



Anexo Ia

Prego de Especificacións Técnicas

Campus Móbil

Traballo coordinado polo **Equipo TIC da Universidade da Coruña**

Coordinador responsable: **D. Víctor Carneiro**

Técnico responsable: **D. Antonio Ventura**

20/07/2015

V. 2015-04



Universidade de Vigo

 The logo for CIXUG, featuring a stylized 'C' with three overlapping loops in pink, blue, and green, followed by the text 'CIXUG' in a bold, green, sans-serif font.	20/07/2015 V. 2015-04
Denominación: • Campus Móbil, Backend xenérico e aplicación de notificacións	
Unidade Responsable: • Universidade da Coruña ○ Resp. Coordinación: Víctor Carneiro ○ Resp. Técnico: Antonio Ventura	
Obxecto: • Este documento describe as funcionalidades propostas pola Universidade da Coruña, como unidade responsable do proxecto, ao módulo de Notificacións Móbil da Xestión Universitaria para o alumnado, PDI e PAS das Universidades que conforman o Sistema Universitario Galego.	
Lexislación de referencia: • Non aplica.	

Índice

1	Introdución	1
1.1	Obxectivo e alcance	1
2	Solución proposta	2
2.1	Contorna e arquitectura.....	2
2.1.1	Tecnoloxías Base	2
2.2	Backend xenérico.....	2
2.2.1	Tipos de notificacións	4
2.2.2	Mecanismos de notificación	4
2.2.3	API de rexistro e configuración de servizos	5
2.2.4	API de publicación	5
2.2.5	API de consumo e configuración	6
2.2.6	Servizo de usuarios e grupos	6
2.2.7	Servizo de SMS	6
2.2.8	Servizo de correo	6
2.2.9	Servizo de notificacións PUSH	6
2.2.10	Arquitectura do core do backend	6
2.2.11	Outras consultas	9
2.3	Implantación do backend na SUG.....	10
2.3.1	Adaptadores	10
2.3.2	Aplicación web de administración	10
2.3.3	Aplicación web de usuarios	11
2.4	Apps móbiles.....	12

Ilustracións

Ilustración 1, Visión global do sistema	3
Ilustración 2, Arquitectura do core.....	7
Ilustración 3, Interface de administración.....	11
Ilustración 4, Interface do usuario	12
Ilustración 5, Interface de configuración do usuario	13
Ilustración 6, Apps móbil.....	14

1 Introducción

Este documento recolle a proposta do SUG para as universidades do Sistema Universitario Galego (SUG) a través do Consorcio para o desenvolvemento de aplicacións para a Xestión Universitaria (CIXUG), para o desenvolvemento dun backend xenérico de notificación e unha serie de aplicacións web e móbiles para consultar e operar sobre as devanditas notificacións.

1.1 Obxectivo e alcance

O obxectivo principal do proxecto é proporcionar un mecanismo xenérico que permita aos diferentes sistemas informáticos do SUG xerar notificacións e que estas poidan ser entregadas aos usuarios do SUG (alumnos, PDI, PAS, etc) por diferentes mecanismos como SMS, correo electrónico ou mediante unhas aplicacións específicas para web e smartphones.

Este sistema debe ser administrable, configurable, modular, escalable e tolerante a fallos. A primeira implementación do mesmo debe dar soporte ás aplicacións do SUG que xa están a realizar notificacións, respectando os seus mecanismos de notificación actuais de correo electrónico, SMS, etc.

O alcance do proxecto, explicado con detalle no corpo do seguinte documento inclúe:

- Desenvolvemento e despregamento dun backend xenérico cos seus APIs para permitir a configuración, publicación e consumo de notificacións.
- Desenvolvemento dunha web de administración.
- Desenvolvemento dunha web para usuarios.
- Desenvolvemento dunha app para usuarios con smartphone iOS.
- Desenvolvemento dunha app para usuarios con smartphone Android.

As aplicacións desenvolvidas terán branding do SUG e utilizarán a infraestrutura xenérica do backend de notificación.

