GAMING KEYS

Documentació

Sergio Martos Forniés
31 de Desembre de 2018
Treball Final de Màster – Comerç electrònic
Universitat Oberta de Catalunya

Índex

1. Actualitzacions

Canvis respecte versions anteriors del document.

- **Revisió 1**
  - Correccions menors en tot el document.
- **Revisió 2**
  - Reorganització de la informació de l'apartat 2.
  - S'afegeixen aspectes relacionats amb seguretat en l'apartat 2.
  - Correcció i ampliació de les tasques de l'apartat de Planificació.
  - Sistema de puntuació queda fora de l'abast d'aquest projecte i és mogut a l'apartat d'extensió.
  - Afegida la descripció del sistema LAMP en l'apartat 2.3.
  - S'afegeix l'apartat 4 d'anàlisi del projecte.
  - Canvi de l'apartat "Webgrafia" per "Bibliografia", juntament amb la millora de format.
  - Correccions menors en tot el document.
- **Revisió 3**
  - Disseny *responsive,* sistema multi idioma i registres en logs queden fora de l'abast. Correccions corresponents en tot el document.
  - Canvi d'entorn *back-end* LAMP a WAMP. Correccions corresponents en tot el document.
  - Canvi de contingut en l'apartat Diagrames de seqüència.
  - S'afageix un nou subapartat anomenat Anàlisi de seguretat en l'apartat d'anàlisi.
  - Nou apartat anomenat Desenvolupament.
  - Nou apartat anomenat Proves.

- Canvi d'estructura i contingut de l'apartat Extensions.
- Correccions menors en tot el document.

2. Introducció

Aquest document determina la memòria de treball de *Gaming Keys*. *Gaming Keys* és una botiga en línia especialitzada en la venda de productes digitals i en el subministrament de claus digitals, oferint els millors preus possibles i entregant el producte instantàniament a través d'un sistema únic automatitzat d'entregues.

2.1. Concepte de la botiga

Gaming Keys és una botiga en línia en la qual es mostren una sèrie de productes dins d'un catàleg, amb un disseny senzill, on el client pot visualitzar tota la informació relacionada amb els productes i seleccionar aquells que desitja comprar.

El client podrà realitzar el pagament, de manera segura, seleccionant el mètode de pagament preferit dins de les opcions disponibles per garantir la seva satisfacció. Un cop verificada que l'operació de compra ha tingut èxit, l'usuari rep, tant en l'històric de compres del compte del sistema com en el seu correu electrònic, la clau o les claus digitals corresponents als seus productes.

2.2. Objectiu principal i abast

L'objectiu principal és oferir als consumidors la quantitat més gran de productes en un entorn dinàmic dins d'un sistema integrat amb diferents mètodes de pagament per fer l'experiència de l'usuari i el procés de compra el més fàcil i còmode possible.

La usabilitat que presenta i la facilitat que proporciona el sistema per realitzar les compres en qüestió de minuts fa que *Gaming Keys* estigui adaptat a qualsevol consumidor d'un rang ampli d'edat que disposi d'un dispositiu i tingui accés a Internet.

A més a més, aquest sistema web ha d'estar implementat de manera que permeti realitzar qualsevol canvi en la botiga de la manera més fàcil i en el mínim de temps possible, a la mateixa vegada que ha de permetre als usuaris garantir una navegació intuïtiva i un procés de compra ràpid i segur. Posteriorment, es poden implementar noves millores esmentades més endavant.

2.3. Seguretat en el comerç electrònic

En l'actualitat, el món empresarial està totalment introduït en la xarxa i qualsevol empresa ofereix com a mínim un servei web, ja sigui una pàgina web d'informació fins un servei complet com, per exemple, una botiga en línia.

Instaurat el comerç electrònic en les nostres vides, sorgeixen noves formes de vulnerar la seguretat informàtica de les empreses. Per tant, de la mateixa manera que una botiga física ha de complir amb unes regles mínimes per garantir la seva seguretat, una botiga en línia no està exempta de problemàtiques similars, degut que només canvia el context, que és Internet.

A continuació, es descriuen algunes tècniques existents en la actualitat per atacar sistemes web.

SQL Injection

L'Atac per injecció SQL (*Structured Query Language Injection*) és una tècnica que consisteix en modificar una consulta de base de dades mitjançant la injecció de codi en la consulta. Aquesta tècnica és principalment utilitzada quan es permet a l'usuari introduir text, com per exemple en els formularis.

Es tracta d'un atac molt fàcil d'executar, degut que només requereix un ordinador i coneixements bàsics sobre bases de dades. Amb aquest atac es pot obtenir accés a les taules de bases de dades i extreure tota la informació necessària incloent, per exemple, informació d'usuaris i contrasenyes.

DoS o DDoS

La Denegació de Servei (DoS) o Denegació de Servei Distribuït (DDoS) són les tècniques més comunes per congelar el funcionament d'un sistema web. Aquests atacs intenten inundar un servei web amb sol·licituds externes, fent que aquest servei no estigui disponible pels usuaris reals.

La diferència principal entre aquestes dues tècniques és que la primera només utilitza un dispositiu amb connexió a Internet per inundar un servidor amb paquets, mentre que la segona s'utilitzen molts dispositius per inundar el servei amb moltes peticions.

Els atacs DdoS es divideixen en 3 tipologies diferents:

- Atacs de volum, on s'intenta desbordar l'amplada de banda en un servei concret.
- Atacs de protocol, on els paquets intenten consumir serveis o recursos de la

xarxa.

- Atacs a aplicacions, on les peticions es realitzen amb l'intenció de col·lapsar el servidor web, mitjançant la capa d'aplicació.

Força bruta

En aquesta tècnica s'intenta provar totes les combinacions possibles de credencials d'usuaris en una pàgina web per tenir accés a les funcionalitats d'aquesta. Aquests atacs requereixen contrasenyes dèbils per poder dur a terme l'operació en un temps raonable.

En aquesta tècnica és molt usual la utilització de diccionaris per intentar reduir al màxim el temps del procés. Aquests diccionaris són una recopilació de contrasenyes o combinacions d'elles més comunes, que s'utilitzen per comprovar si cap d'aquests registres coincideix amb la contrasenya en qüestió, abans de provar totes les combinacions possibles.

Degut que el temps que triga el procés per esbrinar la contrasenya augmenta exponencialment com més segura és la contrasenya i que cada cop els dispositius són més i més potents, les contrasenyes també han d'anar sent cada cop més segures.

Cross Site Scripting (XSS) i Cross Site Request/Reference Forgery (CSRF)

Els atacants utilitzen aquestes tècniques per injectar scripts maliciosos en els sistemes web. En aquest cas, un atacant pot introduir codi en la pàgina web per utilitzar-ho d'atac per altres usuaris que visitin la mateixa pàgina. D'aquesta manera, els usuaris interactuen amb el codi maliciós i així poder obtenir informació continguda, per exemple, en les *cookies* o *tokens* de sessió.

Es poden agrupar en dos grups:

- XSS reflectits, on el codi modificat s'elimina al carregar la pàgina web de nou.

- XSS persistent, on el codi modificat queda emmagatzemant en la web.

Una tècnica més evolucionada que l'anterior és l'anomenada Cross Site Request/Reference Forgery. En aquest cas, s'explotarà la confiança del servidor sobre el client. Per tant, l'atacant es farà passar per un client legítim, utilitzant dades parcials d'aquest.

Aquesta vulnerabilitat està present en formularis. Quan les dades són enviades al servidor, el sistema ha de comprovar que la petició és legítima i que ha sigut realitzada per l'usuari.

En relació amb l'anterior, es poden agrupar els diferents tipus segons on s'aconsegueix injectar el codi desitjat:

- DOM Cross Site Scripting (DOM XSS).
- Cross Site Flashing (XSF).
- Cross Flame Scripting (XFS).
- Cross Zone Scripting (XZS).
- Cross Agent Scripting (XAS).
- Cross Referer Scripting (XRS).

Atacs d'inclusió LFI i RFI

Aquests tipus de tècniques (Local File Inclusion i Remote File Inclusion) consisteixen a incloure arxius que es troben en el mateix o un altre servidor i que es produeix com a conseqüència d'una errada en la programació de la pàgina, filtrant inadequadament el que s'introdueix mitjançant mètodes per incloure arxius.

En resum, s'han pogut veure alguns dels diferents tipus d'atacs que poden rebre els sistemes web i l'impacte que pot tenir en la informació. Per tant, com es detallarà en apartats posteriors, s'aplicaran un seguit de mesures de seguretat per protegir la botiga en línia de qualsevol tipus d'atac.

2.4. Requisits funcionals

Els requisits funcionals que s'han de complir en aquest projecte són els següents:

- **Disseny senzill**: La complexitat és mínima. Les diferents interfícies han de contenir els elements mínims necessaris per fer que l'experiència de l'usuari sigui el més fàcil i agradable possible.
- **Idioma internacional:** L'aplicació ha d'oferir l'idioma més popular a escala mundial. Aleshores, l'idioma principal serà l'Anglès. No obstant això, el sistema serà fàcilment adaptable per la introducció de nous idiomes per captar més clients.
- **Gestió d'usuaris:** El client ha de ser capaç de poder registrar-se en el sistema, iniciar sessió i finalitzar-la. La informació mínima requerida en el procés de registrament ha de ser el nom, cognom, correu electrònic i la contrasenya. Posteriorment, l'usuari podrà activar, mitjançant l'introducció del codi d'activació enviat al correu electrònic corresponent, el compte per poder

començar a realitzar les seves compres.

- **Producte:** Ha de contenir tota la informació necessària per garantir a l'usuari que el producte que està visualitzant és correcte. Aquest producte passarà a ser una clau digital un cop finalitzat el procés de compra.
    - **Informació del producte:** La informació principal ha de ser: Nom, descripció, plataforma, tipus, gènere, tràiler, preu de compra i altres detalls.
    - **Tipus de productes:** Videojocs, paquets d'expansió i llistats de continguts descarregables (DLC).
- **Clau digital:** Correspon al número de sèrie o el codi del producte, compost per una combinació de lletres i números que normalment es troba en la caixa física o imprès en la fitxa de referència d'aquest, que s'utilitza per enregistrar i activar un producte en la plataforma corresponent.
- **Cerca de productes:** L'usuari ha de disposar d'una eina que permeti cercar el producte desitjat. Aquesta eina ha de ser visible durant tot el procés de visualització dels productes en la plana principal.
    - **Cerca per text**: L'usuari ha de ser capaç de cercar els productes desitjats mitjançant un camp de text.
    - **Cerca per menú de navegació**: L'usuari també podrà realitzar la cerca mitjançant els filtres disponibles en el menú de navegació. Aquests filtres seran: categoria, plataforma i jocs nous o especials.
- **Gestió del carret de compra:** L'usuari ha de poder visualitzar el llistat de productes seleccionats i tenir a l'abast les opcions necessàries per gestionar la seva compra. En aquest cas, l'usuari podrà eliminar els productes que ja no desitgi.
- **Procés de compra:** El sistema ha de mostrar el formulari de pagament corresponent al mètode seleccionat per habilitar l'ompliment de la informació necessària per part del client per duu a terme l'operació de pagament. El sistema ha de verificar que aquest procés s'ha dut a terme correctament.
- **Mètodes de pagament:** El client ha de disposar de diferents mètodes de pagament seleccionables per donar la possibilitat que triï l'opció més adient. Els mètodes de pagament poden ser targetes, Paypal, Bitcoin, etc.
- **Sistema d'entrega:** Aquest sistema d'entrega de claus digitals ha de permetre realitzar l'assignació d'aquestes de manera automàtica i instantània. Un cop el

client realitza el pagament i aquest es verifica, la clau ha d'aparèixer en el seu compte d'usuari. D'aquesta manera s'aconsegueix que els clients experimentin un servei ràpid, eficient i personal.

- **Històric de compres:** L'usuari disposa de tota la informació que fa referència a les compres realitzades. Aquesta informació ha d'incloure la clau digital corresponent a cada producte.

- **Ampliació:** *Gaming Keys* ha de ser ampliable amb nous productes i noves funcionalitats de forma senzilla. El funcionament de la botiga serà lo més independent possible del contingut. D'aquesta manera, es podrà afegir noves actualitzacions en el menor temps possible.

- **Navegabilitat:** El disseny de *Gaming Keys* ha d'estar implementat per ser simple i eficient. La navegació entre pàgines ha de ser ràpida i intuitiva, degut que l'usuari ha de saber en tot moment en quin estat del procés de compra està, què vol visualitzar i com aconseguir-ho.

- **Estil visual:** *Gaming Keys* tindrà un estil senzill i cap plana web del sistema contindrà un excés d'elements i colors diferents. Amb aquest fet es pretén aconseguir que l'usuari no pateixi una excessiva càrrega visual i pugui realitzar les seves operacions sense cap problema.

2.5. Requisits de seguretat

La seguretat de la informació és el conjunt de mesures preventives de les organitzacions i dels sistemes tecnològics que permeten protegir la informació buscant mantenir la confidencialitat, disponibilitat i integritat de les dades.

La informació és poder i pot ser mal utilitzada, robada, esborrada o sabotejada. Aquests fets poden tenir greus conseqüències, tant per l'usuari com per l'empresa. Per exemple, l'accés a la informació no protegida pot afectar no només a la privacitat de l'usuari, sinó també pot tenir altres conseqüències com, per exemple, econòmiques (en cas de robatori d'informació de targetes).

Per tant, els requisits de seguretat que s'han de complir en aquest projecte són els següents:

- **Confidencialitat:** és la propietat que impedeix la divulgació d'informació a persones, entitats o processos no autoritzats. Per tant, l'accés de la informació només s'assegura a aquelles persones autoritzades.

  Per exemple, el procés de compra requereix dades d'entitats bancàries com números de targeta, codis, contrasenyes, etc. També, hi han altres dades desades en la base de dades com, per exemple comptes d'usuari amb les seves

contrasenyes corresponents.

Per tant, el sistema ha de mantenir la confidencialitat mitjançant el xifrat de totes les dades sensibles.

- **Integritat:** és la propietat que manté la informació com es va generar i desar, sense haver sigut manipulada ni alterada per persones o processos no autoritzats.

  Aquest concepte garanteix que les dades només puguin ser modificades per personal autoritzat. La integritat de la informació es garanteix adjuntant altres conjunts de dades com, per exemple, firmes digitals, cadenes *hash*, contrasenyes, etc.

  Per exemple, en el cas d'una sol·licitud de pagament, el sistema pot comprovar quines són les dades que s'envien juntament amb la cadena *hash* generada a partir d'aquestes, i comparar-les en el servidor juntament amb la cadena *hash* calculada en aquest.

- **Disponibilitat:** és la propietat de la informació de trobar-se a disposició de les persones o processos autoritzats que així ho requereixin.

  Per garantir aquesta propietat en el sistema implica la prevenció d'atacs de denegació de serveis. Per tant, cal crear mètodes alternatius per garantir l'accés a la informació o, altrament, oferir respostes alternatives.

- **Autenticació:** és la propietat que permet identificar la persona o procés que genera la informació en el sistema. Quan s'ha rebut una petició cal assegurar que ha sigut l'usuari en qüestió i no una tercera persona que l'hagi suplantat la identitat.

  Per tant, cal aplicar un control d'accés a usuaris mitjançant comptes d'usuaris prèviament registrats en el sistema.

- **No repudi:** és la propietat que permet verificar la participació de cadascuna de les parts en la comunicació, proporcionant així protecció contra la interrupció d'aquesta.

  Aquesta propietat ha de provar que l'emissor de la comunicació l'ha enviat (no repudi d'origen) i que el receptor l'ha rebut correctament (no repudi en destí). D'aquesta manera, cap de les parts implicades en la comunicació poden negar la transmissió d'un missatge.

  Per tant, en el sistema s'han d'aplicar un seguit de mesures que permetin que l'emissor pugui comprovar que el missatge ha sigut rebut pel suposat receptor i que aquest ha rebut el missatge de l'emissor en qüestió.

- **Intercanvi equitatiu:** és la propietat que garanteix una resposta a qualsevol sol·licitud dins del sistema.

Resumidament, tots aquests requisits s'han d'aplicar en el sistema per garantir la seguretat de la informació. Aquests procediments es veurà amb més detall en posteriors apartats.

2.6. Tecnologia

Per desenvolupar el codi del projecte s'utilitzarà principalment l'editor de text *Sublime Text [1]*. Aquest editor de text és un dels editors de codi gratuïts més populars del mercat. Aquest editor és molt lleuger i suporta un gran nombre de llenguatges, ja siguin de programació, maquetació, bases de dades, etc.

Com a sistema *back-end* s'utilitzarà el conegut acrònim WAMP. Aquest sistema fa referència a un conjunt de subsistemes de software necessaris per, entre altres, oferir serveis web. En aquest cas, l'acrònim fa referència a *Windows*, *Apache HTTP Server[2]*, *MySQL[3]* i *PHP[4]*.

- **Windows:** És un dels sistemes operatius més utilitzats del mercat, tant en l'àmbit d'usuari com en l'àmbit empresarial.

- **Apache HTTP Server:** És un dels servidors web lliures i de codi obert més populars del mercat i s'utilitza com a referència pel disseny i avaluació d'altres servidors web.

- **MySQL:** És un sistema de gestió de bases de dades relacional que utilitza SQL, multiprocés i multiusuari.

- **PHP (Hypertext Preprocessor):** És un llenguatge de programació dissenyat, principalment, per produïr webs dinàmiques, tot i que també es pot utilitzar des d'una línia de comandes o com aplicació d'escriptori.

Quant a *frameworks*, no es descarta l'ús d'aquests per implementar funcionalitat de manera segura, per tal de reduir el cost del desenvolupament i garantir la seguretat de les operacions.

Finalment, s'integrarà el sistema desenvolupat amb les corresponents APIs que ofereixin els diferents mètodes de pagament, per poder realitzar les operacions de pagament correctament.

3. Planificació

En aquesta secció s'entrarà més en detall en les fites que s'han d'assolir durant el desenvolupament de *Gaming Keys.* S'exposaran totes les tasques principals que determinen la trajectòria que ha de seguir el desenvolupament juntament amb la durada corresponent.

Tenint en compte que la dedicació d'hores al dia pot ser molt variable segons les circumstàncies, l'estimació de la durada de les tasques es mesura en dies en comptes d'hores, i la duració d'aquestes es pot veure notablement afectades per la dedicació esmentada anteriorment.

Aleshores, la planificació de les tasques, a alt nivell, és la següent:

Nom de la tasca	**Dies**
Instal·lació de l'entorn	**1**
Maquetació web	**12**
Definició de maquetació i estils	2
Maquetació	5
Aplicació d'estils	5
Disseny de la base de dades	**3**
Definició de les taules i camps d'informació	1
Scripts de creació (base de dades, usuaris, taules, claus, etc)	1
Scripts de la inserció de dades	1
Cerca de productes	**7**
Definició de filtres	1
Implementació dels filtres i paginació de resultats	6
Gestió d'usuaris	**7**
Definició de requeriments	1
Implementació de les operacions (alta, modificació i eliminació d'usuari)	6
Gestió de la sessió	**7**
Definició	1
Implementació de les operacions d'inici de sessió, manteniment i finalitzar sessió	6
Gestió del carret de compra	**7**
Definició de l'informació	1

Implementació de les operacions d'afegir, modificar i eliminar productes	6
Històric de compres	**7**
Definició de l'informació	1
Implementació de la visualització	6
Integracions amb mètodes de pagament	**28**
Definició dels mètodes	1
Implementació del procés de compra	7
Procés de pagament	7
Procés de resposta del pagament	4
Verificacions i implementació de mesures de seguretat	9
Proves	**7**
TOTAL	**86**

Cal afegir, que en totes les tasques van incloses tant la implementació de les mesures de seguretat corresponents a aquelles tasques on s'hagin d'aplicar com les proves necessàries per verificar-les.

4. Anàlisi

Anàlisi, per definició, és el procés de construcció d'un model o especificació detallada del problema a resoldre. En aquesta construcció no es té en consideració el disseny ni la implementació.

Per tant, en aquesta secció entrarem més en detall en les funcionalitats de Gaming Keys. S'exposaran tots els pilars fonamentals que sostenen la seva navegabilitat així com les accions que podrà duu a terme l'usuari dins de la botiga.

La notació que s'utilitzarà és la proporcionada per l'estàndard UML. En aquest cas, es veuran afectats el diagrama de casos d'ús i els de seqüència.

4.1. Diagrama de flux

En aquesta secció es detallarà el transcurs de la navegació típica a *Gaming Keys*. Es comentaran els passos que el client ha de seguir des de entrar en la pàgina web fins a completar un procés de compra. Per tant, aquí es mostrarà la navegació entre pàgines i més endavant es definirà detalladament el contingut de les interfícies i les funcionalitats disponibles.

Primerament, cal aclarir que totes les interfícies contenen una capçalera, juntament amb un menú i un peu de pàgina:

- Capçalera: segons l'estat de la sessió de l'usuari, aquest component disposa, entre altres funcionalitats, de navegacions diferents.

  Per una banda, en cas que l'usuari no hagi iniciat sessió, la capçalera disposa d'un enllaç cap a la interfície d'inici de sessió. En cas contrari, aquest enllaç mostrarà un llistat d'opcions d'usuari que, entre altres, ofereix la possibilitat de visualitzar l'històric de compres realitzades.

  Per altra banda, en cas que l'usuari hagi afegit productes o no al carret, aquest podrà començar el procés de pagament visualitzant el resum de compra.

  A més a més, mitjançant el logo de la botiga, permet tornar a la pàgina principal. Per tant, qualsevol interfície que contingui capçalera, disposa de l'opció de tornar a la pàgina principal.

- Menú: és un component que es visualitza sempre independentment de l'estat de l'usuari. Aquest component permet la visualització dels productes del catàleg segons l'opció seleccionada.

- Peu de pàgina: aquest component, com l'anterior, també es visualitza sempre. A més a més, aquest no disposa de cap mena de funcionalitat, només conté

informació per l'usuari.

Principalment, el client entra en la pàgina principal i se li presenta el catàleg de productes. En aquesta pàgina, entre les diferents funcionalitats disponibles, pot visualitzar el detall del producte seleccionat. En aquesta pàgina, el client pot tornar enrere per continuar visualitzant el catàleg.

Per una banda, en cas que l'usuari desitgi iniciar sessió, aquesta interfície permet la navegació a dos punts diferents:

- Creació de compte d'usuari: el procediment del registre del nou usuari permet l'opció de tornar a la pàgina d'inici de sessió o avançar amb el procés, on el sistema redirigeix cap a la verificació de la creació del compte d'usuari. En cas d'haver triat la segona opció, aquesta vista permet tornar cap a la interfície d'inici de sessió.

- Recuperació de compte d'usuari: en aquest pas, l'usuari pot retornar cap a la pàgina anterior.

Per altra banda, en cas que l'usuari vulgui començar amb el procés de compra, pot visualitzar la vista amb el resum de la compra. Seguidament, en cas que l'usuari hagi iniciat sessió, pot navegar cap al següent pas del procés de compra i visualitzar la interfície del pagament. També, pot tornar enrere per seguir afegint productes al carret de compra.

Un cop en la vista de pagament, segons el mètode de pagament, el sistema pot realitzar una redirecció cap a una interfície externa o anar directament cap a la vista de la confirmació de pagament. En cas de visualitzar una vista externa, un cop el client hagi acabat amb l'operació de pagament, el sistema realitzarà automàticament la redirecció cap a la vista de la confirmació de pagament.

Finalment, en aquesta última vista del procediment de compra, l'usuari pot tornar a la pàgina principal per seguir comprant, visualitzar les claus dels productes adquirits, etc.

Un esquema general de la navegació entre pàgines pot ser el següent:

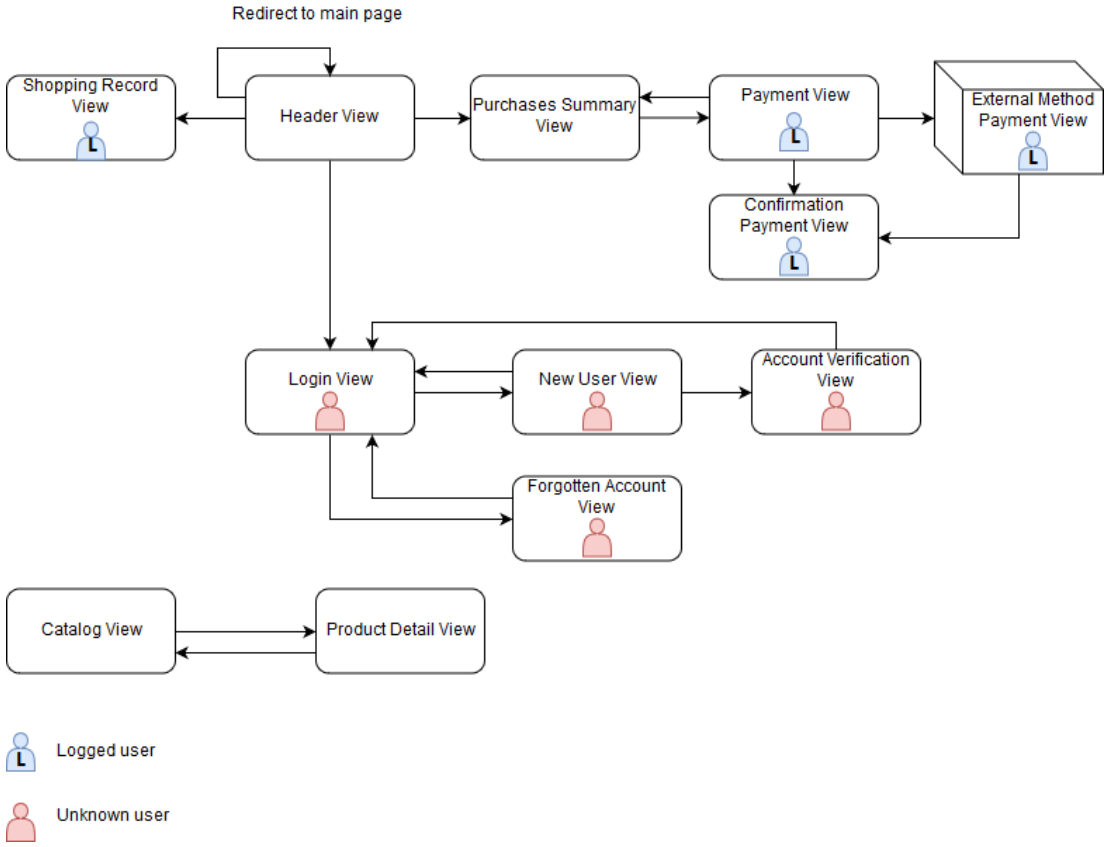


Figura 1: Diagrama de flux

Com es pot veure en la figura 1, s'ha obviat de mostrar tots els retorns cap a la pàgina principal (catàleg), degut que totes les vistes, exceptuant les externes, contenen la capçalera i pot empitjorar la visualització de l'esquema i, per conseqüència, l'enteniment d'aquest.

També, s'ha especificat quines són les interfícies que el client pot visualitzar segons l'estat de la sessió. Per tant, en cas que un usuari no hagi iniciat sessió en el sistema i entri en alguna de les vistes directament mitjançant el navegador, el sistema s'encarrega de realitzar la redirecció cap a la pàgina principal. Aquesta redirecció es realitza en cas contrari, quan un client amb la sessió iniciada intenta accedir a una vista on es requereix que l'usuari no estigui identificat.

