto en PHP coma en C/C++
- Utilidades xerais de XesLab (por exemplo, os módulos de acceso a base de datos e de redes)
- Os métodos `PREPARACION_COMANDOS::PREPARAR_COMANDO_GPIB` e `PREPARACION_COMANDOS::PREPARAR_RESPUESTA_GPIB` de XesLab, xa que o seu correcto funcionamento se verifica noutros métodos da mesma clase.
- Os métodos `COMUNICACION_SOCKET_IP::CONEXION_SOCKET`, `COMUNICACION_SOCKET_IP::ENVIAR` e `COMUNICACION_SOCKET_IP::DESCONEXION_SOCKET`, xa que son esencialmente *wrappers* da librería de sockets de PHP.
- A xestión de fíos na pasarela, debido á dificultades asociadas a probar a concurrencia dos sistemas.

Si se realizaron probas informales durante o proceso de desenvolvemento e depuración nas que se comprobou que non existían fallos evidentes ou obvios.

### 7.1.3. Criterios de paso ou fallo

Debido a que o software forma parte dun proxecto en desenvolvemento para uso interno polo persoal do laboratorio, entre o que se atopan os desenvolvedores do sistema, existe unha maior marxe de aceptación, xa que calquera erro pode ser corrixido en etapas posteriores.



Porcentaxe de fallos	Aceptación do software
0 % - 10 %	Software aceptable, pasa as probas cun número de fallos razoable. Deben pasarse todas as probas nas que o contexto é un caso habitual, e non pode fallar ningunha das utilidades imprescindibles para a execución dos casos habituais (por exemplo, a comprobación do estado dos equipos).
10 % - 25 %	Software aceptable, pero que debe ser revisado tan pronto como sexa posible, debido á gran cantidade de defectos que mostra. Non poderá fallar ningunha das utilidades imprescindibles para a execución dos casos habituais.
>25 %	Software non aceptable, non pasa as probas xa que ten un número de fallos demasiado elevado.

## 7.2. Deseño de probas

### 7.2.1. Estratexia de construción

Debido a que as técnicas habituais de construción de probas dan lugar a unha gran cantidade de casos de proba e neste caso cóntase cun tempo moi limitado para a verificación, optouse por deseñar contextos de execución equivalentes a nivel de proba e dende un coñecemento profundo da aplicación; por exemplo, a utilidade que comproba o estado de rede dos equipos ten a mesma resposta se un equipo está conectado por Ethernet ou a través dunha pasarela, pero interiormente compórtase de forma distinta.

Este mesmo enfoque se aplicou ao deseñar as probas para a pasarela. Ao estar o seu funcionamento enteiramente determinado polas mensaxes que recibe, decidiuse deseñar unha serie de mensaxes representativas dos contextos de execución posibles e asociar esas mensaxes ás probas xeradas.

No tocante aos módulos de comunicación de XesLab, a pequena cantidade de unidades de código permitiu probar cada método independentemente.

### 7.2.2. Probas xeradas

Para a descrición das probas empregouse unha plantilla en forma de táboa cos seguintes campos.

- Número único de identificación da proba
- Elemento probado (método ou módulo)
- Obxectivo que se pretende acadar ca execución da proba

- Casos de proba derivados desa proba
- Criterios a partir dos cales se decide cando se pasa ou falla a proba
- Os diversos contextos de execución que afectan ao comportamento do elemento probado ou á súa resposta
- Comentarios, se aplica

### Módulo de procesamento de comandos

<b>Identificador</b>	<b>P-1</b>
<b>Elemento probado</b>	PROCESADO_COMANDOS::ESTA_ONLINE (\$OBJETO_EQUIPO), módulo de sockets, software da pasarela
<b>Obxectivo da proba</b>	Comprobar que se detecta correctamente o estado de conexión do equipo, tanto a través de Ethernet como de GPIB.
<b>Casos de proba</b>	CP-1, CP-2, CP-3, CP-4
<b>Criterios de paso e fallo</b>	Pasa a proba se pasan todos os seus casos de proba, e falla se non pasa algún deles.
<b>Contextos de execución</b>	<p>O equipo está online:</p> <ul style="list-style-type: none"> <li>▪ e conectado por Ethernet.</li> <li>▪ e conectado por GPIB, a través dunha pasarela.</li> </ul> <p>O equipo está offline:</p> <ul style="list-style-type: none"> <li>▪ porque non está conectado por Ethernet.</li> <li>▪ porque a súa pasarela está conectada por Ethernet, pero o equipo non está conectado á pasarela.</li> </ul>
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>P-2</b>
<b>Elemento probado</b>	PROCESADO_COMANDOS::TIPO_CONEXION (\$OBJETO_EQUIPO), módulo de sockets, software da pasarela
<b>Obxectivo da proba</b>	Comprobar que se detecta correctamente o tipo de conexión do equipo se está conectado a través dunha pasarela.
<b>Casos de proba</b>	CP-5, CP-6, CP-7
<b>Criterios de paso e fallo</b>	Pasa a proba se pasan todos os seus casos de proba, e falla se non pasa algún deles.
<b>Contextos de execución</b>	<ul style="list-style-type: none"> <li>▪ O equipo está conectado por GPIB.</li> <li>▪ O equipo non está conectado por GPIB.</li> </ul>
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>P-3</b>
<b>Elemento probado</b>	PROCESADO_COMANDOS::BLOQUEAR(\$OBJETO_EQUIPO, \$NOMBRE_ELEMENTO, \$ID_ELEMENTO)
<b>Obxectivo da proba</b>	Comprobar que o equipo se bloquea correctamente se está nas condicións adecuadas, e se mostra un erro se non.
<b>Casos de proba</b>	CP-8, CP-9, CP-10
<b>Criterios de paso e fallo</b>	Pasa a proba se pasa todos os casos de proba, e falla se falla algún deles.
<b>Contextos de execución</b>	<p>O equipo está no dereito de ser bloqueado pola entidade que o solicita:</p> <ul style="list-style-type: none"> <li>▪ porque non está bloqueado por ninguén.</li> <li>▪ porque xa está bloqueado por esa mesma entidade.</li> </ul> <p>O equipo non pode ser bloqueado pola entidade que o solicita se xa foi bloqueado por outra entidade.</p>
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>P-4</b>
<b>Elemento probado</b>	PROCESADO_COMANDOS::DESBLOQUEAR(\$OBJETO_EQUIPO, \$NOMBRE_ELEMENTO, \$ID_ELEMENTO)
<b>Obxectivo da proba</b>	Comprobar que só se unha entidade ten un equipo bloqueado pode desbloquealo.
<b>Casos de proba</b>	CP-11, CP-12, CP-13
<b>Criterios de paso e fallo</b>	Pasa a proba se pasa todos os casos de proba, e falla se falla algún deles.
<b>Contextos de execución</b>	<p>O equipo pode ser desbloqueado se estaba bloqueado pola mesma entidade que solicita o desbloqueo.</p> <p>O equipo non pode ser desbloqueado se:</p> <ul style="list-style-type: none"> <li>▪ non estaba bloqueado.</li> <li>▪ estaba bloqueado por outra entidade que non é a que solicita o desbloqueo.</li> </ul>
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>P-5</b>
<b>Elemento probado</b>	PROCESADO_COMANDOS::ESTA_BLOQUEADO_POR_ELEMENTO(\$OBJETO_EQUIPO, \$NOMBRE_ELEMENTO, \$ID_ELEMENTO)
<b>Obxectivo da proba</b>	Comprobar que se detecta correctamente se un equipo concreto está bloqueado por unha entidade concreta.
<b>Casos de proba</b>	CP-14, CP-15
<b>Criterios de paso e fallo</b>	Pasa a proba se pasa todos os casos de proba, e falla se falla algún deles.
<b>Contextos de execución</b>	<ul style="list-style-type: none"> <li>▪ O equipo está bloqueado por esa entidade.</li> <li>▪ O equipo non está bloqueado por esa entidade.</li> </ul>
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>P-6</b>
<b>Elemento probado</b>	PROCESADO_COMANDOS::ESTA_BLOQUEADO (\$OBJETO_EQUIPO)
<b>Obxectivo da proba</b>	Comprobar que se detecta correctamente se un equipo concreto está bloqueado por unha entidade concreta.
<b>Casos de proba</b>	CP-16, CP-17
<b>Criterios de paso e fallo</b>	Pasa a proba se pasan todos os casos de proba, e falla a proba en caso contrario.
<b>Contextos de execución</b>	<ul style="list-style-type: none"><li>▪ O equipo está bloqueado.</li><li>▪ O equipo non está bloqueado.</li></ul>
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>P-7</b>
<b>Elemento probado</b>	PROCESADO_COMANDOS::ENVIAR_COMANDO (\$NOMBRE_ELEMENTO, \$ID_ELEMENTO, \$OBJETO_EQUIPO, \$OBJETO_COMANDO, \$VARIABLES)
<b>Obxectivo da proba</b>	Comprobar que os comandos se envían correctamente nas condicións adecuadas, e se notifican os erros ao invocador se hai algún problema.
<b>Casos de proba</b>	CP-18, CP-19, CP-20, CP-21, CP-22, CP-23, CP-24, CP-25
<b>Criterios de paso e fallo</b>	Pasa a proba se pasa todos os seus casos de proba. Falla en caso contrario.
<b>Contextos de execución</b>	<p>Enviáse con éxito un comando:</p> <ul style="list-style-type: none"> <li>▪ sen resposta.</li> <li>▪ con resposta.</li> <li>▪ a un equipo que non está bloqueado pola entidade que quere enviar o comando, polo que o equipo se bloquea consigo mesmo.</li> </ul> <p>Non se pode enviar o comando ou recibir a resposta porque:</p> <ul style="list-style-type: none"> <li>▪ o equipo está bloqueado por unha entidade diferente.</li> <li>▪ o equipo non está online.</li> <li>▪ o código que prepara o comando para o envío non é válido.</li> <li>▪ non está rexistrado o tipo de resposta.</li> <li>▪ sucede un erro na rede que impide recibir a resposta.</li> <li>▪ o código que prepara a resposta para ser interpretada non é válido.</li> </ul>
<b>Comentarios</b>	As clases de equivalencia derivadas dos distintos tipos de mensaxe que pode implementar a pasarela refléxanse na sección relativa ás probas da pasarela.

## Módulo de preparación de comandos

<b>Identificador</b>	<b>P-8</b>
<b>Elemento probado</b>	PREPARACION_COMANDOS::PREPARAR_COMANDO (\$OBJETO_COMANDO, \$OBJETO_EQUIPO)
<b>Obxectivo da proba</b>	Comprobar que se prepara o comando correctamente para a súa interpretación en destino, independentemente da conexión hardware.
<b>Casos de proba</b>	CP-26, CP-27
<b>Criterios de paso e fallo</b>	Pasa a proba se pasan todos os seus casos de proba, e falla en caso contrario
<b>Contextos de execución</b>	<ul style="list-style-type: none"> <li>▪ O equipo está conectado por Ethernet.</li> <li>▪ O equipo está conectado por GPIB.</li> </ul>
<b>Comentarios</b>	Non se contempla o caso de que o equipo non estea conectado, xa que esta comprobación se realiza previamente na proba de ESTA_ONLINE.

<b>Identificador</b>	<b>P-9</b>
<b>Elemento probado</b>	PREPARACION_COMANDOS::PREPARAR_RESPUESTA (\$STRING_RESPUESTA, \$OBJETO_EQUIPO)
<b>Obxectivo da proba</b>	Comprobar que se extrae o contido da resposta correctamente, independentemente da conexión hardware.
<b>Casos de proba</b>	CP-28, CP-29
<b>Criterios de paso e fallo</b>	Pasa a proba se pasa todos os casos de proba, falla en caso contrario.
<b>Contextos de execución</b>	<ul style="list-style-type: none"> <li>▪ O equipo está conectado por Ethernet.</li> <li>▪ O equipo está conectado por GPIB.</li> </ul>
<b>Comentarios</b>	Non se contempla o caso de que o equipo non estea conectado, xa que esta comprobación se realiza previamente na proba de ESTA_ONLINE.

## Módulo de sockets

<b>Identificador</b>	<b>P-10</b>
<b>Elemento probado</b>	COMUNICACION_SOCKET_IP: :RECIBIR(&\$\$SOCKET)
<b>Obxectivo da proba</b>	Comprobar que se reciben correctamente as mensaxes enviadas por TCP.
<b>Casos de proba</b>	CP-30, CP-31, CP-32
<b>Criterios de paso e fallo</b>	Pasa a proba se pasa todos os seus casos de proba, e falla en caso contrario.
<b>Contextos de execución</b>	<ul style="list-style-type: none"> <li>▪ A mensaxe recíbese correctamente, e non excede o tamaño do <i>buffer</i>.</li> <li>▪ A mensaxe recíbese correctamente, e excede o tamaño do <i>buffer</i>.</li> <li>▪ Expira o <i>time-out</i> durante a recepción da mensaxe.</li> </ul>
<b>Comentarios</b>	Ningún

Cadro 7.1: Probas xeradas para os módulos de comunicación de XesLab

## Software da pasarela

Debido á gran cantidade de capas polas que pasan as mensaxes unha vez chegadas á pasarela, e o tempo limitado do que se dispón para a verificación, considerouse que o máis adecuado non era probar cada método unitariamente senón valorar os contextos de execución que ofrecen os distintos tipos de mensaxes e crear casos de proba de acordo a eses contextos. Ditos contextos son os seguintes.

Situación	Resposta que chega ao servidor	Tipo de contexto
<b>Erros de rede</b>		
Erro ao establecer a conexión co servidor	Mensaxe de control: erro de rede	Erro
Erro ao enviar unha mensaxe ao servidor		Erro
Erro ao recibir unha mensaxe do servidor		Erro



<b>Mensaxes válidas</b>		
Mensaxe baleira	Ningunha	Normal
*IDN? en texto plano	PASARELA_TELEVES	Normal
Array JSON, serializado e codificado, cun enderezo GPIB válido, resposta con valor <b>true</b> , tipo con valor GPIB, e contido con valor XESLAB_ALIVE	Array JSON, serializado e codificado, co mesmo enderezo GPIB que o comando, tipo con valor GPIB, e contido con valor <b>true</b> se hai un equipo operativo nese enderezo e <b>false</b> en caso contrario	Normal
Array JSON, serializado e codificado, cun enderezo GPIB válido, resposta con valor <b>true</b> , tipo con valor GPIB, e contido cun comando interpretable polo equipo e ao que o equipo responda	Array JSON, serializado e codificado, co mesmo enderezo GPIB que o comando, tipo con valor GPIB, e contido con valor igual á resposta que dera o equipo	Normal
Array JSON, serializado e codificado, cun enderezo GPIB válido, resposta con valor <b>false</b> , tipo con valor GPIB, e contido cun comando interpretable polo equipo que non teña unha resposta asociada	Array JSON, serializado e codificado, co mesmo enderezo GPIB que o comando, tipo con valor GPIB, e contido con valor OK	Normal
<b>Erros de parse</b>		
<i>String</i> que non é a cadea vacía, o comando de identificación, ou o resultado de serializar e codificar un obxecto JSON (por exemplo: <i>outro_string</i> )	Mensaxe de control: erro de <i>parse</i>	Erro
Array JSON, serializado e codificado, cun campo tipo que non se corresponde con ningún dos valores posibles		Erro
Array JSON, serializado e codificado, cun campo tipo con valor GPIB, pero que non ten algún dos campos resposta, contido ou <i>direccion_equipo</i>		Erro
Array JSON, serializado e codificado, cun campo tipo cun valor posible pero que é un pseudo-tipo (por exemplo: TIPO_NO_VALIDO)		Erro

Parámetros incorrectos		
Array JSON, serializado e codificado, co campo tipo con valor GPIB e cun enderezo GPIB non válido	Mensaxe de control: erro GPIB	Erro
Array JSON, serializado e codificado, co campo tipo con valor GPIB, cun enderezo GPIB válido e cun comando non válido para o equipo		Erro
Array JSON, serializado e codificado, co campo tipo con valor GPIB, cun enderezo GPIB válido, cun comando sen resposta válido para o equipo e co campo respuesta con valor <b>true</b> <sup>1</sup>		Erro

Cadro 7.2: Cadro de clasificación de respostas aos mensaxes enviadas á pasarela.