2 Solución proposta

2.1 Contorna e arquitectura

2.1.1 Tecnoloxías Base

- Contorna de desenvolvemento
 - Git para desenvolvemento local e repositorio de integración de código do SUG (Git ou SVN).
- Backend
 - Linguaxe JAVA.
 - Xestor de colas
 - APIs de publicación/consumo de notificacións seguindo a paradigma REST e o formato de datos JSON.
- Frontend web
 - Linguaxe JAVA, framework web baseado en tecnoloxías JAVA
 - HTML5, CSS3 e JavaScript.
 - Aplicacións web integradas co servidor de autenticación centralizada do SUG (CAS).
- App iOS
 - Linguaxe de programación Swift.
 - OS soportados: iOS7 e superiores.
- App Android
 - Linguaxe Java.
 - OS soportados: KitKat (Versión 4.4) e superiores.
- Fontes de datos
 - Base de datos específica para o backend: Oracle ou MySQL.
 - LDAP do sistema de autenticación única do SUG.

2.2 Backend xenérico

A continuación defínense os conceptos craves do sistema e descríbense as diferentes partes que compoñen o backend, tanto a nivel funcional como de arquitectura. A Ilustración 1 ofrece unha visión global do sistema:

- Existirá un backend xenérico encargado de procesar as notificacións.

- Este backend publicará diferentes APIs REST para que outros servizos poidan enviar notificacións, para que unha aplicación de administración poida dar de alta devanditos servizos e os seus permisos, e para que outras aplicacións (web, móbiles, etc.) permitan aos usuarios visualizar as notificacións e configurar cales e como as reciben.
- O backend adaptará unha serie de servizos externos que lle permitan solicitar información de usuarios e utilizar mecanismos concretos de notificación.
- Ademais das notificacións, o sistema contemplará opcións para obter outro tipo de información dinámica ou estática como:
 - Consulta do expediente académico
 - Consulta do plano de estudos
 - Consulta dos horarios de clase e titorías
 - ...

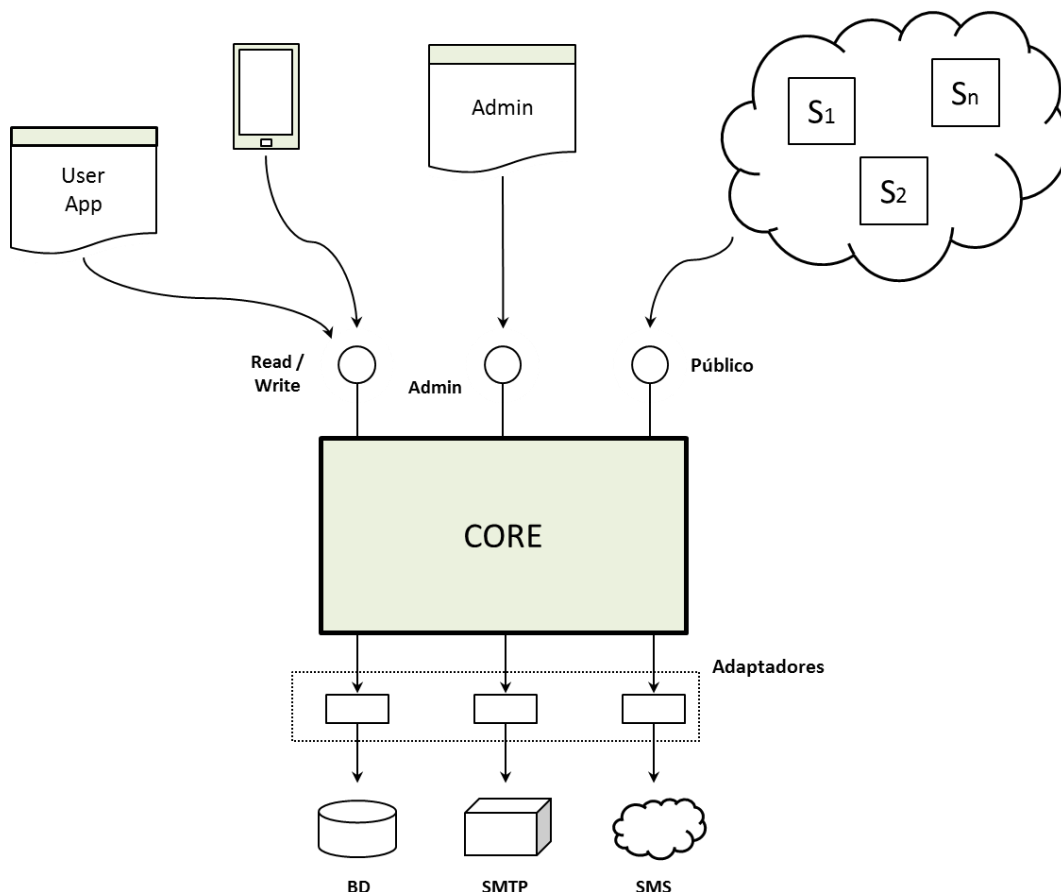


Ilustración 1, Visión global del sistema

2.2.1 Tipos de notificacións

Defínense tres tipos:

- Notificación
- Aviso
- Evento

As notificacións e avisos utilizaranse para mensaxes que deben ser enviados instantaneamente ao usuario, como poden ser as calificacións dunha materia. Semanticamente, un aviso é máis importante que unha notificación e utilizarase cando se desexe algún tipo de feedback por parte do usuario (por exemplo, marcar como lida nas aplicacións web/móviles). Os eventos definirán unha data de inicio e utilizaranse para notificar eventos de interese futuro como, por exemplo, un curso ou unhas xornadas. Os eventos comezarán a notificarse no momento en que o backend procésoos (sen ter en conta a data de inicio, que se usa só a título informativo).

Para todos os tipos de notificacións definirase un texto informativo (con opcións básicas de formato) e, opcionalmente, unha URL onde obter máis información.

Poderanse definir textos informativos específicos para os diferentes servizos de notificación para que as aplicacións que emitan notificacións poidan construír mensaxes adecuadas aos límites do servizo (p.ex. limitacións de caracteres no caso de SMS).

2.2.2 Mecanismos de notificación

Defínense inicialmente os seguintes mecanismos de notificación:

- Aplicación
 - A notificación pódese ver a través dunha aplicación, xa sexa unha web ou unha app para smartphone.
- Push
 - O usuario recibe unha notificación no seu móbil, cando pulsa na notificación ábrese a app móbil.
- E-mail
 - A notificación envíase a través do correo electrónico.
- SMS
 - A notificación envíase nun SMS.

Como se indicou no apartado anterior, os tipos de notificacións conterán un

texto informativo xenérico que se adaptará automaticamente ao mecanismo de notificación. A aplicación emisora poderá, no entanto, definir un texto específico para un mecanismo. Por exemplo, o texto recortarase se se envía vía SMS para axustalo a un límite de caracteres de maneira automática a menos que se defina un texto específico por parte da aplicación cliente.

2.2.3 API de rexistro e configuración de servizos

API REST (JSON) que proporcionará operacións para dar de alta servizos no sistema así como configuralos.

Para cada servizo indicarse que notificacións, e de que tipo, pode utilizar, e para cada unha delas, que mecanismos de notificación poderá usar e cales usará por defecto (se o usuario que recibe as notificacións non configurou as súas preferencias).

Unha vez que un servizo sexa rexistrado xeraranse unhas credenciais (tokens) que o servizo terá que utilizar para autenticarse á hora de utilizar o API de publicación.

2.2.4 API de publicación

Os diferentes servizos rexistrados no sistema utilizarán este API REST (JSON) para publicar notificacións. Para iso deberán rexistrarse previamente co API de rexistro e utilizar as credenciais adecuadas.

O API permitirá as seguintes operacións:

- Enviar unha notificación a un usuario concreto
 - Enviar unha notificación a un grupo de usuarios
 - Enviar un aviso a un usuario concreto
 - Enviar un aviso a un grupo de usuarios
 - Crear un evento para un usuario concreto
 - Crear un evento para un grupo de usuarios

O tipo de notificación que se queira publicar debe estar rexistrado no sistema previamente. A identidade do usuario así como os usuarios que corresponden a grupos extraeranse dun servizo xenérico de usuarios e grupos.

Con este API os servizos poderían realizar operacións como:

- Avisa CALIFICACION("8 en EDI... ") a "juan nadie"
- Notifica ACTAS("apertura de Redes... ") a "juand .nadie"
- Crea evento MATRICULACION("1er prazo de matrícula... ", 01/09/2015) para o grupo "alumnado".
- Crea evento CONSELLO_DE_FACULTADE("Febrero... ", 10/02/2015) para o grupo "pdi_facultade_ff".

Backend xenérico e aplicación de notificacións

2.2.5 API de consumo e configuración

As aplicacións web e as apps móbiles necesitan un API REST (JSON) para consumir notificacións ou eventos, e para configurar as notificacións que o usuario desexa recibir.

