# Seguretat de la pàgina web
# V1.0

*VIA - Vilanova Intelligent Airport*

**Index:**

## Introducció:

En aquest document s'explicaran els requeriments de seguretat que es proposen per implementar a la pàgina web, els quals s'aniran modificant al rebre el feedback del equip.
També servirà per mantenir un changelog amb els canvis que es vagin fent. (Al final del document).

## HTTPs:

**Per què ho volem:**

És imprescindible l'ús d'HTTPs en comptes de HTTP. HTTPs xifra les dades que s'envien entre el client i el servidor, cosa que HTTP no fa.

Per altra banda, cada cop més navegadors, adverteixen que no es segur al entrar en una web HTTP. Això porta al usuari a desconfiar de la web, cosa que no volem.

Finalment, Google dóna prioritat la les pàgines HTTPs a l'hora de mostrar els resultats d'una cerca.

**Com implementar-lo:**

Per poder utilitzar HTTPs, necesitem un certificat SSL. Un certificat SSL, el que fa, és autenticar l'identitat de la pàgina web (garantitzant als usuaris que no estan en una web falsa) i xifra la informació.

Recomanem utilitzar la web Let's Encrypt (https://letsencrypt.org/es/)
Es una web gratuita, oberta i automatitzada, que compta amb el suport i patrocini de gegants tecnològics com Google, Mozilla, Cisco...

Un cop tinguem el certificat SSL, farà falta implementar-lo a la web. Podem utilitzar aquesta implementació d'un client que automatitza el procés en PHP (https://community.letsencrypt.org/t/a-lets-encrypt-php-client-for-complete-automation-issue-renew-and-install-of-free-ssl-certificates-in-cpanel-shared-hosting/75599).

## Logins:

**Usuaris:**

Hi haurà dos tipus d'usuaris:

-Clients de l'aeroport:

·User: Hauran d'utilitzar el correu amb el que s'han registrat.

·Pass: La contrasenya haurà de tenir les següents característiques:

-Ha de tenir com a mínim 8 caràcters.

-Ha de tenir com a mínim una lletra minúscula i una majúscula.

-Ha de tenir com a mínim un número.

-S'accepten símbols.

·Quan un client hagi de crear la seva contrasenya, hi ha un sistema que comprovi que tots els camps anterior es compleixen.

-Administradors:

·User: Pot ser un nom d'usuari identificatiu. (no te sentit que sigui un correu).

·Pass: La contrasenya haurà de tenir les següents característiques:

-Ha de tenir com a mínim 12 caràcters.

-Ha de tenir una barreja de lletres minúscules i majúscules, símbols, números.

-No pot contenir paraules explícites.

·A diferència dels clients, no hi haurà un sistema de comprovació al introduir una contrasenya d'administrador. És responsabilitat d'aquest establir una contrasenya segura seguint les pautes anteriorment exposades.

**Captcha:**

Per altra banda, és necessària l'implementació d'un captcha que s'ha de validar cada cop que un usuari vulgui fer login (ja sigui client o administrador). Això evitarà atacs externs que intentin petar contrasenyes utilitzant força bruta.

El captcha més utilitzat actualment és el Google Captcha, i es pot implementar en PHP. En aquesta web s'explica al detall com fer-ho: (https://www.esthersola.com/ejemplo-implementar-google-captcha-php/)

I si no volem utilitzar google captcha i preferim utilitzar un captcha tradicional (de codi), ho podem fer utilitzant aquesta web: (https://code.tutsplus.com/es/tutorials/build-your-own-captcha-and-contact-form-in-php--net-5362).

## Json:

Considerem que l'ús de Json pel pas d'informació i intercanvi de dades és correcte, ja que json és actualment (junt amb XML) el format més utilitzat i amb més suport. No haurieu de descartar però, que per a algunes coses potser necesiteu utilitzar XML, però, com dit anteriorment, no és un problema.

Un cop l'usuari hagi fer login, podrà accedir al seu perfil, on podrà veure la seva informació personal, els seus vols… En cap cas es podrà veure sense loguejar i això implica que un usuari mai podrà veure informació de cap tipus d'un altre usuari.

## Jquery:

És imprescindible contar amb un CDN. Aquest ens permet tenir diferents servidors entre els que es reparteix la feina.
En el cas que patisim un atac DDoS, al tenir aquest sistema, els servidors que no han sigut atacats mantindrien el sistema funcional mentre el servidor caigut tingues temps a recuperar-se.

Amb Jquery es pot implementar un CDN, per tant creiem adient que l'utilitzeu.
Podeu trobar més informació de com fer-ho en aquesta web:
(https://code.jquery.com/).

## Cookies:

Seria interessant donar la opció al usuari de recordar la seva informació de login, per a que quan torni a entrar a la web, no hagi de tornar a introduir les seves dades.

Per fer-ho, utilitzarem cookies. Necessitem una cookie que contingui el nom d'usuari i un nombre suficientment llarg (mínim 32 caràcters numèrics) que l'identifiqui. Per seguretat, aquesta cookie no pot contenir la contrasenya, per tant, quan l'usuari entra a la web, el servidor té les associacions usuari-nombre per comprovar que la cookie en concret és vàlida, i si ho es, permet el login automàtic.

Podeu trobar més informació de com implementar-ho en PHP en aquesta web:
(https://www.baulphp.com/login-con-la-opcion-recordarme-php-session-y-cookies/).

## Altres:

·Pàgina per comprovar seguretat de la web(Requereix Mozilla):
https://observatory.mozilla.org/analyze/www.facebook.com

## Consideracions:

·Falta informació sobre protocols utilitzats, sobretot amb la comunicació amb base de dades i FOG/arquitectura.
·Es poden incloure noves cookies per afegir funcionalitats a la pàgina web.

## Changelog:

**V1.0:**
-Creació del document.