A táboa 7.2.2 mostra a única proba xerada para a pasarela, que engloba todos os contextos mencionados no cadro anterior.

### 7.3. Casos de proba

Os casos de proba detállanse usando unha plantilla en forma de táboa cos seguintes campos.

- Número identificador do caso de proba dentro da lista
- Proba que deu lugar á creación dese caso de proba
- Método probado
- Entradas proporcionadas
- Saída esperada para esa entrada
- Requisitos da contorna na que se executa o caso de proba
- Comentarios, se aplica

<sup>1</sup>O caso contrario, no que o comando espera resposta pero o parámetro respuesta é **false**, non constitúe un erro para a pasarela, xa que simplemente ignora a resposta do equipo. Este erro debe ser comprobado no servidor.

<b>Identificador</b>	<b>P-11</b>
<b>Elemento probado</b>	Software da pasarela
<b>Obxectivo da proba</b>	Comprobar que a pasarela devolve a mensaxe correcta para cada contexto
<b>Casos de proba</b>	CP-33, CP-34, CP-35, CP-36, CP-37, CP-38, CP-39, CP-40, CP-41, CP-42, CP-43, CP-44, CP-45, CP-46, CP-47
<b>Criterios de paso e fallo</b>	Pasa a proba se pasa todos os seus casos de proba, e falla en caso contrario.
<b>Contextos de execución</b>	<ul style="list-style-type: none"> <li>■ Non se pode establecer a conexión co servidor de XesLab.</li> <li>■ Non se pode enviar unha mensaxe ao servidor.</li> <li>■ Establécese unha conexión co servidor, pero este non envía ningunha mensaxe.</li> <li>■ Recíbese unha mensaxe baleira (lonxitude cero).</li> <li>■ Recíbese un comando de identificación sen empacotar.</li> <li>■ Recíbese unha consulta sobre o estado dun equipo, e o equipo está online.</li> <li>■ Recíbese unha consulta sobre o estado dun equipo, e o equipo está offline.</li> <li>■ Recíbese un comando (que non require unha resposta) para un equipo conectado.</li> <li>■ Recíbese un comando (que require unha resposta) para un equipo conectado.</li> <li>■ Recíbese un <i>string</i> de texto aleatorio; non é unha mensaxe especial, nin unha mensaxe normal correctamente encapsulada.</li> <li>■ A mensaxe non ten un tipo válido.</li> <li>■ Falta algún dos campos dunha mensaxe normal.</li> <li>■ O tipo da mensaxe é un pseudo-tipo.</li> <li>■ O enderezo GPIB do equipo ao que vai dirixido a mensaxe non é válido.</li> <li>■ O contido da mensaxe non é recoñecido polo seu equipo destinatario.</li> <li>■ A mensaxe indica que se debe esperar por unha resposta, pero o contido non é un comando con resposta.</li> </ul>
<b>Comentarios</b>	Ningún

Cadro 7.3: Proba xerada para a pasarela

## 7.3.1. Casos de proba de XesLab

<b>Identificador</b>	<b>CP-1</b>
<b>Proba</b>	P-1
<b>Descrición</b>	O equipo está online e conectado por Ethernet.
<b>Método</b>	PROCESADO_COMANDOS::ESTA_ONLINE (\$OBJETO_EQUIPO)
<b>Entradas</b>	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>▪ Id: 522</li> <li>▪ RED["ESTADO"]: ONLINE</li> </ul> <p>Ademais, ao enviarlle un comando *IDN? a resposta é distinta de PASARELA_TELEVES.</p>
<b>Saídas</b>	<b>true</b>
<b>Requisitos da contorna</b>	O equipo con identificador 522 (MXG Vector Signal Generator) debe estar acendido e conectado á rede local.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-2</b>
<b>Proba</b>	P-1
<b>Descrición</b>	O equipo está offline.
<b>Método</b>	PROCESADO_COMANDOS::ESTA_ONLINE (\$OBJETO_EQUIPO)
<b>Entradas</b>	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>▪ Id: 522</li> <li>▪ RED["ESTADO"]: OFFLINE</li> </ul>
<b>Saídas</b>	<b>false</b>
<b>Requisitos da contorna</b>	O equipo con identificador 522 (MXG Vector Signal Generator) debe estar apagado ou desconectado da rede local.
<b>Comentarios</b>	Este caso de proba ten o mesmo comportamento se o equipo é unha pasarela.

<b>Identificador</b>	<b>CP-3</b>
<b>Proba</b>	P-1
<b>Descrición</b>	O equipo está online e conectado por GPIB a unha pasarela.
<b>Método</b>	PROCESADO_COMANDOS::ESTA_ONLINE (\$OBJETO_EQUIPO)
<b>Entradas</b>	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>■ Id: 522</li> <li>■ RED["ESTADO"]: ONLINE</li> </ul> <p>Ademais, ao enviarlle un comando *IDN? a resposta é PASARELA_TELEVES. PROCESADO_COMANDOS::TIPO_CONEXION devolve GPIB para este equipo.</p>
<b>Saídas</b>	<b>true</b>
<b>Requisitos da contorna</b>	O equipo con identificador 522 (MXG Vector Signal Generator) debe estar acendido e conectado por GPIB a unha pasarela, e esta debe estar acendida e conectada á rede local.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-4</b>
<b>Proba</b>	P-1
<b>Descrición</b>	A pasarela está conectada á rede, pero o equipo non está conectado a esta.
<b>Método</b>	PROCESADO_COMANDOS::ESTA_ONLINE (\$OBJETO_EQUIPO)
<b>Entradas</b>	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>■ Id: 522</li> <li>■ RED["ESTADO"]: ONLINE</li> </ul> <p>Ademais, ao enviarlle un comando *IDN? a resposta é PASARELA_TELEVES. PROCESADO_COMANDOS::TIPO_CONEXION devolve null para este equipo.</p>
<b>Saídas</b>	<b>false</b>
<b>Requisitos da contorna</b>	O equipo con identificador 522 (MXG Vector Signal Generator) debe estar apagado ou desconectado dunha pasarela, e esta debe estar acendida e conectada á rede local.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-5</b>
<b>Proba</b>	P-2
<b>Descrición</b>	O equipo está conectado a unha pasarela por GPIB.
<b>Método</b>	PROCESADO_COMANDOS::TIPO_CONEXION (\$OBJETO_EQUIPO)
<b>Entradas</b>	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>▪ Id: 522</li> <li>▪ RED["ESTADO"]: ONLINE</li> <li>▪ GPIB_Conexion: 19</li> </ul> <p>Ademais, ao enviarlle un comando *IDN? a resposta é PASARELA_TELEVES.</p> <p>Ao enviar a mensaxe tipo: GPIB, direccion_equipo: \$EQUIPO-&gt;GPIB_Direccion, contenido: "XESLAB_ALIVE", tipo_respuesta: <b>true</b> a resposta é tipo: GPIB, direccion_equipo: \$EQUIPO-&gt;GPIB_Direccion, contenido: <b>true</b>, tipo_respuesta: <b>false</b></p>
<b>Saídas</b>	GPIB
<b>Requisitos da contorna</b>	O equipo con identificador 522 (MXG Vector Signal Generator) debe estar acendido e conectado a unha pasarela, e esta debe estar acendida e conectada á rede local.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-6</b>
<b>Proba</b>	P-2
<b>Descrición</b>	O equipo non ten enderezo secundario (enderezo GPIB).
<b>Método</b>	PROCESADO_COMANDOS::TIPO_CONEXION (\$OBJETO_EQUIPO)
<b>Entradas</b>	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>▪ Id: 522</li> <li>▪ RED["ESTADO"]: ONLINE</li> <li>▪ GPIB_Conexion: null (non existe)</li> </ul> <p>Ademais, ao enviarlle un comando *IDN? a resposta é PASARELA_TELEVES.</p> <p>Ao enviar a mensaxe tipo: GPIB, direccion_equipo: \$EQUIPO-&gt;GPIB_Direccion, contenido: "XESLAB_ALIVE", tipo_respuesta: <b>true</b> a resposta é tipo: GPIB, direccion_equipo: \$EQUIPO-&gt;GPIB_Direccion, contenido: <b>false</b>, tipo_respuesta: <b>false</b></p>
<b>Saídas</b>	null
<b>Requisitos da contorna</b>	O equipo con identificador 522 (MXG Vector Signal Generator) debe estar acendido e conectado a unha pasarela, e esta debe estar acendida e conectada á rede local.
<b>Comentarios</b>	Ningún

Identificador	CP-7
Proba	P-2
Descrición	O equipo non está conectado á pasarela.
Método	PROCESADO_COMANDOS::TIPO_CONEXION (\$OBJETO_EQUIPO)
Entradas	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>■ Id: 522</li> <li>■ RED["ESTADO"]: ONLINE</li> <li>■ GPIB_Conexion: 19</li> </ul> <p>Ademais, ao enviarlle un comando *IDN? a resposta é PASARELA_TELEVES.</p> <p>Ao enviar a mensaxe tipo: GPIB, direccion_equipo: \$EQUIPO-&gt;GPIB_Direccion, contenido: "XESLAB_ALIVE", tipo_respuesta: <b>true</b> a resposta é tipo: GPIB, direccion_equipo: \$EQUIPO-&gt;GPIB_Direccion, contenido: <b>false</b>, tipo_respuesta: <b>false</b></p>
Saídas	null
Requisitos da contorna	O equipo con identificador 522 (MXG Vector Signal Generator) debe estar apagado ou desconectado dunha pasarela, e esta debe estar acendida e conectada á rede local.
Comentarios	Ningún

Identificador	CP-8
Proba	P-3
Descrición	Bloquéase un equipo que estaba libre.
Método	PROCESADO_COMANDOS::BLOQUEAR(\$OBJETO_EQUIPO, \$NOMBRE_ELEMENTO, \$ID_ELEMENTO)
Entradas	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>■ Id: 522</li> </ul> <p>\$NOMBRE_ELEMENTO con valor EQUIPO. \$ID_ELEMENTO con valor \$EQUIPO-&gt;Id.</p>
Saídas	<b>true.</b>
Requisitos da contorna	<p>Antes da execución, <b>non</b> hai na táboa BLOQUEOS da base de datos un rexistro que contén o valor \$EQUIPO-&gt;Id na columna Id_Equipo.</p> <p>Despois da execución, hai un rexistro na táboa BLOQUEOS da base de datos que contén o valor \$NOMBRE_ELEMENTO na columna Elemento, \$ID_ELEMENTO na columna Id_Elemento e \$EQUIPO-&gt;Id na columna Id_Equipo.</p>
Comentarios	Despois da execución, PROCESADO_COMANDOS::ESTA_BLOQUEADO_POR_ELEMENTO(\$OBJETO_EQUIPO, \$NOMBRE_ELEMENTO, \$ID_ELEMENTO) devolve <b>true</b> .

<b>Identificador</b>	<b>CP-9</b>
<b>Proba</b>	P-3
<b>Descrición</b>	Bloquéase un equipo que xa estaba bloqueado por ese elemento (éxito silencioso).
<b>Método</b>	PROCESADO_COMANDOS::BLOQUEAR(\$OBJETO_EQUIPO, \$NOMBRE_ELEMENTO, \$ID_ELEMENTO)
<b>Entradas</b>	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>▪ Id: 522</li> </ul> <p>\$NOMBRE_ELEMENTO con valor EQUIPO. \$ID_ELEMENTO con valor \$EQUIPO-&gt;Id.</p>
<b>Saídas</b>	<b>true</b>
<b>Requisitos da contorna</b>	<p>Antes da execución, hai un rexistro na táboa BLOQUEOS da base de datos que contén o valor \$NOMBRE_ELEMENTO na columna Elemento, ID_ELEMENTO na columna Id_Elemento e \$EQUIPO-&gt;Id na columna Id_Equipo. Despois da execución, segue existindo ese rexistro.</p>
<b>Comentarios</b>	<p>Antes e despois da execución, PROCESADO_COMANDOS::ESTA_BLOQUEADO _POR_ELEMENTO(\$OBJETO_EQUIPO, \$NOMBRE_ELEMENTO, \$ID_ELEMENTO) devolve <b>true</b>.</p>

<b>Identificador</b>	<b>CP-10</b>
<b>Proba</b>	P-3
<b>Descrición</b>	Inténtase bloquear un equipo que está bloqueado por outro elemento.
<b>Método</b>	PROCESADO_COMANDOS::BLOQUEAR(\$OBJETO_EQUIPO, \$NOMBRE_ELEMENTO, \$ID_ELEMENTO)
<b>Entradas</b>	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>▪ Id: 522</li> </ul> <p>\$NOMBRE_ELEMENTO con valor EQUIPO. \$ID_ELEMENTO con valor \$EQUIPO-&gt;Id.</p>
<b>Saídas</b>	<b>false</b>
<b>Requisitos da contorna</b>	<p>Antes da execución, hai un rexistro na táboa BLOQUEOS da base de datos que contén o valor EQUIPO na columna Elemento, 492 na columna Id_Elemento e 522 na columna Id_Equipo. Despois da execución, segue existindo ese rexistro.</p>
<b>Comentarios</b>	Ningún