O API proporcionará as seguintes operacións:

- Listaxe de avisos (con algún mecanismo de paxinado)
- Detalle de aviso
- Listaxe de notificacións (con algún mecanismo de paxinado)
- Detalle de notificación
- Listaxe de eventos (nun rango de datas)
- Detalle de evento
- Listaxe de notificacións posibles e configuración para o usuario
- Activación de tipo de notificación
- Desactivación de tipo de notificación
- Asociar un dispositivo móbil a un identificador de usuario

2.2.6 Servizo de usuarios e grupos

Permitirá obter de forma xenérica información acerca de usuarios e grupos. A implementación concreta dependerá da implantación. As operacións deste servizo son:

- Obter información de contacto dun usuario asociada a un identificador de usuario.
- Obter información de contacto de todos os usuarios pertencentes a un grupo asociado a un identificador de grupo.

2.2.7 Servizo de SMS

Permitirá enviar de forma xenérica un SMS a un teléfono móbil. A implementación concreta dependerá da implantación.

2.2.8 Servizo de correo

Permitirá enviar de forma xenérica un correo a unha dirección de correo en formato texto e/ou formato HTML.

2.2.9 Servizo de notificacións PUSH

Permitirá enviar unha notificación a un dispositivo móbil.

2.2.10 Arquitectura do core do backend

A continuación coméntase a arquitectura do backend xenérico (Ilustración 2).

a) Base de datos e modelo

Modelo de datos con BD propia que almacenará as notificacións publicadas polo sistema e a configuración dos diferentes servizos rexistrados así como as preferencias de notificación dos diferentes usuarios do sistema.

b) API RESTs

APIs REST JSON protexidas vía https para:

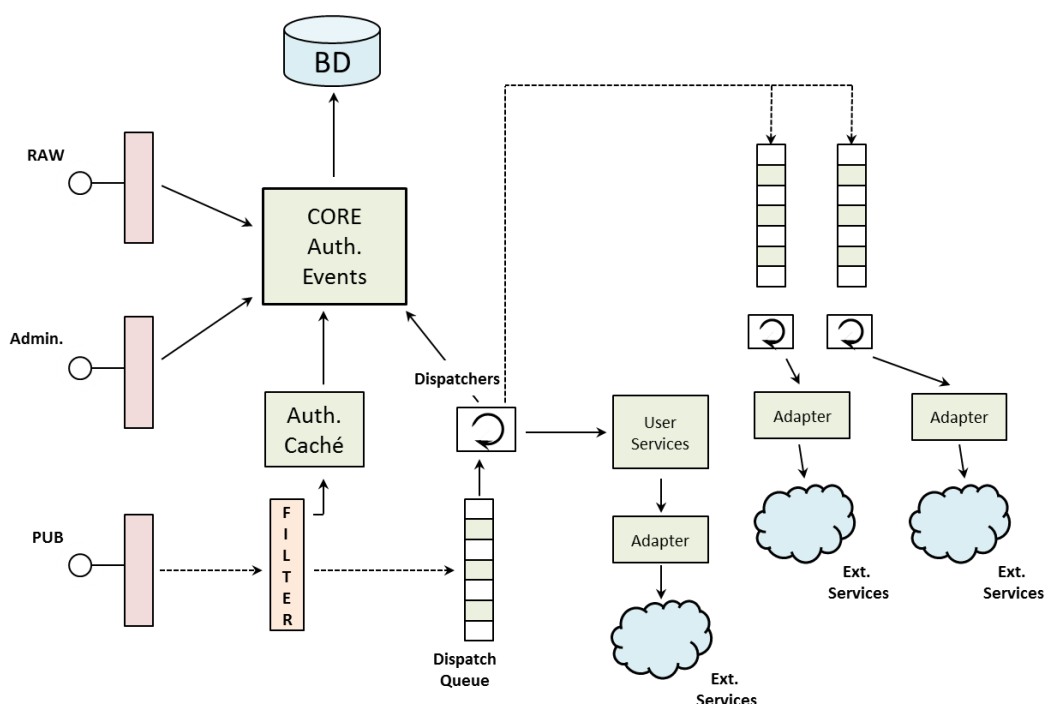


Ilustración 2, Arquitectura do core

- Rexistro e configuración de servizos no core. Utilizará un mecanismo de autenticación mediante intercambio de tokens de sesión de usuario.
- Consultar notificacións/avisos/eventos e configurar preferencias dun usuario. Utilizará un mecanismo de autenticación mediante intercambio de tokens de sesión de usuario.
- Publicar notificacións/avisos/eventos para usuarios ou grupos. Utilizará un mecanismo de autenticación mediante tokens de servizo.