Pel que fa la resta d'interfícies, no hi ha cap restricció de sessió i el client pot visualitzar i navegar lliurement.

Existeixen restriccions més específiques en certes vistes de la botiga que no són reflexades en l'esquema i són les següents:

- Verificació de la creació de compte d'usuari: en aquest cas, l'usuari només té

accés en cas d'haver realitzat l'operació de creació de compte.

- Formulari de pagament: aquesta vista només pot ser visualitzada en cas d'haver afegit productes al carret. A més a més, un cop el client crida l'operació de pagament, aquest no podrà realitzar cap tipus de navegació més que la dictada pel sistema, assegurant així que no hi hagi cap problema.

- Resum de la compra: com en el cas anterior, aquesta vista només pot ser visualitzada en cas d'haver afegit productes al carret.

- Confirmació de pagament: segons els passos del procés de compra, només pot ser visualitzada en cas d'haver realitzat l'operació de compra.

4.2. Diagrama de casos d'ús

Els diagrames de casos d'ús determinen la funcionalitat del sistema utilitzant actors i casos d'ús. Un cas d'ús és una descripció de les accions d'un sistema des del punt de vista de l'usuari, mentrestant els actors són les persones o entitats que realitzen aquestes accions.

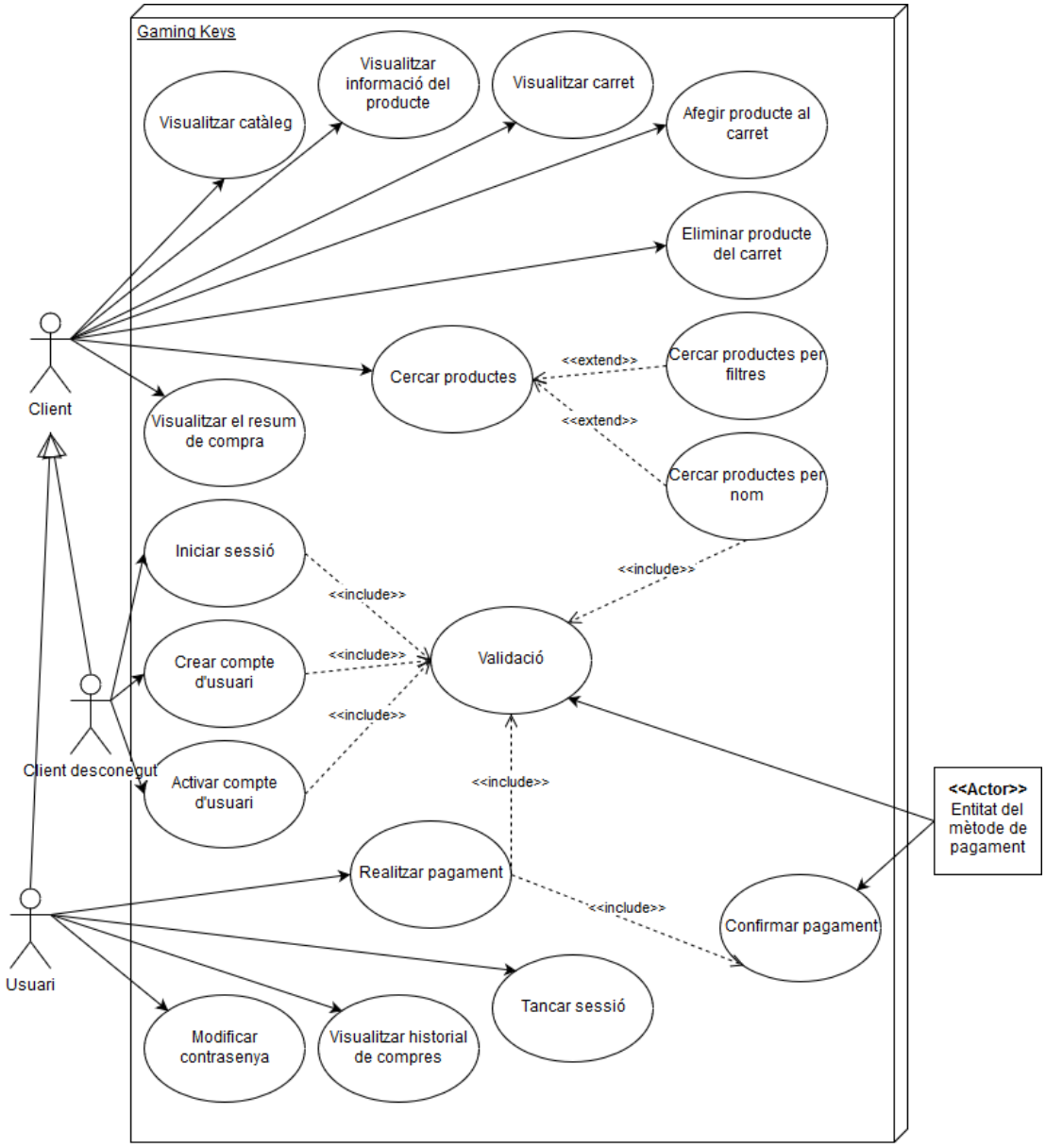


Figura 2: Diagrama de casos d'ús

Primerament, els actors de Gaming Keys són el client i l'entitat externa del mètode de pagament corresponent. A més a més, cal sumar dos actors més que seran especialitzacions de l'actor client, i són el client identificat i el client que encara no ha iniciat sessió.

En la figura anterior es mostren les funcionalitats de tots els actors. Començant per l'usuari, en qualsevol dels seus estats, les accions que pot realitzar són: visualitzar el catàleg, veure el detall d'un producte, veure el carret de la compra, afegir i eliminar un producte del carret, visualitzar el resum de la compra i cercar productes. Aquesta

última acció depèn de la manera de com és realitzada, pot ser tant per cerca de nom com de categoria.

Seguidament, l'especialització de l'usuari sense identificar realitza les següents accions: iniciar sessió, creació i activació d'un compte d'usuari. En canvi, les accions que pot duu a terme l'usuari identificat són: modificar contrasenya, visualitzar l'historial de compres, tancar sessió i realitzar el pagament. Pel que fa a l'actor extern, aquest serà l'encarregat de validar, realitzar, confirmar el pagament, enviant al nostre sistema el resultat d'aquest.

Finalment, com es pot veure en el diagrama, en moltes de les accions sinclou l'acció de validació realitzada pel mateix sistema. Això és degut que s'ha de controlar qualsevol tipus de missatge o acció no desitjada per part dels actors per garantir la seguretat en el sistema. Hi ha diferents tipus de validacions:

- Validació de missatge d'entrada.
- Validació de sessió.
- Validació d'autenticació.
- Validació de pagament.
- Validació de navegació

Cadascuna d'aquestes validacions es veuran amb més detall en posteriors apartats.

4.3. Diagrames de seqüència

Els diagrames de seqüència són un tipus de diagrama d'interacció que descriu el comportament dinàmic del sistema d'informació per un determinat escenari, fent enfasis en la seqüència dels missatges intercanviats pels objectes.

Per tant, segons els escenaris plantejats anteriorment, el següent pas és descriure com es comportarà el sistema segons el tipus d'acció realitzat.

Escenari: visualitzar catàleg

L'usuari, independentment de l'estat de la sessió, des de qualsevol interfície que contingui la capçalera pot tornar a la pàgina principal i visualitzar el catàleg de productes.

Per tant, el sistema cercarà tots els productes existents en el catàleg i la vista els mostrarà:

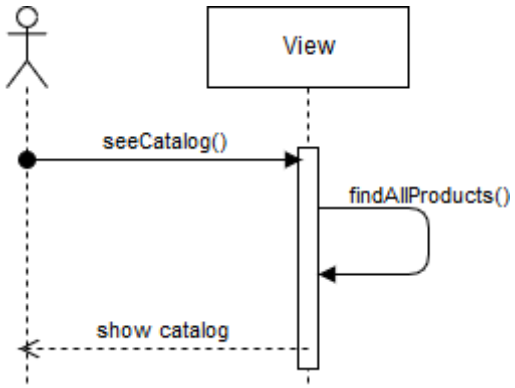


Figura 3: Diagrama de seqüència - visualitzar catàleg

En aquest cas, cal valorar l'opció de realitzar la cerca de productes de manera paginada. D'aquesta manera s'aconsegueix reduir el temps d'espera de l'execució de la consulta a la base de dades i oferint una resposta ràpida a l'usuari.

Escenari: cercar productes

L'usuari, independentment de l'estat de la sessió, des de qualsevol interfície que contingui el menú o la capçalera pot realitzar cerques de productes en el catàleg. Aquest escenari es pot dividir en dos escenaris més específics depenent del mètode de cerca triat per l'usuari. Els dos escenaris més específics són: cerca per categoria i cerca per introducció de text.

El primer cas és quan utilitza la barra de menú, on el client pot fer clic a qualsevol de les opcions disponibles i el sistema realitzarà la cerca amb el criteri seleccionat (tipus de producte, categoria, etc.):

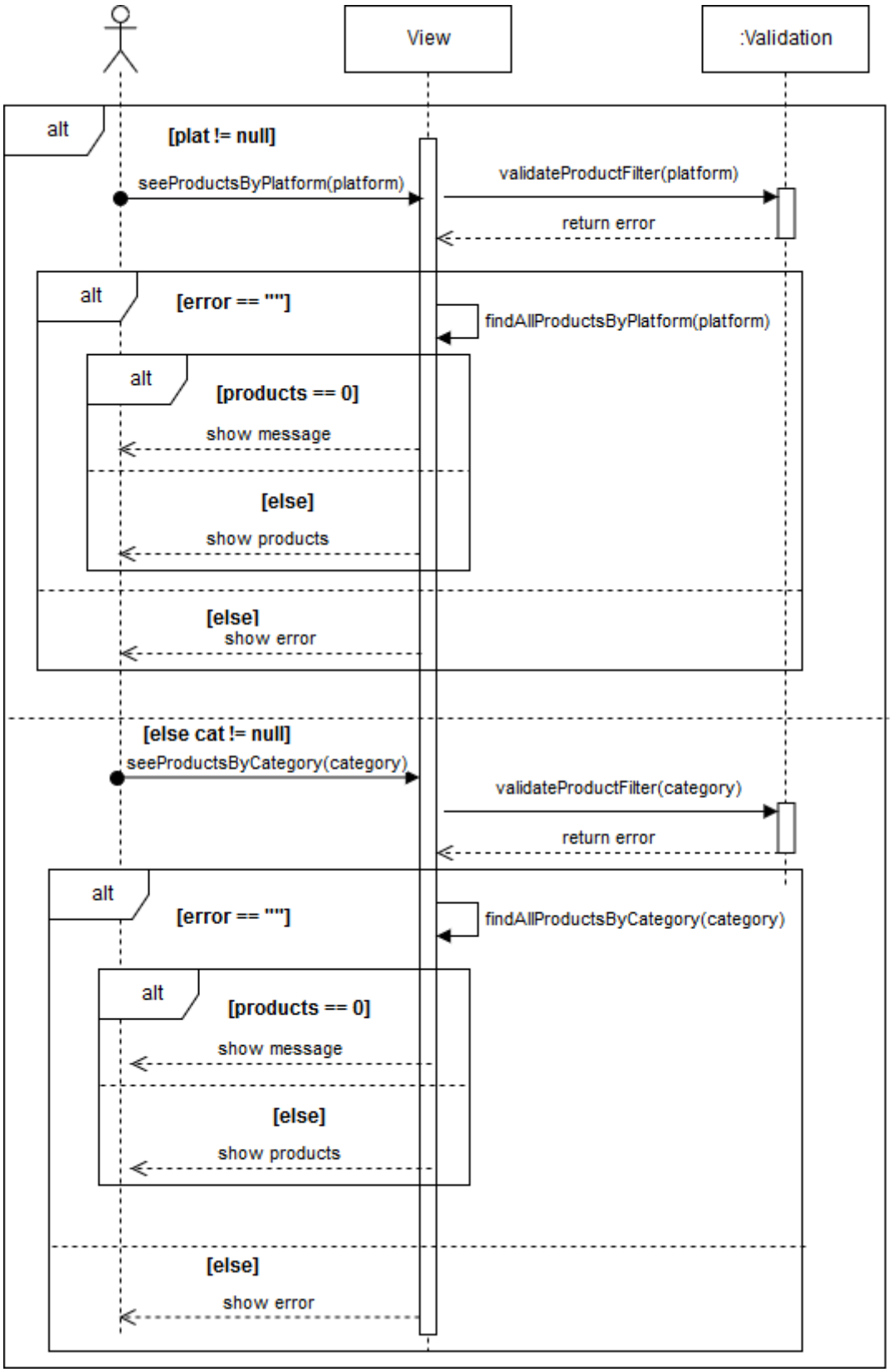


Figura 4: Diagrama de seqüència - cercar productes per filtres

El segon cas, en canvi, és quan l'usuari utilitza el camp de text de la capçalera per realitzar la cerca de productes.

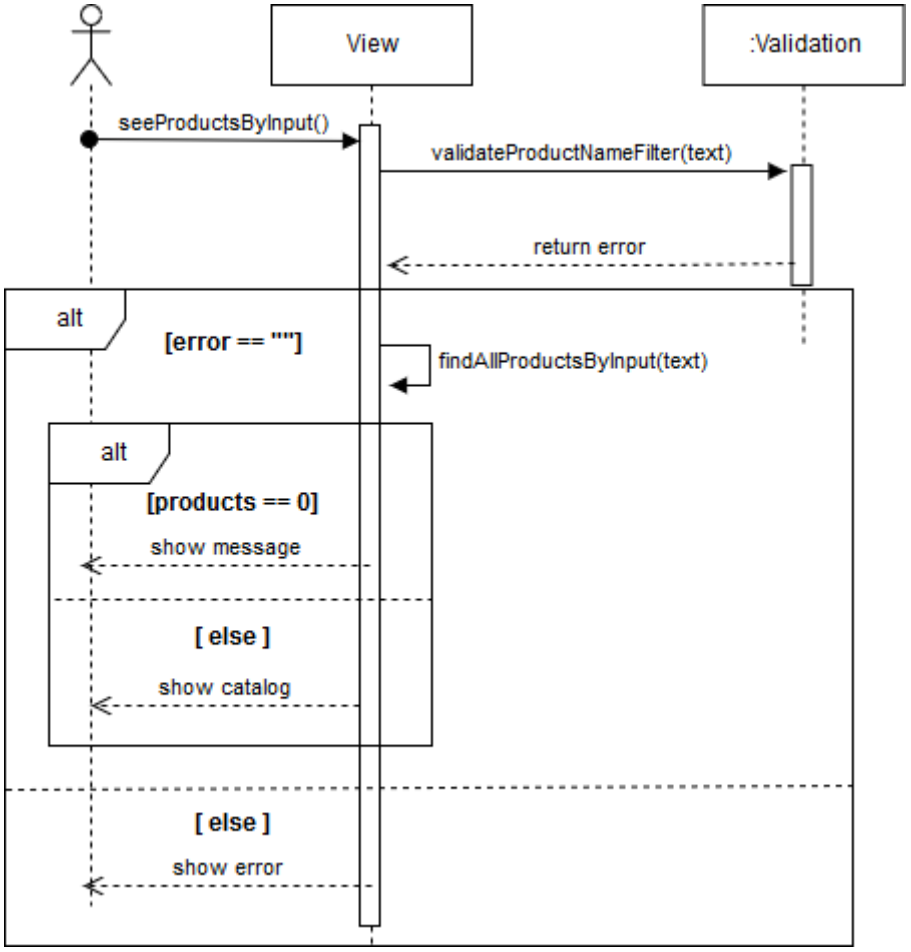


Figura 5: Diagrama de seqüència - cercar productes per nom

En qualsevol cas, l'usuari malintencionadament pot introduir missatges no desitjats al sistema i realitzar atacs d'injecció. El sistema ha de controlar i verificar totes les cadenes de text que pugui introduir l'usuari en el sistema. Per tant, el sistema realitzarà prèviament una validació de la cadena de text introduïda, per controlar qualsevol mena de caràcter no desitjat.

Escenari: visualitzar informació d'un producte

L'usuari, independentment de l'estat de la sessió, mentre visualitza el catàleg pot fer clic en un producte i navegar a la següent pàgina amb tota la informació que fa referència a l'article seleccionat.

Per tant, el sistema cercarà el producte seleccionat mitjançant l'identificador d'aquest:

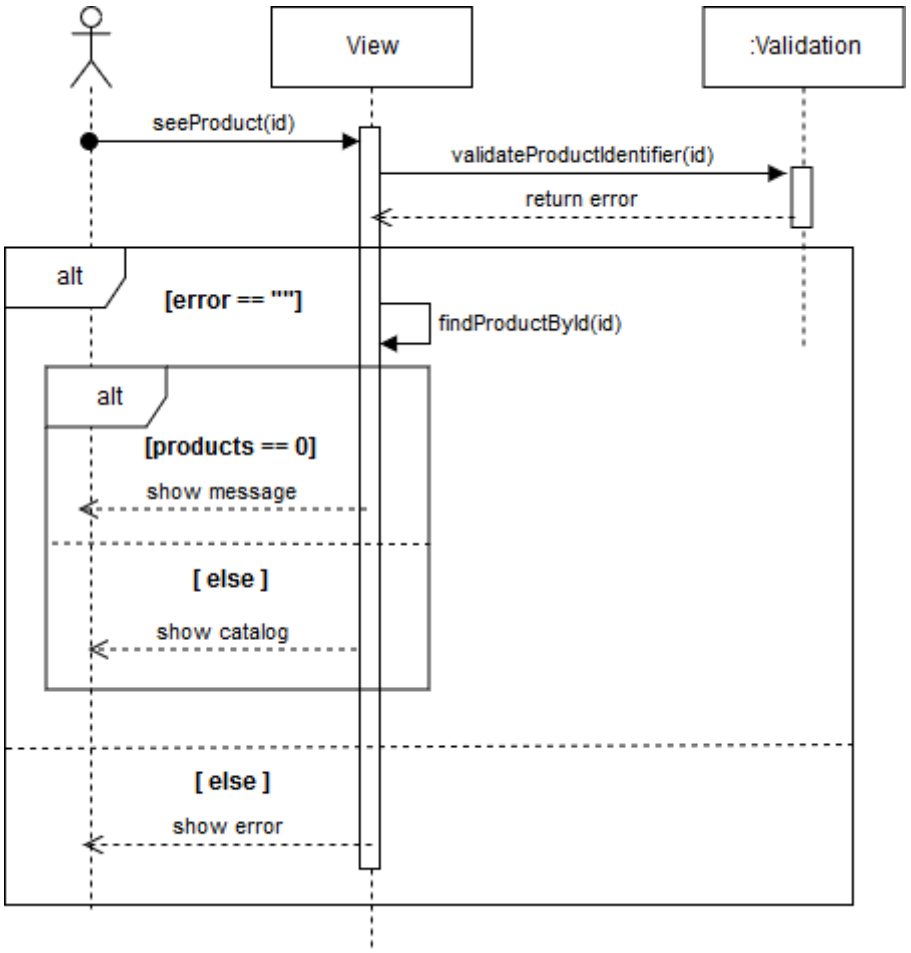


Figura 6: Diagrama de seqüència - visualitzar informació d'un producte

En aquest cas, cal realitzar una validació de l'identificador del producte, de manera que verifiqui que només es tracta d'un valor numèric, evitant així, també, qualsevol tipus d'atac d'injecció.

Escenari: visualitzar carret de la compra

L'usuari, independentment de l'estat de la sessió, des de qualsevol interfície que contingui la capçalera pot fer clic al carret i visualitzar els productes que conté en aquell instant.

A diferència dels anteriors, en aquest cas la informació ja està carregada en la vista, degut que les operacions d'afegir o eliminar producte van actualitzant el carret, com es podrà veure en els següents escenaris. Per tant, en cas que l'usuari hagi afegit prèviament productes al carret, aquests es podran veure i, en cas contrari, es mostrarà el missatge corresponent:

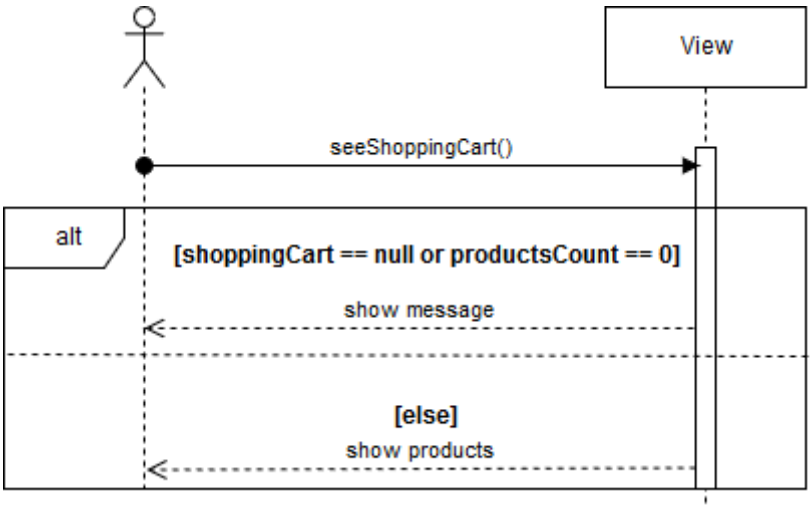


Figura 7: Diagrama de seqüència - visualitzar carret de la compra

Escenari: afegir producte al carret

L'usuari, independentment de l'estat de la sessió, mentre visualitza el catàleg o el detall d'un producte pot fer clic en el botó de comprar per afegir el producte al carret.

Per tant, el sistema afegirà el producte al carret mitjançant l'identificador i actualitzarà la vista per mostrar el nou element en el llistat de productes:

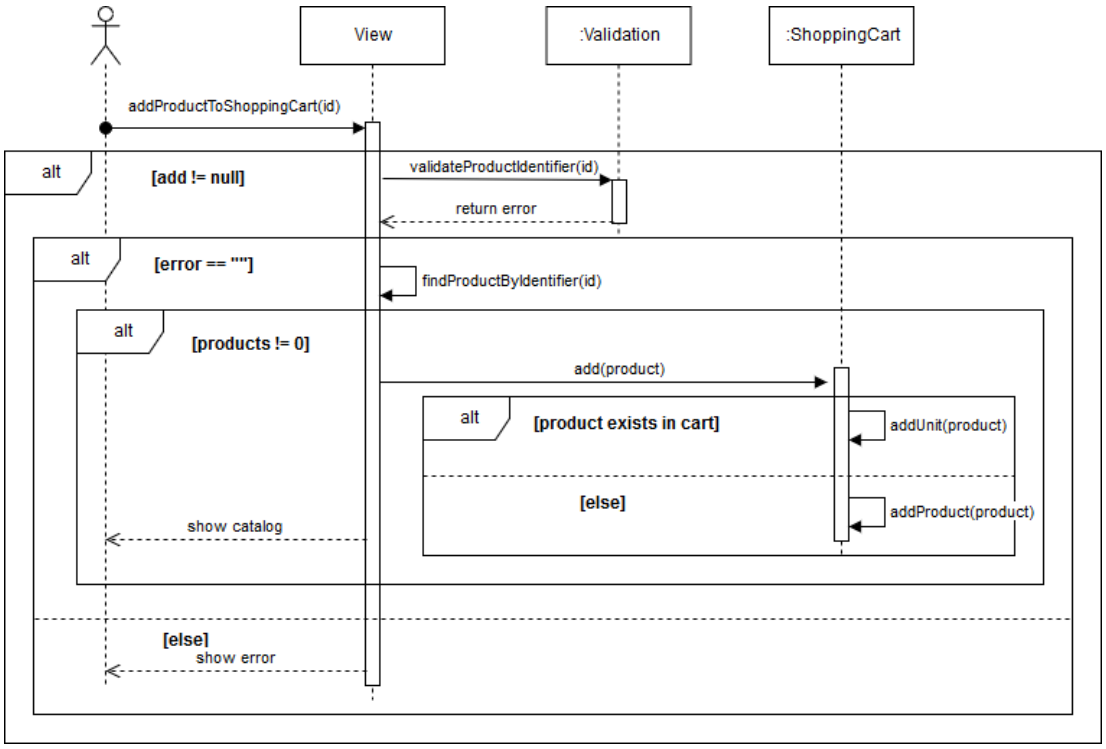


Figura 8: Diagrama de seqüència - afegir producte al carret

En cas que el producte que es desitja afegir ja existeixi prèviament en el carret, es suma una unitat del mateix i, en cas contrari, afegeix el nou producte al carret.

També, cal aplicar un control dels valors que conté l'identificador del producte, que en cas que sigui invàlid, es mostrarà el missatge d'error corresponent.

Escenari: eliminar producte del carret

L'usuari, independentment de l'estat de la sessió, pot treure qualsevol producte dels llistats, tant en el carret com en el resum de la compra.

Per tant, semblant al cas anterior, el sistema eliminar el producte del carret mitjançant l'identificador i actualitzarà la vista per mostrar l'eliminació de l'element en el llistat de productes:

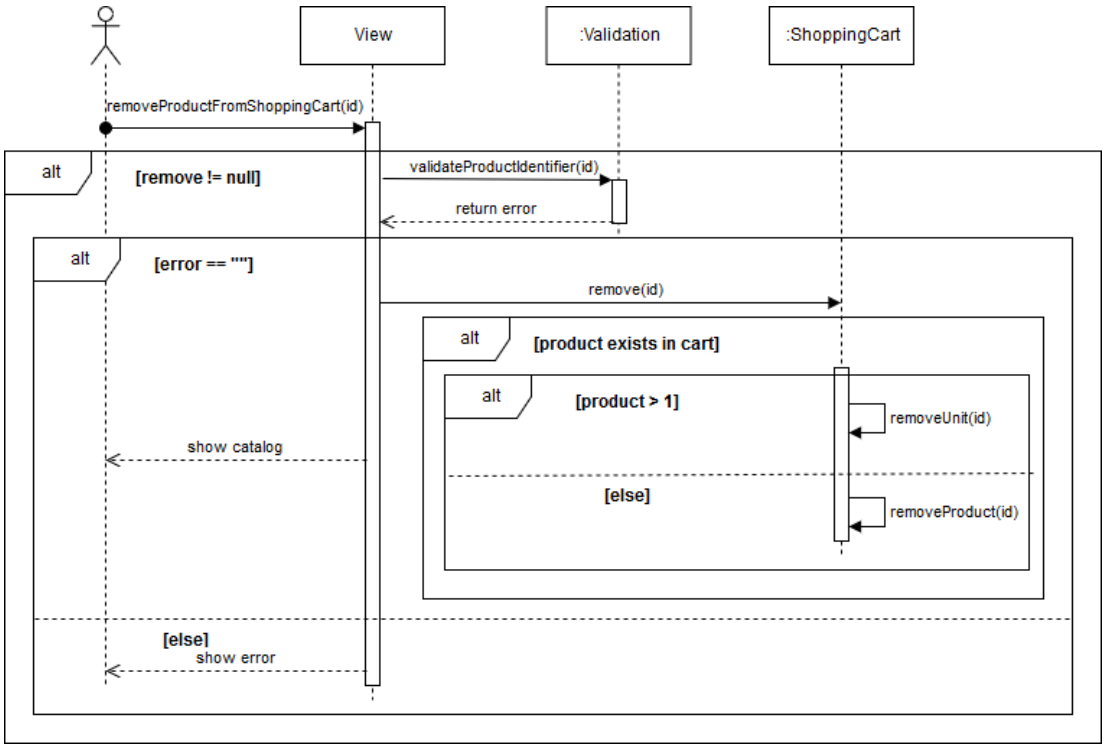


Figura 9: Diagrama de seqüència - eliminar producte del carret

També, com en el cas anterior, cal aplicar el mateix control de valors que arriben al sistema per garantir que no es rep cap intent malintencionat.