<b>Identificador</b>	<b>CP-11</b>
<b>Proba</b>	P-4
<b>Descrición</b>	Desbloquéase un equipo que estaba bloqueado por ese elemento.
<b>Método</b>	PROCESADO_COMANDOS::DESBLOQUEAR(\$OBJETO_EQUIPO, \$NOMBRE_ELEMENTO, \$ID_ELEMENTO)
<b>Entradas</b>	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>■ Id: 522</li> </ul> <p>\$NOMBRE_ELEMENTO con valor EQUIPO. \$ID_ELEMENTO con valor \$EQUIPO-&gt;Id.</p>
<b>Saídas</b>	<b>true</b>
<b>Requisitos da contorna</b>	<p>Antes da execución, hai un rexistro na táboa BLOQUEOS da base de datos que contén o valor \$NOMBRE_ELEMENTO na columna Elemento, ID_ELEMENTO na columna Id_Elemento e \$EQUIPO-&gt;Id na columna Id_Equipo.</p> <p>Despois da execución, xa non existe ese rexistro.</p>
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-12</b>
<b>Proba</b>	P-4
<b>Descrición</b>	O desbloqueo falla porque o equipo está bloqueado por outro elemento.
<b>Método</b>	PROCESADO_COMANDOS::DESBLOQUEAR(\$OBJETO_EQUIPO, \$NOMBRE_ELEMENTO, \$ID_ELEMENTO)
<b>Entradas</b>	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>■ Id: 522</li> </ul> <p>\$NOMBRE_ELEMENTO con valor EQUIPO. \$ID_ELEMENTO con valor \$EQUIPO-&gt;Id.</p>
<b>Saídas</b>	<b>false</b>
<b>Requisitos da contorna</b>	<p>Antes da execución, <b>non</b> hai na táboa BLOQUEOS da base de datos un rexistro que contén o valor \$EQUIPO-&gt;Id na columna Id_Equipo, pero hai un rexistro que contén o valor EQUIPO na columna Elemento, 492 na columna Id_Elemento e 492 na columna Id_Equipo.</p> <p>Despois da execución, o primeiro rexistro segue sen existir e o segundo segue existindo.</p>
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-13</b>
<b>Proba</b>	P-4
<b>Descrición</b>	O desbloqueo falla porque o equipo está libre.
<b>Método</b>	PROCESADO_COMANDOS: :DESBLOQUEAR(\$OBJETO_EQUIPO, \$NOMBRE_ELEMENTO, \$ID_ELEMENTO)
<b>Entradas</b>	Obxecto \$EQUIPO con campos: <ul style="list-style-type: none"> <li>▪ Id: 522</li> </ul> \$NOMBRE_ELEMENTO con valor EQUIPO. \$ID_ELEMENTO con valor \$EQUIPO->Id.
<b>Saídas</b>	<b>false</b>
<b>Requisitos da contorna</b>	Antes da execución, <b>non</b> hai na táboa BLOQUEOS da base de datos un rexistro que contén o valor \$EQUIPO->Id na columna Id_Equipo. Despois da execución, segue sen existir.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-14</b>
<b>Proba</b>	P-5
<b>Descrición</b>	O equipo está bloqueado por ese elemento.
<b>Método</b>	PROCESADO_COMANDOS: :ESTA_BLOQUEADO_POR _ELEMENTO(\$OBJETO_EQUIPO, \$NOMBRE_ELEMENTO, \$ID_ELEMENTO)
<b>Entradas</b>	Obxecto \$EQUIPO con campos: <ul style="list-style-type: none"> <li>▪ Id: 522</li> </ul> \$NOMBRE_COMANDO con valor EQUIPO. \$ID_ELEMENTO con valor 522.
<b>Saídas</b>	<b>true</b>
<b>Requisitos da contorna</b>	Antes da execución, hai un rexistro na táboa BLOQUEOS da base de datos que contén o valor \$NOMBRE_ELEMENTO na columna Elemento, \$ID_ELEMENTO na columna Id_Elemento e \$EQUIPO->Id na columna Id_Equipo.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-15</b>
<b>Proba</b>	P-5
<b>Descrición</b>	O equipo non está bloqueado por ese elemento.
<b>Método</b>	PROCESADO_COMANDOS::ESTA_BLOQUEADO_POR _ELEMENTO(\$OBJETO_EQUIPO, \$NOMBRE_ELEMENTO, \$ID_ELEMENTO)
<b>Entradas</b>	Obxecto \$EQUIPO con campos: <ul style="list-style-type: none"> <li>▪ Id: 522</li> </ul> \$NOMBRE_ELEMENTO con valor EQUIPO. \$ID_ELEMENTO con valor 522.
<b>Saídas</b>	<b>false</b>
<b>Requisitos da contorna</b>	Antes da execución, non hai un rexistro na táboa BLOQUEOS da base de datos que contén o valor \$NOMBRE_ELEMENTO na columna Elemento, \$ID_ELEMENTO na columna Id_Elemento e \$EQUIPO->Id na columna Id_Equipo.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-16</b>
<b>Proba</b>	P-6
<b>Descrición</b>	O equipo está bloqueado por algún elemento.
<b>Método</b>	PROCESADO_COMANDOS::ESTA_BLOQUEADO (\$OBJETO_EQUIPO)
<b>Entradas</b>	Obxecto \$EQUIPO con campos: <ul style="list-style-type: none"> <li>▪ Id: 522</li> </ul>
<b>Saídas</b>	<b>true</b>
<b>Requisitos da contorna</b>	Antes da execución, hai na táboa BLOQUEOS da base de datos un rexistro que contén o valor \$EQUIPO->Id na columna Id_Equipo.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-17</b>
<b>Proba</b>	P-6
<b>Descrición</b>	O equipo non está bloqueado por ningún elemento.
<b>Método</b>	PROCESADO_COMANDOS::ESTA_BLOQUEADO (\$OBJETO_EQUIPO)
<b>Entradas</b>	Obxecto \$EQUIPO con campos: <ul style="list-style-type: none"> <li>▪ Id: 522</li> </ul>
<b>Saídas</b>	<b>false</b>
<b>Requisitos da contorna</b>	Antes da execución, non hai na táboa BLOQUEOS da base de datos un rexistro que contén o valor \$EQUIPO->Id na columna Id_Equipo.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-18</b>
<b>Proba</b>	P-7
<b>Descrición</b>	Enviar un comando sen resposta a un equipo, con éxito
<b>Método</b>	PROCESADO_COMANDOS::ENVIAR_COMANDO (\$NOMBRE_ELEMENTO, \$ID_ELEMENTO, \$OBJETO_EQUIPO, \$OBJETO_COMANDO, \$VARIABLES)
<b>Entradas</b>	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>▪ Id: 522</li> <li>▪ RED["ESTADO"]: ONLINE</li> </ul> <p>Ademais, ao enviarlle un comando *IDN? a resposta é distinta de PASARELA_TELEVES.</p> <p>Obxecto \$COMANDO con campos:</p> <ul style="list-style-type: none"> <li>▪ Id: 2</li> <li>▪ Comando_PHP: \$COMANDO_ENVIAR='FREQ_'. \$FRECUENCIA;</li> <li>▪ Respuesta_Tipo: SIN RESPUESTA</li> </ul> <p>\$NOMBRE_ELEMENTO con valor EQUIPO. \$ID_ELEMENTO con valor \$EQUIPO-&gt;Id.</p>
<b>Saídas</b>	A cadea vacía
<b>Requisitos da contorna</b>	<p>Antes da execución, hai un rexistro na táboa BLOQUEOS da base de datos que contén o valor \$NOMBRE_ELEMENTO na columna Elemento, \$ID_ELEMENTO na columna Id_Elemento e \$EQUIPO-&gt;Id na columna Id_Equipo.</p> <p>O equipo con identificador 522 (MXG Vector Signal Generator) debe estar acendido e conectado á rede local.</p>
<b>Comentarios</b>	Este caso de proba é independente de se o equipo está conectado directamente á rede ou a través da pasarela.

<b>Identificador</b>	<b>CP-19</b>
<b>Proba</b>	P-7
<b>Descrición</b>	Enviar un comando con resposta a un equipo, con éxito
<b>Método</b>	PROCESADO_COMANDOS::ENVIAR_COMANDO (\$NOMBRE_ELEMENTO, \$ID_ELEMENTO, \$OBJETO_EQUIPO, \$OBJETO_COMANDO, \$VARIABLES)
<b>Entradas</b>	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>▪ Id: 522</li> <li>▪ RED["ESTADO"]: ONLINE</li> </ul> <p>Ademais, ao enviarlle un comando *IDN? a resposta é distinta de PASARELA_TELEVES.</p> <p>Obxecto \$COMANDO con campos:</p> <ul style="list-style-type: none"> <li>▪ Id: 2</li> <li>▪ Comando_PHP: \$COMANDO_ENVIAR='*IDN?';</li> <li>▪ Respuesta_Tipo: STRING</li> <li>▪ Respuesta_Codigo: \$RESPUESTA=\$RESPUESTA_SIN_PROCESAR;</li> </ul> <p>\$NOMBRE_ELEMENTO con valor EQUIPO. \$ID_ELEMENTO con valor \$EQUIPO-&gt;Id.</p>
<b>Saídas</b>	Agilent Technologies, N5182B, MY53051062, B.01.58
<b>Requisitos da contorna</b>	<p>Antes da execución, hai un rexistro na táboa BLOQUEOS da base de datos que contén o valor \$NOMBRE_ELEMENTO na columna Elemento, \$ID_ELEMENTO na columna Id_Elemento e \$EQUIPO-&gt;Id na columna Id_Equipo.</p> <p>O equipo con identificador 522 (MXG Vector Signal Generator) debe estar acendido e conectado á rede local.</p>
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-20</b>
<b>Proba</b>	P-7
<b>Descrición</b>	Enviar un comando sen resposta a un equipo que non está bloqueado por ningunha entidade
<b>Método</b>	PROCESADO_COMANDOS::ENVIAR_COMANDO (\$NOMBRE_ELEMENTO, \$ID_ELEMENTO, \$OBJETO_EQUIPO, \$OBJETO_COMANDO, \$VARIABLES)
<b>Entradas</b>	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>▪ Id: 522</li> <li>▪ RED["ESTADO"]: ONLINE</li> </ul> <p>Ademais, ao enviarlle un comando *IDN? a resposta é distinta de PASARELA_TELEVES.</p> <p>Obxecto \$COMANDO con campos:</p> <ul style="list-style-type: none"> <li>▪ Id: 2</li> <li>▪ Comando_PHP: \$COMANDO_ENVIAR='FREQ_'. \$FRECUENCIA;</li> <li>▪ Respuesta_Tipo: SIN RESPUESTA</li> </ul> <p>\$NOMBRE_ELEMENTO con valor EQUIPO. \$ID_ELEMENTO con valor \$EQUIPO-&gt;Id.</p>
<b>Saídas</b>	null
<b>Requisitos da contorna</b>	<p>Antes da execución, non hai un rexistro na táboa BLOQUEOS da base de datos que contén o valor \$EQUIPO-&gt;Id na columna Id_Equipo.</p> <p>O equipo con identificador 522 (MXG Vector Signal Generator) debe estar acendido e conectado á rede local.</p>
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-21</b>
<b>Proba</b>	P-7
<b>Descrición</b>	Inténtase enviar un comando a un equipo que non está online
<b>Método</b>	PROCESADO_COMANDOS::ENVIAR_COMANDO (\$NOMBRE_ELEMENTO, \$ID_ELEMENTO, \$OBJETO_EQUIPO, \$OBJETO_COMANDO, \$VARIABLES)
<b>Entradas</b>	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>▪ Id: 522</li> <li>▪ RED["ESTADO"]: OFFLINE</li> </ul> <p>Obxecto \$COMANDO con campos:</p> <ul style="list-style-type: none"> <li>▪ Id: 2</li> <li>▪ Comando_PHP: \$COMANDO_ENVIAR='FREQ_'. \$FRECUENCIA;</li> <li>▪ Respuesta_Tipo: SIN RESPUESTA</li> </ul> <p>\$NOMBRE_ELEMENTO con valor EQUIPO. \$ID_ELEMENTO con valor \$EQUIPO-&gt;Id.</p>
<b>Saídas</b>	EQUIPO Generador de Senal MXG X-Series OFFLINE
<b>Requisitos da contorna</b>	<p>Antes da execución, hai un rexistro na táboa BLOQUEOS da base de datos que contén o valor \$NOMBRE_ELEMENTO na columna Elemento, \$ID_ELEMENTO na columna Id_Elemento e \$EQUIPO-&gt;Id na columna Id_Equipo.</p> <p>O equipo con identificador 522 (MXG Vector Signal Generator) debe estar apagado ou desconectado da rede local.</p>
<b>Comentarios</b>	Tamén é válido se o equipo está desconectado da súa pasarela, xa que depende de ESTA_ONLINE.

<b>Identificador</b>	<b>CP-22</b>
<b>Proba</b>	P-7
<b>Descrición</b>	Inténtase enviar un comando a un equipo que está bloqueado
<b>Método</b>	PROCESADO_COMANDOS::ENVIAR_COMANDO (\$NOMBRE_ELEMENTO, \$ID_ELEMENTO, \$OBJETO_EQUIPO, \$OBJETO_COMANDO, \$VARIABLES)
<b>Entradas</b>	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>▪ Id: 522</li> <li>▪ RED["ESTADO"]: ONLINE</li> </ul> <p>Ademais, ao enviarlle un comando *IDN? a resposta é distinta de PASARELA_TELEVES.</p> <p>Obxecto \$COMANDO con campos:</p> <ul style="list-style-type: none"> <li>▪ Id: 2</li> <li>▪ Comando_PHP: \$COMANDO_ENVIAR='FREQ_'. \$FRECUENCIA;</li> <li>▪ Respuesta_Tipo: SIN RESPUESTA</li> </ul> <p>\$NOMBRE_ELEMENTO con valor EQUIPO. \$ID_ELEMENTO con valor \$EQUIPO-&gt;Id.</p>
<b>Saídas</b>	EQUIPO Generador de Senal MXG X-Series YA BLOQUEADO.
<b>Requisitos da contorna</b>	<p>Antes da execución, hai un rexistro na táboa BLOQUEOS da base de datos que contén o valor EQUIPO na columna Elemento, 492 na columna Id_Elemento e 522 na columna Id_Equipo.</p> <p>O equipo con identificador 522 (MXG Vector Signal Generator) debe estar acendido e conectado á rede local.</p>
<b>Comentarios</b>	Ningún



<b>Identificador</b>	<b>CP-23</b>
<b>Proba</b>	P-7
<b>Descrición</b>	O código que combina o comando cos seus parámetros non é válido
<b>Método</b>	PROCESADO_COMANDOS::ENVIAR_COMANDO (\$NOMBRE_ELEMENTO, \$ID_ELEMENTO, \$OBJETO_EQUIPO, \$OBJETO_COMANDO, \$VARIABLES)
<b>Entradas</b>	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>■ Id: 522</li> <li>■ RED["ESTADO"]: ONLINE</li> </ul> <p>Ademais, ao enviarlle un comando *IDN? a resposta é distinta de PASARELA_TELEVES.</p> <p>Obxecto \$COMANDO con campos:</p> <ul style="list-style-type: none"> <li>■ Id: 2</li> <li>■ Comando_PHP: ERROR;</li> <li>■ Respuesta_Tipo: SIN RESPUESTA</li> </ul> <p>\$NOMBRE_ELEMENTO con valor EQUIPO. \$ID_ELEMENTO con valor \$EQUIPO-&gt;Id.</p>
<b>Saídas</b>	ERROR EN LA EJECUCION DE Comando_PHP
<b>Requisitos da contorna</b>	<p>Antes da execución, hai un rexistro na táboa BLOQUEOS da base de datos que contén o valor \$NOMBRE_ELEMENTO na columna Elemento, \$ID_ELEMENTO na columna Id_Elemento e \$EQUIPO-&gt;Id na columna Id_Equipo.</p> <p>O equipo con identificador 522 (MXG Vector Signal Generator) debe estar acendido e conectado á rede local.</p>
<b>Comentarios</b>	Este caso de proba é independente de se o equipo está conectado directamente á rede ou a través da pasarela.