Todas as APIs utilizarán estratexias de versionado e de autenticación por token usando cabeceiras HTTP. Así mesmo, cando sexa conveniente por rendemento se implementarán caches intermedias para evitar o acceso á BD (por exemplo, cache de credenciais e configuración de servizos no API de publicación).

c) Servizo de usuarios

Servizo de directorio que permite obter información de usuarios e grupos. Utilizará un adaptador configurable e enchufable para consumir a información dalgún servizo externo (base de datos, LDPA, ActiveDirectory, etc).

d) Cola e sistema de dispatching

As peticións recibidas a través do API de publicación serán encoladas nunha cola específica para que un ou máis procesos de dispatching procesenas: en función da configuración do servizo e das preferencias de usuario (obtidas do servizo de usuarios) as peticións serán traducidas e reencoladas en colas específicas para cada mecanismo de notificación.

e) Colas e mecanismos de notificación

Cada mecanismo de notificación disporá dunha cola propia, que irá consumindo. Os mecanismos de notificación utilizarán diferentes adaptadores (configurables e enchufables) para acceder ao servizo final de notificación (por exemplo, un servidor SMTP de correo ou un gateway SMS).

f) Exemplo de fluxo dentro do backend

Con esta arquitectura, unha invocación como:

- “Crea evento MATRICULACION (“1er prazo de matrícula...”, 01/09/2015) para o grupo alumnado” desencadearía o seguinte proceso no sistema:
 - Recíbese a mensaxe dende un servizo externo.
 - Chega a petición dende un servizo do SUG.
 - Confírmase que o token do servizo é correcto.
 - Compróbase a configuración do servizo e búscase o tipo MATRICULACION.
 - Encolase a petición na cola de dispatching e envíase un ACK ao servizo.
- Asincronamente:
 - O dispatcher colle a petición da cola e en función da configuración da notificación (neste caso, por exemplo vía correo electrónico), e a información de usuarios asociada ao grupo “alumno”, engade á cola do mecanismo de correo unha petición por usuario do grupo.
 - O procesador da cola de correo vai consumindo as peticións e enviando un correo a cada usuario.

g) Vantaxes da arquitectura

Esta arquitectura proporciona, entre outras vantaxes:

- Escalabilidade e tolerancia a fallos
 - Todos os compoñentes do sistema son replicables e distribuíbles.
- O tempo de resposta do API de publicación é constante e baixo grazas ao procesado asíncrono das notificacións. Os pescozos de botella do sistema asíncrono poderían ser:
 - A BD: con políticas de caché adecuadas e no futuro algún mecanismo de “clustering”/“sharding” podería liquidarse.
 - Os servizos finais de notificación: SMTP de correo e gateway de SMS. Pode ser necesario implementar mecanismos de “throttling”. Nun sistema baseado en colas o “throttling” vén de serie e só é necesario configurar a frecuencia coa que se procesan os traballos encolados.
- Soporte para reintentos.
 - En caso de fallo dos mecanismos de notificación poden impleamentarse de maneira transparente mecanismo de reintento.
- Implantación progresiva
 - O sistema é modular e unha vez montado o core do sistema é posible ir incorporando mecanismos de notificación progresivamente. Así mesmo, pódese pospor a implantación das aplicacións web de usuario ou as apps móbiles xa que son elementos satélite do sistema.

2.2.11 Outras consultas

- Ademais das notificacións, o sistema contemplará opcións para obter, a lo menos, a seguinte información:
- Consulta do expediente académico previo control de acceso
- Consulta do plano de estudos
- Presentación dos distintos centros, servizos e órganos da universidade, así como datos de contacto.
- Visualización de información sobre transportes públicos para poder chegar aos diversos campus
- Mapas de localización de POIs de cada universidade
- Acceso ao directorio telefónico e de contactos dos distintos campus
- Consulta dos horarios de clase e titorías
- Consulta de reserva de espazos

A aplicación deberá poder establecer idioma, activar e desactivar notificacións e avisos así como recordar os datos de acceso de usuario.

2.3 Implantación do backend na SUG

Para implantar o sistema é necesario desenvolver os adaptadores apropiados tanto para consumir información de usuarios e grupos como para utilizar mecanismos específicos de notificación, e desenvolver as aplicacións web/móviles para consumir eventos e configurar o sistema.

Inicialmente só será necesario dar soporte aos casos de uso de notificación presentes actualmente na SUG: notificacións vía correo electrónico e SMS.