Escenari: visualitzar resum de la compra

L'usuari, independentment de l'estat de la sessió, des de qualsevol interfície que contingui la capçalera pot fer clic al botó que hi ha dins del carret per començar el procés de compra i visualitzar el resum de productes escollits.

Tot i que aquesta operació només estarà disponible quan el client hagi afegit al carret de la compra com a mínim un producte, el sistema realitzarà una validació per comprovar que realment existeixen productes:

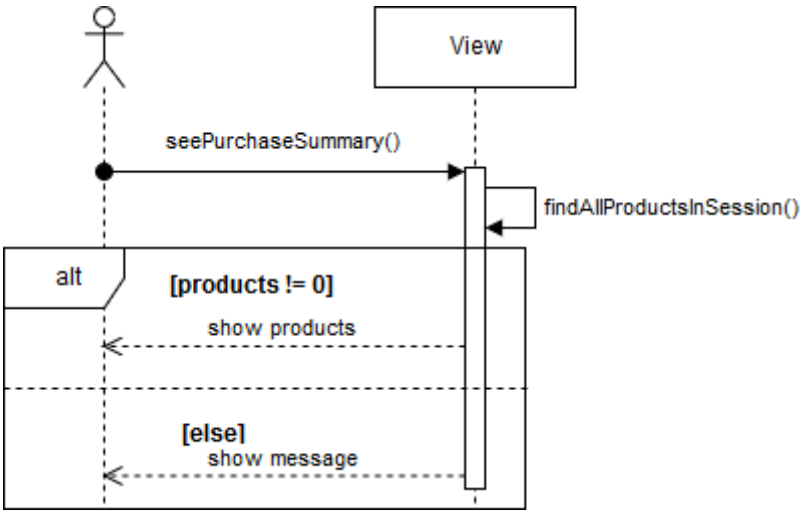


Figura 10: Diagrama de seqüència - visualitzar resum de la compra

D'aquesta manera, s'evita la navegació malintencionada, per part de l'usuari, cap a la vista del resum de la compra en cas de no tenir cap article afegit.

Escenari: crear compte d'usuari

L'usuari, si no té la sessió iniciada, des de la interfície de creació de comptes d'usuaris, pot realitzar l'operació i registrar-se en el sistema amb les dades requerides:

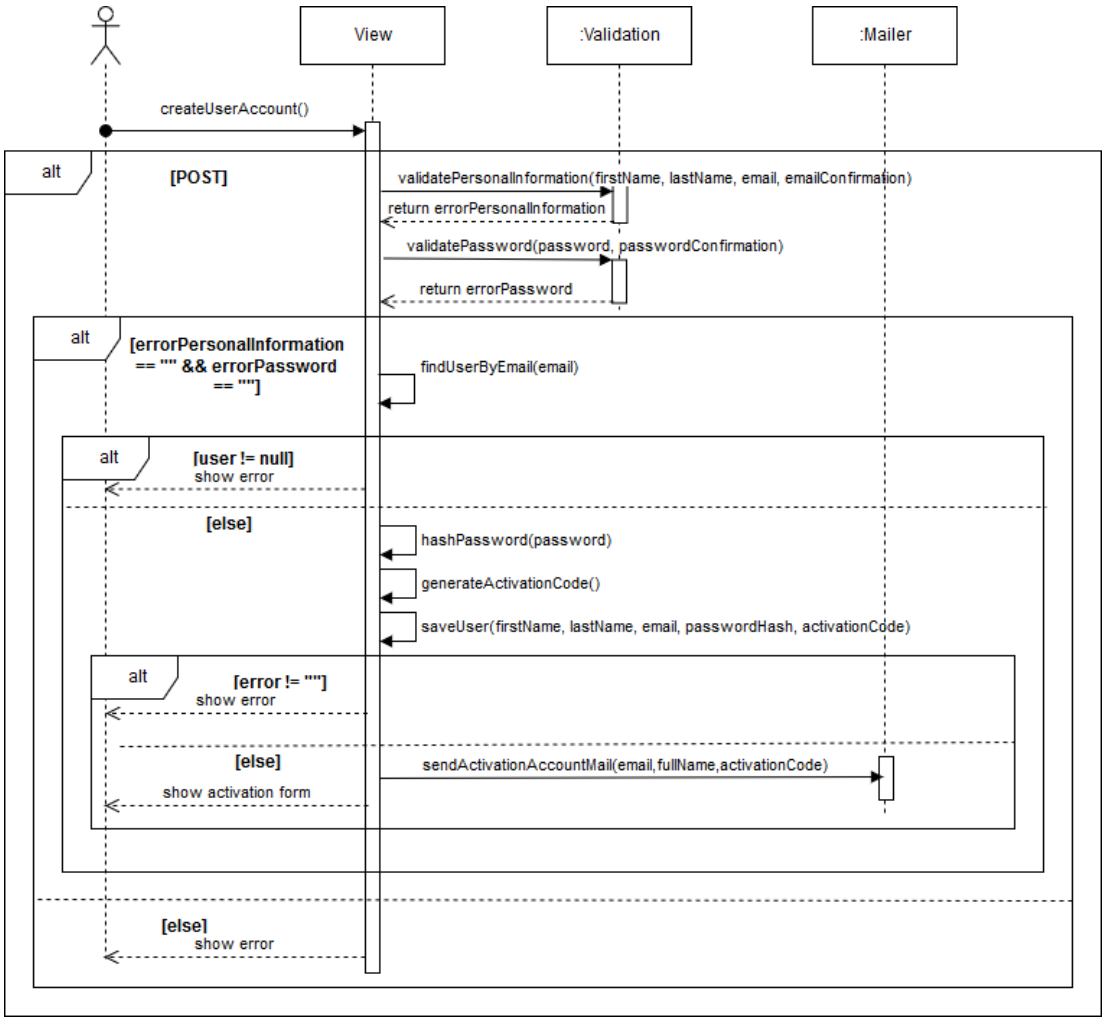


Figura 11: Diagrama de seqüència - crear compte d'usuari

Seguint els passos del diagrama, el primer pas és realitzar les validacions corresponents a cada cadena de text. Aquestes validacions són:

- Longitud de les cadenes de text.
- Valors sense caràcters especials malintencionats.
- Correu electrònic amb el format adient.
- Contrasenyes mínimanent segures
- Usuari ja registrat.

Seguidament, si no són vàlides, es mostrarà un error en el camp de text corresponent per informar a l'usuari. En cas contrari, només en cas que l'usuari no

existeixi en la base de dades, el sistema xifrarà la contrasenya introduïda i desarà el nou usuari en la base de dades juntament amb el codi d'activació generat. Finalment, el sistema enviarà un email a l'adreça corresponent, on s'inclou el codi d'activació, i mostrarà el següent formulari per activar el nou compte.

Escenari: activar compte d'usuari

Seguint amb el procés de la creació de compte d'usuari, aquest si no té la sessió iniciada, des de la interfície d'activació de comptes d'usuaris, l'usuari pot introduir el codi rebut al correu electrònic i duu a terme l'activació.

Per tant, abans de consultar a la base de dades si el codi introduït és correcte, la cadena de text serà validada de la següent manera:

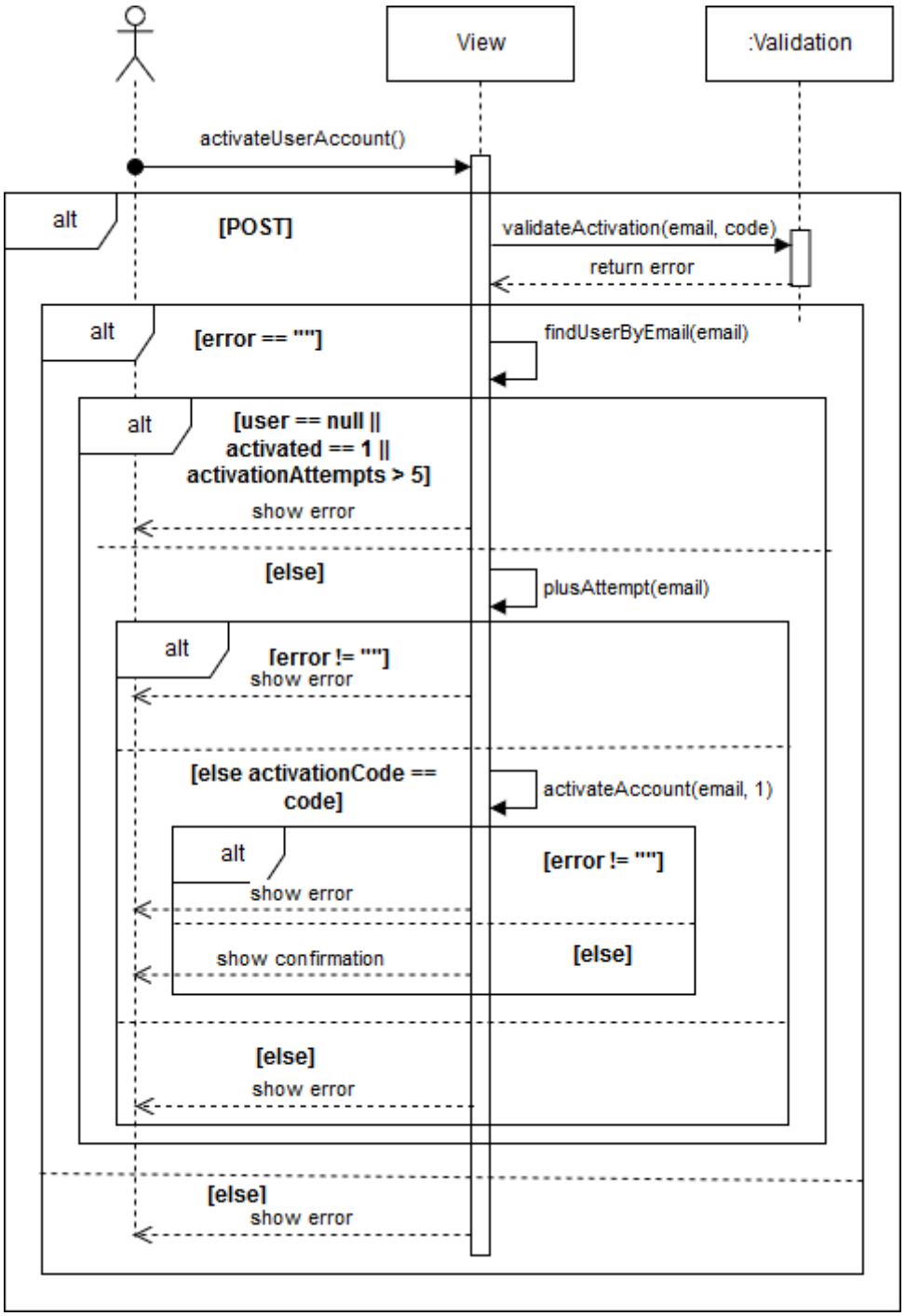


Figura 12: Diagrama de seqüència - activar compte d'usuari

En aquest cas, el sistema abans d'activar l'usuari sumarà un intent a l'usuari i comprovarà si el codi introduït per l'usuari coincideix amb el generat prèviament pel sistema. En cas de no ser així, es mostrarà el corresponent missatge d'error en el formulari. En cas contrari, es mostrarà el missatge de confirmació i l'usuari disposarà

del compte activat fent que ja pugui iniciar sessió.

Com en casos anteriors, també es tindrà en compte el nombre d'intents fallits. D'aquesta manera s'aconsegueix protegir el sistema contra atacs de força bruta o amb diccionaris.

Escenari: iniciar sessió

L'usuari, si no té la sessió iniciada, des de qualsevol interfície a la qual tingui accés i contingui la capçalera, pot inicia sessió introduint les credencials en el formulari.

Com s'ha vist en el cas anterior, en qualsevol camp de text s'ha de controlar les cadenes de text que introdueix l'usuari. Per tant, abans de consultar a la base de dades si les credencials són correctes, aquestes cadenes seran netejades de la següent manera:

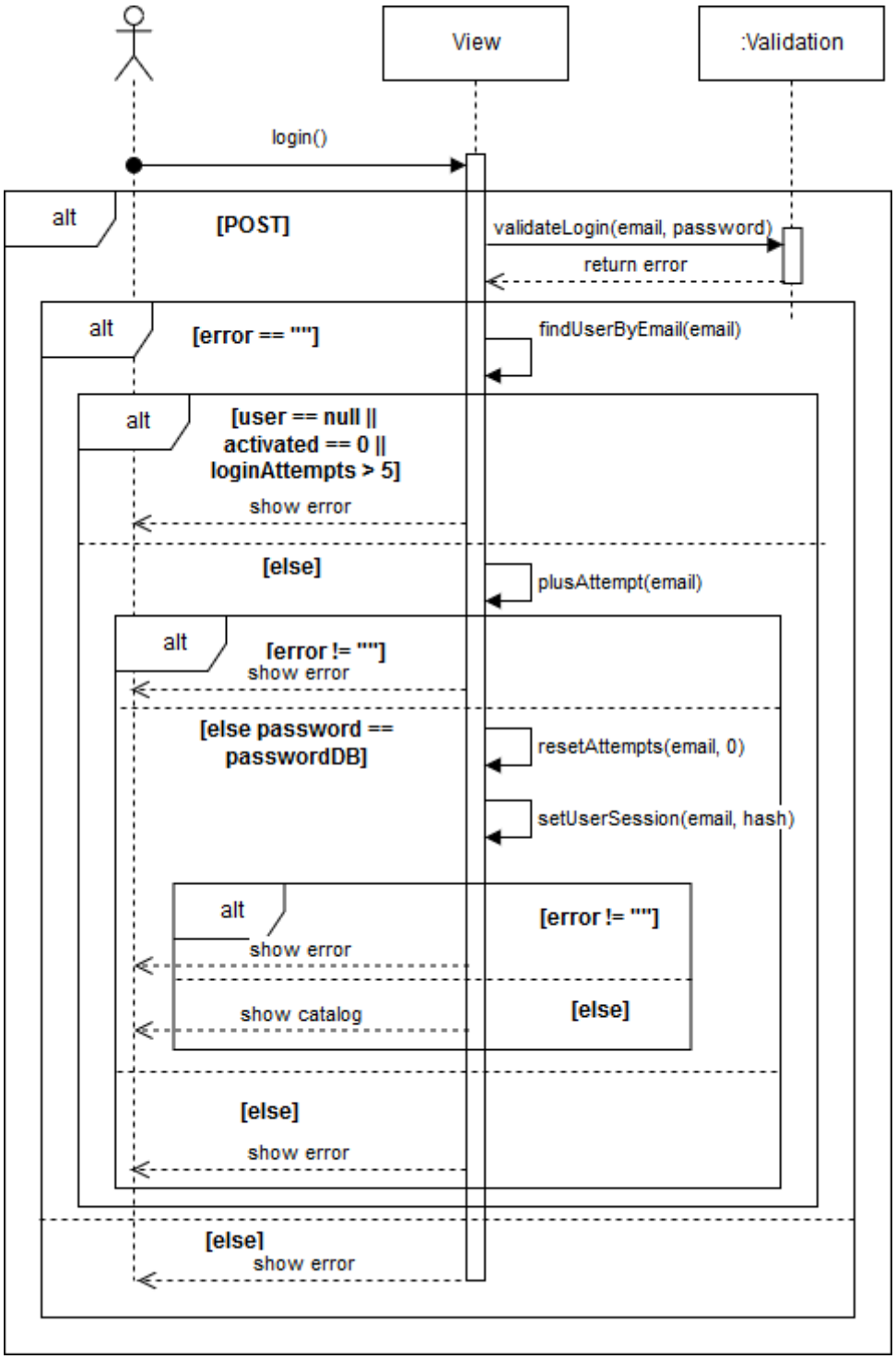


Figura 13: Diagrama de seqüència – iniciar sessió

Seguint els passos del diagrama, el primer pas és validar les cadenes de text introduïdes en el formulari. Seguidament, cal calcular la cadena *hash* de la

contrasenya introduïda, degut que, com es podrà veure en el següent escenari, en la base de dades no es desa la contrasenya sense xifrar, ja que suposaria un error greu de la seguretat de la informació.

Finalment, el sistema comprova l'existència de l'usuari en la base de dades i en cas de coincidir les credencials, i tenir el compte activat, el sistema inicia sessió.

En aquest cas, també s'ha de comptar el nombre d'intents fallits. Això és degut que l'usuari pot aplicar atacs de força bruta o de diccionari per intentar accedir al sistema amb un altre compte d'usuari. Per tant, es bloquejarà temporalment l'accés al sistema un cop arribi al màxim nombre d'intents.

Escenari: modificar contrasenya

L'usuari, si no té la sessió iniciada, des de el formulari d'inici de sessió pot recuperar i assignar una nova contrasenya fent clic a l'enllaç corresponent.

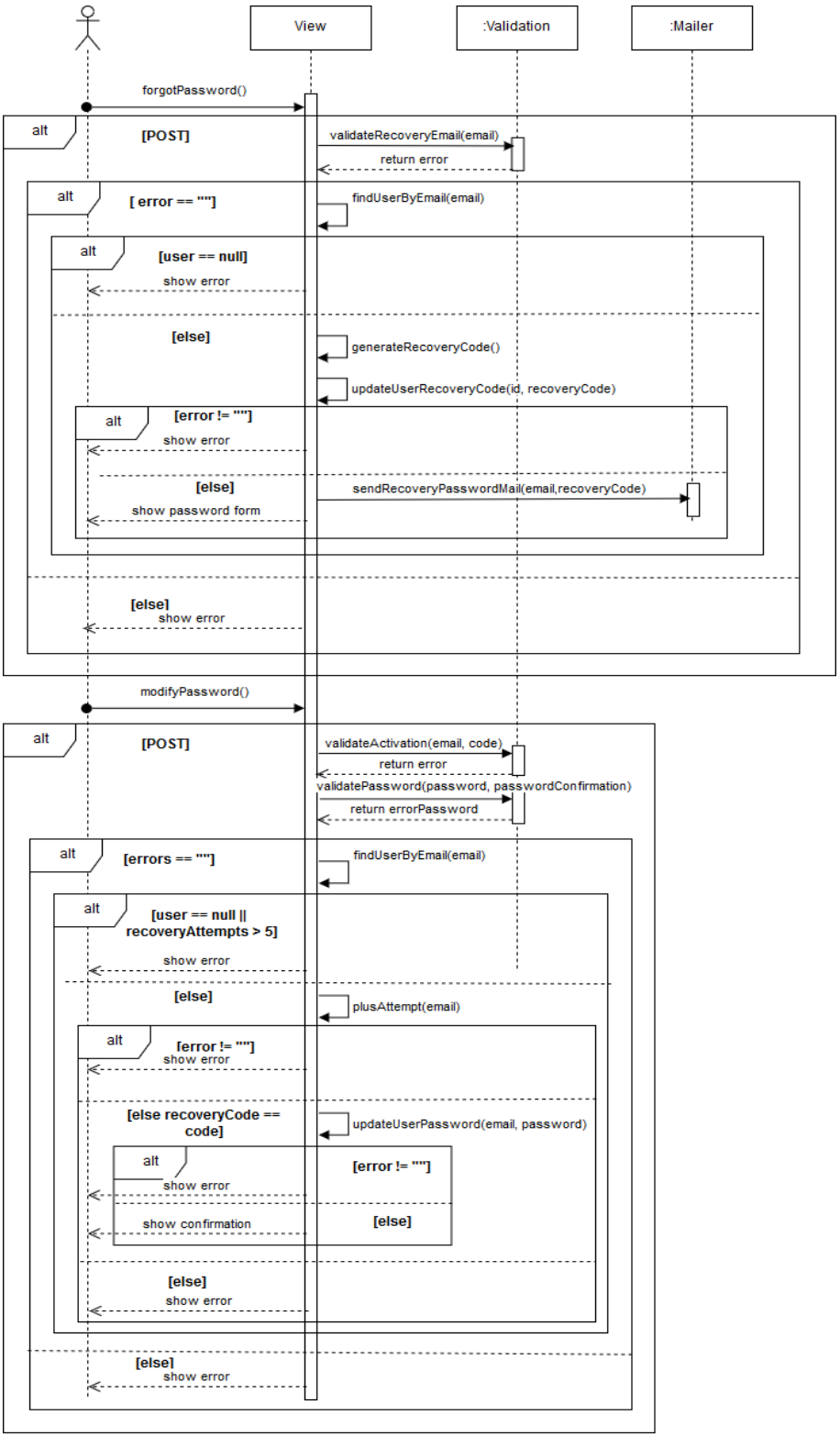


Figura 14: Diagrama de seqüència – modificar contrasenya

Primerament, el sistema realitzarà una validació per comprovar que l'usuari ha introduït un correu electrònic amb el format adient. Seguidament, en cas que no hi hagi cap error, es comprovarà l'existència de l'usuari mitjançant el correu. En cas de no existir, es mostrarà el missatge d'error corresponent. En canvi, en cas d'existir es generarà un codi únic de recuperació i s'enviarà per correu. Finalment, el sistema mostrarà el següent formulari.

Un cop rebut el correu electrònic, l'usuari pot assignar la nova contrasenya. Com en tots els casos on el client pot enviar informació, es validaran totes les dades rebudes en el servidor. A més a més, es tornarà a comprovar l'existència d'aquest. En cas de no existir cap error, el sistema tindrà en compte el nombre d'intents fallits.

Finalment, si tot ha anat bé, el sistema sumarà un intent de recuperació de compte i comprovarà que el codi enviat coincideix amb el que existeix en la base de dades. En cas d'existir, l'usuari tindrà actualitzada la nova contrasenya.

Escenari: realitzar pagament

L'usuari, si té la sessió iniciada i ha afegit productes al carret, des de la interfície del resum de compra pot accedir al formulari per realitzar el pagament corresponent.

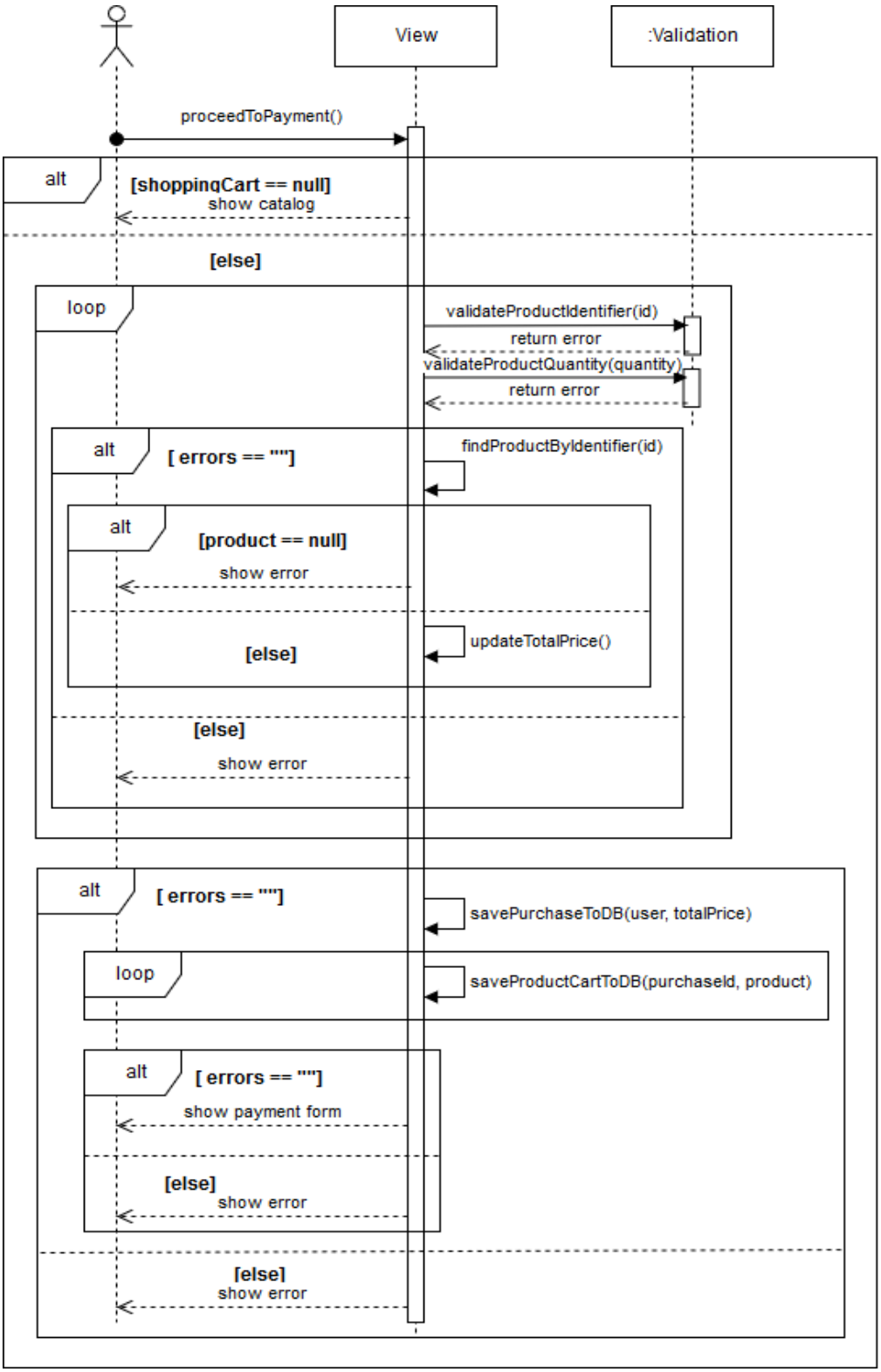


Figura 15: Diagrama de seqüència – realitzar pagament, primera part

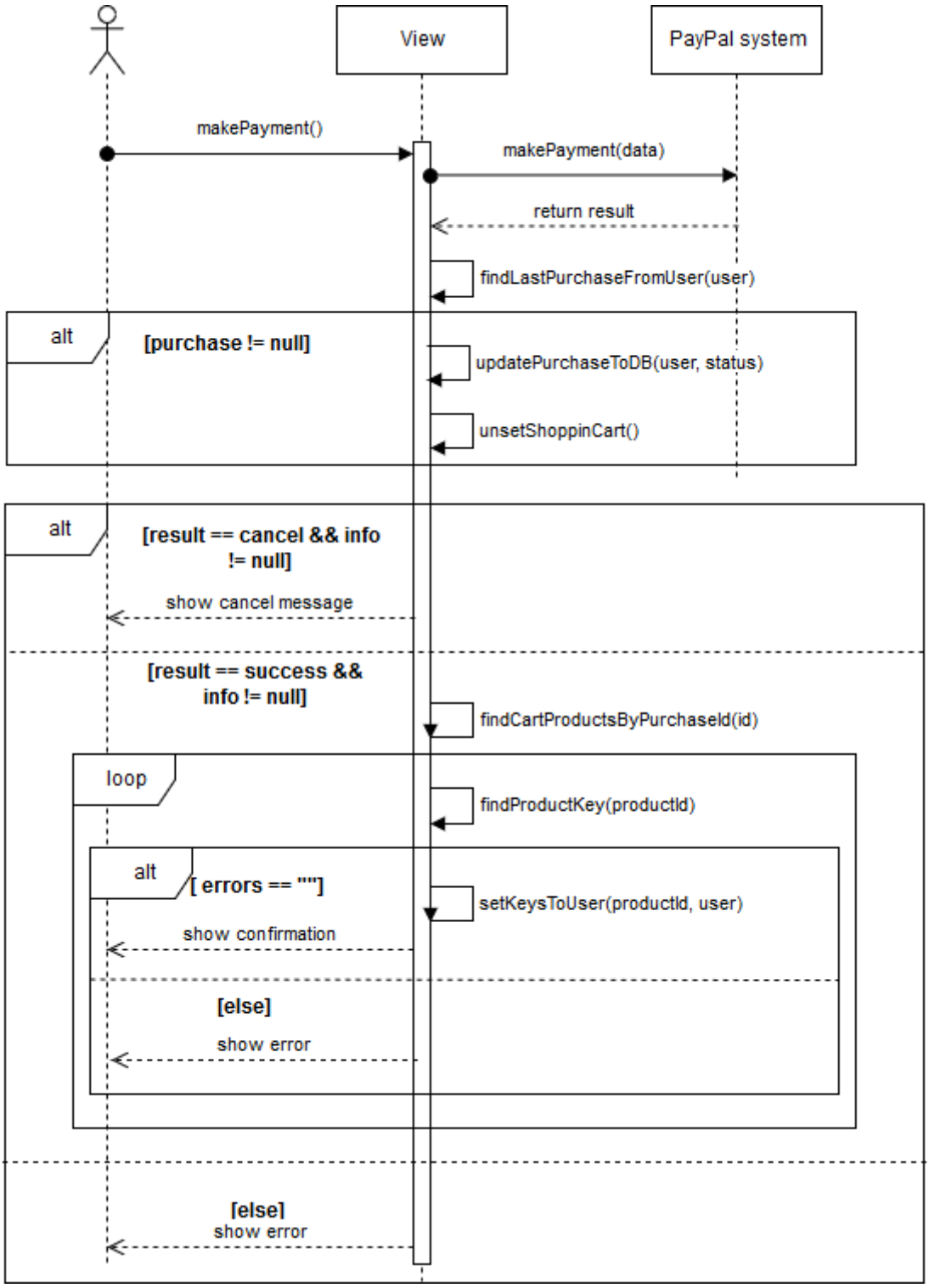


Figura 16: Diagrama de seqüència – realitzar pagament, segona part

Primerament, el sistema comprova l'existència dels productes del carret en la base

de dades i obté els preus desats en aquesta. D'aquesta manera s'evita que l'usuari modifiqui, al seu gust, els preus i pugui realitzar compres gratuïtes. Seguidament, sistema desa el procés del pagament juntament amb els articles que es desitgen adquirir.