<b>Identificador</b>	<b>CP-24</b>
<b>Proba</b>	P-7
<b>Descrición</b>	O código que procesa a resposta de forma específica para ese comando non é válido.
<b>Método</b>	PROCESADO_COMANDOS::ENVIAR_COMANDO (\$NOMBRE_ELEMENTO, \$ID_ELEMENTO, \$OBJETO_EQUIPO, \$OBJETO_COMANDO, \$VARIABLES)
<b>Entradas</b>	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>▪ Id: 522</li> <li>▪ RED["ESTADO"]: ONLINE</li> </ul> <p>Ademais, ao enviarlle un comando *IDN? a resposta é distinta de PASARELA_TELEVES.</p> <p>Obxecto \$COMANDO con campos:</p> <ul style="list-style-type: none"> <li>▪ Id: 2</li> <li>▪ Comando_PHP: \$COMANDO_ENVIAR='*IDN?';</li> <li>▪ Resposta_Tipo: STRING</li> <li>▪ Resposta_Codigo: ERROR;</li> </ul> <p>\$NOMBRE_ELEMENTO con valor EQUIPO. \$ID_ELEMENTO con valor \$EQUIPO-&gt;Id.</p>
<b>Saídas</b>	ERROR EN LA EJECUCION DE Resposta_Codigo
<b>Requisitos da contorna</b>	<p>Antes da execución, hai un rexistro na táboa BLOQUEOS da base de datos que contén o valor \$NOMBRE_ELEMENTO na columna Elemento, \$ID_ELEMENTO na columna Id_Elemento e \$EQUIPO-&gt;Id na columna Id_Equipo.</p> <p>O equipo con identificador 522 (MXG Vector Signal Generator) debe estar acendido e conectado á rede local.</p>
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-25</b>
<b>Proba</b>	P-7
<b>Descrición</b>	O tipo de resposta non é válido.
<b>Método</b>	PROCESADO_COMANDOS::ENVIAR_COMANDO (\$NOMBRE_ELEMENTO, \$ID_ELEMENTO, \$OBJETO_EQUIPO, \$OBJETO_COMANDO, \$VARIABLES)
<b>Entradas</b>	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>▪ Id: 522</li> <li>▪ RED["ESTADO"]: ONLINE</li> </ul> <p>Ademais, ao enviarlle un comando *IDN? a resposta é distinta de PASARELA_TELEVES.</p> <p>Obxecto \$COMANDO con campos:</p> <ul style="list-style-type: none"> <li>▪ Id: 2</li> <li>▪ Comando_PHP: \$COMANDO_ENVIAR='*IDN?';</li> <li>▪ Respuesta_Tipo: OTRO</li> <li>▪ Respuesta_Codigo: \$RESPUESTA=\$RESPUESTA_SIN_PROCESAR;</li> </ul> <p>\$NOMBRE_ELEMENTO con valor EQUIPO. \$ID_ELEMENTO con valor \$EQUIPO-&gt;Id.</p>
<b>Saídas</b>	TIPO DE RESPUESTA NO VALIDO
<b>Requisitos da contorna</b>	<p>Antes da execución, hai un rexistro na táboa BLOQUEOS da base de datos que contén o valor \$NOMBRE_ELEMENTO na columna Elemento, \$ID_ELEMENTO na columna Id_Elemento e \$EQUIPO-&gt;Id na columna Id_Equipo.</p> <p>O equipo con identificador 522 (MXG Vector Signal Generator) debe estar acendido e conectado á rede local.</p>
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-26</b>
<b>Proba</b>	P-8
<b>Descrición</b>	Preparar un comando para o seu envío a un equipo conectado por Ethernet.
<b>Método</b>	PREPARACION_COMANDOS: :PREPARAR_COMANDO (\$OBJETO_COMANDO, \$OBJETO_EQUIPO)
<b>Entradas</b>	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>▪ Id: 522</li> <li>▪ RED["ESTADO"]: ONLINE</li> <li>▪ RED["TIPO_CONEXION"]: ETHERNET</li> </ul> <p>Ademais, ao enviarlle un comando *IDN? a resposta é distinta de PASARELA_TELEVES.</p> <p>Obxecto \$COMANDO con campos:</p> <ul style="list-style-type: none"> <li>▪ Id: 2</li> <li>▪ Comando: \$FREQ 2GHz;</li> </ul>
<b>Saídas</b>	FREQ 2GHz
<b>Requisitos da contorna</b>	O equipo con identificador 522 (MXG Vector Signal Generator) debe estar acendido e conectado á rede local.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-27</b>
<b>Proba</b>	P-8
<b>Descrición</b>	Preparar un comando para o seu envío a un equipo conectado por GPIB a unha pasarela.
<b>Método</b>	PREPARACION_COMANDOS::PREPARAR_COMANDO (\$OBJETO_COMANDO, \$OBJETO_EQUIPO)
<b>Entradas</b>	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>■ Id: 522</li> <li>■ RED["ESTADO"]: ONLINE</li> <li>■ RED["TIPO_CONEXION"]: GPIB</li> <li>■ GPIB_Direccion: 19</li> </ul> <p>Ademais, ao enviarlle un comando *IDN? a resposta é PASARELA_TELEVES.</p> <p>Obxecto \$COMANDO con campos:</p> <ul style="list-style-type: none"> <li>■ Id: 2</li> <li>■ Comando: \$FREQ 2GHz;</li> <li>■ Respuesta_Tipo: SIN RESPUESTA</li> </ul>
<b>Saídas</b>	{tipo: "GPIB", direccion_equipo: 19, contenido: 'FREQ_2GHz', tipo_respuesta: <b>false</b> }, serializado e codificado en Base64
<b>Requisitos da contorna</b>	O equipo con identificador 522 (MXG Vector Signal Generator) debe estar acendido e conectado por GPIB a unha pasarela, que debe estar conectada á rede local.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-28</b>
<b>Proba</b>	P-9
<b>Descrición</b>	Preparar unha resposta dun equipo conectado por Ethernet.
<b>Método</b>	PREPARACION_COMANDOS::PREPARAR_RESPUESTA (\$STRING_RESPUESTA, \$OBJETO_EQUIPO)
<b>Entradas</b>	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>▪ Id: 522</li> <li>▪ RED["ESTADO"]: ONLINE</li> <li>▪ RED["TIPO_CONEXION"]: ETHERNET</li> </ul> <p>Ademais, ao enviarlle un comando *IDN? a resposta é distinta de PASARELA_TELEVES. <i>String</i> \$RESPUESTA con valor Agilent Technologies, N5182B, MY53051062, B.01.58</p>
<b>Saídas</b>	Agilent Technologies, N5182B, MY53051062, B.01.58
<b>Requisitos da contorna</b>	O equipo con identificador 522 (MXG Vector Signal Generator) debe estar acendido e conectado á rede local.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-29</b>
<b>Proba</b>	P-8
<b>Descrición</b>	Preparar unha resposta chegada dun equipo conectado por GPIB a unha pasarela.
<b>Método</b>	PREPARACION_COMANDOS::PREPARAR_RESPUESTA (\$STRING_RESPUESTA, \$OBJETO_EQUIPO)
<b>Entradas</b>	<p>Obxecto \$EQUIPO con campos:</p> <ul style="list-style-type: none"> <li>■ Id: 522</li> <li>■ RED["ESTADO"]: ONLINE</li> <li>■ RED["TIPO_CONEXION"]: GPIB</li> <li>■ GPIB_Direccion: 19</li> </ul> <p>Ademais, ao enviarlle un comando *IDN? a resposta é PASARELA_TELEVES.</p> <p><i>String</i> \$RESPUESTA con valor {tipo: "GPIB", direccion_equipo: 19, contenido: "Agilent_Technologies,_N5182B,_MY53051062,_B.01.58", tipo_respuesta: <b>false</b>} codificado en Base64</p>
<b>Saídas</b>	Agilent Technologies, N5182B, MY53051062, B.01.58
<b>Requisitos da contorna</b>	O equipo con identificador 522 (MXG Vector Signal Generator) debe estar acendido e conectado por GPIB a unha pasarela, que debe estar conectada á rede local.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-30</b>
<b>Proba</b>	P-10
<b>Descrición</b>	Recibir menos de \$TAM_BUFFER bytes a través dun socket.
<b>Método</b>	COMUNICACION_SOCKET_IP::RECIBIR(&\$SOCKET)
<b>Entradas</b>	5025
<b>Saídas</b>	holamundo
<b>Requisitos da contorna</b>	<p>A pasarela establece unha conexión válida co servidor no porto 5025.</p> <p>Antes de que transcorran 5 segundos, lánzase un proceso telnet na pasarela que envía a cadea holamundo ao porto 5025 do servidor.</p>
<b>Comentarios</b>	No momento da execución das probas, \$TAM_BUFFER está establecido a 1024.

<b>Identificador</b>	<b>CP-31</b>
<b>Proba</b>	P-10
<b>Descrición</b>	Recibir máis de \$TAM_BUFFER bytes a través dun socket.
<b>Método</b>	COMUNICACION_SOCKET_IP::RECIBIR(&\$SOCKET)
<b>Entradas</b>	5025
<b>Saídas</b>	<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla consequat neque at elit sodales, nec porta nisl gravida. Curabitur varius justo vitae velit eleifend, vel maximus lacus bibendum. Nullam dictum laoreet justo, ac rutrum sapien facilisis a. Sed finibus, mi non tempor faucibus, sapien libero sagittis dui, sit amet gravida sapien leo vel orci. Vestibulum egestas, nisl sed luctus bibendum, massa justo facilisis libero, a dictum lorem risus quis nisl. In sed mollis nunc, nec semper lorem. Donec in molestie erat. Aliquam enim lorem, egestas eget odio id, lacinia mattis leo. Nunc rutrum quam nec rutrum finibus. Aliquam id sollicitudin ligula. Donec metus ante, facilisis at efficitur sit amet, aliquet eu sem. Vestibulum dignissim placerat aliquam. Nulla bibendum nisl id ornare pharetra. Nulla facilisi. Nulla facilisi. Pellentesque congue malesuada rhoncus. Donec a euismod est. Sed lorem purus, fermentum eget ipsum sit amet, posuere pellentesque felis. Suspendisse potenti. Fusce aliquam urna nec imperdiet aliquet. In hac massa nunc.</p>
<b>Requisitos da contorna</b>	<p>A pasarela establece unha conexión válida co servidor no porto 5025.</p> <p>Antes de que transcorran 5 segundos, lánzase un proceso telnet na pasarela que envía a cadea holamundo ao porto 5025 do servidor.</p>
<b>Comentarios</b>	No momento da execución das probas, \$TAM_BUFFER está establecido a 1024.

<b>Identificador</b>	<b>CP-32</b>
<b>Proba</b>	P-10
<b>Descrición</b>	Sucede un <i>time-out</i> durante a lectura do socket.
<b>Método</b>	COMUNICACION_SOCKET_IP::RECIBIR(&\$SOCKET)
<b>Entradas</b>	5025
<b>Saídas</b>	<p>null. Lánzase unha excepción coa mensaxe TIMEOUT EN RECEPCION POR SOCKET   [11] Resource temporarily unavailable</p>
<b>Requisitos da contorna</b>	A pasarela establece unha conexión válida co servidor no porto 5025. Nunca se envían datos ao socket.
<b>Comentarios</b>	No momento da execución das probas, \$TAM_BUFFER está establecido a 1024.



### 7.3.2. Casos de proba da pasarela

Considérase que as entradas son as mensaxes entrantes para a pasarela, e as saídas son as respostas que esta envía de volta para o servidor de XesLab. Ademais, asúmese que son requisitos da contorna para todos os casos de proba:

- A pasarela está conectada á rede local de Ladetel.
- O equipo 522 está asignado en base de datos ao enderezo IP da pasarela.
- O equipo 522 está configurado co enderezo GPIB 19.

Ademais disto, óbvianse as clases P-11 e P-12. Debido a que tratan erros de rede, son difíciles de probar; hai que forzar un fallo de rede para poder simular o estado, pero para poder recibir a mensaxe no servidor a rede debe de estar restaurada. Si ten sentido probar a clase P-13, que é o escenario dun *time-out* por parte do servidor.

En todos os casos o punto de entrada é o mesmo: o método `GestorModulos::procesar_mensaje()`, do que colga toda a xestión dos casos especiais de mensaxes, e a transferencia das mensaxes normais ao módulo correcto. Debido a isto, todos os casos de proba indican que se proba este método, pero é preciso ter en conta que non é só este método o que se está probando, senón que é unha proba de integración que proba a funcionalidade principal da pasarela.

<b>Identificador</b>	<b>CP-33</b>
<b>Proba</b>	P-11
<b>Descrición</b>	Erro na recepción de datos.
<b>Método</b>	<code>Conexion::recibir()</code>
<b>Entradas</b>	Ningunha
<b>Saídas</b>	Excepción <code>ExcepcionRed</code> coa mensaxe <code>"No_se_ha_podido_recibir_el_mensaje._Error:_Resource_temporarily_unavailable."</code>
<b>Requisitos da contorna</b>	Débese establecer unha conexión válida coa pasarela, pero non se debe enviar ningún dato.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-34</b>
<b>Proba</b>	P-11
<b>Descrición</b>	Recíbese un <i>string</i> baleiro.
<b>Método</b>	<code>GestorModulos::procesar_mensaje()</code>
<b>Entradas</b>	Un <i>string</i> baleiro
<b>Saídas</b>	Excepción <code>ExcepcionMensajeVacio</code>
<b>Requisitos da contorna</b>	Ningún
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-35</b>
<b>Proba</b>	P-11
<b>Descrición</b>	Recíbese un comando de identificación sen encapsular.
<b>Método</b>	GestorModulos::procesar_mensaje()
<b>Entradas</b>	*IDN?
<b>Saídas</b>	PASARELA_TELEVES
<b>Requisitos da contorna</b>	Ningún
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-36</b>
<b>Proba</b>	P-11
<b>Descrición</b>	Recíbese unha consulta sobre o estado do equipo, e o equipo está conectado.
<b>Método</b>	GestorModulos::procesar_mensaje()
<b>Entradas</b>	{tipo: "GPIB", direccion_equipo: 19, contenido:"XESLAB_ALIVE", tipo_respuesta: <b>true</b> }
<b>Saídas</b>	{tipo: "GPIB", direccion_equipo: 19, contenido: <b>true</b> , tipo_respuesta: <b>false</b> }
<b>Requisitos da contorna</b>	O equipo con identificador 522 (MXG Vector Signal Generator) debe estar fisicamente acendido e conectado a través de GPIB á pasarela.
<b>Comentarios</b>	O valor de tipo_respuesta non é importante nas mensaxes de resposta, xa que se descarta ao chegar ao servidor. Porén, estableceuse que en todas as mensaxes de resposta este valor sería <b>false</b> , por consistencia.

<b>Identificador</b>	<b>CP-37</b>
<b>Proba</b>	P-11
<b>Descrición</b>	Recíbese unha consulta sobre o estado do equipo, e o equipo está desconectado.
<b>Método</b>	GestorModulos::procesar_mensaje()
<b>Entradas</b>	{tipo: "GPIB", direccion_equipo: 19, contenido:"XESLAB_ALIVE", tipo_respuesta: <b>true</b> }
<b>Saídas</b>	{tipo: "GPIB", direccion_equipo: 19, contenido: <b>false</b> , tipo_respuesta: <b>false</b> }
<b>Requisitos da contorna</b>	O equipo con identificador 522 (MXG Vector Signal Generator) non debe estar fisicamente acendido ou non debe estar conectado a través de GPIB á pasarela.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-38</b>
<b>Proba</b>	P-11
<b>Descrición</b>	Recíbese un comando ao que o equipo responde.
<b>Método</b>	GestorModulos::procesar_mensaje()
<b>Entradas</b>	{tipo: "GPIB", direccion_equipo: 19, contenido: "*IDN?", tipo_respuesta: <b>true</b> }
<b>Saídas</b>	{tipo: "GPIB", direccion_equipo: 19, contenido: "Agilent_ Technologies,_N5182B,_MY53051062,_B.01.58\n", tipo_respuesta: <b>true</b> }
<b>Requisitos da contorna</b>	O equipo con identificador 522 (MXG Vector Signal Generator) debe estar fisicamente acendido e conectado a través de GPIB á pasarela.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-39</b>
<b>Proba</b>	P-11
<b>Descrición</b>	Recíbese un comando ao que o equipo non responde.
<b>Método</b>	GestorModulos::procesar_mensaje()
<b>Entradas</b>	{tipo: "GPIB", direccion_equipo: 19, contenido: "FREQ_ 2GHz", tipo_respuesta: <b>false</b> }
<b>Saídas</b>	{tipo: "GPIB", direccion_equipo: 19, contenido: "OK", tipo_respuesta: <b>true</b> }
<b>Requisitos da contorna</b>	O equipo con identificador 522 (MXG Vector Signal Generator) debe estar fisicamente acendido e conectado a través de GPIB á pasarela.
<b>Comentarios</b>	Enviar a este equipo o comando FREQ? despois da execución do caso de proba debería devolver +2.000000000000E+09.

<b>Identificador</b>	<b>CP-40</b>
<b>Proba</b>	P-11
<b>Descrición</b>	Recíbese un <i>string</i> ilexible; non está baleiro, non é a cadea de identificación, e non está encapsulada en JSON e Base64.
<b>Método</b>	GestorModulos::procesar_mensaje()
<b>Entradas</b>	outro_string
<b>Saídas</b>	{tipo: "CONTROL", contenido: "No_se_ha_podido_ deserializar_el_mensaje._Error:_parse_error_at_2:_ unexpected_end_of_input;_expected_string_literal"}
<b>Requisitos da contorna</b>	O equipo con identificador 522 (MXG Vector Signal Generator) debe estar fisicamente acendido e conectado a través de GPIB á pasarela.
<b>Comentarios</b>	Ningún .