2.3.1 Adaptadores

Desenvolveranse os seguintes adaptadores:

a) Adaptador de usuarios e grupos:

Permitirá obter información dos usuarios (correo, teléfono móbil, etc.) mediante un identificador único. Así mesmo permitirá obter información agregada de todos os usuarios pertencentes a un grupo. Utilizarase o LDAP de Federico ou calquera outro utilizado no SUG como fonte de datos, usando o ide único de usuario. Utilizarase como identificador de grupo calquera rol presente en xestor de identidade.

b) Adaptador do servizo de correo

Permitirá enviar un correo (indicando destinatario, asunto, corpo de texto e corpo HTML) usando o SMTP do SUG.

c) Adaptador do servizo de SMS

Permitirá enviar un SMS (indicando destinatario e contido) usando o servizo web de envío de SMSs utilizado actualmente polo sistema de notificación de cualificacións.

2.3.2 Aplicación web de administración

Esta plataforma (Ilustración 3) permitirá a un administrador do sistema definir novas aplicacións ou servizos emisores de notificacións. Ao dar de alta cada aplicación/servizo xeraranse unhas credenciais de autenticación necesarias para que se poidan utilizar o API de notificacións do backend. Cada aplicación/servizo poderá definir un conxunto de tipos de avisos, notificacións ou eventos. Para cada tipo fixarase que mecanismos de notificación pode elixir o usuario para recibir as notificacións e cales se usan por defecto se o usuario non establece a súa propia configuración. Esta aplicación utilizará o API REST de administración.

Tanto a autenticación como a autorización da interface estarán integradas co CAS do SUG, e manterase o look & feel e branding do resto de aplicacións de cada universidade.

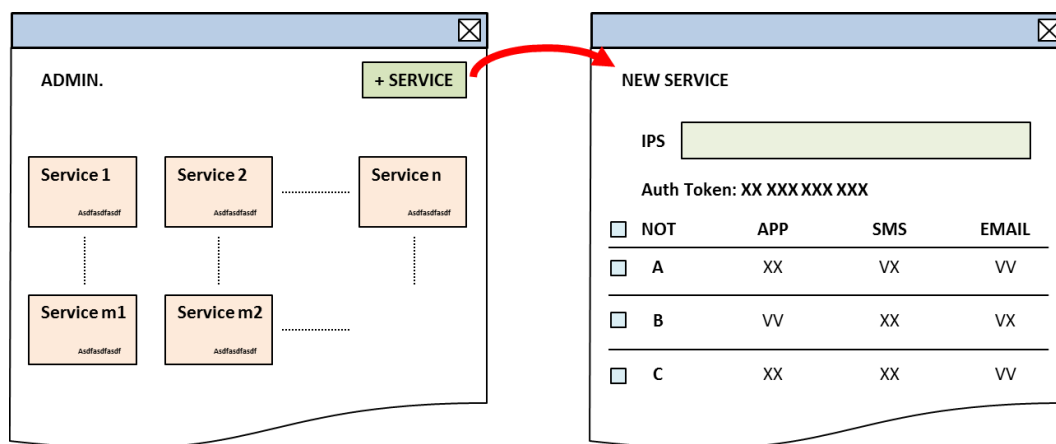


Ilustración 3, Interface de administración

2.3.3 Aplicación web de usuarios

Esta aplicación (ver Ilustración 4 e Ilustración 5) permitirá aos usuarios:

- Navegar por unha lista de avisos relevantes para o usuario e marcalos como lidos/non lidos.
- Navegar por unha lista de notificacións relevantes para o usuario.
- Visualizar nun calendario unha lista de eventos relevantes para o usuario.
- Ver o detalle dun aviso/notificación/evento.
- Configurar que avisos, notificacións e eventos son notificados ao usuario e como (é dicir, mediante que mecanismos).
- Acceder á información indicada no apartado 2.2.11.

Tanto a autenticación como a autorización da aplicación web estarán integradas co CAS do SUG, e manterase o look & feel e branding do resto de aplicacións do SUG.

Esta aplicación utilizará o API REST de consumo e configuración.