Un cop l'usuari hagi seleccionat el mètode de pagament, el sistema crearà i enviarà la petició de pagament al sistema de l'entitat del mètode de pagament seleccionat. Un cop rebuda la resposta externa, el sistema desarà totes les dades del procés i notificarà a l'usuari amb un error o una confirmació. Si, i només si, el pagament ha sigut realitzat amb èxit, el sistema procedeix a assignar les claus corresponents als productes a l'usuari. En cas que les existències s'hagin esgotat, s'informarà a l'usuari mitjançant el missatge corresponent. En cas contrari, les claus quedaran assignades a aquest i les podrà visualitzar en el registre de compres.

Escenari: veure historial de compres

L'usuari, si té la sessió iniciada, des de qualsevol interfície que contingui la capçalera pot fer clic a les opcions d'usuari i visualitzar l'historial de compres realitzades.

Per tant, el sistema cercarà totes les compres realitzades per l'usuari en el sistema i la vista els mostrarà:

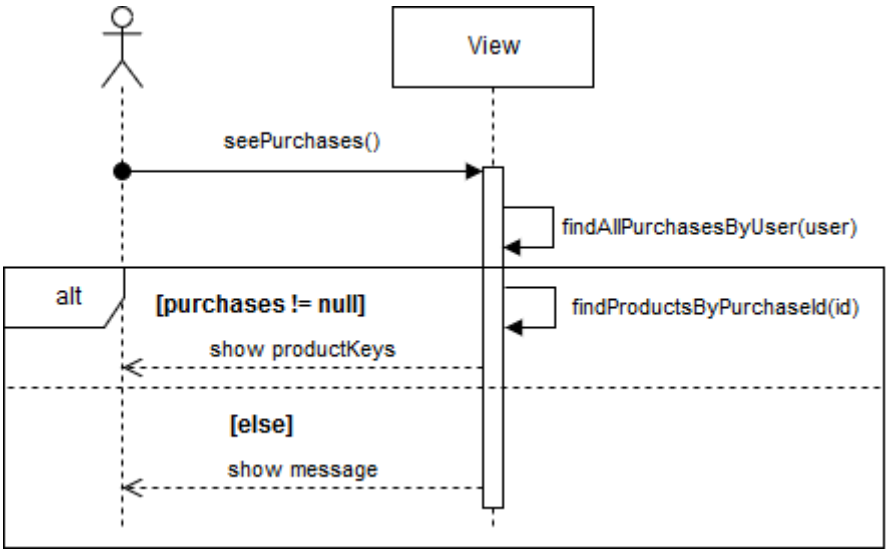


Figura 17: Diagrama de seqüència – visualitzar l'historial de compres

Escenari: tancar sessió

L'usuari, si té la sessió iniciada, des de qualsevol interfície que contingui la capçalera pot fer clic a les opcions d'usuari i tancar la sessió actual. Per tant, un cop

tancada la sessió, el sistema s'encarregarà de realitzar una redirecció cap a la pàgina principal de la botiga:

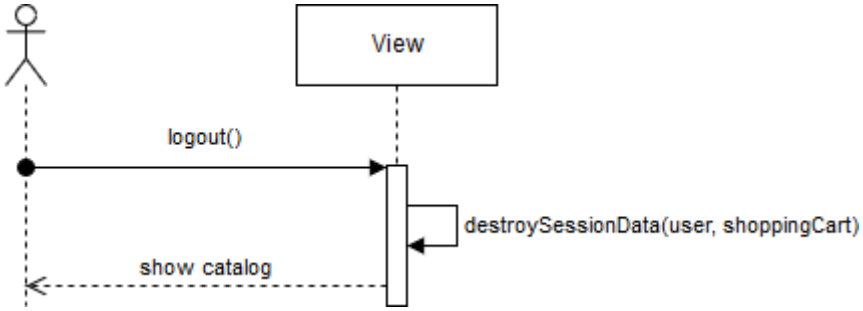


Figura 18: Diagrama de seqüència – tancar sessió

Escenari: control de sessió

Aquest escenari correspon al control d'accés a les vistes de la botiga que es poden veure en la figura 1. Per tant, segons els tipus de vista s'aplica un control d'accés o un altre:

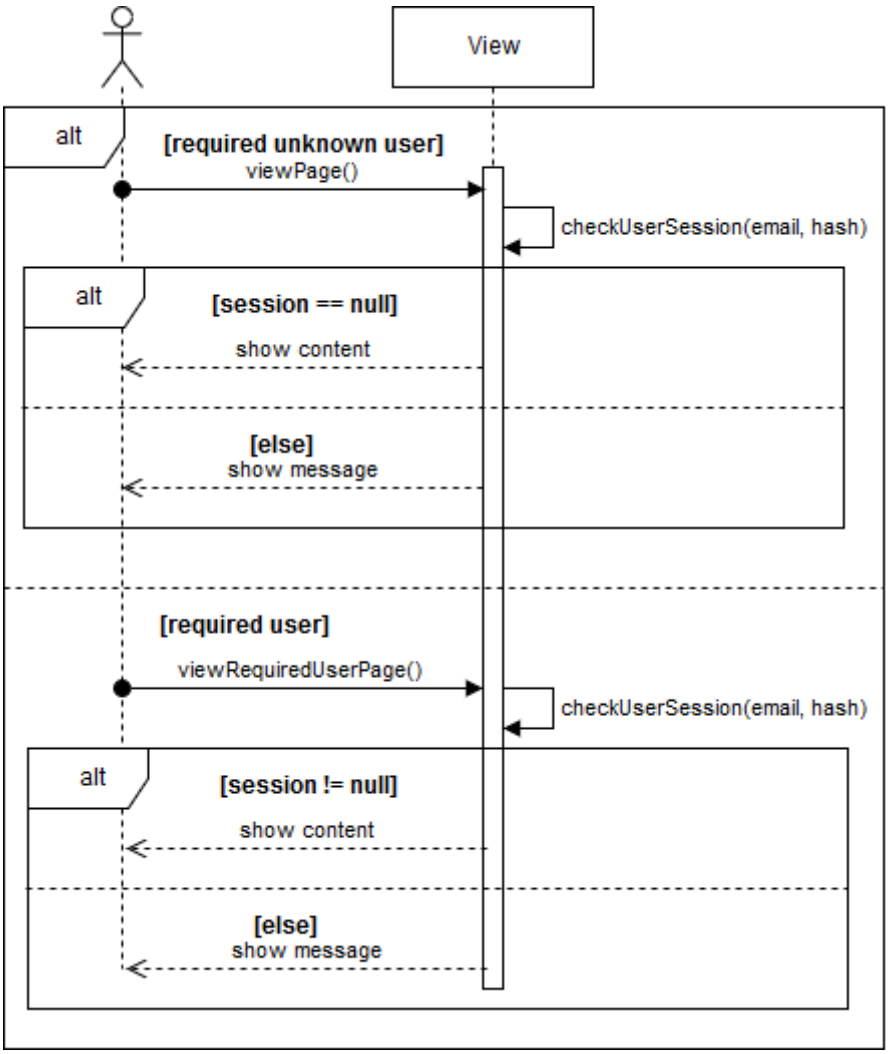


Figura 19: Diagrama de seqüència – control de la sessió

4.4. Anàlisi de seguretat

Un cop vistos tots els diagrames i amb una imatge de l'anàlisi del funcionament de la botiga, en aquest subapartat s'analitzen els aspectes més rellevants de la solució relacionats amb la seguretat.

Sessió d'usuari

Cada cop que un usuari inicia sessió en la botiga és necessari desar en la sessió del servidor un identificador que permeti al sistema identificar a l'usuari durant la navegació en el sistema.

Aquest identificador ha de ser únic en la sessió i impossible de suplantar, com per exemple un *token* d'usuari o una cadena de caràcters resultats d'un xifrat amb algoritmes com SHA-512 o similars.

D'aquesta manera, es protegeix al sistema controlant les sessions dels usuaris evitant l'accés no desitjat als recursos d'aquest.

Control d'accés

Com s'ha pogut observar en el diagrama de flux de la figura 1, existeixen casos on és necessari controlar l'accés a aquestes vistes. Segons l'estat de la sessió, esmentada anteriorment, el client tindrà accés, o no, a la vista determinada.

Per tant, el sistema ha de verificar, segons el cas, l'estat de la sessió de l'usuari realitzant una comparació del valor actual desat en sessió amb el generat en el moment de la comprovació. Només en cas que els valors coincideixin, el sistema pot determinar que l'usuari té la sessió iniciada.

D'aquesta manera, com en el cas anterior, es dur a terme un control evitant l'accés no desitjat a les vistes.

Validacions de formularis

Com s'ha anat esmentant anteriorment, cal realitzar una validació de tots els valors d'entrada possibles en el sistema, tant dels valors enviats per formularis com altres valors propis de les vistes.

Els tipus de validacions que s'han de realitzar han de ser de quatre tipus:

- Validacions de contingut: han de verificar que un camp requerit contingut valor.

- Validacions de longitud: han de verificar que la longitud del valor no excedeix la màxima del camp corresponent en la base de dades, evitant així errors corresponents en inserir informació en aquesta.

- Validacions d'igualtat: han de verificar que dos camps contenen el mateix valor. Aquest pot ser el cas, per exemple, els camps que fan referència a la contrasenya del formulari de creació de comptes, on es requereix que l'usuari escrigui dos cops el mateix valor.

- Validacions de format: han de verificar que el valor conté un format específic, evitant així qualsevol tipus d'atac d'injecció. Aquests valors es poden controlar mitjançant l'ús d'expressions regulars o qualsevol altra tècnica de patrons.

Bloqueig de compte d'usuari

Per protegir el sistema, i els comptes d'usuaris existents, contra persones malintencionades que, mitjançant atacs de força bruta o mitjançant diccionaris, intenten aconseguir l'accés als recursos protegits de la botiga, és important aplicar un bloqueig temporal o permanent als comptes d'usuari que hagin sigut víctimes d'aquests atacs.

Per determinar que un compte està en perill, el sistema ha d'anar comptant els intents fallits que ha realitzat l'usuari i en arribar a una certa quantitat de vegades, es bloqueja l'usuari que estigués realitzant l'operació i ja no ha de poder tornar-la a executar.

Un clar exemple, és en intentar iniciar sessió d'un usuari, on la persona malintencionada pot realitzar intents fallits intentant esbrinar la contrasenya associada al correu electrònic de la víctima.

Verificació d'usuaris

El sistema ha de verificar que l'usuari correspon a l'adreça del correu electrònic que especifica en el formulari de creació del compte d'usuari.

Per tant, el sistema ha de proporcionar, mitjançant un correu electrònic, la possibilitat de verificar la seva identitat, tant en el procés d'activació del compte com en la seva recuperació. Aquest mètode pot ser mitjançant un enllaç que redirigeixi cap al formulari adient o un codi generat únicament per la transacció corresponent.

D'aquesta manera es dificulta l'activació o la recuperació del compte d'usuari a la persona malintencionada, fent que, per aconseguir els seus propòsits, hagi de tenir accés al compte de correu electrònic esmentat.

Correu electrònic

El sistema ha de garantir que l'enviament d'informació, mitjançant un servidor de correus electrònics, es faci de manera segura, protegint l'accés a la informació contra els possibles espies de la xarxa.

Per tant, cal realitzar les configuracions necessàries en el servidor per garantir que aquestes mesures es compleixen i que les dades que arriben a l'usuari no han sigut llegides o modificades.

Les configuracions corresponents es podrà veure més endavant en l'apartat d'implementacions de seguretat.

Procés de pagament

En aquest procés és on els usuaris realitzaran el pagament bancari en la passarel·la de cobraments del banc corresponent. Segons el mètode de pagament triat, el sistema realitzarà una redirecció cap a la passarel·la externa pròpia de l'entitat corresponent al mètode. Per tant, aquesta és responsable d'aplicar les seves pròpies mesures de seguretat i d'emparar a l'usuari en tot el seu procediment fins al retorn cap a la botiga.

No obstant això, el sistema ha de realitzar les verificacions necessàries per comprovar que el procés de pagament s'ha realitzat amb èxit, com per exemple que les transaccions han sigut completades, que els imports cobrats són els esperats, que no s'ha intentat realitzar cap devolució, i en definitiva qualsevol altra validació necessària per garantir la validesa del pagament.

Visualització de contingut sensible

Aquesta mesura es dóna en cas de la visualització de les claus dels productes comprats per part de l'usuari. Aquestes claus són úniques i, en cas de ser copiades per terceres persones, no es poden tornar a utilitzar, deixant a l'usuari sense producte.

Per tant, s'ha d'implementar un mètode perquè l'usuari pugui mostrar, o amagar quan desitgi, les claus dels productes comprats per posteriorment copiar-les i utilitzar-les en les seves corresponents plataformes.

D'aquesta manera s'aconsegueix protegir la informació de caràcter sensible contra persones que tinguin accés físic a la visualització d'aquesta, com per exemple els espais públics.

5. Desenvolupament

En aquesta secció del document entrarem més en detall en el procés de desenvolupament de *Gaming Keys*. En els següents subapartats s'exposaran totes les diferents possibilitats que ofereix aquest sistema, des del disseny i configuracions del projecte fins a les funcionalitats i la seguretat que s'implementa en cadascuna d'elles.

5.1. Arquitectura multicapa

El disseny d'aquest projecte es basa en un conjunt de subsistemes que interaccionen entre si organitzats en una arquitectura de tres capes:

Figura 20: Arquitectura multicapa

- Capa de presentació, també coneguda com la interfície gràfica, s'encarrega de la presentació de la informació a l'usuari i de l'obtenció de les dades d'aquest per enviar-les al servidor. En aquesta capa, s'utilitza qualsevol navegador per accedir a la botiga.

- Capa de negoci o lògica s'encarrega de realitzar totes les operacions necessàries per dur a terme la funcionalitat desitjada.

- Capa de persistència és l'encarregada d'emmagatzemar tota la informació de l'aplicació, a més d'assegurar l'accés a les dades d'una forma controlada i segura. Aquest nivell es compon per la base de dades i el software de gestió de base de dades.

A més a més, la capa de negoci s'integrarà amb dos serveis externs a l'aplicació:

Figura 21: Integracions

- Servidor de correu electrònic és el component que s'encarrega de realitzar tota

la funcionalitat referent als correus electrònics, com la configuració, construcció i l'enviament d'aquests. Per tant, aquest component capacita a la botiga per enviar correus electrònics als usuaris.

- PayPal és el component que conté el sistema del mètode de pagament que s'encarrega de realitzar les operacions necessàries per dur a terme les transaccions bancàries dels usuaris. Per tant, possibilita a la botiga un mètode de pagament fiable i segur perquè els usuaris puguin realitzar les seves compres en el sistema.

5.2. Configuració de l'entorn

En aquest apartat es tractaran tots els aspectes necessaris referents a la instal·lació i les configuracions necessàries per dur a terme el desenvolupament i execució del projecte.

Abans de continuar amb els següents subapartats, cal esmentar que en aquest document, es mostren els tutorials de les versions més actualitzades que existeixen en el moment de la redacció d'aquest apartat i no el de les instal·lacions realitzades per implementar aquest projecte. Per tant, la informació detallada en aquests pot variar segons les versions utilitzades.

5.2.1. Instal·lació i configuració de l'entorn WAMP

Instal·lació

La instal·lació d'aquest sistema resulta relativament fàcil seguint els passos que ofereix l'assistent d'instal·lació. El primer pas és descarregar el programari de l'enllaç següent:

http://www.wampserver.com/en/

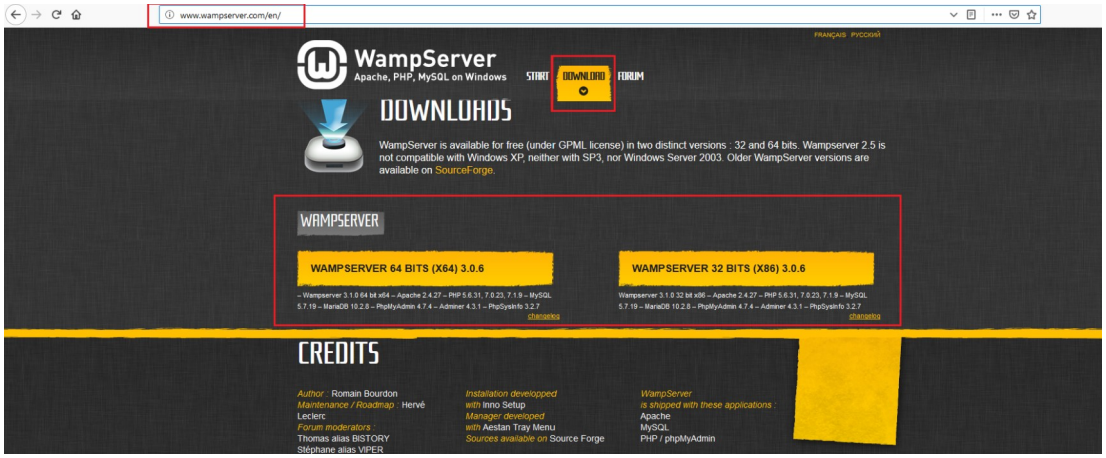


Figura 22: Pàgina de descàrrega de Wamp

Aquest sistema redirigirà cap a una pàgina externa on es pot realitzar la descàrrega del arxiu anomentat *wampserver[versió]_[arquitectura].exe,* on la versió i arquitectura canviarà segons l'opció triada.

Un cop obtingut el fitxer instal·lador, el primer pas és seleccionar l'idioma preferit durant l'instal·lació del programari. Seguidament cal acceptar els termes i condicions d'aquest:

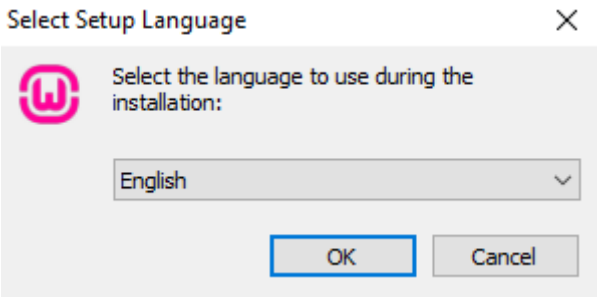


Figura 23: Selecció d'idioma de l'instal·lació de Wamp

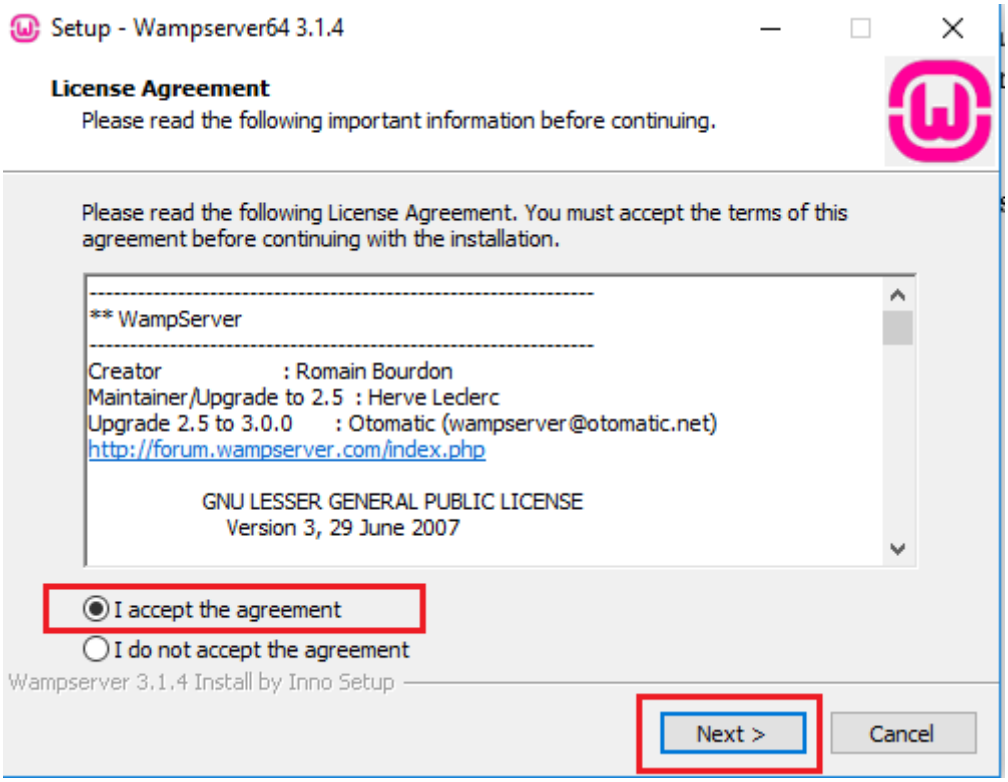


Figura 24: Acceptació de condicions de Wamp

Seguidament, mostra informació corresponent a la pròpia instal·lació i, a continuació, cal seleccionar la carpeta on s'ubicarà tot el contingut:

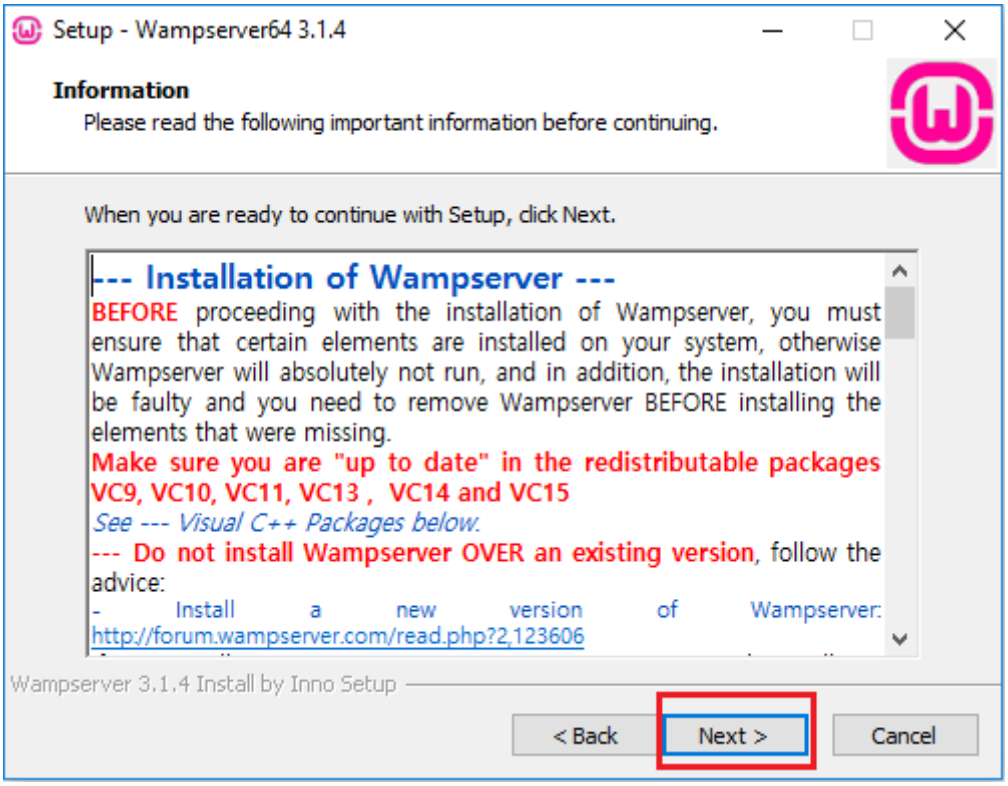


Figura 25: Informació de l'instal·lació de Wamp

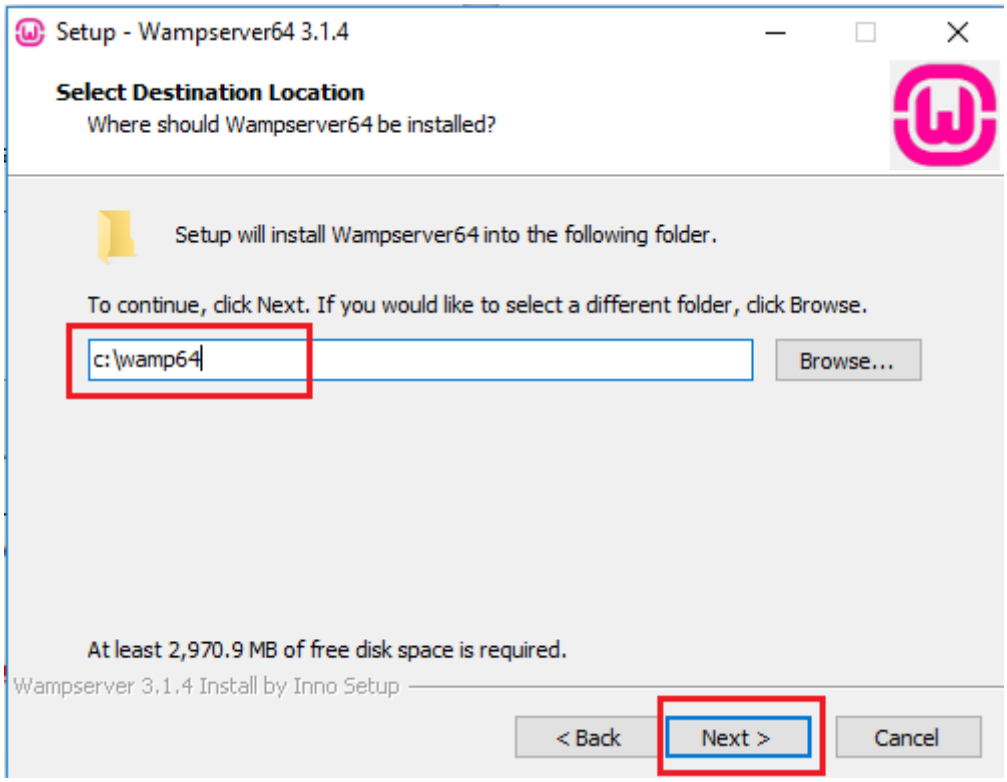


Figura 26: Directori de l'instal·lació de Wamp

Finalment, el fitxer procedirà a la extracció i instal·lació de tots els components necessaris del programari:

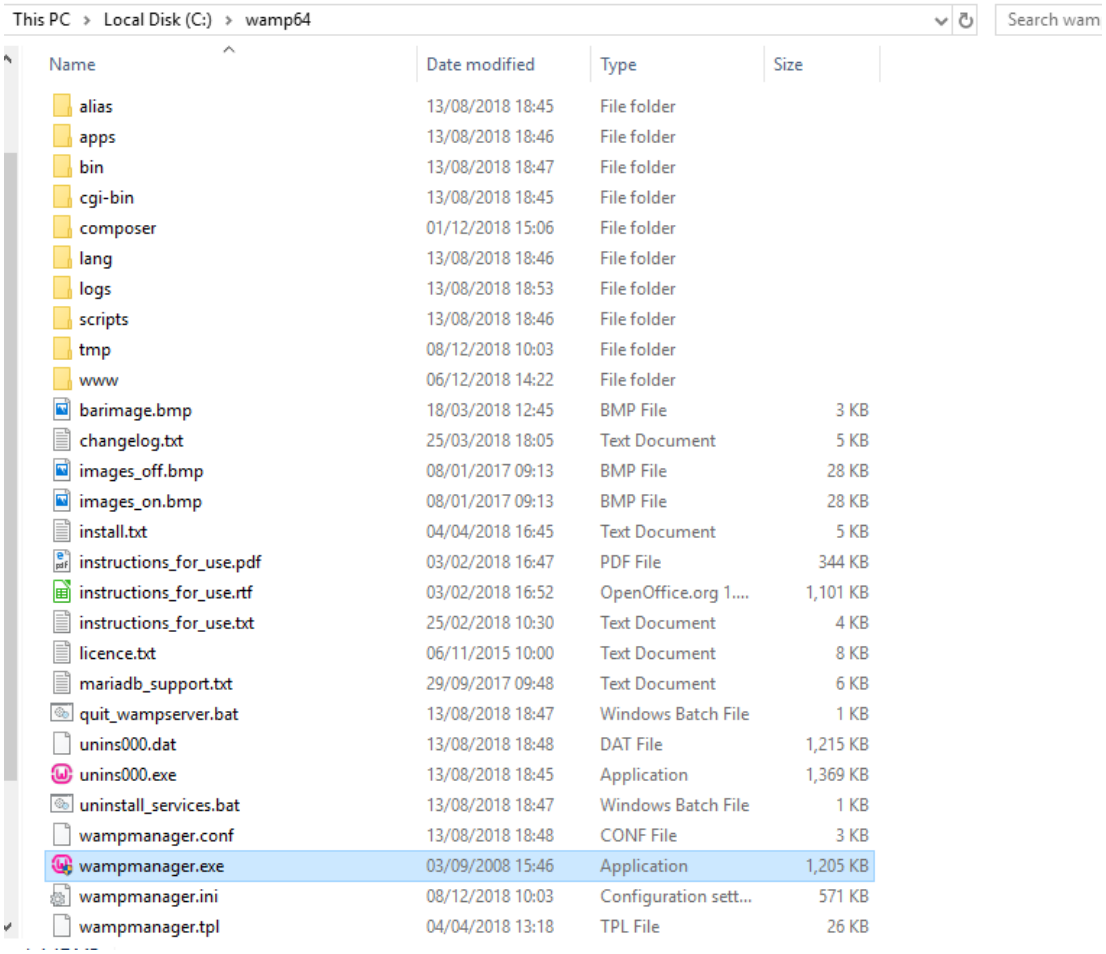


Figura 27: Resultat de l'instal·lació de Wamp

Configuració

Aquest programari, en aquest projecte, no requereix cap configuració especial més que l'execució del programari i l'inici de tots els serveis integrats, els quals s'inicien automàticament amb l'inici del programa. Per tant:

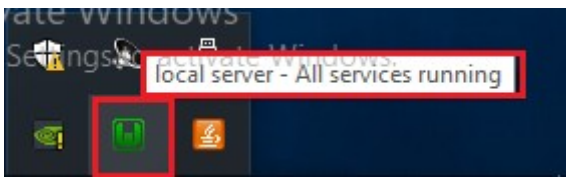


Figura 28: Icona de Wamp

5.2.2. Instal·lació i configuració de l'editor de text Sublime Text

Instal·lació

En aquest cas, tant la instal·lació com la configuració d'aquest editor no té cap mena de complexitat i només cal seguir els passos que ofereix l'assistent d'instal·lació. Per tant, com en el cas anterior, el primer pas és descarregar el programari de l'enllaç següent:

https://www.sublimetext.com/

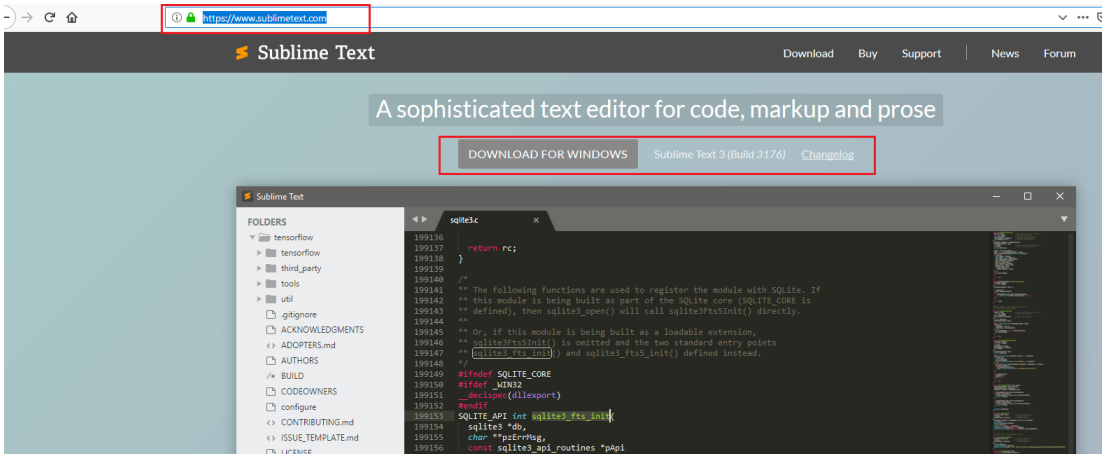


Figura 29: Pàgina de descàrrega de Sublime Text

Un cop obtingut el fitxer instal·lador, el primer pas és seleccionar la carpeta on s'ubicarà la instal·lació del programari:

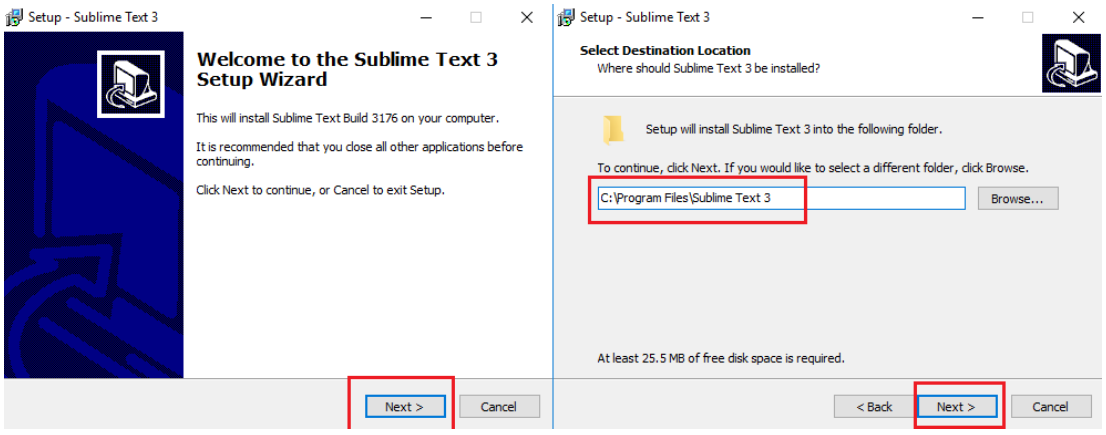


Figura 30: Primers pasos de l'instal·lació de Sublime Text

Seguidament, cal especificar configuracions addicionals sense cap mena de

rellevància i finalment, procedir amb la instal·lació:

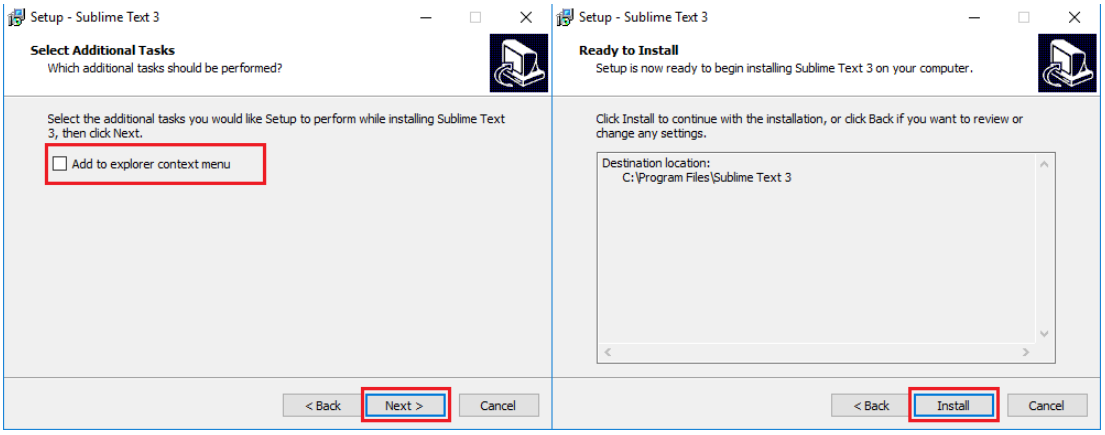


Figura 31: Següents pasos de l'instal·lació de Sublime Text

Configuració

Aquest programari, no requereix cap configuració especial més que l'execució del programari. No obstant això, en aquest editor es poden afegir moltes expansions, eines i configuracions externes per fer que l'editor de text satisfaci totes les necessitats requerides per l'usuari. En aquest projecte en concret no són necessàries.

Figura 32: Interfície de Sublime Text

5.2.3. Instal·lació i configuració de PHPMailer

Degut que el desenvolupament de la funcionalitat que fa referència a l'enviament de correus electrònics requereix un alt nivell de coneixements de l'estàndard SMTP i les diferents vulnerabilitats, en *Gaming Keys* s'ha optat per la utilització d'una de les solucions més populars que existeixen en PHP, anomenada PHPMailer[6].

PHPMailer és una llibreria de codi PHP que permet integrar en el codi la

funcionalitat corresponent amb l'enviament de correus electrònics de manera simple i segura. Aquesta llibreria pot ser integrada en el projecte de diferents maneres. En aquest cas, s'ha utilitzat l'eina anomenada Composer[5].

Què és Composer? Resumidament, és un gestor de dependències que permet la instal·lació de tots els paquets i llibreries necessàries dins d'un directori local, per defecte */vendor*. Cal afegir que aquest programa és capaç de resoldre les dependències que tenen les mateixes dependències de manera automatitzada, fent que la tasca de la descàrrega i instal·lació de les necessitats sigui realment fàcil.

Instal·lació

Primerament cal instal·lar l'eina Composer seguint, un cop més, l'assistent d'instal·lació. Cal esmentar que, en aquesta ocasió, requereix especificar la línia d'ordres de PHP que es vol utilitzar, en aquest cas la instal·lada amb el programari WAMP anteriorment:

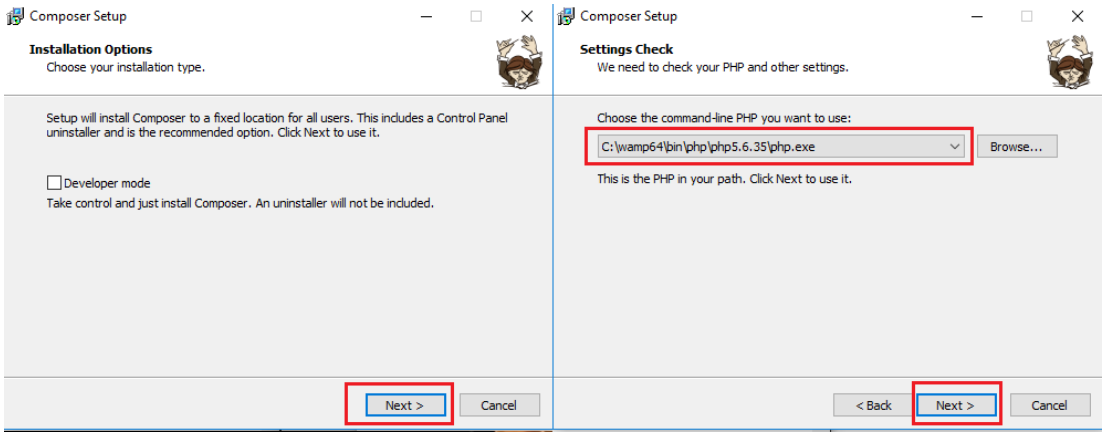


Figura 33: Primers pasos de l'instal·lació de Composer

Seguidament, només en cas d'utilitzar algun tipus de *proxy*, cal especificar-ho en la següent finestra. Com que no és aquest cas, es deixen els valors per defecte de la finestra i, finalment, es procedeix amb la instal·lació del programari:

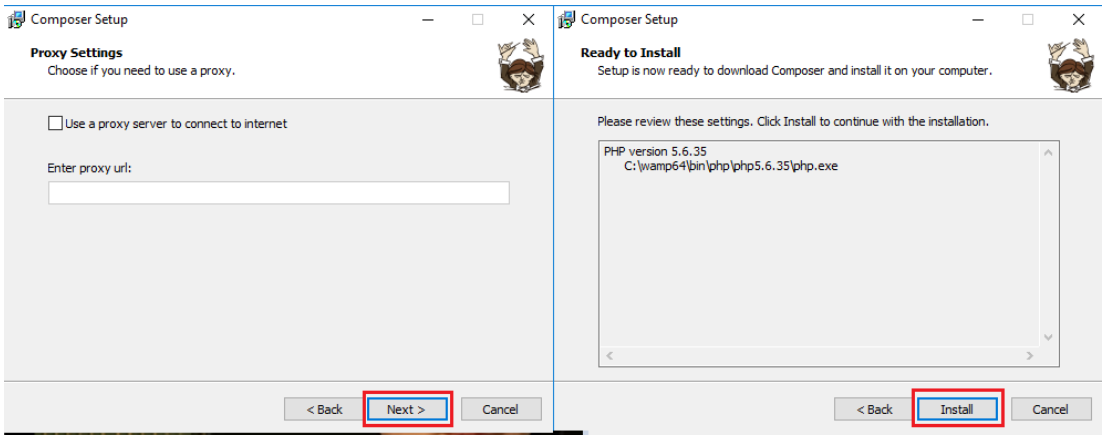


Figura 34: Següents pasos de l'instal·lació de Composer

Al final del procediment, crearà el directori corresponent dins del directori de WAMP:

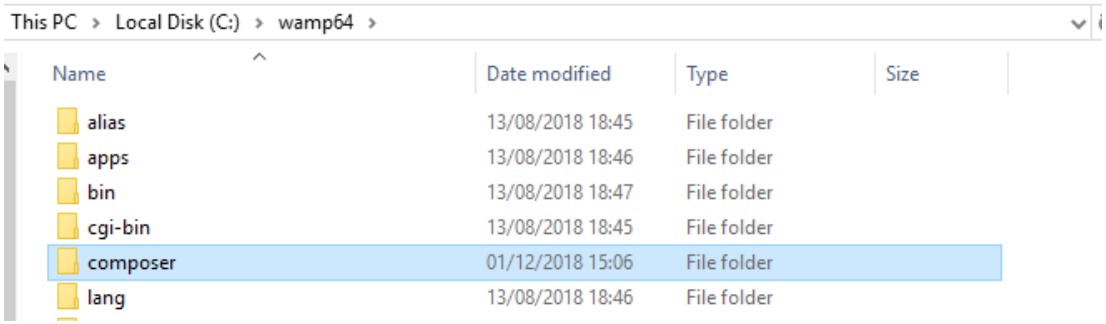


Figura 35: Resultat de l'instal·lació de Composer

Un cop finalitzada la instal·lació, cal importar les dependències que fan referència a la llibreria en qüestió. Composer és el programari encarregat que aquest procediment no tingui cap mena de complexitat. Per tant, només cal obrir la línia d'ordres de Windows, assegurar-se que la ubicació sigui la del directori creat anteriorment i executar la següent ordre:

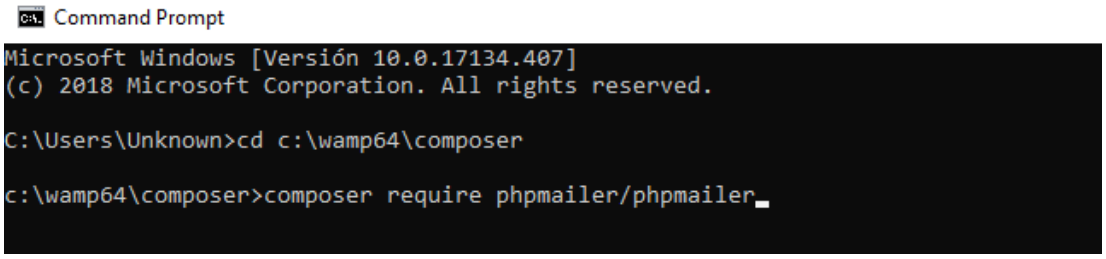


Figura 36: Descàrrega de PHPMailer

El resultat d'aquesta execució ha de ser similar al següent:

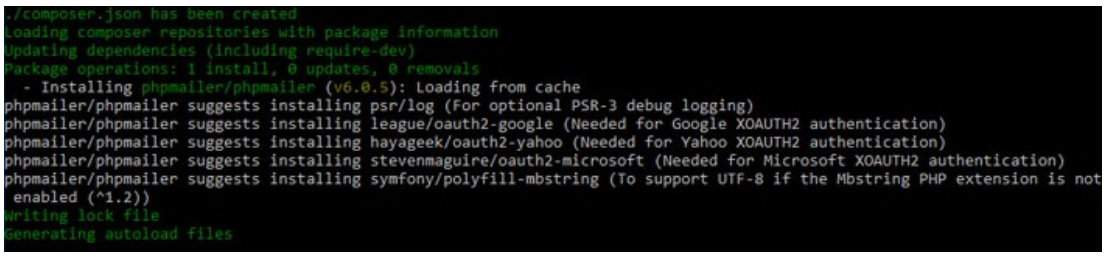


Figura 37: Resultat de la descàrrega de PHPMailer

Configuració

Amb els passos anteriors, s'ha instal·lat en el sistema totes les dependències necessàries per dur a terme la funcionalitat que fa referència al correu electrònic. Com que en aquest projecte es treballa localment i s'implementa la funcionalitat més bàsica per cobrir les necessitats, no es requereix cap configuració extra de la qual ja ve per defecte.

5.2.4. Creació de la base de dades

La següent configuració que cal realitzar és la creació de la base de dades, juntament amb la creació d'usuaris, taules i dades necessàries per l'execució de *Gaming Keys*.

Per tant, simplement cal accedir al servei de PhpMyAdmin que ofereix WAMP, amb les credencials, per defecte, usuari 'root' i sense cap contrasenya, des de les icones de la dreta:

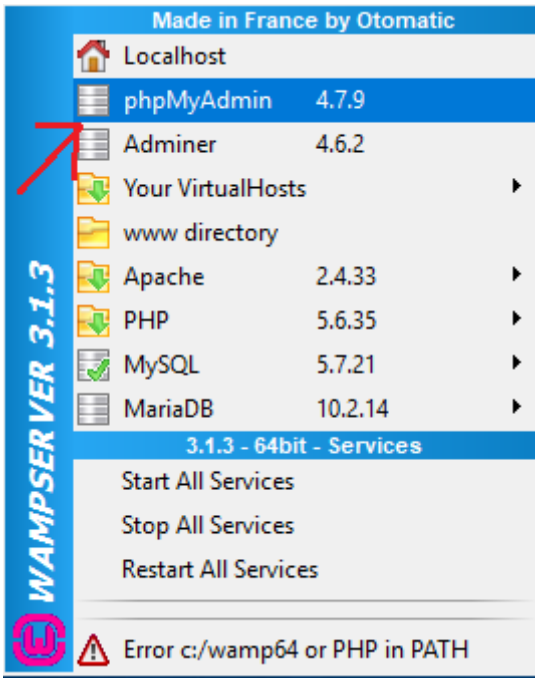


Figura 38: Serveis de Wamp

O des de l'adreça, per defecte: http://localhost/phpmyadmin/

Figura 39: Formulari d'inici de sessió en PhpMyAdmin

Seguidament, cal executar els *scripts* SQL, adjunts a l'apartat d'*scripts* de la base de dades d'aquest document i en l'ordre especificat. Per poder dur a terme aquesta tasca, cal seleccionar la pestanya 'SQL' i executar l'*script* desitjat.

Per l'execució del primer script, el que conté el codi que fa referència a la creació de la base de dades i de l'usuari, no cal seleccionar cap base de dades prèviament. Per tant:

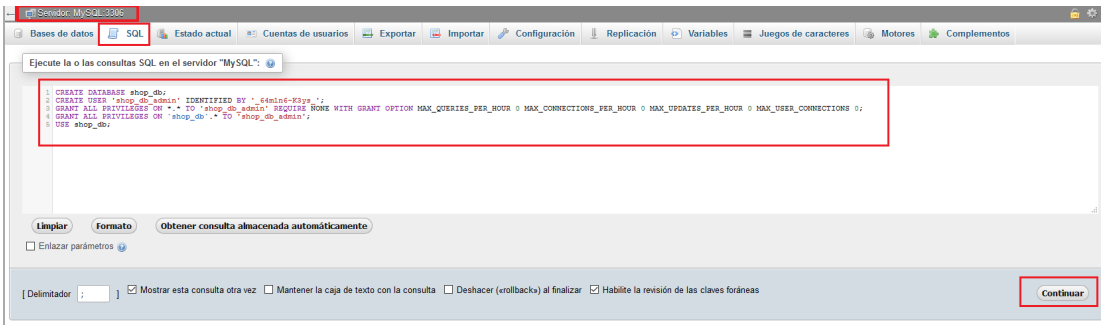


Figura 40: Script de creació de la base de dades en PhpMyAdmin

Si tot ha anat bé, s'ha de visualitzar una pantalla com la següent:

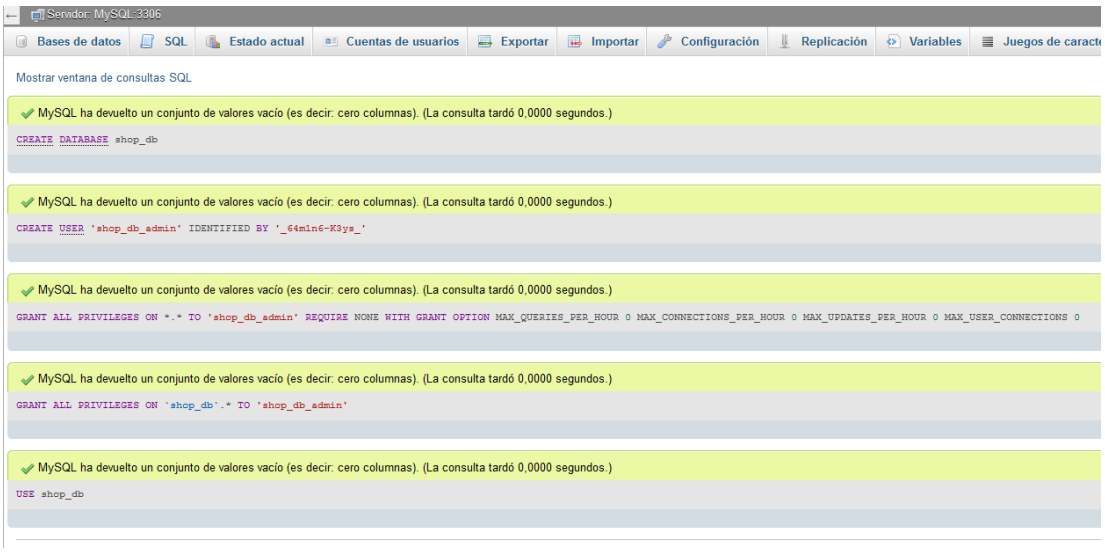


Figura 41: Resultat de l'execució de l'script

Abans de seguir amb els següents scripts, cal esmentar l'existència d'un *bug* molt conegut de MySQL, referent a la creació d'usuaris:

https://bugs.mysql.com/bug.php?id=28331

Resumidament, aquest error sorgeix quan s'intenta crear un usuari existent en el sistema, tot i haver executat la sentència SQL d'eliminació d'usuaris correctament. Això és degut, almenys en aquest cas, que tot i eliminar correctament l'usuari de les taules corresponents de MySQL, no ha quedat completament eliminat del sistema. Per tant, cal eliminar-ho manualment en la següent finestra:

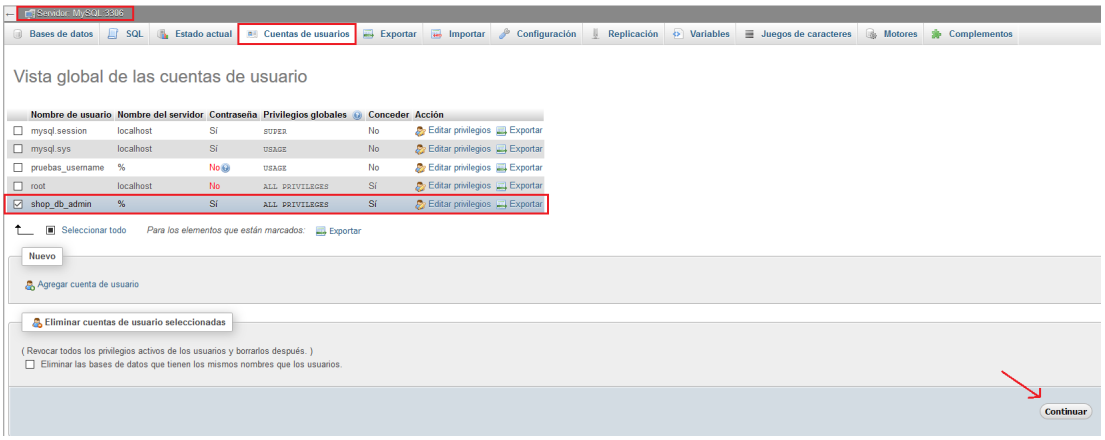


Figura 42: Comptes d'usuari en PhpMyAdmin

Un cop eliminat completament, es pot procedir a la creació de l'usuari mitjançant el codi SQL, en principi, sense cap mena de problema. Seguidament, un cop creada la base de dades i l'usuari administrador corresponent, cal seleccionar la base de dades per l'execució de la resta d'*scripts*:

Figura 43: Execució del següent script

Un cop executats tots els fitxers SQL, la base de dades ja conté la informació necessària per l'execució de la botiga.