<b>Identificador</b>	<b>CP-41</b>
<b>Proba</b>	P-11
<b>Descrición</b>	O campo tipo da mensaxe non é ningún valor contemplado na lista de tipos e pseudo-tipos.
<b>Método</b>	GestorModulos::procesar_mensaje()
<b>Entradas</b>	{tipo: "TIPO_ERROR", direccion_equipo: 19, contenido:"*IDN?", tipo_respuesta: <b>true</b> }
<b>Saídas</b>	{tipo: "CONTROL", contenido:"El_tipo_del_mensaje_no_es_valido"}
<b>Requisitos da contorna</b>	O equipo con identificador 522 (MXG Vector Signal Generator) debe estar fisicamente acendido e conectado a través de GPIB á pasarela.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-42</b>
<b>Proba</b>	P-11
<b>Descrición</b>	A mensaxe non ten campo contenido.
<b>Método</b>	GestorModulos::procesar_mensaje()
<b>Entradas</b>	{tipo: "GPIB", direccion_equipo: 19, tipo_respuesta: <b>false</b> }
<b>Saídas</b>	{tipo: "CONTROL", contenido:"No_se_ha_podido_deserializar_el_mensaje._Error:_key_'contenido'_not_found"}
<b>Requisitos da contorna</b>	O equipo con identificador 522 (MXG Vector Signal Generator) debe estar fisicamente acendido e conectado a través de GPIB á pasarela.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-43</b>
<b>Proba</b>	P-11
<b>Descrición</b>	O campo tipo da mensaxe é un pseudo-tipo, polo que non ten un módulo de procesamento asociado.
<b>Método</b>	GestorModulos::procesar_mensaje()
<b>Entradas</b>	{tipo: "TIPO_NO_VALIDO", direccion_equipo: 19, contenido:"*IDN?", tipo_respuesta: <b>true</b> }
<b>Saídas</b>	{tipo: "CONTROL", contenido:"No_se_ha_podido_deserializar_el_mensaje._Error:_El_tipo_del_mensaje_no_es_valido"}
<b>Requisitos da contorna</b>	O equipo con identificador 522 (MXG Vector Signal Generator) debe estar fisicamente acendido e conectado a través de GPIB á pasarela.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-44</b>
<b>Proba</b>	P-11
<b>Descrición</b>	O enderezo GPIB do equipo ao que vai dirixida a mensaxe non é un enderezo GPIB válido (é menor que 0 ou maior que 30).
<b>Método</b>	GestorModulos::procesar_mensaje()
<b>Entradas</b>	{tipo: "GPIB", direccion_equipo: 31, contenido:"*IDN?", tipo_respuesta: <b>true</b> }
<b>Saídas</b>	{tipo: "CONTROL", contenido:"Error_en_la_comunicacion_GPIB._Error:_You_have_attempted_to_write_data_or_command_bytes,_but_there_are_no_listeners_currently_addressed"}
<b>Requisitos da contorna</b>	Ningún
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-45</b>
<b>Proba</b>	P-11
<b>Descrición</b>	Non hai ningún equipo escoitando no enderezo GPIB ao que vai dirixida a mensaxe.
<b>Método</b>	GestorModulos::procesar_mensaje()
<b>Entradas</b>	{tipo: "GPIB", direccion_equipo: 20, contenido:"*IDN?", tipo_respuesta: <b>true</b> }
<b>Saídas</b>	{tipo: "CONTROL", contenido:"Error_en_la_comunicacion_GPIB._Error:_A_read_or_write_of_data_bytes_has_been_aborted,_possibly_due_to_a_timeout_or_reception_of_a_device_clear_command"}
<b>Requisitos da contorna</b>	Non debe haber ningún dispositivo activo no enderezo GPIB 20 conectado á pasarela.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	<b>CP-46</b>
<b>Proba</b>	P-11
<b>Descrición</b>	O campo contenido da mensaxe non é ningún comando válido para o equipo ao que vai dirixida a mensaxe.
<b>Método</b>	GestorModulos::procesar_mensaje()
<b>Entradas</b>	{tipo: "GPIB", direccion_equipo: 19, contenido:"COMANDO_ERROR", tipo_respuesta: <b>true</b> }
<b>Saídas</b>	{tipo: "CONTROL", contenido:"Error_en_la_comunicacion_GPIB._Error:_A_read_or_write_of_data_bytes_has_been_aborted,_possibly_due_to_a_timeout_or_reception_of_a_device_clear_command"}
<b>Requisitos da contorna</b>	O equipo con identificador 522 (MXG Vector Signal Generator) debe estar fisicamente acendido e conectado a través de GPIB á pasarela.
<b>Comentarios</b>	Ningún

<b>Identificador</b>	CP-47
<b>Proba</b>	P-11
<b>Descrición</b>	O campo <code>tipo_respuesta</code> da mensaxe é <code>true</code> , pero contido contén un comando para o que o equipo non xera unha resposta; estarase esperando por unha resposta que nunca chega.
<b>Método</b>	<code>GestorModulos::procesar_mensaje()</code>
<b>Entradas</b>	<code>{tipo: "GPIB", direccion_equipo: 19, contenido:"FREQ_2GHz", tipo_respuesta: true}</code>
<b>Saídas</b>	<code>{tipo: "CONTROL", contenido:"Error_en_la_comunicacion_GPIB._Error:_A_read_or_write_of_data_bytes_has_been_aborted,_possibly_due_to_a_timeout_or_reception_of_a_device_clear_command"}</code>
<b>Requisitos da contorna</b>	O equipo con identificador 522 (MXG Vector Signal Generator) debe estar fisicamente acendido e conectado a través de GPIB á pasarela.
<b>Comentarios</b>	Ningún

## 7.4. Implementación e execución das probas

### 7.4.1. Implementación

Como librarías de *testing*, utilízase Boost.Test para o código escrito en C++ e PHPUnit para aquel escrito en PHP. As librarías aportan a súa propia función `main`; en cada subsistema creouse un ficheiro de nome `test` que almacena todos os casos de proba.

- Na pasarela, o ficheiro `test.cpp` atópase no paquete `test`. Decláranse os casos de proba coa macro `BOOST_AUTO_TEST_CASE`, que permite que se rexistren de xeito automático. É preciso compilar coa opción `-lboost-test-framework`.

```
BOOST_AUTO_TEST_CASE(comando_con_respuesta) {
    GestorModulos gm;

    nlohmann::json respuesta_esperada_json;
    respuesta_esperada_json["tipo"] = "GPIB";
    respuesta_esperada_json["direccion_equipo"] = 19;
    respuesta_esperada_json["contenido"] = "Agilent_
        Technologies,_N5182B,_MY53051062,_B.01.58\n";
    respuesta_esperada_json["tipo_respuesta"] = false;
    std::string respuesta_esperada =
        respuesta_esperada_json.dump();

    nlohmann::json mensaje_json;
    mensaje_json["tipo"] = "GPIB";
    mensaje_json["direccion_equipo"] = 19;
    mensaje_json["contenido"] = "*IDN?";
```

```

    mensaje_json["tipo_respuesta"] = true;
    std::string mensaje = mensaje_json.dump();

    std::string respuesta = gm.procesar_mensaje(mensaje);

    BOOST_CHECK(respuesta == respuesta_esperada);
}

```

Inclúuse no *makefile* unha regra para compilar as probas correctamente, polo que se poden executar simplemente invocando o executable: `./daemon_pasarela`.

- No servidor, creouse unha clase TEST no ficheiro `test.php` do módulo de equipos. A clase hereda de `TestCase`, presente no `import PHPUnit\framework\TestCase`; os casos de proba defínense como métodos públicos e estáticos desta clase, cuxo nome é precedido pola palabra `test`.

```

public function test_ENVIAR_COMANDO_RESPUESTA_EXITO() {
    echo "\n_ENVIAR_COMANDO_RESPUESTA_EXITO_\n";
    $EQUIPO_522 = new EQUIPO(522, true);
    $COMANDO_IDN = new COMUNICACION_COMANDO_EQUIPO(1);
    $VARIABLES = null;

    PROCESADO_COMANDOS::BLOQUEAR($EQUIPO_522, "EQUIPO",
        $EQUIPO_522->Id);
    $RESPUESTA =
        PROCESADO_COMANDOS::ENVIAR_COMANDO("EQUIPO",
            $EQUIPO_522->Id, $EQUIPO_522, $COMANDO_IDN,
            $VARIABLES);
    PROCESADO_COMANDOS::DESbloQUEAR($EQUIPO_522, "EQUIPO",
        $EQUIPO_522->Id);

    $this->assertSame("Agilent_Technologies ,_N5182B ,_
        MY53051062 ,_B.01.58\n", $RESPUESTA);
}

```

Úsase o método `setUpBeforeClass` para inicializar a contorna. As probas execútanse con `phpunit test`.

## 7.4.2. Incidencias

### Incompatibilidade de Boost.Test con libb64

A librería utilizada para a codificación e decodificación Base64 non funcionou correctamente ao compilar o programa coa opción de test, posiblemente debido a que a súa implementación do algoritmo usa co-rutinas. Debido a isto, a idea orixinal de comparar as mensaxes de saída no estado no que se enviarían pola rede non é factible.

Isto foi un problema, xa que se considera fundamental incluír o método `GestorModulos::procesar_mensaje` nas probas do sistema, xa que contén parte da

funcionalidade relevante (como por exemplo a xestión de casos especiais de mensaxes), e del colgan as invocacións ás funcionalidades de outras clases relevantes (o xestor de mensaxes e o módulo GPIB, entre outros). O problema reside en que dentro de `procesar_mensaje` se invocan ás funcións de empaquetamento e desempaquetamento de mensaxes, que invocan á librería Base64. Para poder probar `procesar_mensaje` prescindindo da librería Base64 sen facer refactorizacións maiores, creouse unha macro de nome `TEST`, que condiciona a chamada ás funcións de codificado e decodificado, e que pode activarse durante a compilación.

### **Necesidade de cambios manuais na contorna**

Todos os casos de proba foron codificados, pero non se puido automatizar totalmente a súa execución debido á necesidade de facer cambios manuais na contorna, concretamente no relativo ao estado online ou offline dos equipos; para forzar o cambio de estado dun equipo, hai que conectar ou desconectar fisicamente a conexión.

No software da pasarela, debido a que a librería `Boost.Test` xera a súa propia función `main` na que se invoca o código de todos os casos de proba, é preciso executar as probas dúas veces; unha delas para probar os casos de proba que requiran unha conexión co equipo (onde fallarán o resto, pero estes fallos non serán resultados útiles, ao non darse os seus requisitos de contexto), e outra para todos os demais. O informe de execución é, entón, o resultado de combinar os resultados útiles das dúas execucións.

### **Imposibilidade de restaurar o contexto en CP-11**

O uso de `PHPUnit` como librería de probas fai que o acceso aos recursos comúns a todo o sistema, como o sistema de usuarios e a base de datos, non se faga a través do punto de acceso habitual (a web) senón a través dun *script* de probas. Debido a que todos os rexistros da base de datos están asociados ao usuario dende cuxa sesión se escribiu ese rexistro, o uso do *script* dá lugar a certas inconsistencias na base de datos.

Por isto, e debido a que a información dos bloqueos sobre os equipos se rexistra na base de datos, foi imposible restaurar o contexto para que se dean as circunstancias nas que o caso de proba CP-11 se execute con éxito. Solvéntase este problema executando o caso de proba por separado, tras baleirar manualmente a táboa conflictiva.

Sucede o mesmo cos casos de proba relacionados coa función `ENVIAR_COMANDO` posteriores a CP-22.

### **Probas de funcionalidades relacionadas coa rede**

Resultou particularmente difícil probar os casos de proba relacionados coas funcionalidades de rede. A idea inicial era a de lanzar, dende o propio caso de pro-



ba e usando funcións similares a `system` ou `exec`, un proceso en paralelo que abraza unha comunicación con `localhost`, de xeito que non sería necesario involucrar a outros sistemas e a proba realizaríase de forma completamente automática.

Porén, as incompatibilidades das librerías fixeron imposible este enfoque, polo que foi necesario comunicarse remotamente. Para probar as funcionalidades de rede da pasarela, creouse unha función no apartado de desenvolvemento de Xes-Lab que executa un pequeno cliente TCP; e para probar as do servidor, escribiuse un servidor TCP en C para ser compilado e executado na pasarela.

## 7.5. Informe de execución

Na táboa 7.4 móstranse os resultados de todos os casos de proba executados. No momento en que se detecta un defecto, soluciónase inmediatamente para evitar que isto afectara aos resultados doutras probas.

Unha vez finalizada a corrección dos defectos, o sistema pasa todas as probas satisfactoriamente, polo que se considera que o software é completamente aceptable.

## 7.6. Validación

### 7.6.1. Envío de comandos a equipos de xeito individual

O sistema permite enviar a cada equipo os comandos do seu repertorio, para executar probas ou calibracións manuais. A pantalla dos datos dos equipos contén un panel no que se poden visualizar os comandos, indicar a parametrización, enviar cada comando ao equipo e mostrar a resposta, se o comando é un comando con resposta.

CP	Método	Descrición	Resultado	Defecto
CP-1	ESTA_ONLINE	O equipo está online e conectado por Ethernet.	Proba pasada	
CP-2	ESTA_ONLINE	O equipo está offline.	Proba fallida	Dentro do método lese unha propiedade que só existe se o equipo está online. Se non está online, esta propiedade non está definida e causa un erro inesperado.
CP-3	ESTA_ONLINE	O equipo está online e conectado por GPIB a unha pasarela.	Proba fallida	Utilizábase <code>property_exists</code> para comprobar a existencia dunha clave nun <i>array</i> , pero a función adecuada para iso é <code>isset</code> .
CP-4	ESTA_ONLINE	A pasarela está conectada á rede, pero o equipo non está conectado a esta.	Proba pasada	
CP-5	TIPO_CONEXION	O equipo está conectado a unha pasarela por GPIB.	Proba pasada	
CP-6	TIPO_CONEXION	O equipo non ten enderezo secundario (enderezo GPIB).	Proba pasada	
CP-7	TIPO_CONEXION	O equipo non está conectado á pasarela.	Proba fallida	A clase de sockets lanza unha excepción por <i>time-out</i> que non se captura no invocador.
CP-8	BLOQUEAR	Bloquéase un equipo que estaba libre.	Proba pasada	
CP-9	BLOQUEAR	Bloquéase un equipo que xa estaba bloqueado por ese elemento (éxito silencioso).	Proba pasada	
CP-10	BLOQUEAR	Inténtase bloquear un equipo que está bloqueado por outro elemento.	Proba pasada	
CP-11	DESBLOQUEAR	Desbloquéase un equipo que estaba bloqueado por ese elemento.	Proba pasada	
CP-12	DESBLOQUEAR	O desbloqueo falla porque o equipo está bloqueado por outro elemento.	Proba pasada	
CP-13	DESBLOQUEAR	O desbloqueo falla porque o equipo está libre.	Proba pasada	
CP-14	ESTA_BLOQUEADO_POR_ELEMENTO	O equipo está bloqueado por ese elemento.	Proba pasada	
CP-15	ESTA_BLOQUEADO_POR_ELEMENTO	O equipo non está bloqueado por ese elemento.	Proba pasada	

CP	Método	Descrición	Resultado	Defecto
CP-16	ESTA_BLOQUEADO	O equipo está bloqueado por algún elemento.	Proba pasada	
CP-17	ESTA_BLOQUEADO	O equipo non está bloqueado por ningún elemento.	Proba pasada	
CP-18	ENVIAR_COMANDO	Enviar un comando sen resposta a un equipo, con éxito.	Proba fallida	Non se comprobou que unha variable estivera definida antes de acceder ao seu valor.
CP-19	ENVIAR_COMANDO	Enviar un comando con resposta a un equipo, con éxito.	Proba pasada	
CP-20	ENVIAR_COMANDO	Enviar un comando sen resposta a un equipo que non está bloqueado por ningunha entidade.	Proba pasada	
CP-21	ENVIAR_COMANDO	Inténtase enviar un comando a un equipo que non está online.	Proba pasada	
CP-22	ENVIAR_COMANDO	Inténtase enviar un comando a un equipo que está bloqueado.	Proba pasada	
CP-23	ENVIAR_COMANDO	O código que combina o comando cos seus parámetros non é válido.	Proba pasada	
CP-24	ENVIAR_COMANDO	O código que procesa a resposta de forma específica para ese comando non é válido.	Proba pasada	
CP-25	ENVIAR_COMANDO	O tipo de resposta non é válido.	Proba fallida	Non se comproba que o tipo de resposta sexa un valor válido, só se comproba se é SIN RESPUESTA ou un valor distinto.
CP-26	PREPARAR_COMANDO	Preparar un comando para o seu envío a un equipo conectado por Ethernet.	Proba pasada	
CP-27	PREPARAR_COMANDO	Preparar un comando para o seu envío a un equipo conectado por GPIB a unha pasarela.	Proba pasada	
CP-28	PREPARAR_RESPUESTA	Preparar unha resposta dun equipo conectado por Ethernet.	Proba pasada	
CP-29	PREPARAR_RESPUESTA	Preparar unha resposta chegada dun equipo conectado por GPIB a unha pasarela.	Proba pasada	