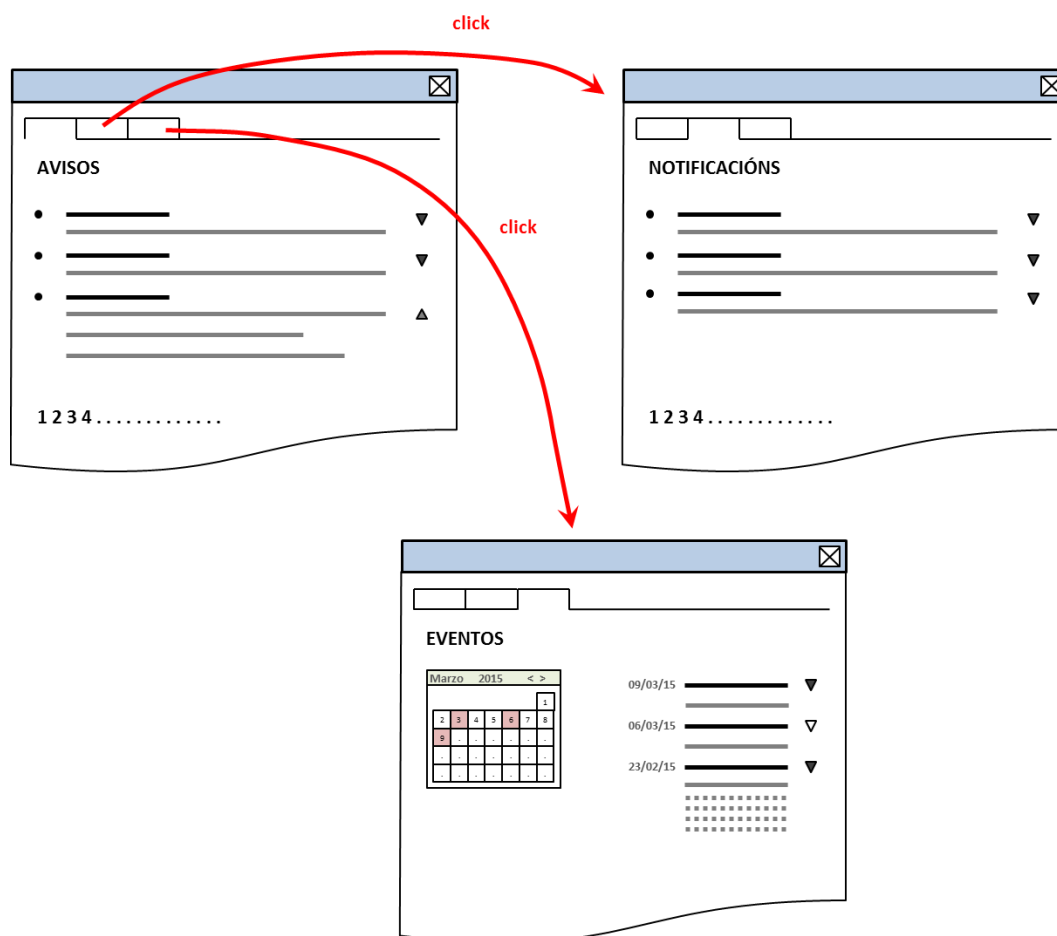
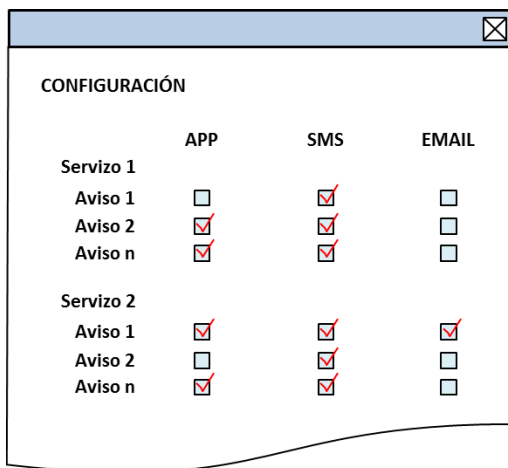


Ilustración 4, Interface do usuario

2.4 Apps móbiles

Desenvolveranse dúas aplicacións xemelgas para plataformas iOS e Android. Ambas as aplicacións terán a mesma funcionalidade pero os seus interfaces de usuario adaptaranse ás guías de estilos de cada plataforma. Non se desenvolverán compoñentes especiais para a aplicación que modifiquen o look and feel nativo: o branding visual das aplicacións limitarase ao uso dos logotipos e cores corporativas segundo a guía de identidade visual do SUG.