5.2.5. Versions de l'entorn

En aquest subapartat, es llisten les diferents versions de l'entorn on es desenvolupa i s'executa la botiga en línia:

- WAMP: versió 3.1.3 de 64 bits.
  - Apache: versió 2.4.33.
  - MySQL: versió 5.7.21.
  - PHP: versió 5.6.35.
  - PhpMyAdmin: versió 4.7.9.

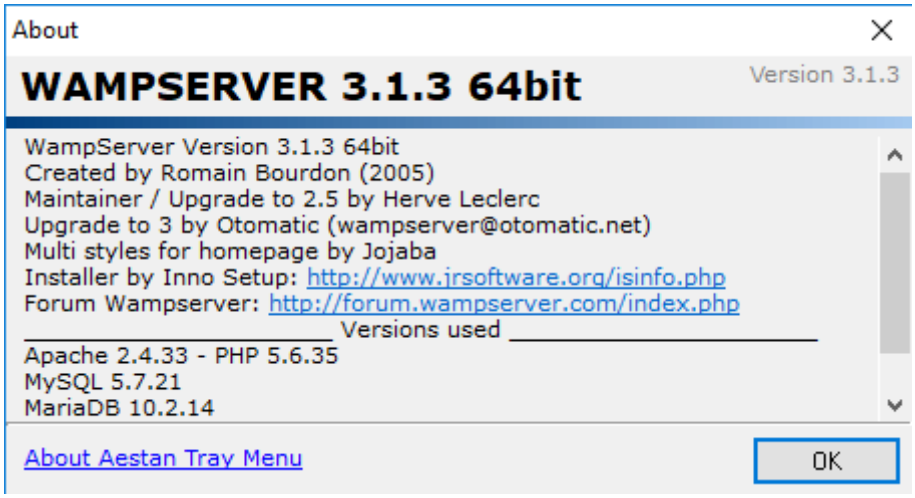


Figura 44: Informació de la versió instal·lada en el sistema de Wamp

- Sublime Text: versió 3.1.1 build 3176.

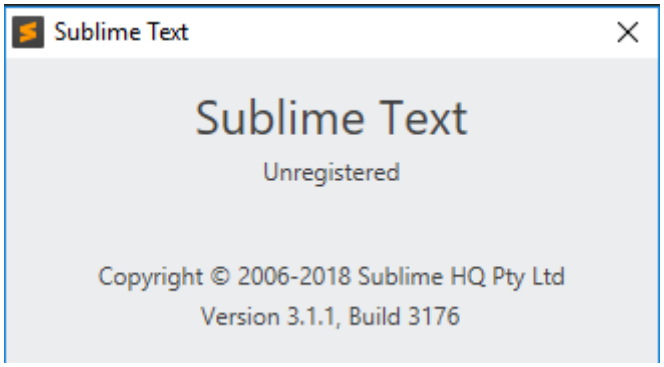


Figura 45: Informació de la versió instal·lada en el sistema de Sublime Text

- PHPMailer: versió 6.0.6.

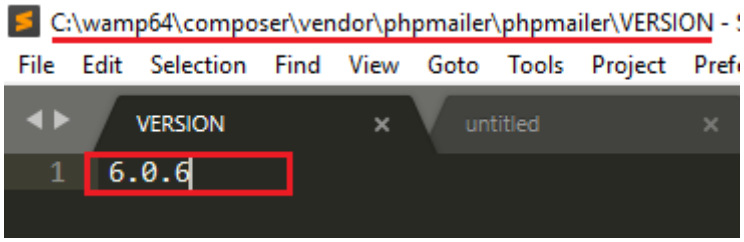


Figura 46: Informació de la versió instal·lada en el sistema de PHPMailer

5.3. PayPal

Actualment, existeixen molts mètodes de pagament per realitzar compres en Internet. Una de les opcions més populars, a escala mundial, és PayPal[7] i, per tant, és l'opció principal escollida per integrar en el sistema i poder realitzar les vendes dels productes. Però, què és PayPal?

PayPal és una empresa del sector del comerç electrònic, el sistema del qual permet als seus usuaris realitzar pagaments i transferències a través d'Internet sense compartir la informació financera amb el destinatari, amb l'únic requeriment que aquests disposin de correu electrònic.

És un sistema ràpid i segur per enviar i rebre diners. PayPal pot ser utilitzat tant com per pagar compres realitzades com per cobrar les vendes realitzades per Internet o realitzar diferents transferències entre particulars.

Pel que fa a l'àmbit de seguretat, PayPal ofereix un mètode segur per realitzar pagaments i transferències de diners perquè utilitza tecnologia d'encriptació SSL per protegir tota la informació confidencial. El destinatari mai rep dades financeres, com per exemple informació de la targeta, o compte bancari, o informació personal. També, ofereix programes de protecció, on el comprador pot demanar la devolució total o parcial dels seus diners segons les circumstàncies.

En apartats posteriors, es mostrarà com es pot realitzar una compra mitjançant aquest mètode de pagament des de la botiga de *Gaming Keys*. No obstant això, el procés serà bastant similar a una operació real en qualsevol botiga en línia.

5.3.1. Configuració de comptes

En aquest subapartat, s'explicarà amb detalls la informació necessària per crear i configurar els comptes d'usuaris de prova, tant pel client com l'empresa.

Paypal té un entorn de desenvolupament anomenat Sandbox, on s'executen les transaccions de prova entre dos tipus de comptes existents en aquest entorn. Aquests dos tipus de comptes són:

- Personal: representa al client o el compte que enviarà els diners en la transacció.

- Business: representa a la botiga o el compte que rebrà els diners de la transacció.

Cal esmentar que aquests comptes només funcionen localment i no es pot realitzar cap transacció real. Per tant, *Gaming Keys* només podrà rebre diners en un entorn de desenvolupament, però no en un entorn real.

Arribats a aquest punt, el primer pas és iniciar sessió en la pàgina següent:

https://developer.paypal.com/

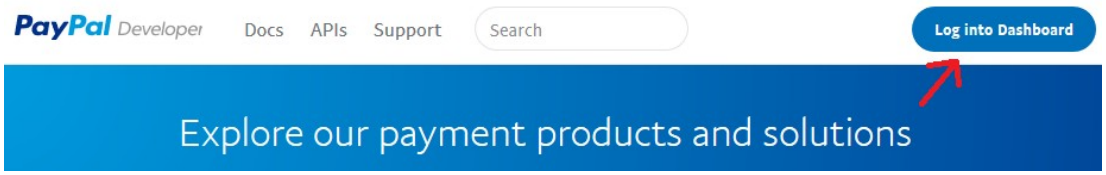


Figura 47: Pàgina principal de desenvolupadors de PayPal

Cal esmentar que es pot accedir a través d'un compte normal de PayPal. En cas de no tenir, cal crear un i, posteriorment, iniciar sessió. PayPal, a través d'un compte d'usuari normal i corrent, ofereix gratuïtament l'entorn de desenvolupament on l'usuari pot realitzar proves en àmbit local. Un cop s'ha accedit al sistema, cal accedir a la part de comptes de Sandbox.

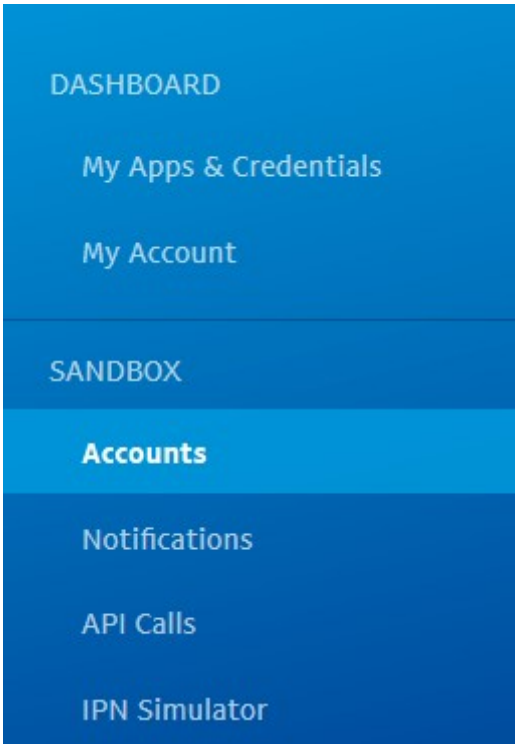


Figura 48: Menú del desenvolupador de PayPal

Un cop allà, es pot visualitzar com PayPal té creat per defecte dos comptes, un de tipus personal i l'altre de tipus business.

Figura 49: Comptes Sandbox creats per defecte

En aquest cas, es crearan dos nous comptes específics per la botiga, de manera que cal fer clic al botó de creació de comptes ubicat a la dreta. En el formulari corresponent, només cal omplir les dades bàsiques necessàries per cadascun i seleccionar el tipus de compte. Les altres configuracions es deixen per defecte.

Un cop creades apareixeran en el llistat de comptes de l'entorn de Sandbox:

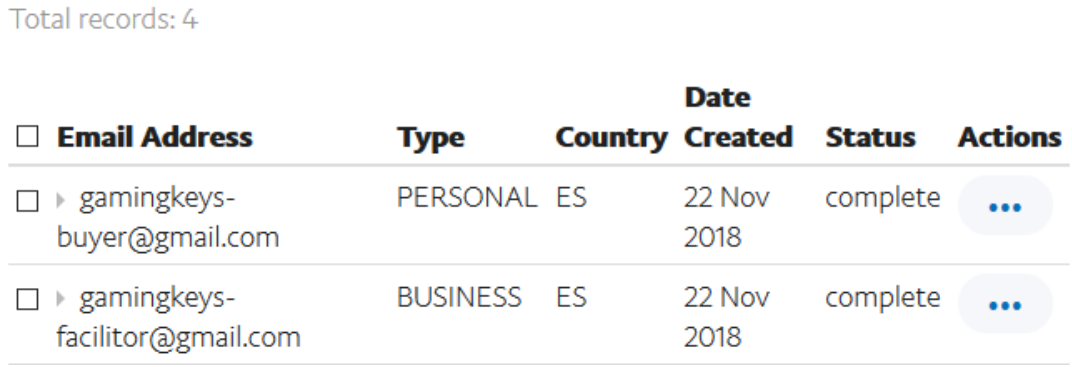


Figura 50: Comptes creats per la botiga

Seguidament, abans de començar a realitzar pagaments en la botiga, cal realitzar les últimes configuracions per completar la integració amb el sistema. Concretament, cal configurar el compte business per fer que PayPal retorni el resultat i la informació de les operacions de manera automàtica a les adreces que s'indiquin. Per tal, cal iniciar sessió amb el nou compte en la pàgina de l'entorn:

https://www.sandbox.paypal.com/es/home

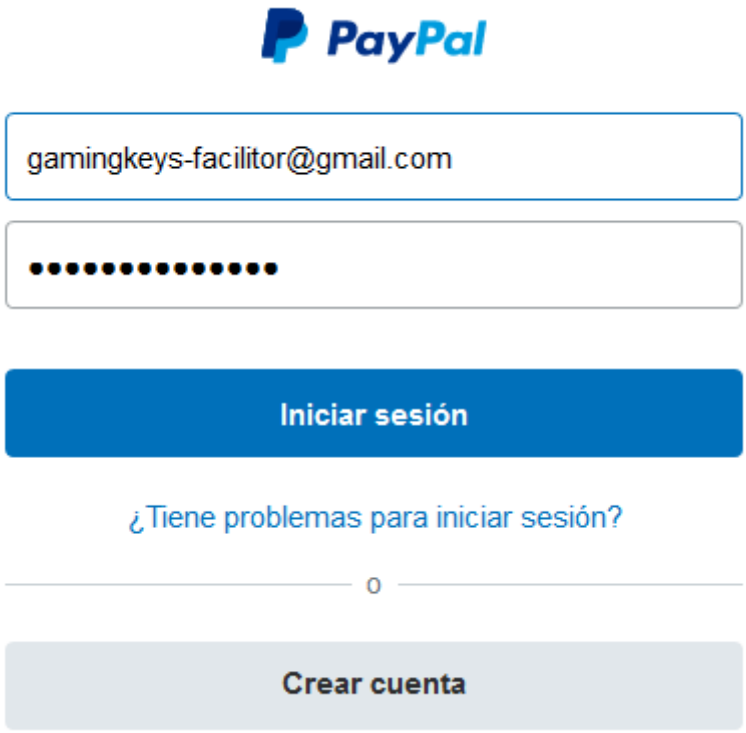


Figura 51: Inici de sessió en Sandbox

Un cop s'ha iniciat sessió, en la pàgina principal es podran veure totes les transaccions rebudes pels clients. El següent pas és fer clic en l'icona de configuració del menú, i seguidament fer clic a les opcions de vendes i actualitzar les preferències web:

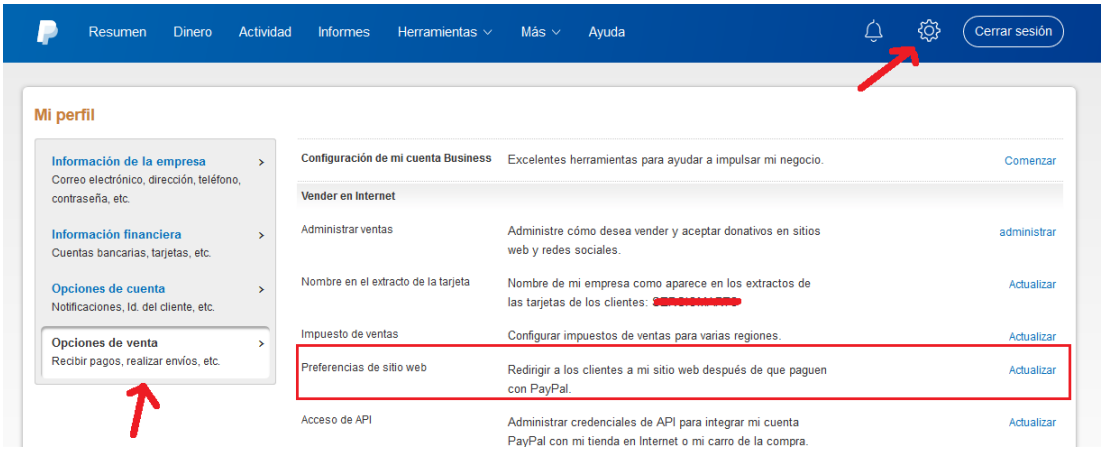


Figura 52: Opcions de configuració del compte business

Un cop s'ha accedit a les preferències, cal activar el retorn automàtic i especificar

l'enllaç on l'usuari serà redirigit, que en aquest cas serà http://localhost/gk_shop/payment_success.php:

Figura 53: Retorn automàtic de les transaccions

Finalment, també cal activar la transferència de dades de pagament per rebre les notificacions i la informació dels pagaments realitzats. Aquestes dades són les que permeten al sistema analitzar el resultat de la transacció i poder prendre les decisions necessàries per completar el procés de compra.

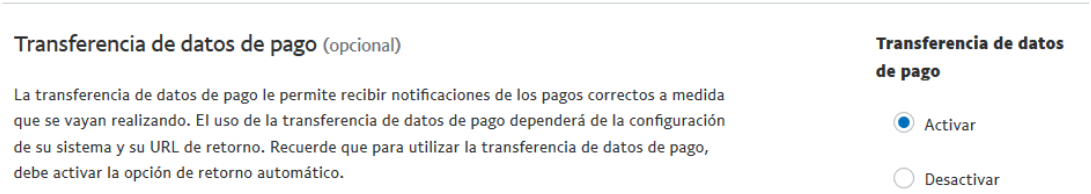


Figura 54: Transferència de dades de pagament

Un cop realitzades les configuracions descrites, els comptes creats ja estan habilitats per realitzar les transaccions dins de la botiga. El procediment de pagament i la comprovació dels moviments de diners es podran veure en apartats posteriors.

5.4. W3.CSS

W3.CSS[8] és un *framework* modern de CSS que permet l'acceleració i la simplificació del desenvolupament web a causa de la facilitat d'aprenentatge i del seu ús en comparació a altres *frameworks*. Per defecte, suporta el disseny responsive, i el

seu ús és totalment gratuït i no requereix cap mena de llicència.

En el següent subapartat, s'explicarà amb detalls la informació necessària per instal·lar i configurar la llibreria CSS.

5.4.1. Instal·lació i configuració

Per poder utilitzar la llibreria descrita anteriorment, hi ha dues maneres d'incorporar el seu contingut al projecte. La primera opció és importar-ho directament en el codi HTML mitjançant l'enllaç corresponent. Per exemple:

```
<link rel="stylesheet" href="https://www.w3schools.com/w3css/4/w3.css">
```

Figura 55: Exemple d'integració amb la llibreria w3.css

La segona opció, realitzada en aquest projecte, és realitzant la descàrrega del fitxer CSS que ofereix la pàgina web del *framework* i incloure l'adreça del fitxer corresponent en el codi HTML que sigui necessari el seu ús:

https://www.w3schools.com/w3css/

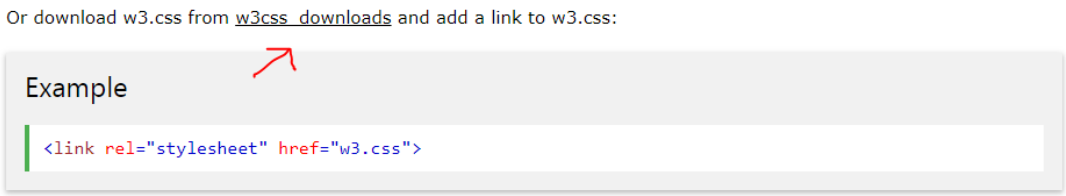


Figura 56: Descàrrega de la llibreria w3.css

Un cop realitzats els passos anteriors ja es pot fer ús de la llibreria i introduir tots els estils necessaris recollits en aquesta.

5.5. Font Awesome

Font Awesome [9] és un *framework* que conté un conjunt d'icones i logos moderns dissenyats amb detall disponibles per incloure en qualsevol part de la pàgina web, degut que suporta l'especificació de la seva mida, adaptant-se així a qualsevol component. Com l'anterior llibreria, el seu ús és totalment gratuït i no requereix cap mena de llicència.

En el següent subapartat, s'explicarà amb detalls la informació necessària per instal·lar i configurar aquest conjunt d'icones.

5.5.1. Instal·lació i configuració

Per poder utilitzar la llibreria descrita anteriorment, com en el cas anterior, existeixen les dues maneres d'incorporar el seu contingut al projecte. Per tant, es realitza la descàrrega del fitxer CSS que ofereix la pàgina web del *framework*:

https://fontawesome.com/start

Figura 57: Descàrrega de la llibreria Font Awesome

I incloure els fitxers necessaris en el codi HTML de la mateixa manera que el cas anterior, integrant així la llibreria i poder fer ús de les seves icones.

5.6. Implementació funcional

En aquest subapartat, s'explicaran les parts més rellevants de tot el desenvolupament que comporten els casos funcionals de *Gaming Keys*, tant els diferents casos d'ús que pot realitzar l'usuari com les operacions més rellevants que es realitzen en el servidor.

Cal esmentar, que tots els aspectes relacionats amb la seguretat s'obviaran, degut que aquests detalls s'explicaran amb més detall en l'apartat d'implementacions de seguretat. A més a més, per veure un funcionament complet de l'aplicació en qualsevol àmbit, es podrà visualitzar detalladament en els apartats de proves.

5.6.1. Composició de les pàgines

Com bé s'ha esmentat en el punt 4.1 d'aquest document, qualsevol de les vistes que conté Gaming Keys està formada per quatre components principals: la capçalera, el menú de navegació, el contingut i el peu de pàgina.

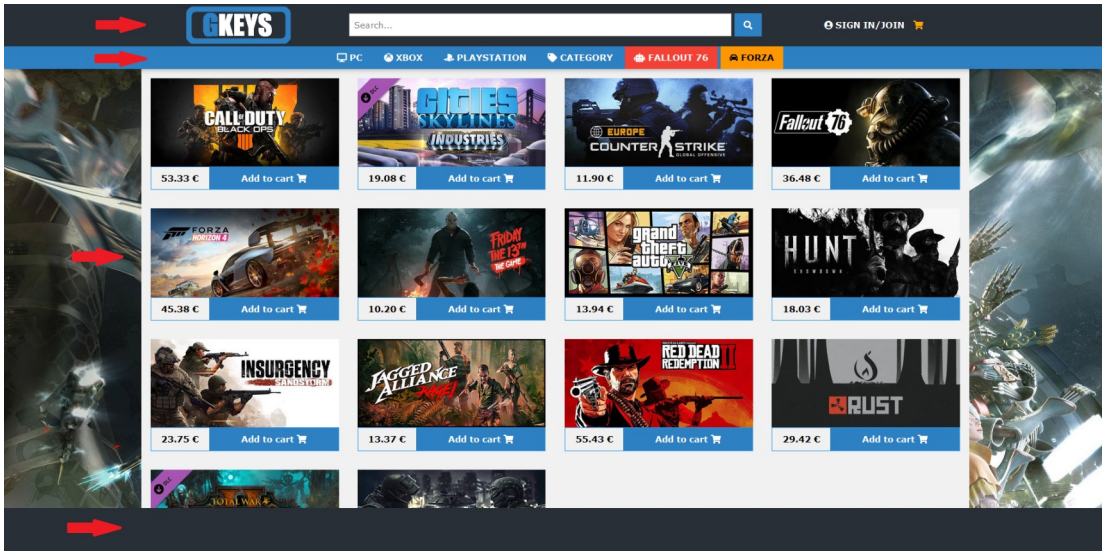


Figura 58: Components de la pàgina principal

Capçalera

És el primer element ubicat en la part de dalt de la pàgina i consta de diferents funcionalitats que es podran veure detalladament en apartats posteriors.

Menú de navegació

Es tracta del següent component que es visualitza seguidament de la capçalera Aquest component permet la visualització dels productes del catàleg segons l'opció seleccionada. El detall d'aquesta funcionalitat també es podrà veure detalladament en els apartats corresponents.

Contingut

Aquest component és el que conté tota la informació corresponent a la vista. Per tant, aquest pot varia segons les necessitats de l'usuari.

Peu de pàgina

Aquest component és el que es situa al final de tot el contingut i no disposa de cap mena de funcionalitat, només es tracta d'un espai per emplenar d'informació per l'usuari, com podria ser mètodes de pagament disponibles, imatges de certificats obtinguts o informació de contacte.

5.6.2. Retorn a la pàgina principal

Una de les funcionalitats principals de la capçalera és el retorn a la pàgina principal de la botiga, on l'usuari pot fer clic a l'enllaç que es presenta en forma d'imatge amb el logo de la botiga.

Figura 59: Logo de la botiga

Un cop s'ha fet clic en la imatge, el sistema redirigeix cap a la pàgina principal que conté el catàleg de productes, com es pot veure en el següent apartat.

```
<a class="c-link" href="index.php" title="GamingKeys"><img src="img/logo1.png" alt="
GamingKeys"></a>
```

Figura 60: Enllaç de retorn a la pàgina principal

5.6.3. Catàleg

La pantalla principal que es presenta en la botiga conté el catàleg de productes existents:

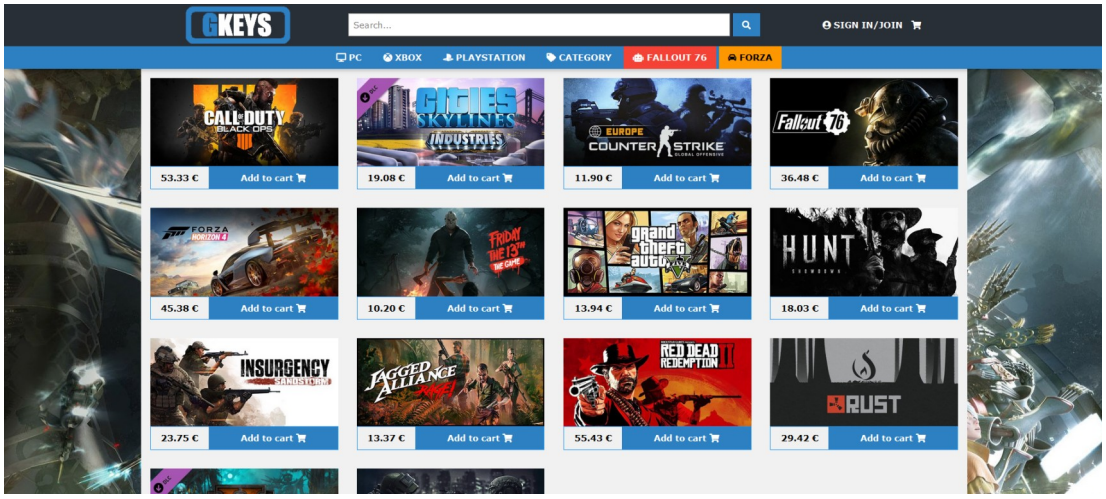


Figura 61: Catàleg de productes

En cas de no existir cap mena d'error previ, aquests productes són obtinguts de la base de dades i, seguidament, es mostra la corresponent informació de cadascun en un llistat dinàmic.

```
if($result == "" && $error == ""){
    //GET ALL THE PRODUCTS FROM DB
    $sql = "SELECT * FROM Product";
    $result = mysqli_query($conn,$sql);
}
```

Figura 62: Sentència de cerca de productes en la base de dades

Com s'ha pogut observar en la figura anterior, no s'especifica cap mena de criteri en la cerca dels productes de manera que es mostren tots en el mateix ordre en el qual s'obtenen de la base de dades. En el següent apartat, es pot veure com es realitzen aquestes consultes amb diferents criteris segons el tipus de cerca que realitzi l'usuari.

5.6.4. Cercar productes

L'usuari pot realitzar dos tipus de cerques de productes: cerques per filtres o cerques per nom del producte.