CP	Método	Descrición	Resultado	Defecto
CP-30	RECIBIR	Recibir menos de \$TAM_BUFFER bytes a través dun socket.	Proba pasada	
CP-31	RECIBIR	Recibir máis de \$TAM_BUFFER bytes a través dun socket.	Proba pasada	
CP-32	RECIBIR	Sucede un <i>time-out</i> durante a lectura do socket.	Proba pasada	
CP-33	recibir	Erro na recepción de datos.	Proba fallida	O caso de proba falla porque non se lanza unha excepción en <i>time-out</i> , pero o comportamento do programa trata o caso como o de unha mensaxe vacía, e modificar o programa para corrixir o defecto causarían problemas no caso desta mensaxe especial; polo tanto, e como o programa non se queda esperando por unha mensaxe que nunca chega, non se considerou necesario facer cambios.
CP-34	procesar_comando	Recíbese un <i>string</i> baleiro.	Proba pasada	
CP-35	procesar_comando	Recíbese un comando de identificación sen encapsular.	Proba pasada	
CP-36	procesar_comando	Recíbese unha consulta sobre o estado do equipo, e o equipo está conectado.	Proba pasada	
CP-37	procesar_comando	Recíbese unha consulta sobre o estado do equipo, e o equipo está desconectado.	Proba pasada	
CP-38	procesar_comando	Recíbese un comando que o equipo responde.	Proba pasada	
CP-39	procesar_comando	Recíbese un comando que o equipo non responde.	Proba pasada	
CP-40	procesar_comando	Recíbese un <i>string</i> ilexible; non está baleiro, non é a cadea de identificación, e non está encapsulada en JSON e Base64.	Proba pasada	

CP	Método	Descrición	Resultado	Defecto
CP-41	procesar_comando	O campo tipo da mensaxe non é ningún valor contemplado na lista de tipos e pseudo-tipos.	Proba pasada	
CP-42	procesar_comando	A mensaxe non ten campo contenido.	Proba fallida	A excepción que lanza a librería JSON non é capturada, e causa unha terminación do programa.
CP-43	procesar_comando	O campo tipo da mensaxe é un pseudo-tipo polo que non ten un módulo de procesamento asociado.	Proba pasada	
CP-44	procesar_comando	O enderezo GPIB do equipo ao que vai dirixida a mensaxe non é un enderezo GPIB válido (é menor que 0 ou maior que 30).	Proba pasada	
CP-45	procesar_comando	Non hai ningún equipo escoitando no enderezo GPIB ao que vai dirixida a mensaxe.	Proba pasada	
CP-46	procesar_comando	O campo contenido da mensaxe non é ningún comando válido para o equipo ao que vai dirixida a mensaxe.	Proba pasada	
CP-47	procesar_comando	O campo tipo_respuesta da mensaxe é <b>true</b> , pero contenido contén un comando para o que o equipo non xera unha resposta; estase esperando por unha resposta que nunca chega.	Proba pasada	

Cadro 7.4: Resultados das probas

## DATOS COMUNICACIÓN EQUIPO ☰

FRECUENCIA	
FREQ \$FRECUENCIA	2 GHz
	 ENVIAR
Nº de Traza	
TRAC? \$TRAZA	-
	 ENVIAR
*IDN?	Agilent Technologies, N5182B, MY53051062, B.01.58
	 ENVIAR
FREQ?	+2.0000000000000E+09
	 ENVIAR

Figura 7.1: Resultados de comunicaciones co equipo 522

### 7.6.2. Execución de medidas

Demostrouse que o sistema construído cumpre as necesidades dos usuarios implementando en XesLab unha medida de proba, representativa dunha das categorías de medidas que se realizan habitualmente no laboratorio, e executándoa. Trátase dunha medida de apantallamento, na que un xerador de sinal de radiofrecuencia transmite a un receptor de sinal.

O panel de execución da medida carga os valores establecidos polo procedemento, e permite modificalos. Selecciona automaticamente, de entre os equipos que están conectados á rede local, aqueles que son adecuados para a execución da medida; os que son do tipo que esta necesita e que poden executar os comandos de medida. Empregouse o xerador de sinal Keysight MXG, o mesmo instrumento que se usou para desenvolvemento e probas durante todo o proxecto, e o receptor de sinal Ronde & Schwarz ESCI.

The screenshot shows the 'MEDIDA TÉCNICA (COMPROBACIÓN PEE)' configuration screen. It features a sidebar with procedure details and a main area with a table of variables and a receiver selection dropdown.

VARIABLES DE LA MEDIDA					
Longitud del Tubo (m)	Frecuencia [INCIQ] (Hz)	Frecuencia [FIN] (Hz)	Nº de Puntos	Impedancia del cable	Generador
2	30000000.0	2150000000.0	801	75	Generador de Señal MXG X-Series [Id: 522]

Receptor: Receptor de medida R&S ESCI (9KHz-3GHz) [Id: 75]

ENVIAR

© 2019. LADETEL - TELEVES

Figura 7.2: Pantalla dunha medida técnica

Nesta medida, o xerador de sinal MXG (figura 7.4 aumenta progresivamente a frecuencia dende os 30 MHz ata os 2,15 GHz e o receptor (figura 7.5 toma unha medida da frecuencia cada poucos segundos. Todos os comandos empregados para orquestrar esta medida se envían dende XesLab.

Na parte inferior do panel da medida técnica pódense ver os resultados da medida en tempo real.



Figura 7.3: Panel de ejecución da medida

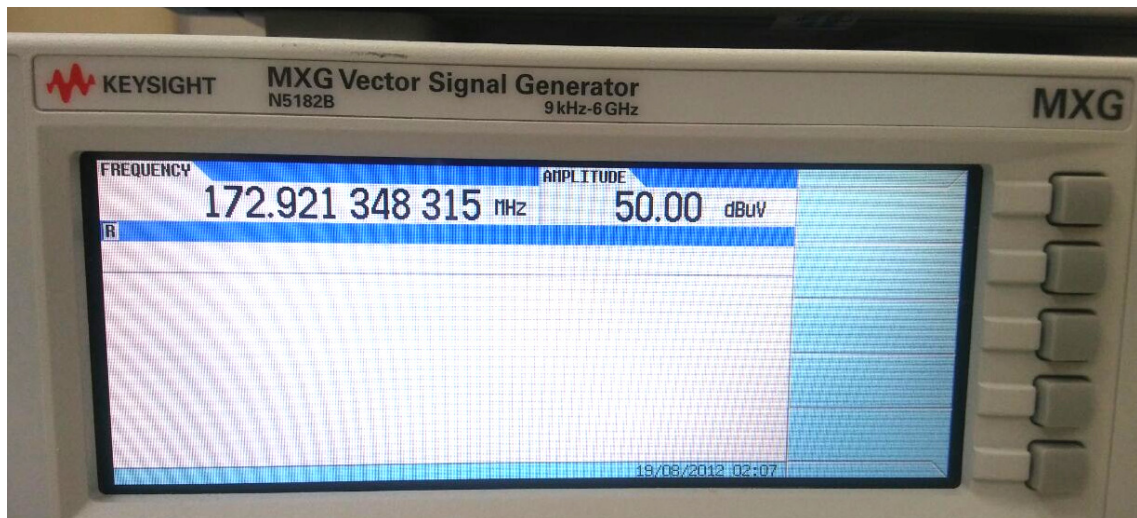


Figura 7.4: Pantalla do equipo MXG



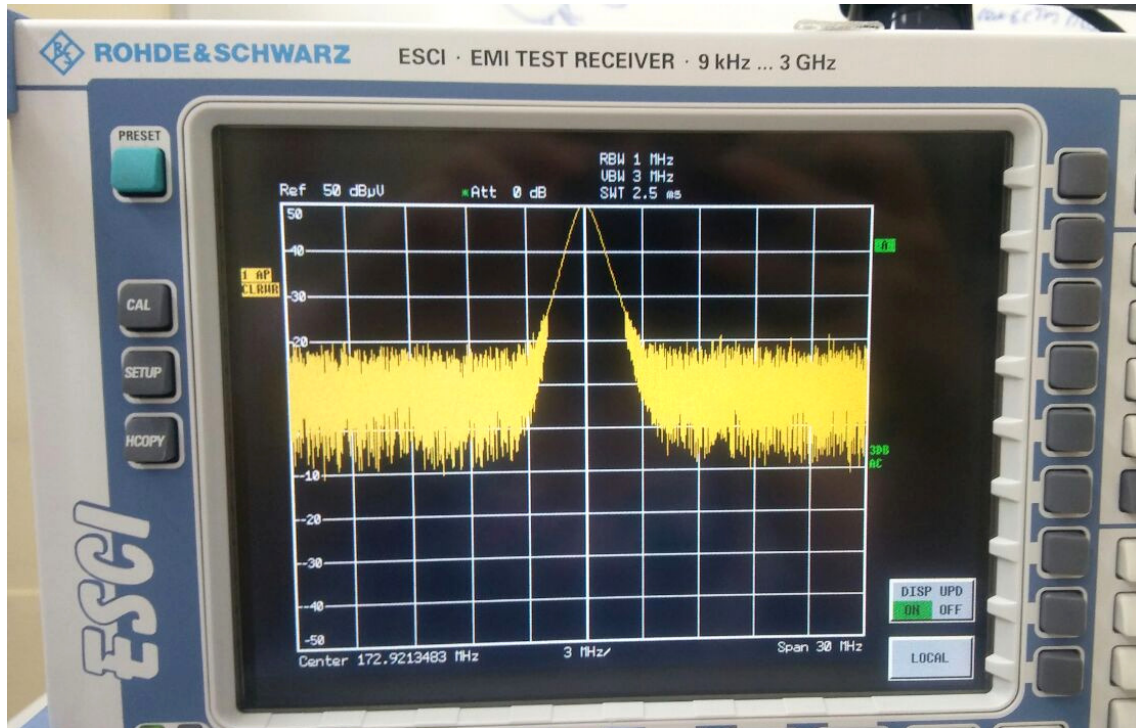


Figura 7.5: Pantalla do equipo ESCI

MEDIDA TÉCNICA (COMPROBACIÓN PEE)

VARIABLES DE LA MEDIDA

Longitud del Tubo (m)	Frecuencia [INICIO] (Hz)	Frecuencia [FIN] (Hz)	Nº de Puntos	Impedancia del cable	Generador
2	30000000.0	2150000000.0	801	50	Generador de Señal MXG X-Series [id: 522]

Receptor

Receptor de medida R&S ESCI (9KHz-3GHz) [id: 75]

ENVIAR

- 30000000.0Hz
- 32646691.635456Hz
- 35293383.270911Hz
- 37940074.906367Hz
- 40586766.541823Hz
- 43233458.17278Hz
- 45880149.812734Hz
- 48526841.44819Hz
- 51173533.083645Hz
- 53820224.719101Hz
- 56466916.354557Hz
- 59113607.990013Hz
- 61760299.625468Hz

Figura 7.6: Resultados da medida, obtidos en tempo real

Esta medida é xustificadamente similar a un experimento real, e por isto os usuarios certifican que, en efecto, o sistema cumpre cos requisitos especificados.

# Capítulo 8

## Conclusións e posibles ampliacións

### 8.1. Conclusións

Para lograr a acreditación conforme á norma 17025 de requisitos de competencia en laboratorios de calibración e ensaio, plantexouse en Ladetel a elaboración dun sistema informático que cubra tanto a xestión da información xerada a partir do traballo diario, como a automatización das medicións. Un sistema como este reduciría o erro resultante da intervención humana nas medicións, ademais de liberar aos técnicos dunha carga de traballo importante, propiciando a mellora da produtividade.

O produto final do proxecto é o sistema de automatización xunto coa súa documentación; o sistema de xestión da información implementouse de forma previa como parte dunha aplicación web (XesLab), creando unha base técnica e documental que se aproveitou durante a execución deste proxecto.

As medidas son automatizables aproveitando as capacidades de control remoto que teñen a maior parte dos equipos; manéxanse enviándolles comandos a través de algunha das súas interfaces hardware. O sistema desenvolto é o resultado da colaboración de dous subsistemas que resolven as necesidades de automatización do laboratorio.

- Un conxunto de módulos de comunicación en XesLab, que permiten a comunicación a través de TCP/IP cos equipos que contan con interface de tipo Ethernet.
- Unha serie de dispositivos físicos coñecidos como pasarelas, que actúan como intermediarias entre a rede local e os equipos sen capacidades IP.

O proxecto foi realizado nun contexto empresarial, a diferenza doutros traballos de fin de grao, e isto tivo un impacto nas tarefas de xestión; algúns dos

riscos que se materializaron foron consecuencia da súa realización nesta contorna. Considérase que, de todos modos, a proximidade do problema e do seu entorno tivo un impacto positivo na realización do proxecto e nas leccións aprendidas deste.

Foi necesario realizar unha replanificación debido á manifestación dun risco de alta exposición, pero logrouse entregar un produto de calidade satisfactoria. A nivel de requisitos, certificouse que o sistema cumpre todos os casos de uso. Durante a etapa de verificación e validación, comprobouse que o sistema pasa todas as probas implementadas, e a funcionalidade principal foi validada polos usuarios nun contexto de traballo habitual; a execución dunha medida que involucra a varios equipos. Debido a que o desenvolvemento do produto continuará nos próximos meses, será posible ampliar as funcionalidades e refinar as existentes.

Dende un punto de vista académico, o proxecto resultou ser tremendamente enriquecedor para a desenvolvedora; non só se colaborou nun proxecto real e nunha contorna próxima ao problema e aos usuarios, senón que se empregaron tecnoloxías que a desenvolvedora non utilizara antes. Aplicáronse e melloráronse moitos coñecementos de deseño de software e xestión de proxectos.

Deste xeito, conclúese que, a pesar do retraso, o proxecto concluíu con éxito, de forma satisfactoria para os involucrados.

## 8.2. Posibles ampliacións

Ao ser un software en desenvolvemento, os módulos desenvoltos para este proxecto serán moi probablemente revisados e mellorados, como parte da evolución de XesLab. O software de pasarela presenta un abanico particularmente grande de posibilidades de ampliación, xa que está especificamente deseñada para que se poida estender con facilidade.

- Incluir un módulo para o control de sensores de temperatura e humidade.
- Ampliar o soporte a máis tipos de interfaces hardware, como pode ser Bluetooth, RS232, etc. Este punto engloba ao punto anterior.
- Desenvolver un módulo de control para agrupar as funcionalidades especiais da pasarela, e engadir soporte para comandos de estado, que permitan coñecer os datos do estado desta.
- Implementar soporte para conexión cunha placa de relés USB que permita controlar dispositivos a través da sinal de 220V.
- Implementar un sistema que permita actualizar o software das pasarelas dende o servidor.
- Implementar en XesLab unha consola SSH para poder acceder ás pasarelas a través deste protocolo.

- Engadir funcións de *logging* ás pasarelas, para apoiar os logs xerais de XesLab e como medida para a consecución dos requisitos non funcionais relacionados coa seguridade.
- Cifrar as comunicacións entre o servidor e a pasarela.
- Engadir á pasarela unha pantalla, co obxectivo de mostrar un panel de información.
- Engadir á pasarela unha cámara, para poder capturar imaxes dos ensaios de xeito automatizado.
- Cargar configuracións na pasarela por medio dun ficheiro de configuración, de xeito que se poidan indicar dinamicamente parámetros como os datos de rede ou o número e tipo de módulos a cargar.
- Implementar tempos límite para os bloqueos dos dispositivos.

Estas ampliacións teñen unha parte de implementación correspondente a XesLab e outra na pasarela, xa que a idea é que a pasarela exerza como hardware de control de apoio a XesLab, e os resultados de todas as operacións deberían ser consultables dende este.



# Apéndice A

## Manuais técnicos

### A.1. Manual de instalación

As librarías `libb64` e `nlohmann::json` distribúense como código fonte, polo que non é preciso instalar nada no sistema; compílanse xunto co código do programa principal.