	APP	SMS	EMAIL
Servizo 1			
Aviso 1	☐	☒	☐
Aviso 2	☒	☒	☐
Aviso n	☒	☒	☐
Servizo 2			
Aviso 1	☒	☒	☒
Aviso 2	☐	☒	☐
Aviso n	☒	☒	☐

Ilustración 5, Interface de configuración do usuario

As aplicacións permitirán aos usuarios (Ilustración 6):

- Vincular o dispositivo móbil á súa conta de usuario do xestor de identidade: o usuario se autenticará utilizando as súas credenciais únicas.
- Navegar por unha lista de avisos relevantes para o usuario e marcalas como lidas/non lidas.
- Navegar por unha lista de notificacións relevantes para o usuario.
- Visualizar nun calendario unha lista de eventos relevantes para o usuario.
- Ver o detalle dun aviso/notificación/evento.
- Configurar que avisos, notificacións e eventos son notificados ao usuario e como (de modo análogo a como se fai na interface web).
- Configurar de maneira rápida como se recibe unha notificación na app dende a vista de detalle (“Non ver máis este tipo de notificación”).
- Acceder á información indicada no apartado 2.2.11.

Ambas as aplicacións utilizarán o API REST de consumo e configuración.

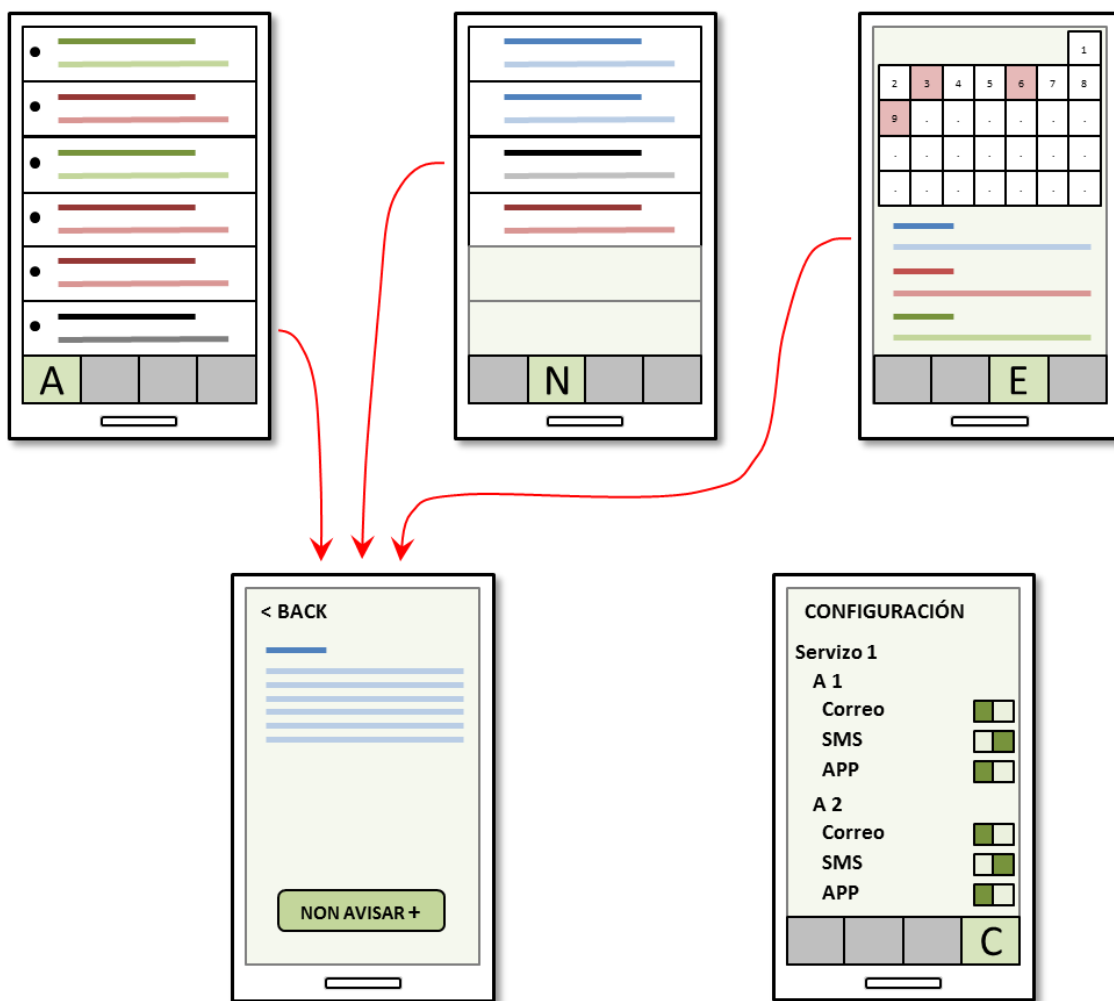


Ilustración 6, Apps móbil