Cerca de productes per filtres

Aquesta opció está disponible en la barra de menú de navegació, on l'usuari pot fer clic al filtre desitjat i visualitzar els productes corresponents. Els criteris que es poden utilitzar són:

- Plataforma
  - PC
    - Steam
    - Battlenet
    - Origin
    - Uplay
  - Xbox
  - PlayStation

- Categoria
  - Action
  - Adventura
  - MMO
  - RPG
  - Simulation
  - Gore
  - Racing
  - Strategy
- Productes destacats
  - Fallout 76
  - Forza

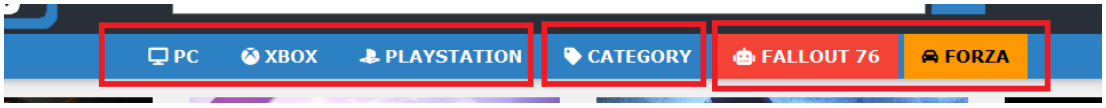


Figura 63: Organització de filtres de cerca

Un cop l'usuari ja ha decidit quin criteri utilitzar, només cal fer clic en l'enllaç corresponent. La pàgina web realitzarà una petició GET a la pàgina principal amb el valor corresponent segons el criteri seleccionat.

```
<div class="c-center-horizontal c-content-space">
    <div class="w3-dropdown-hover">
        <a href="#" class="w3-bar-item w3-btn c-blue c-hover-extra-dark-gray"><b><i class="
        fa fa-desktop"></i> PC</b></a>
        <div class="w3-dropdown-content w3-bar-block w3-card-4 c-blue">
            <a href="index.php?plat=Steam" class="w3-bar-item w3-btn c-hover-extra-dark-gray"
            >Steam</a>
            <a href="index.php?plat=Battlenet" class="w3-bar-item w3-btn
            c-hover-extra-dark-gray">Battlenet</a>
            <a href="index.php?plat=Origin" class="w3-bar-item w3-btn c-hover-extra-dark-gray
            ">Origin</a>
            <a href="index.php?plat=Uplay" class="w3-bar-item w3-btn c-hover-extra-dark-gray"
            >Uplay</a>
        </div>
    </div>
    <a href="index.php?plat=Xbox" class="w3-bar-item w3-btn c-hover-extra-dark-gray"><b><i
    class="fab fa-xbox"></i> XBOX</b></a>
    <a href="index.php?plat=PlayStation" class="w3-bar-item w3-btn c-hover-extra-dark-gray"><
    b><i class="fab fa-playstation"></i> PLAYSTATION</b></a>
    <div class="w3-dropdown-hover">
        <a href="#" class="w3-bar-item w3-btn c-hover-extra-dark-gray"><b><i class="fa fa-tag
        "></i> CATEGORY</b></a>
        <div class="w3-dropdown-content w3-bar-block w3-card-4 c-blue">
            <a href="index.php?cat=Action" class="w3-bar-item w3-btn c-hover-extra-dark-gray"
            >Action</a>
            <a href="index.php?cat=Adventure" class="w3-bar-item w3-btn
            c-hover-extra-dark-gray">Adventure</a>
            <a href="index.php?cat=MMO" class="w3-bar-item w3-btn c-hover-extra-dark-gray">
            MMO</a>
            <a href="index.php?cat=RPG" class="w3-bar-item w3-btn c-hover-extra-dark-gray">
            RPG</a>
            <a href="index.php?cat=Simulation" class="w3-bar-item w3-btn
            c-hover-extra-dark-gray">Simulation</a>
            <a href="index.php?cat=Gore" class="w3-bar-item w3-btn c-hover-extra-dark-gray">
            Gore</a>
            <a href="index.php?cat=Racing" class="w3-bar-item w3-btn c-hover-extra-dark-gray"
            >Racing</a>
            <a href="index.php?cat=Strategy" class="w3-bar-item w3-btn
            c-hover-extra-dark-gray">Strategy</a>
        </div>
    </div>
    <a href="index.php?id=4" class="w3-bar-item w3-btn c-hover-extra-dark-gray w3-red"><b><i
    class="fa fa-robot"></i> FALLOUT 76</b></a>
    <a href="index.php?id=5" class="w3-bar-item w3-btn c-hover-extra-dark-gray w3-orange"><b>
    <i class="fa fa-car-alt"></i> FORZA</b></a>
</div>
```

Figura 64: Enllaços del menú

Seguidament el sistema realitza una validació dels valors rebuts en el servidor i, només en cas de ser vàlids, realitza la corresponent consulta a la base de dades segons el criteri utilitzat i mostra els resultats obtinguts.

```php
//VALIDATE PRODUCT FILTER
if(isset($_GET["plat"])){
    $error = Validation::validateProductFilter($_GET['plat']);
    if($error == ""){
        //SEARCH PRODUCT BY FILTER
        $sql = "SELECT * FROM Product WHERE platform='".$_GET['plat']."'";
        $result = mysqli_query($conn,$sql);
    }
}else if(isset($_GET["cat"])){
    $error = Validation::validateProductFilter($_GET['cat']);
    if($error == ""){
        //SEARCH PRODUCT BY FILTER
        $sql = "SELECT * FROM Product WHERE genre='".$_GET['cat']."'";
        $result = mysqli_query($conn,$sql);
    }
}else if(isset($_GET["id"])){
    $error = Validation::validateProductIdentifier($_GET['id']);
    if($error == ""){
        //SEARCH PRODUCT BY FILTER
        $sql = "SELECT * FROM Product WHERE id='".$_GET['id']."'";
        $result = mysqli_query($conn,$sql);
    }
```

Figura 65: Consultes de productes segons el criteri

Per exemple, en cas de seleccionar Xbox, els resultats que es mostren són els següents:

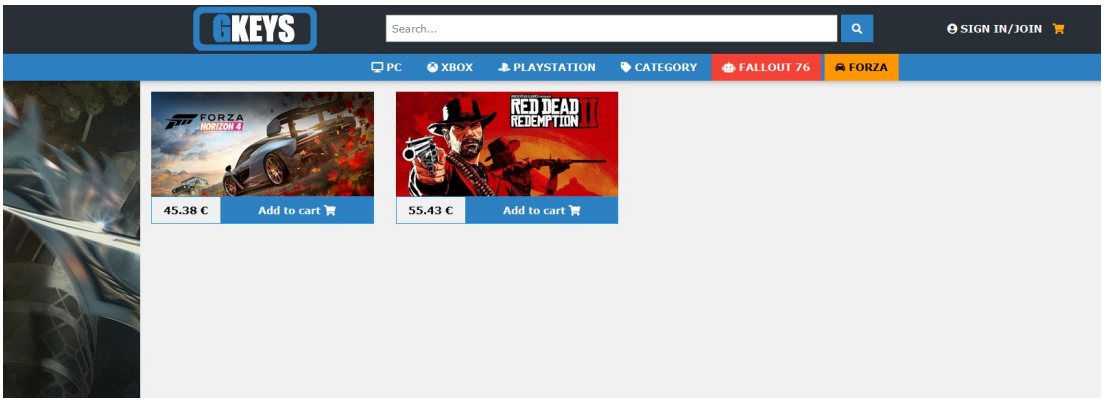


Figura 66: Resultat de la cerca de productes de la plataforma Xbox

En cas d'error de validació, de no trobar resultats o qualsevol mena d'error, el sistema mostra un missatge d'error avisant a l'usuari que no existeixen productes amb la cerca realitzada, juntament amb un enllaç per tornar al catàleg.

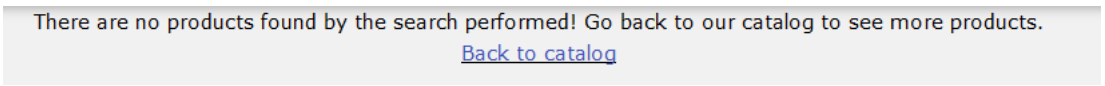


Figura 67: Missatge d'error en la cerca de productes

Cerca de productes per nom

En aquest cas, l'usuari pot realitzar una cerca del producte desitjat introduint el nom mitjançant el camp de text disponible en la capçalera i fent clic en el botó de cerca. La pàgina web realitzarà una petició POST a la pàgina principal amb el valor corresponent.

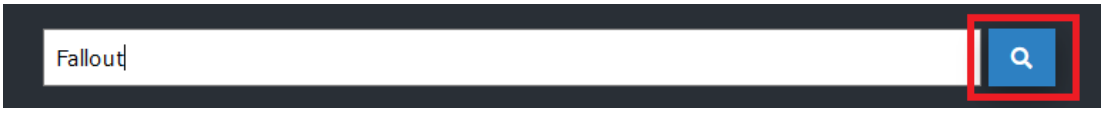


Figura 68: Cerca de producte per nom

Seguidament el sistema realitza una validació del valor rebut en el servidor i, només en cas de ser vàlids, realitza la corresponent consulta a la base de dades en la qual es cerca tots els productes el nom dels quals continguin el text introduït.

```php
if($_SERVER["REQUEST_METHOD"] == "POST") {
    //IF SEARCHING BY INPUT, VALIDATE PRODUCT NAME FILTER
    if(isset($_POST["header-search-input"])){
        $error = Validation::validateProductNameFilter($_POST["header-search-input"]
            );
        if($error == ""){
            //SEARCH PRODUCT BY FILTER
            $sql = "SELECT * FROM Product WHERE name LIKE '%" .$_POST["
                header-search-input"]. "%'";
            $result = mysqli_query($conn,$sql);
        }
    }
```

Figura 69: Consulta de productes per nom

Finalment, exactament com en el cas anterior, es mostren els productes que coincideixen amb la cerca i en cas contrari, el missatge d'error corresponent.

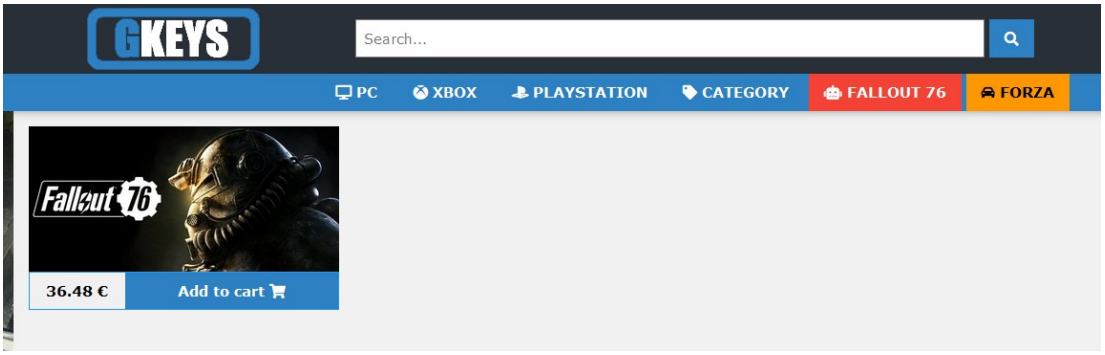


Figura 70: Resultats de la cerca per nom realitzada

5.6.5. Visualitzar producte

En qualsevol punt del catàleg, en cas de desitjar veure més informació sobre un producte, l'usuari pot fer clic en la imatge corresponent per veure tota la informació disponible i acabar de decidir si l'interessa el producte.

Per exemple, fent clic en el producte de la figura anterior, la pàgina web realitza una petició GET a la pàgina de detall del producte passant com a valor l'identificador d'aquest.

```
//VALIDATE PRODUCT IDENTIFIER
if(isset($_GET['id'])){
    $productID = $_GET['id'];
    $error = Validation::validateProductIdentifier($productID);
    //IF EVERYTHING OK, SEARCH THE PRODUCT
    if($error == ""){
        //GET THE PRODUCT FROM DB
        $sql = "SELECT * FROM Product WHERE id=".$productID." limit 1";
        $result = mysqli_query($conn,$sql);
        $row = mysqli_fetch_array($result,MYSQLI_ASSOC);
        $count = mysqli_num_rows($result);
        if($count != 1){
            $error = "The product does not exists in our catalog! Go back to our
                catalog to see more products.";
        }
    }
```

Figura 71: Procediment per la cerca de productes per identificador

Seguidament el sistema realitza una validació de l'identificador rebut en el servidor i, només en cas de ser vàlid, realitza la consulta a la base de dades en la qual es cerca el producte amb l'identificador corresponent. Finalment, es mostra tota la informació disponible del producte corresponent.

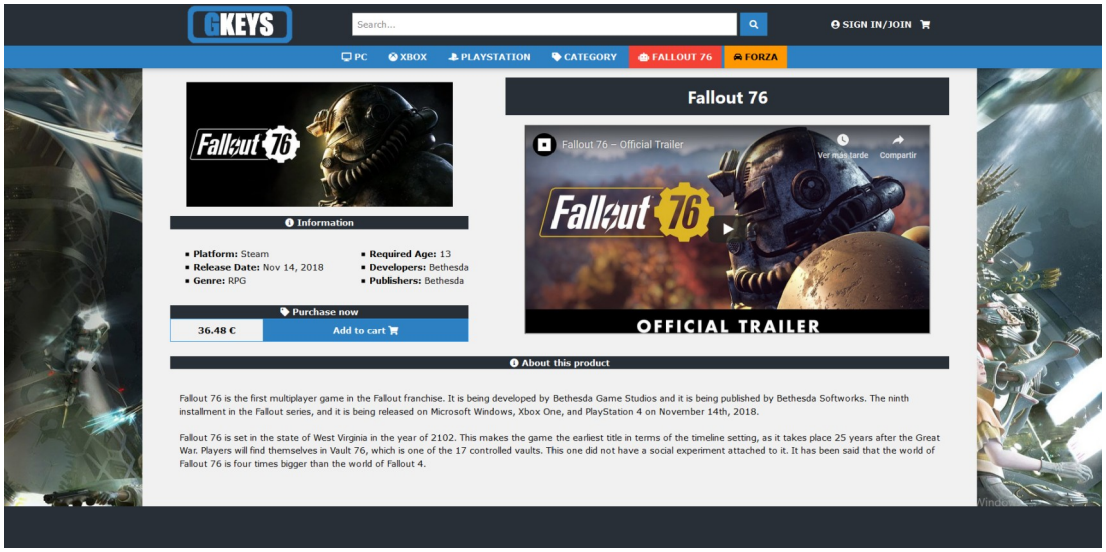


Figura 72: Detall del producte Fallout 76

En cas d'error de validació, de no trobar resultats o qualsevol mena d'error, el sistema mostra un missatge d'error avisant a l'usuari que no existeix el producte en el catàleg, juntament amb un enllaç per tornar a aquest.

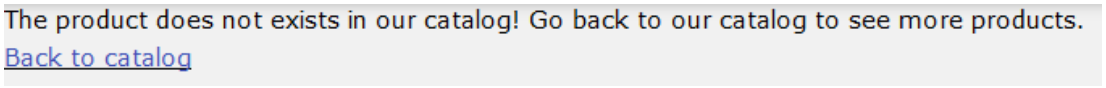


Figura 73: Missatge d'error en la visualització d'informació d'un producte

5.6.6. Visualitzar el carret de la compra

Aquesta funcionalitat està disponible en la icona del carret de la compra situada a la part dreta de la capçalera. Per veure el seu contingut, només cal passar el cursor per sobre de l'icona i es farà visible una nova capa amb el contingut d'aquesta.

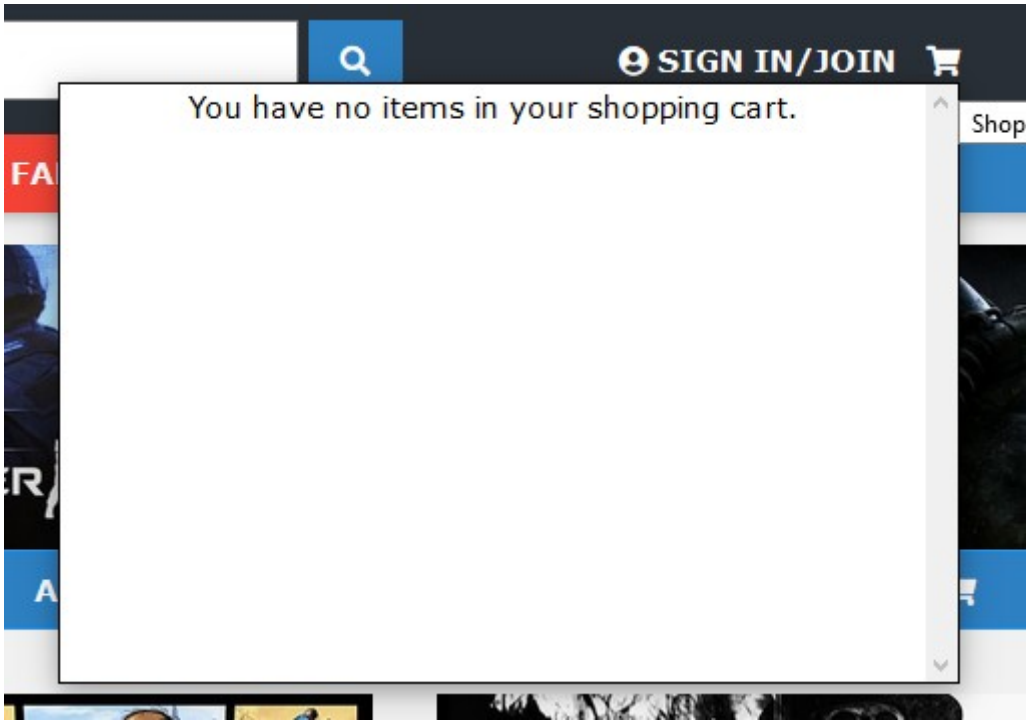


Figura 74: Carret de la compra buit

En cas de no tenir productes desats en el carret, la icona d'aquest és de color blanc i en el seu contingut es mostra un missatge advertint que no existeixen productes desats en el carret.

En cas contrari, la icona del carret és de color taronja i els productes es mostren en un llistat, juntament amb la quantitat, el preu per unitat, un botó per eliminar-lo i, al final de tot, el preu total del carret juntament amb un botó per procedir amb el pagament.

Figura 75: Carret de la compra amb productes

Aquest llistat de productes es troben desats en sessió, degut que prèviament han sigut afegits per l'usuari, com es podrà veure en el següent apartat. Per tant, com que el sistema ja té carregada la informació en sessió, no cal consultar-la a la base de dades i només cal mostrar-la.

```php
if(!isset($_SESSION["shopping-cart"]) || count($_SESSION["shopping-cart"]
    ) == 0){
    echo "<span>You have no items in your shopping cart.</span>";
}else{
    $products = $_SESSION["shopping-cart"];
    $count = count($products);
    echo '<div id="header-cart-content-products">';
    for($i=0; $i<$count; $i++){
        echo '
        <div class="w3-row">
            <div class="w3-margin">
                <div>
                    <div class="w3-cell-row c-center-horizontal-text">
                        <div id="cart-product-image" class="w3-cell
                            w3-cell-middle">
                            <img src="img/product/'.$products[$i]["img"].
                                '" class="c-product-image">
                        </div>
                        <div class="w3-cell w3-cell-middle">
                            <b>'.$products[$i]["quantity"].' x '.$
                                products[$i]["price"].' €</b>
                        </div>
                        <div class="w3-cell w3-cell-middle
                            c-center-horizontal-text">
                            <a class="w3-btn w3-block w3-hover-red
                                c-blue" href="'.htmlspecialchars($_SERVER
                                ["PHP_SELF"]).'?remove='.$products[$i]["
                                id"].'" title="Remove product">
                                <b><i class="fas fa-trash"></i></b>
                            </a>
                        </div>
                    </div>
                </div>
            </div>
        </div>';
    }
```

Figura 76: Procediment que mostra el contingut del carret de la compra

5.6.7. Afegir producte al carret

Tant en el catàleg com en la visualització de la informació del producte, l'usuari té disponible el botó per afegir el producte corresponent al carret de la compra, com s'ha pogut veure en les figures del catàleg o en la figura 72 de la pàgina d'informació del producte. Per tant, un cop l'usuari fa clic en aquest botó la pàgina web realitza una petició GET on s'envia l'identificador del producte que es vol desar en el carret de la compra.

```
}else if(isset($_GET["add"])){
    $error = Validation::validateProductIdentifier($_GET['add']);
    if($error == ""){
        //SEARCH PRODUCT BY ID
        $sql = "SELECT * FROM Product WHERE id='".$_GET['add']."'";
        $result = mysqli_query($conn,$sql);
        $row = mysqli_fetch_array($result,MYSQLI_ASSOC);
        $count = mysqli_num_rows($result);
        //IF PRODUCT EXISTS, ADD PRODUCT TO SESSION CART
        if($count == 1){
            $product = array(
                "id" => $row["id"],
                "name" => $row["name"],
                "img" => $row["cover"],
                "price" => $row["price"],
                "quantity" => 1,
            );
            ShoppingCart::add($product);
            header("Location: index.php");
        }
    }
```

Figura 77: Procediment per afegir un producte al carret

Seguidament el sistema realitza una validació de l'identificador rebut en el servidor i, només en cas de ser vàlid, realitza la consulta a la base de dades en la qual es cerca el producte amb l'identificador corresponent. Finalment, s'afegeix el producte al carret de la següent manera:

```php
public static function add($product){

    //IF EXISTS SHOPPING CART PREVIOUSLY, ADD THE PRODUCT
    if(isset($_SESSION["shopping-cart"])){
        $products = $_SESSION["shopping-cart"];
        //IF THE PRODUCT EXISTS, ADD QUANTITY
        $count = count($products);
        $found = false;
        for($i=0; $i<$count; $i++){
            if($products[$i]["id"] == $product["id"]){
                $products[$i]["quantity"] += 1;
                $_SESSION["total-price"] += $products[$i]["price"];
                $_SESSION["shopping-cart"] = $products;
                $found = true;
            }
        }
        // IF IT IS A NEW PRODUCT IN THE CART, ADD IT
        if($found == false){
            $_SESSION["shopping-cart"][] = $product;
            $_SESSION["total-price"] += $product["price"];
        }
    }else{
        $_SESSION["shopping-cart"] = array();
        $_SESSION["shopping-cart"][] = $product;
        $_SESSION["total-price"] = $product["price"];
    }
}
```

Figura 78: Procediment per afegir el producte a la sessió

Resumidament, es comprova que el producte existeixi prèviament en el carret. De ser així, es suma una unitat del producte corresponent i, seguidament, s'actualitza el preu total de la compra. En cas contrari, afegeix la informació del nou producte al carret. Un exemple del resultat d'aquestes operacions és el contingut que mostra la figura 75 de l'apartat anterior.

Finalment, en cas d'error de validació, de no trobar el producte en la base de dades o qualsevol mena d'error, el sistema no realitza cap més operació, deixant el carret en l'estat en què es trobava.

5.6.7. Eliminar producte

Tant en el catàleg com en la visualització de la informació del producte, sempre que el client hagi afegit prèviament productes al carret de compra, apareix un botó corresponent a cada producte afegit dins del carret, que fent clic permet a l'usuari eliminar-lo:

Figura 79: Botó d'eliminar producte del carret

Un cop s'ha fet clic, la pàgina web realitza una petició GET a si mateixa on s'envia l'identificador del producte que s'ha d'eliminar del carret. Seguidament es realitzen les operacions següents:

```
}else if(isset($_GET["remove"])){
    $error = Validation::validateProductIdentifier($_GET['remove']);
    if($error == ""){
        ShoppingCart::remove($_GET['remove']);
        header("Location: index.php");
    }
}
```

Figura 80: Procediment per eliminar un producte del carret

El sistema realitza una validació de l'identificador rebut en el servidor i, només en cas de ser vàlid, realitza procés d'eliminar el producte de la sessió.

```php
public static function remove($id){

    if(isset($_SESSION["shopping-cart"])){
        //SEARCH PRODUCT IN SESSION BY ID
        $products = $_SESSION["shopping-cart"];
        $count = count($products);
        for($i=0; $i<$count; $i++){
            //IF PRODUCT EXISTS, REMOVE IT
            if($products[$i]["id"] == $id){
                $_SESSION["total-price"] -= $products[$i]["price"];
                if($products[$i]["quantity"] > 1){
                    $products[$i]["quantity"] -= 1;
                }else{
                    unset($products[$i]);
                    array_splice($products,$i,0);
                }
                $_SESSION["shopping-cart"] = $products;
            }
        }
    }
}
```

Figura 81: Procediment per eliminar el producte de la sessió

Aquest mètode és totalment invers al mètode de l'apartat anterior. Resumidament, es comprova que el producte existeixi prèviament en el carret. De ser així, si la quantitat d'unitats és superior a 1, es resta una unitat del producte corresponent. En cas contrari, s'elimina la informació del producte de la sessió i es reorganitza el llistat de productes d'aquesta. En cas d'eliminar tots els productes del carret, el resultat d'aquestes operacions és el contingut que mostra la figura 74.

Finalment, com en el cas anterior, el sistema no realitza cap més operació, deixant el carret en l'estat en què es trobava.

5.6.8. Resum de la compra

Un cop l'usuari hagi afegit productes al carret, aquest pot començar el procés per realitzar la compra. El primer pas del procediment de compra és visualitzar el resum d'aquesta. Per tant, l'usuari ha de fer clic al botó 'Purchase now' situat dins del carret de la compra, i només visible en cas d'haver-hi productes en aquest.

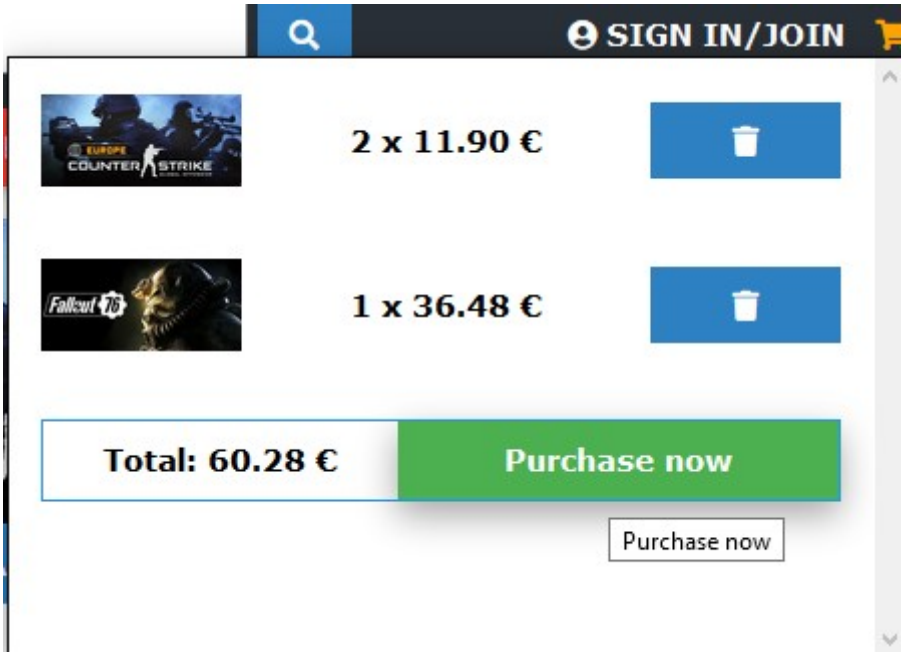


Figura 82: Botó per procedir a visualitzar el resum de compra

I seguidament, l'enllaç redirigeix cap a la pàgina web que mostra la informació del carret, desada en la sessió, en un formulari juntament amb la descripció del següent pas en el procediment de compra i el botó que possibilita a l'usuari a avançar en aquest.