Non se mostra como instalar e configurar XesLab xa que está fóra do alcance deste proxecto.

#### A.1.1. Instalación dos drivers de GPIB

Adxúntase para a instalación da librería `linux-gpib` o seguinte *script*. Instala os drivers de GPIB de xeito compatible cunha Raspberry Pi 3 B+ con sistema operativo Raspbian Stretch 9.8 e versión de kernel 4.14.98-v7+.

O *script* instala a utilidade `rpi-source` para poder obter as cabeceiras do *kernel* de Raspbian, necesarias para compilar os drivers GPIB. Para iso, activa o repositorio de código fonte para APT, instala as dependencias de `rpi-source`, e instala e executa a utilidade.

```
# Activamos apt-get source
echo "deb-src_ftp://raspbian.raspberrypi.org/raspbian/_stretch_
main_contrib_non-free_rpi" >> /etc/apt/sources.list

# Instalamos dependencias
apt-get update
apt-get install libncurses5-dev bc tk-dev build-essential texinfo
texi2html libcwidget-dev libx11-dev bison flex binutils-dev
libusb-1.0-0 libusb-dev libmpfr-dev libexpat1-dev tofrodos
subversion autoconf automake libtool mercurial

# Instalamos y ejecutamos rpi-source para descargar las cabeceras
wget
https://raw.githubusercontent.com/notro/rpi-source/master/rpi-source
```

```

    -0 /usr/bin/rpi-source && sudo chmod +x /usr/bin/rpi-source &&
    /usr/bin/rpi-source --q --tag-update
rpi-source

```

Realiza un *checkout* do repositorio SVN de linux-gpib para obter o código fonte.

```

# Descargamos linux-gpib
mkdir linux-gpib
cd linux-gpib
svn checkout svn://svn.code.sf.net/p/linux-gpib/code/trunk
    linux-gpib-code
export VERBOSE=1
export LINUX_SRCDIR="/root/linux"

```

Constrúe as cabeceiras e compila o módulo de kernel de linux-gpib.

```

cd /root/linux
make oldconfig && make prepare

# Instalamos el módulo del kernel
cd /root/linux-gpib/linux-gpib-code/linux-gpib-kernel
make
make install

```

A continuación, compila as utilidades de usuario.

```

# Instalamos las utilidades de usuario
cd ../linux-gpib-user
./bootstrap
./configure
make
make install
cp util/templates/gpib.conf /etc/gpib.conf

```

Unha vez instalado linux-gpib, é preciso configurar e cargar o módulo GPIB adecuado. Hai que indicar no ficheiro de configuración `gpib.conf`, do que se pode atopar unha plantilla en `/etc` e que debe colocarse en `/usr/local/etc/gpib.conf`, o tipo de placa GPIB usado. Neste caso usouse o adaptador NI-USB-HS, polo que hai que asignar `board_type` ao valor `ni_usb_b` (o nome da familia de adaptadores).

A configuración do módulo GPIB cárgase co comando `gpib-config`, e o driver para a placa cárgase co comando `modprobe ni_usb_gpib`. Hai que modificar `crontab` para que en cada inicio do sistema se cargue o módulo e o driver.

```

# Configuramos y cargamos el módulo de GPIB
sed -i 's/*board_type = "ni_pci"*/board_type = "ni_usb_b"/'
    /etc/gpib.conf
cp /etc/gpib.conf /usr/local/etc/gpib.conf

modprobe ni_usb_gpib
gpib_config

crontab -l > crontab_old

```



```

echo "@reboot_/usr/local/sbin/gpib_config" >> crontab_old
echo "@reboot_modprobe_ni_usb_gpib" >> crontab_old
crontab crontab_old
rm crontab_old

```

O *script* eleva privilexios internamente xa que ten que acceder e modificar ficheiros do sistema.

### A.1.2. Instalación das librarías de Boost

Empréganse para as probas do software da pasarela. É preciso obter a versión máis recente de Boost da páxina [www.boost.org](http://www.boost.org) (para o desenvolvemento empregouse a 1.70.0). A efectos deste manual, asúmese que o directorio da instalación é /root.

```

# descargar boost
wget
  https://dl.bintray.com/boostorg/release/1.70.0/source/boost_1_70_0.tar.bz2
tar xjf boost_1_70_0.tar.bz2

```

Realízase a instalación habitual de Boost.

```

# instalar boost
cd boost_1_70_0
./bootstrap.sh
./b2 headers
./b2

```

É preciso actualizar a configuración de GCC e do enlazador dinámico para que se compilen e enlacen as novas librarías correctamente.

```

# actualizar configuración gcc
export CPATH=/root/boost_1_70_0:$CPATH
export LIBRARY_PATH=/root/boost_1_70_0/stage/lib:$LIBRARY_PATH
export LIBRARY_PATH=/root/boost_1_70_0/:$LIBRARY_PATH

# actualizar configuración ld
export LD_LIBRARY_PATH=/root/boost_1_70_0/stage/lib:$LD_LIBRARY_PATH

```

Engádense as configuracións do compilador e o enlazador a crontab.

```

crontab -l > crontab_old
echo "@reboot_export_CPATH=/root/boost_1_70_0:$CPATH" >> crontab_old
echo "@reboot_export_
  LIBRARY_PATH=/root/boost_1_70_0/stage/lib:$LIBRARY_PATH" >>
  crontab_old
echo "@reboot_export_
  LD_LIBRARY_PATH=/root/boost_1_70_0/stage/lib:$LD_LIBRARY_PATH"
  >> crontab_old
crontab crontab_old
rm crontab_old

```

### A.1.3. Compilación do código da pasarela

Supoñemos que o código C++ da pasarela está no directorio `/root/daemon_pasarela`.

Adxúntase unha *makefile* con diversas regras para unha compilación dirixida á depuración do programa ou á execución de probas.

```

MAIN=daemon_pasarela
CXX=g++
CXXFLAGS=-std=c++17 -Wno-narrowing -Wno-psabi -O0 -g
DEPS=b64/cdecode.h b64/cencode.h b64/decode.h b64/encode.h
      ethernet/GestorConexiones.h ethernet/Conexion.h
      ethernet/ExcepcionRed.h json/json.hpp mensajes/gestorMensajes.h
      mensajes/Mensaje.h mensajes/MensajeControl.h
      mensajes/MensajeGPIB.h modulos/GestorModulos.h modulos/Modulo.h
      modulos/ModuloGPIB.h controlador/Controlador.h
      controlador/ControladorHilo.h controlador/TipoComunicacion.h
      controlador/Utilidades.h controlador/Hilo.h test/test.h
LIBS=-lgpib -lpthread

%.o: %.c $(DEPS)
      $(CXX) -c -o $@ $< $(CXXFLAGS) $(LIBS)

%.o: %.cpp $(DEPS)
      $(CXX) -c -o $@ $< $(CXXFLAGS) $(LIBS)

```

### Depuración

Compílese coa regra `main`.

```

main: main.o b64/cencode.o b64/cdecode.o
      ethernet/GestorConexiones.o ethernet/Conexion.o
      mensajes/GestorMensajes.o mensajes/MensajeControl.o
      mensajes/MensajeGPIB.o modulos/GestorModulos.o
      modulos/ModuloGPIB.o controlador/Controlador.o
      controlador/ControladorHilo.o controlador/TipoComunicacion.o
      controlador/Utilidades.o controlador/Hilo.o
      $(CXX) $(CXXFLAGS) $(LIBS) -o $(MAIN) main.o b64/cencode.o
      b64/cdecode.o ethernet/GestorConexiones.o
      ethernet/Conexion.o mensajes/GestorMensajes.o
      mensajes/MensajeControl.o mensajes/MensajeGPIB.o
      modulos/GestorModulos.o modulos/ModuloGPIB.o
      controlador/Controlador.o controlador/ControladorHilo.o
      controlador/TipoComunicacion.o controlador/Utilidades.o
      controlador/Hilo.o

```

### Probas

Compílese coa regra `test`. Cabe destacar que non se enlaza o ficheiro `main.o` (xa que a libraría de probas provee a súa propia función `main`) e fai uso da opción `-lboost_unit_test_framework`.

Debido a certos problemas coa definición de macros a través de liña de comandos, é preciso activar manualmente a macro TEST do ficheiro Utilidades.h

```
test: test/test.o b64/cencode.o b64/cdecode.o
      ethernet/GestorConexiones.o ethernet/Conexion.o
      mensajes/GestorMensajes.o mensajes/MensajeControl.o
      mensajes/MensajeGPIB.o modulos/GestorModulos.o
      modulos/ModuloGPIB.o controlador/Controlador.o
      controlador/ControladorHilo.o controlador/TipoComunicacion.o
      controlador/Utilidades.o controlador/Hilo.o
      $(CXX) $(CXXFLAGS) $(LIBS) -lboost_unit_test_framework -o
      $(MAIN) test/test.o b64/cencode.o b64/cdecode.o
      ethernet/GestorConexiones.o ethernet/Conexion.o
      mensajes/GestorMensajes.o mensajes/MensajeControl.o
      mensajes/MensajeGPIB.o modulos/GestorModulos.o
      modulos/ModuloGPIB.o controlador/Controlador.o
      controlador/ControladorHilo.o
      controlador/TipoComunicacion.o controlador/Utilidades.o
      controlador/Hilo.o
```

## A.2. Engadir un novo módulo

Para engadir ao sistema a funcionalidade de comunicarse cun novo tipo de interface hardware, é preciso seguir os pasos que se relatan a continuación. De agora en diante, considerarase que <nome\_interface> contén o nome da interface.

- En XesLab:

1. Modificar a táboa de equipos na base de datos, para incluír un campo que almacene o enderezo secundario do equipo (ou calquera outro dato que sexa necesario para a comunicación a través desa interface). Modificar na base de datos os equipos que contén con esta interface para incluír os enderezos secundarios. O nome da columna que os almacene debe ser <nome\_interface>\_Direccion.
2. Modificar a táboa SISTEMA.RED da base de datos para que os equipos que se vaian a conectar a través da pasarela teñan a IP da pasarela.
3. Engadir "<nome\_interface>" a PROCESADO\_COMANDOS::\$ARRAY\_PRIORIDADES, na posición que se queira.
4. Crear o método PREPARACION\_COMANDOS::\$PREPARAR\_COMANDO\_<nome\_interface>. Debe devolver un obxecto JSON serializado e codificado en Base64, e ter polo menos os campos tipo (co valor <nome\_interface>) e contenido.

- Na pasarela:

1. Engadir <nome\_interface> aos posibles valores de TipoComunicacion. Editar os métodos `string_to_enum` e `enum_to_string` para contemplar o novo valor.
2. Instalar as librerías necesarias para esa interface e encapsular a invocación das súas funcións cunha clase chamada `Modulo<nome_interface>`, que estenda a clase abstracta `Modulo`.
3. Editar o construtor de `GestorModulos` para que se instancie `Modulo<nome_interface>`.
4. Escribir a clase `Mensaje<nome_interface>` que estenda a clase abstracta `Mensaje`.
5. Editar `GestorMensajes::desempaquetar` para que realice instanciación de `Mensaje<nome_interface>`.

# Apéndice B

## Manuais de usuario

Pódese acceder a XesLab a través da url `xeslab.ladetel`, dende un equipo conectado á rede de Televés.

### B.1. Envío de comandos de forma individual

Para enviar comandos de forma individual a un equipo, accédese á pantalla de equipos a través do menú lateral, pinchando en "Equipos" e posteriormente en "Todos", no menú despregable.

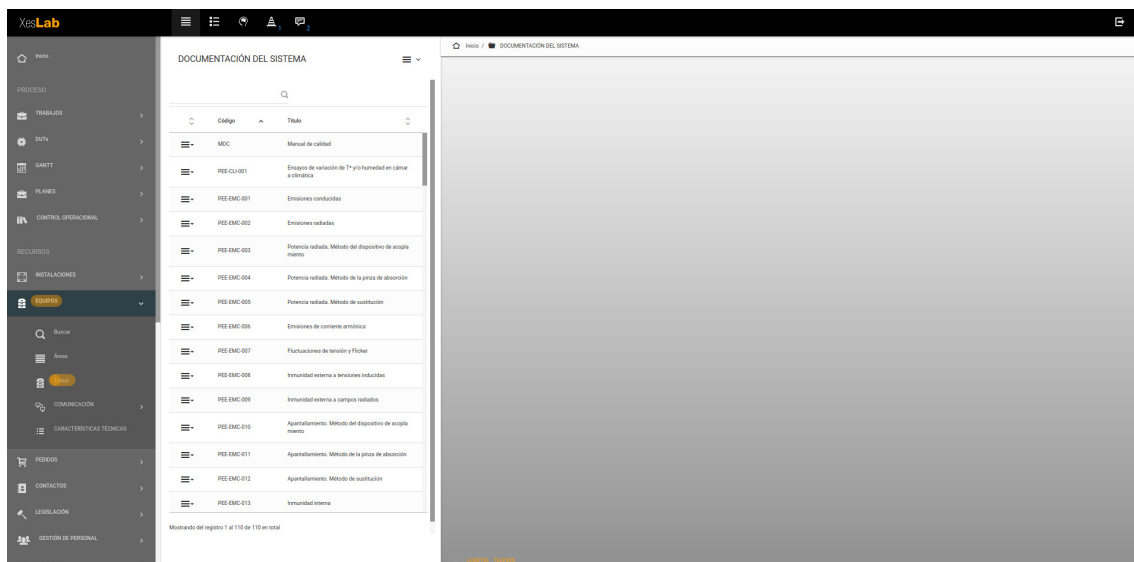


Figura B.1: Acceso á pantalla de equipos

A pantalla de equipos mostra a lista de todos os equipos do laboratorio, onde se poden buscar os equipos polo seu número de identificación, nome, ou outros campos.

-	Id	Nombre	Marca	Estado
☰	522	Generador de Señal MXG X-Series	Keysight Technologies	ACTIVO

Figura B.2: Lista de equipos e resultados do filtrado

No menú contextual do equipo, situado á esquerda, pódense ver todos os paneis de datos relacionados co equipo. Para enviar comandos, emprégase o panel titulado "Datos comunicación equipo".

**DATOS EQUIPO**

Estado: **ACTIVO** Id: 522

Nombre: **Generador de Señal MXG X-Series**

Nombre [Inglés]: **MXG X-Series RF Vector Signal Generator**

Marca: **Keysight Technologies** Modelo: **N5182B**

Nº de Serie: **MYS3051062**

Descripción: **Generador de señal MXG X-Series, 9kHz-60Hz.**

Código de inmovilizado: **MEM10036**

Software:

Nº de Carpetas: **1** Código de Inmovilizado:

Observaciones:

---

**INVENTARIO**

Fecha de Alta: **2017-02-02**

Observaciones al Alta:

Fecha de Baja:

**DATOS COMUNICACIÓN EQUIPO**

RED

Puerto de Comunicación: **5025**

MAC	IP	ESTADO	PUERTOS
00:e0:4b:48:91:26	192.168.180.44	OFFLINE	

**COMANDOS**

Comando	Variables	Resultado
*IDN?		<b>ENVIAR</b>

Mostrando del registro 1 al 1 de 1 en total

**DOCUMENTOS DE CALIBRACIÓN**

Código: **CC1701000** Origen: **EXTERNO** Información: **CERTIFICADO DE CALIBRACIÓN EXTERNO** Título:

Mostrando del registro 1 al 1 de 1 en total

**CARACTERÍSTICAS TÉCNICAS**

**ELEMENTOS DEL SISTEMA DE GESTIÓN**

**IMÁGENES**

**COMANDOS COMUNICACIÓN [EQUIPO]**

**ELEMENTOS RELACIONADOS**

**HISTORIAL**

**PROCEDIMIENTOS**

**ARCHIVOS**

**ENLACES**

Figura B.3: Datos do equipo 522

Este panel, mostrado na figura B.4, contén datos do equipo necesarios para a comunicación con el. Na táboa "Comandos", os botóns de cor laranxa permiten enviar ao equipo o comando indicado na columna da esquerda. Se é necesario

Inicio / EQUIPOS / Generador de Señal MXG X-Series

### DATOS COMUNICACIÓN EQUIPO

RED

Puerto de Comunicación: 5025

MAC	IP	ESTADO	PUERTOS	DESCRIPCIÓN
00:w04b:48:91:26	192.168.180.50	ONLINE	<ul style="list-style-type: none"> <li>SSH (25) ACCESIBLE</li> <li>PUERTO DE COMUNICACIÓN (5025) ACCESIBLE</li> </ul>	JUMP INDUSTRIELLE COMPUTERTECHNIK GmbH

GPIB

Dirección GPIB: 19

COMANDOS (4)

Comando	Variables	Resultado
FREQ? FREQUENCIA	<input type="text"/>	-
TRAC? STRAZA	<input type="text"/>	-
*IDN?	<input type="text"/>	-
FREQ?	<input type="text"/>	-

Mostrando del registro 1 al 4 de 4 en total

Figura B.4: Panel de comunicaciones do equipo 522

engadir unha parametrización, pódese escribir o parámetro no campo de texto da columna central.

Se o equipo devolve unha resposta, esta móstrase na columna da dereita.

Comando	Variables	Resultado
FREQ? FREQUENCIA	<input type="text" value="2 GHz"/>	OK
TRAC? STRAZA	<input type="text"/>	-
*IDN?	<input type="text"/>	Agilent Technologies, N5182B, MY53051062, B.01.58
FREQ?	<input type="text"/>	+2.000000000000000E+09

Figura B.5: Resultados do envío de comandos

Os erros poden mostrarse na columna de resultados, se proveñen da pasarela,

ou como unha alerta de excepción, se se trata dun erro de XesLab.

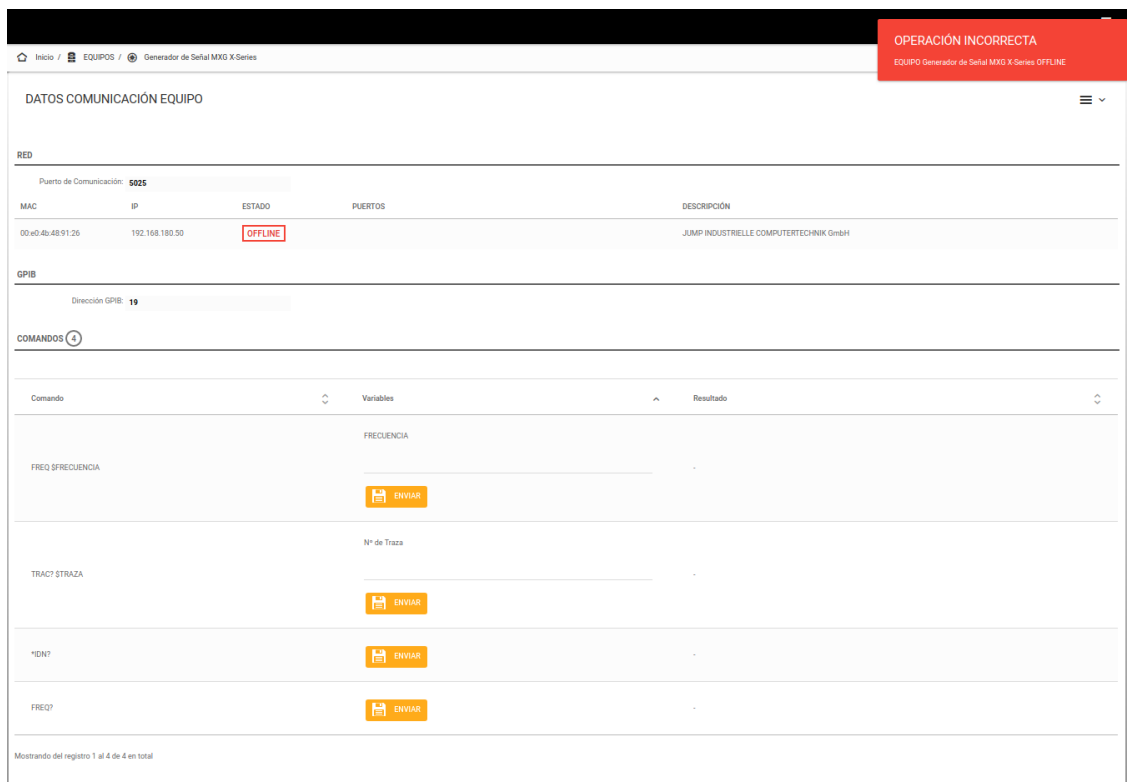


Figura B.6: Erro nos módulos de comunicacións en XesLab

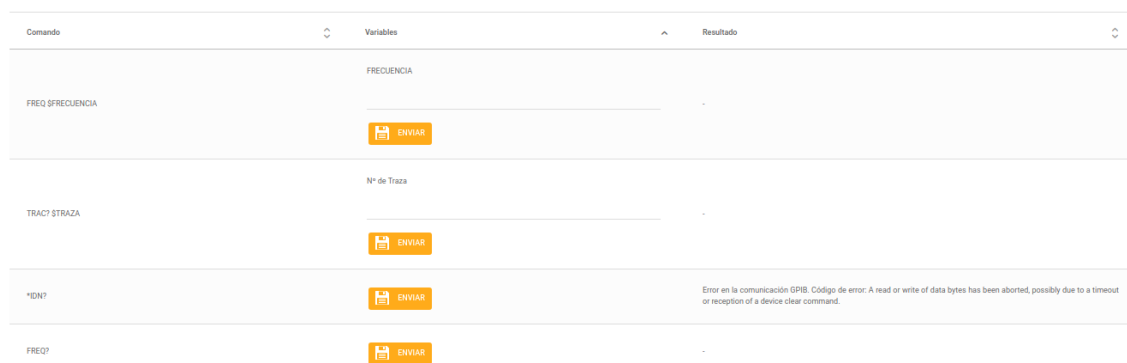


Figura B.7: Erro na pasarela

## B.2. Execución de medidas

A lista de procedementos móstrase na páxina principal de XesLab, `xeslab.ladetel`, unha vez iniciada a sesión. Tamén se pode acceder a ela a través do



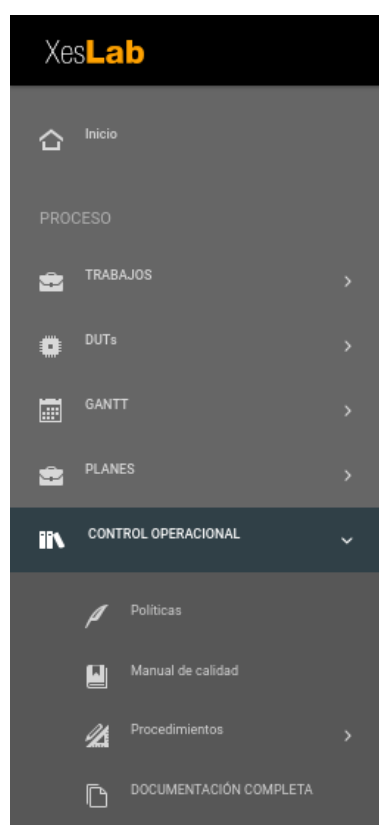


Figura B.8: Acceso á lista de procedimientos a través do menú contextual

menú lateral; é preciso abrir o despregable titulado “Control operacional”, na sección “Proceso”, e pinchar sobre “Documentación completa”. Para acceder aos datos dun procedemento, é preciso abrir o menú contextual premendo no icono que se atopa á esquerda do rexistro, e seleccionar “Ver”.

DOCUMENTACIÓN DEL SISTEMA ☰ ▾

Q

↕	Código	^	Título	↕
☰	MDC		Manual de calidad	
☰	PEE-CLI-001		Ensayos de variación de Tª y/o humedad en cámara climática	
☰	PEE-EMC-001		Emisiones conducidas	
☰	PEE-EMC-002		Emisiones radiadas	
☰	PEE-EMC-003		Potencia radiada. Método del dispositivo de acoplamiento	
☰	PEE-EMC-004		Potencia radiada. Método de la pinza de absorción	

Figura B.9: Lista de procedimientos

DOCUMENTACIÓN DEL SISTEMA ☰ ▾

emp Q

↕	Código	^	Título	↕
☰	PEE-EMP-001		Atenuación de apantallamiento en cables. Método triaxial.	

Ver

Figura B.10: Menú contextual do procedimiento

Na pantalla do procedemento móstranse varios paneis con información relativa a este elemento; só están despregados por defecto o panel de datos básicos, o panel de implementación e o panel da medida técnica asociada a ese procedemento. Neste último é posible establecer valores para os parámetros da medida; no momento en que se carga o panel énchese o formulario cos valores que establece o procedemento, pero os campos son modificables.

Os equipos que encaixan co perfil da medida (son do tipo que a medida solicita, e poden executar os comandos que a medida impón para ese tipo de equipo) que se atopan online no momento actual móstranse en selectores despregables neste mesmo panel, de xeito que nunca se pode executar unha medida con instrumentos que non sexan adecuados. Pódese seleccionar calquera dos instrumentos mostrados no despregable.

The screenshot shows a web-based interface for managing measurement procedures. The top navigation bar includes 'Inicio', 'DOCUMENTACIÓN DEL SISTEMA', and 'PEE-EMP-001'. The main content area is organized into several panels:

- DATOS PROCEDIMIENTO:** Contains fields for 'Tipo' (PROCEDIMIENTO ESPECIFICO DE ENSAYO), 'Área' (Electromagnéticos), 'SubÁrea' (Prestaciones EM), 'Título' (Atenuación de apantallamiento en cables. Método triaxial), 'DOCUMENTO' (PEE-EMP-001), and 'Descripción' (Atenuación de apantallamiento en cables. Método triaxial).
- CONTABILIDAD: CONFIGURACIÓN DE LÍNEAS:** A panel for line configuration.
- ZONAS/PEQs:** A panel for zones and points of interest.
- REQUISITOS DE CARACTERÍSTICAS TÉCNICAS:** A panel for technical characteristics requirements.
- REQUISITOS DE CARACTERÍSTICAS DE INSTALACIÓN:** A panel for installation characteristics requirements.
- ELEMENTOS DEL SISTEMA DE GESTIÓN:** A panel for system management elements.
- ELEMENTOS RELACIONADOS:** A panel for related elements.
- ARCHIVOS:** A panel for files.
- HISTORIAL:** A panel for history.
- IMPLEMENTACIONES DEL PEE:** A panel for PEE implementations.
- MEDIDA TÉCNICA (COMPROBACIÓN PEE):** A panel for technical measurement, containing:
  - VARIABLES DE LA MEDIDA:** A table with columns: Longitud del Tubo (m), Frecuencia [INICIO] (Hz), Frecuencia [FIN] (Hz), Nº de Puntos, Impedancia del cable, and Generador.
  - Receptor:** A dropdown menu showing 'Receptor de medida R&S ESCI (9KHz-3GHz) [id: 75]'.
  - ENVIAR:** An orange button to submit the data.

At the bottom left, there is a copyright notice: © 2015 LAOETEL-TELEVES.

Figura B.11: Pantalla do procedemento

MEDIDA TÉCNICA (COMPROBACIÓN PEE) ☰

---

**VARIABLES DE LA MEDIDA**

Longitud del Tubo (m)	Frecuencia [NFCIO] (Hz)	Frecuencia [FR] (Hz)	Nº de Puntos	Impedancia del cable	Generador
2	30000000.0	2150000000.0	801	50	▼ Generador de Señal MXG X-Series [Id: 522] ▼

Receptor

Receptor de medida R&S ESCI (9KHz-3GHz) [Id: 75] ▼

ENVIAR

Figura B.12: Panel da medida técnica

No momento en que se ordena a execución da medida, os resultados obtidos ata o momento comezan a mostrarse na parte inferior do panel.

MEDIDA TÉCNICA (COMPROBACIÓN PEE)

---

**VARIABLES DE LA MEDIDA**

Longitud del Tubo (m)	Frecuencia [NFCIO] (Hz)	Frecuencia [FR] (Hz)	Nº de Puntos	Impedancia del cable	Generador
2	30000000.0	2150000000.0	801	50	▼ Generador de Señal MXG X-Series [Id: 522] ▼

Receptor

Receptor de medida R&S ESCI (9KHz-3GHz) [Id: 75] ▼

ENVIAR

30000000.0Hz  
 32646691.635456Hz  
 35293383.270911Hz  
 37940074.906367Hz  
 40586716.541823Hz  
 43233458.172778Hz  
 45880149.812734Hz  
 48526841.44819Hz  
 51173533.083645Hz  
 53820224.719101Hz  
 56466916.354557Hz  
 59113607.990013Hz  
 61760299.625468Hz

Figura B.13: Resultados da medida, obtidos en tempo real

# Bibliografía

- [1] ISO: *General requirements for the competence of testing and calibration laboratories. ISO 17025:2017*. Estándar, International Organization for Standardization, Xenebra, Abril 2018. <https://www.iso.org/standard/66912.html>.
- [2] Daponte Villanueva, Olalla e Otero López, Pilar: *Especificaciones de la herramienta de calidad del laboratorio acorde a ISO 17025*. Informe técnico, Televés Corporation, Santiago de Compostela, Outubro 2018.
- [3] IEEE: *IEEE Standard For Higher Performance Protocol for the Standard Digital Interface for Programmable Instrumentation*. Estándar, Institute of Electrical and Electronics Engineers, Piscataway, Xuño 2003. <https://www.iso.org/standard/66912.html>.
- [4] IEEE: *IEEE Standard Codes, Formats, Protocols, and Common Commands for Use With IEEE Std 488.1-1987, IEEE Standard Digital Interface for Programmable Instrumentation*. Estándar, Institute of Electrical and Electronics Engineers, Piscataway, Xuño 1992. <https://standards.ieee.org/standard/488.2-1992.html>.
- [5] *Gestión de riesgos en proyectos software*. Informe técnico.
- [6] Daponte Villanueva, Olalla: *Control de estado CORP4FUTURE*. Informe técnico, Xuño 2019.
- [7] Servizo de Xestión de Persoal da USC: *Táboas de retribucións de persoal docente e investigador*. <http://www.usc.es/gl/servizos/sxp/taboas/tabPDI19.html>, [En liña; consultado o 25 de xuño de 2019], 2019.
- [8] *Resolución pola que se inscribe no Rexistro e se da publicidade ao convenio colectivo da empresa Televés, S.A.* Boletín Oficial de A Coruña, (39), Febreiro 2017. [En liña; consultado o 25 de xuño de 2019].
- [9] *Manual de PHP*, Xullo 2019. <https://www.php.net/manual/es/index.php>, [En liña; consultado o 11 de xullo de 2019].

- [10] Stroustrup, Bjarne: *The C++ Programming Language*. Pearson Education, Upper Saddle River, 4ª edición, 2013, ISBN 978-0-321-56384-2.
- [11] *Using the National Instruments GPIB-USB-B on Linux*. [https://www.cl.cam.ac.uk/~osc22/tutorials/gpib\\_usb\\_linux.html](https://www.cl.cam.ac.uk/~osc22/tutorials/gpib_usb_linux.html), [En liña; consultado o 15 de abril de 2019].
- [12] Venter, Chris: *libb64: Base64 Encoding/Decoding Routines*. <http://libb64.sourceforge.net/>, [En liña; consultado o 18 de abril de 2019].
- [13] Rozental, Gennadiy e Enficiaud, Raffi: *Boost.Test - 1.70.0*, Abril 2019. [https://www.boost.org/doc/libs/1\\_70\\_0/libs/test/doc/html/index.html](https://www.boost.org/doc/libs/1_70_0/libs/test/doc/html/index.html), [En liña; consultado o 8 de xullo de 2019].
- [14] Lohmann, Niels: *JSON for Modern C++*, 2019. <https://nlohmann.github.io/json/>, [En liña; consultado o 23 de maio de 2019].
- [15] Bergmann, Sebastian: *PHPUnit 6.0 Manual*, 2017. <https://phpunit.de/manual/6.5/en/index.html/>, [En liña; consultado o 8 de xullo de 2019].