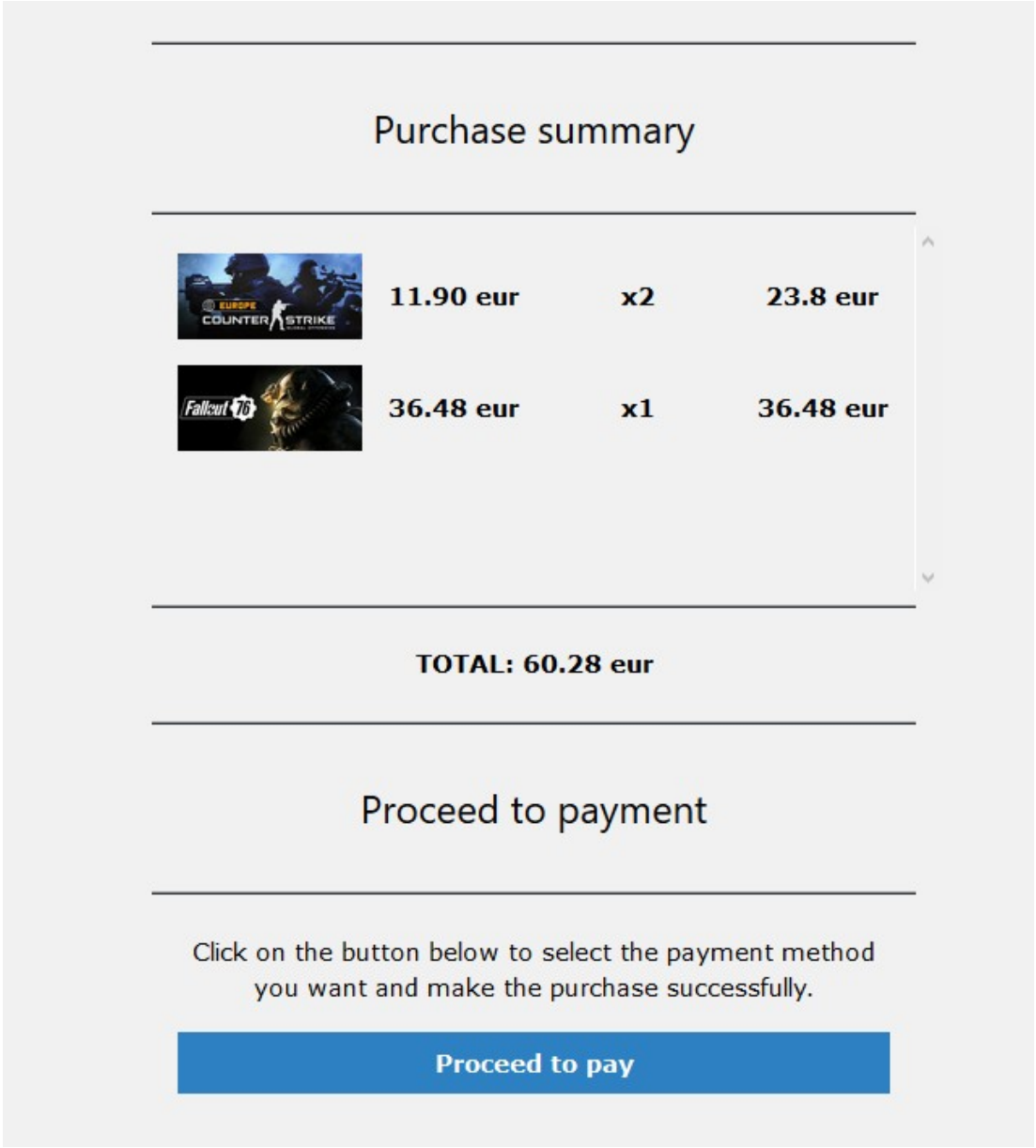


Figura 83: Resum de compra

Cal esmentar que, en cas que el client encara no hagi iniciat sessió en el sistema o no tingui compte d'usuari, al fer clic en el botó per visualitzar el següent pas del procediment, el sistema determinarà que el client no té accés al següent pas i mostrarà el formulari d'inici de sessió per identificar a l'usuari.

5.6.9. Crear compte d'usuari

Qualsevol client, si no té la sessió iniciada, pot visualitzar, emplenar les dades requerides i enviar el formulari de creació de comptes d'usuari.

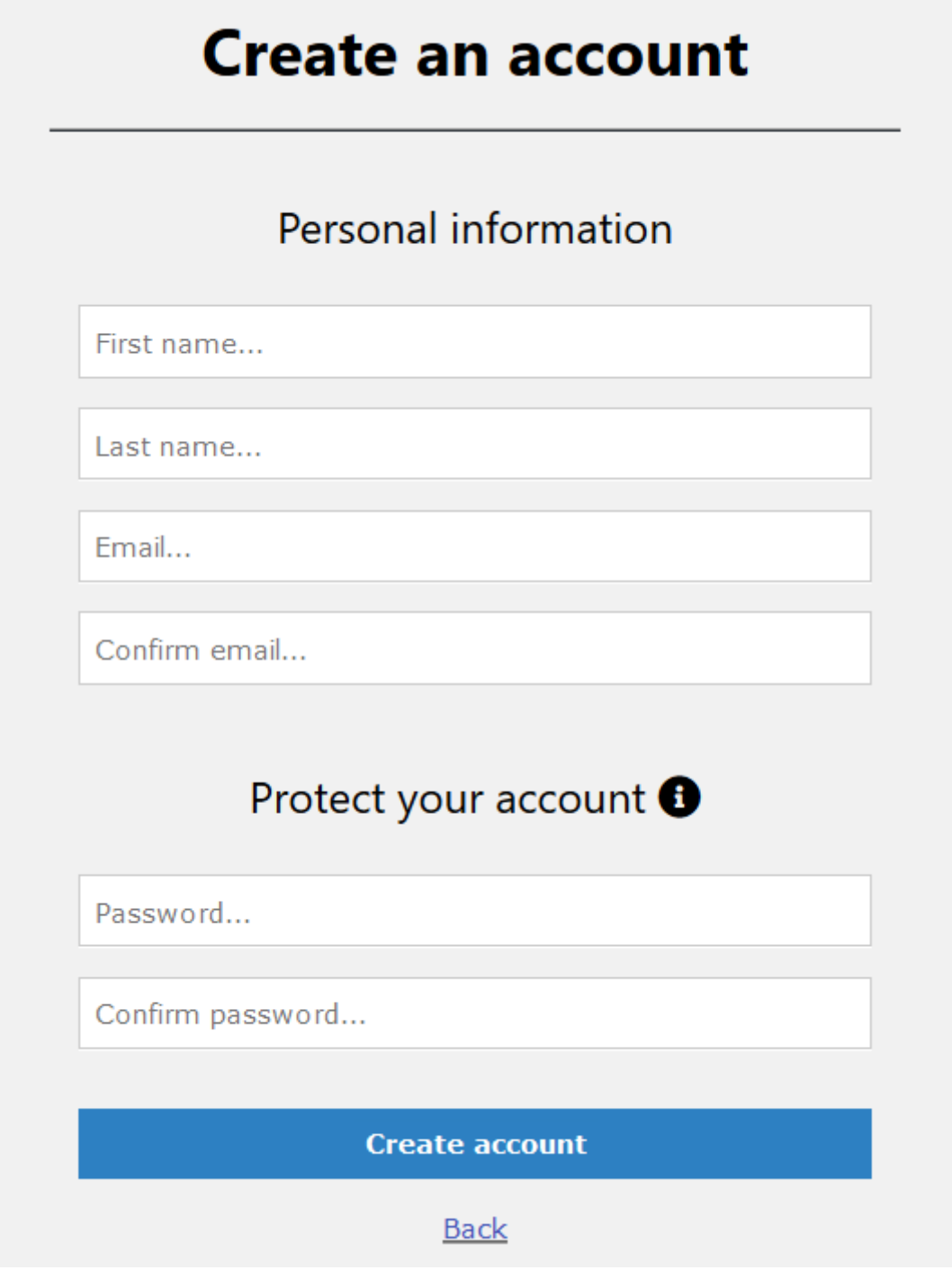


Figura 84: Formulari de creació de comptes d'usuari

Les dades necessàries per poder enregistrar-se en el sistema són: el nom, cognom, correu electrònic, la confirmació del correu electrònic, la contrasenya i la seva confirmació. Aquestes confirmacions són necessàries per verificar que l'usuari no s'ha confós i desitja realitzar la creació del compte amb el correu electrònic i la contrasenya enviada. A més a més, l'usuari pot situar el cursor a sobre de la icona d'informació per veure els requeriments mínims necessaris per establir una contrasenya segura.

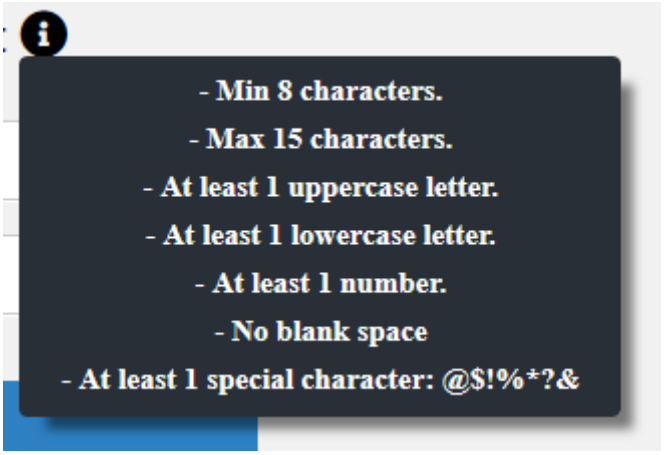


Figura 85: Recomanacions a l'usuari per establir una contrasenya segura

Un cop enviada la informació, el sistema realitzarà les validacions corresponents, tant en el client com en el servidor.

```
function validatePersonalInformation(firstName, lastName, email, emailConfirmation){

    //VALIDATE REQUIRED FIELDS
    if(!validateFieldRequired(firstName) || !validateFieldRequired(lastName) || !validateFieldRequired(email) || !validateFieldRequired(emailConfirmation)){
        return "Please fill the information. All the fields are mandatory!";
    }
    //VALIDATE FIELDS LENGTH
    if(!validateFieldLength(firstName, 255)){
        return "The length of the first name cannot be over 255 characters!";
    }else if(!validateFieldLength(lastName, 255)){
        return "The length of the last name cannot be over 255 characters!";
    }else if(!validateFieldLength(email, 255)){
        return "The length of the email cannot be over 255 characters!";
    }
    //VALIDATE FULL NAME FORMAT
    if(!validateFirstNameFormat(firstName)){
        return "The first name has not a valid format!";
    }else if(!validateLastNameFormat(lastName)){
        return "The last name has not a valid format!";
    }
    //VALIDATE EMAIL FORMAT
    if(!validateEmailFormat(email)){
        return "The email has not a valid format!";
    }
    //VALIDATE EMAIL EQUALITY
    if(!validateFieldEquality(email,emailConfirmation)){
        return "The value of fields email and email confirmation must be equals!";
    }
    //IF EVERYTHING OK, RETURN EMPTY ERROR
    return "";

}

function validatePassword(password, passwordConfirmation){

    //VALIDATE REQUIRED FIELDS
    if(!validateFieldRequired(password) || !validateFieldRequired(passwordConfirmation)){
        return "Please set the passwords. All the fields are mandatory!";
    }
    //VALIDATE SECURE PASSWORD
    if(!validatePasswordFormat(password)){
        return "Set a secure password. See requirements specified in the info icon!";
    }
    //VALIDATE PASSWORD EQUALITY
    if(!validateFieldEquality(password,passwordConfirmation)){
        return "The value of fields password and password confirmation must be equals!";
    }
    //IF EVERYTHING OK, RETURN EMPTY ERROR
    return "";
}
```

Figura 86: Validacions en el client del formulari de creació d'usuari

En cas que alguna de les dades enviades no sigui vàlida, es mostrarà el missatge d'error corresponent en el formulari, com per exemple el següent cas:

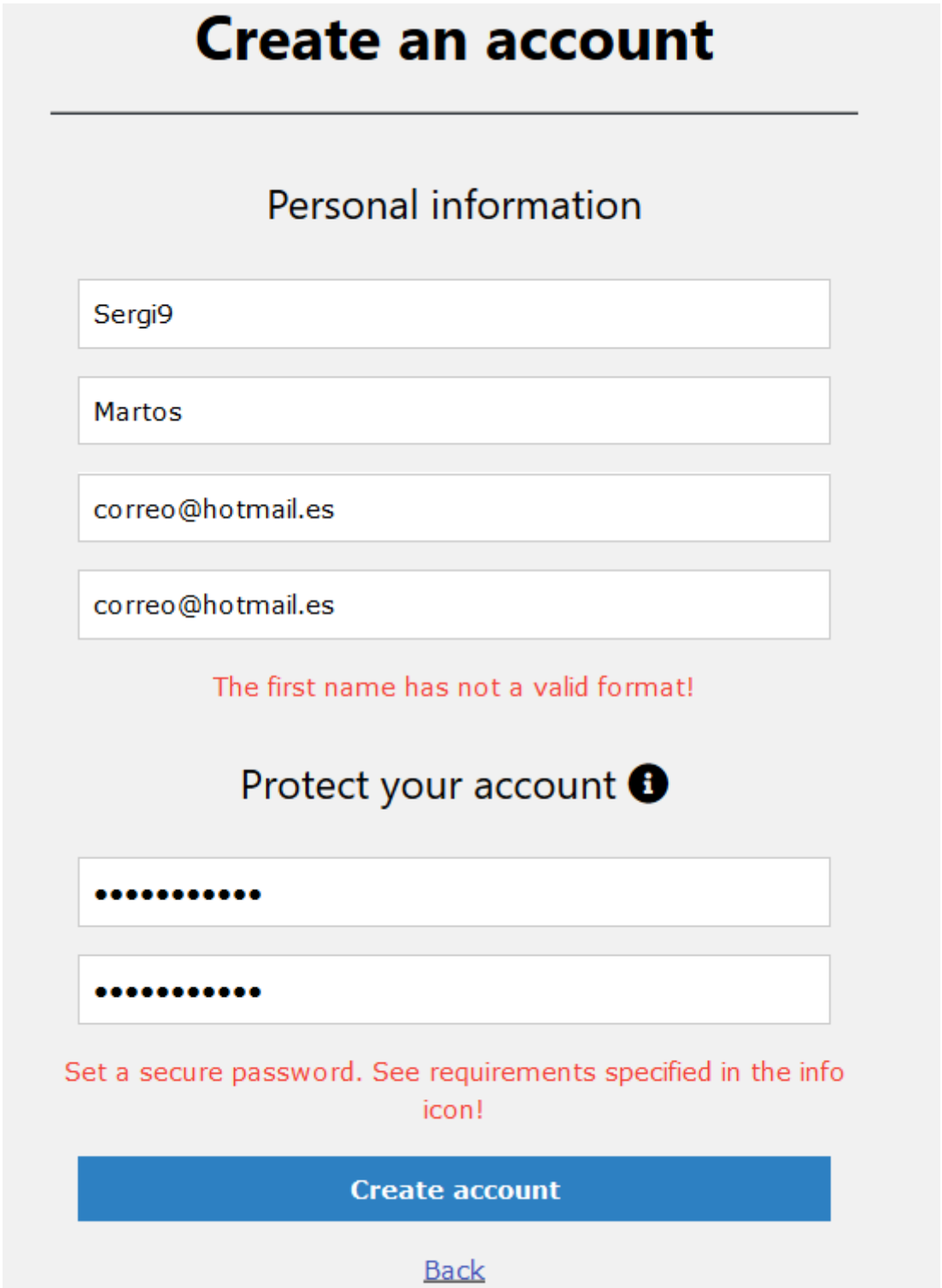


Figura 87: Exemple de formulari de creació d'usuaris amb informació no vàlida

En cas contrari, el sistema comprova que el correu electrònic no hagi sigut enregistrat prèviament i, en cas de no existir, el sistema crea la cadena hash corresponent de la contrasenya, i es genera un codi únic, que s'utilitzarà posteriorment per activar el compte d'usuari.

```
//CHECK IF ACCOUNT EXISTS IN DB
$sql = "SELECT * FROM Customer WHERE email = '".$email."'";
$result = mysqli_query($conn,$sql);
$row = mysqli_fetch_array($result,MYSQLI_ASSOC);
$count = mysqli_num_rows($result);
// IF USER EXISTS
if($count == 1) {
   $error = "The email given has been already registered!";
}else{
   //HASH PASSWORD
   $passwordHash = password_hash($password, PASSWORD_DEFAULT);
   //GENERATE ACTIVATION CODE
   $activationCode = uniqid();
   //SAVE USER
   $sql = "INSERT INTO Customer (first_name, last_name, email, password,
      activation_code) VALUES ('".$firstName."', '".$lastName."', '".$email."
      ', '".$passwordHash."', '".$activationCode."')" ;
   //IF EVERYTHING OK, SEND THE EMAIL AND REDIRECT TO ACTIVATION PAGE
   if ($conn->query($sql) === TRUE) {
      $error = Mailer::sendActivationAccountMail($email,$firstName." ".$
         lastName,$activationCode);
      if($error !== NULL && $error !== ''){
         $error = "Error trying to create new account. Please, try later or
            contact to our support team!";
      }else{
         header("location: activation_account_form.php");
      }
   } else {
      $error = "Error trying to create new account. Please, try later or
         contact to our support team!";
   }
}
```

Figura 88: Procediment de creació d'usuaris

El sistema desa el nou usuari inserint en la base de dades el nom, cognom, correu electrònic, la cadena *hash* referent a la contrasenya, per no desar la contrasenya en la base de dades en text pla, i el codi d'activació. Finalment, s'enviarà un correu electrònic al destinatari especificat per enviar a l'usuari el codi generat anteriorment que haurà d'enviar en el següent formulari d'activació de comptes d'usuari.

```php
public static function sendActivationAccountMail($email,$name,$code) {

    /* Include the Composer generated autoload.php file. */
    require("C:/wamp64/composer/vendor/autoload.php");
    require("C:/wamp64/composer/vendor/phpmailer/phpmailer/src/Exception.php");
    require("C:/wamp64/composer/vendor/phpmailer/phpmailer/src/PHPMailer.php");
    require("C:/wamp64/composer/vendor/phpmailer/phpmailer/src/SMTP.php");

    $shopEmail = "gamingkeysshop@hotmail.com";
    $shopEmailPass = "Gaming_Keys_99";
    $shopName = "Gaming Keys";

    $mail = new PHPMailer\PHPMailer\PHPMailer(TRUE);
    try {
        $mail->IsSMTP();
        $mail->CharSet = 'UTF-8';
        $mail->Host = "smtp.live.com";
        $mail->SMTPAuth= true;
        $mail->Port = 587;
        $mail->Username= $shopEmail;
        $mail->Password= $shopEmailPass;
        $mail->SMTPSecure = 'tls';
        $mail->From = $shopEmail;
        $mail->FromName= $shopName;
        $mail->isHTML(true);
        $mail->Subject = 'Activation user account';
        $mail->Body = '<span>Please, copy the code in the activation form to activate your
            account: <strong>'.$code.'</strong></span>';
        $mail->addAddress($email);

        $mail->send();

    } catch (PHPMailer\PHPMailer\Exception $e) {
       return $e->errorMessage();
    }
}
```

Figura 89: Exemple de procediment per enviar un correu electrònic

Finalment, en cas que tot hagi anat bé, el client haurà rebut el correu electrònic i podrà visualitzar el següent formulari d'activació de comptes d'usuari.

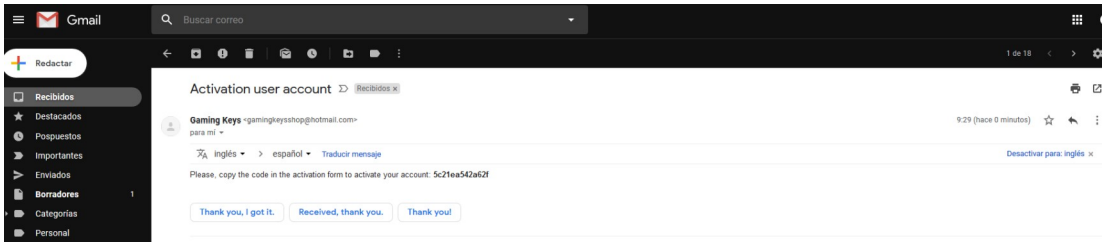


Figura 90: Exemple de correu electrònic enviat amb el codi d'activació

5.6.10. Activar compte d'usuari

Qualsevol client, si no té la sessió iniciada, pot visualitzar, emplenar les dades requerides i enviar el formulari d'activació de comptes d'usuari.

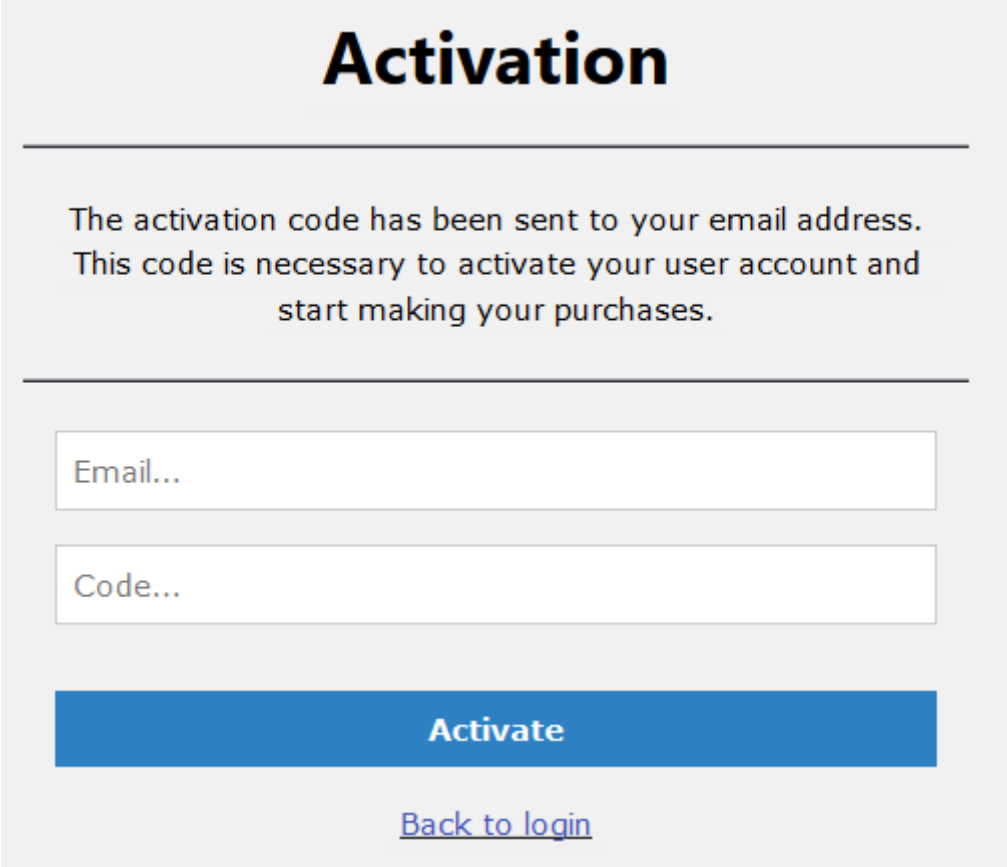


Figura 91: Formulari d'activació de comptes d'usuari

En aquest formulari, l'usuari ha d'introduir el correu electrònic que ha registrat en el sistema juntament amb el codi d'activació que li ha arribat a l'adreça indicada, com per exemple el de la figura 90. Un cop enviada la informació, el sistema realitzarà les validacions corresponents, tant en el client com en el servidor.

```
function validateActivation(email, code){

    //VALIDATE REQUIRED FIELDS
    if(!validateFieldRequired(email) || !validateFieldRequired(code)){
        return "Please fill the information. All the fields are mandatory!";
    }
    //VALIDATE FIELDS LENGTH
    if(!validateFieldLength(email, 255)){
        return "The length of the email cannot be over 255 characters!";
    }else if(!validateFieldLength(code, 255)){
        return "The length of the activation code cannot be over 255 characters!";
    }
    //VALIDATE EMAIL FORMAT
    if(!validateEmailFormat(email)){
        return "The email has not a valid format!";
    }
    //VALIDATE CODE FORMAT
    if(!validateCodeFormat(code)){
        return "The activation code has not a valid format!";
    }
    //IF EVERYTHING OK, RETURN EMPTY ERROR
    return "";

}
```

Figura 92: Validacions en el client del formulari d'activació

En cas que alguna de les dades enviades no sigui vàlida, es mostrarà el missatge d'error corresponent en el formulari, com per exemple el següent cas:

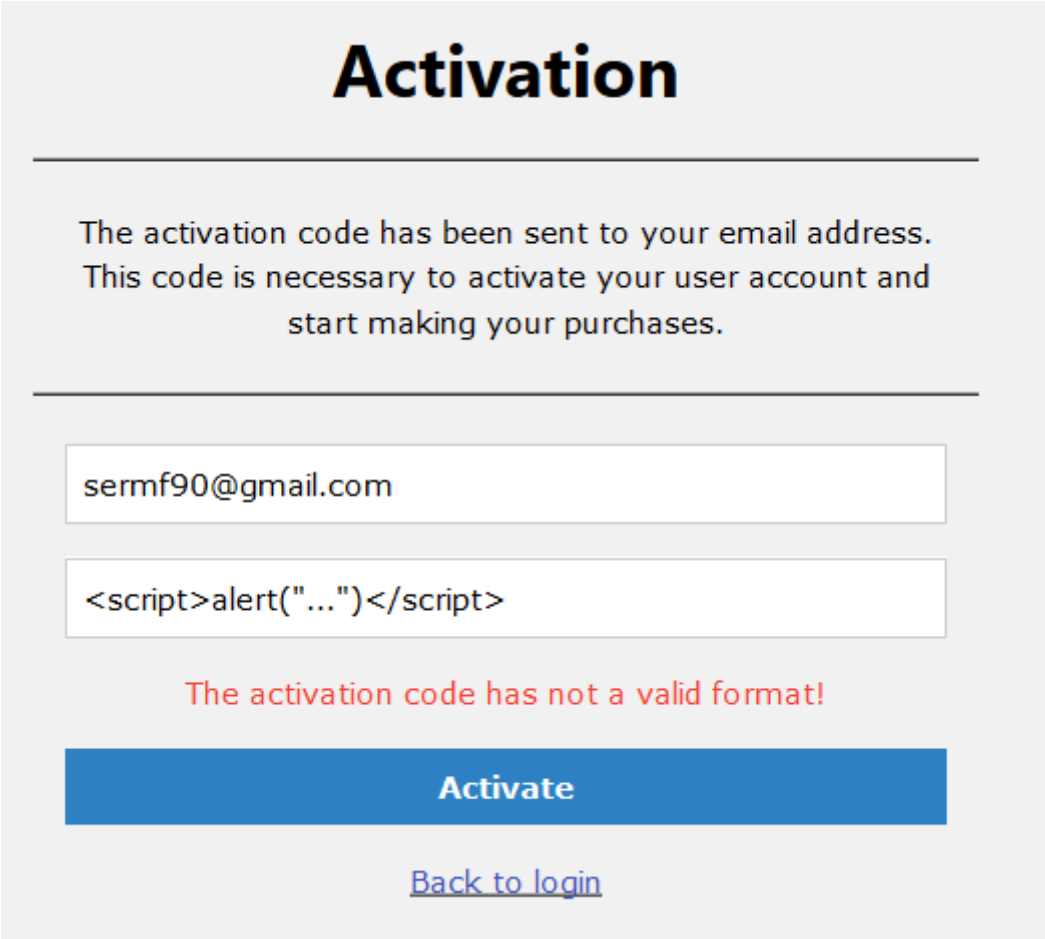


Figura 93: Exemple d'error de validació en el formulari d'activació

En cas contrari, el sistema comprova que el correu electrònic hagi sigut enregistrat prèviament, no hagi sigut activat prèviament i que no hagi sigut bloquejat pels intents realitzats. Si no és el cas, el sistema suma un intent a l'usuari que vol activar el compte i comprova que el codi d'activació coincideixi amb el generat prèviament.

En cas de no coincidir, es mostrarà el missatge d'error corresponent. En cas contrari, es mostrarà un missatge de confirmació i l'usuari ja podrà iniciar sessió en el sistema per realitzar les seves compres.

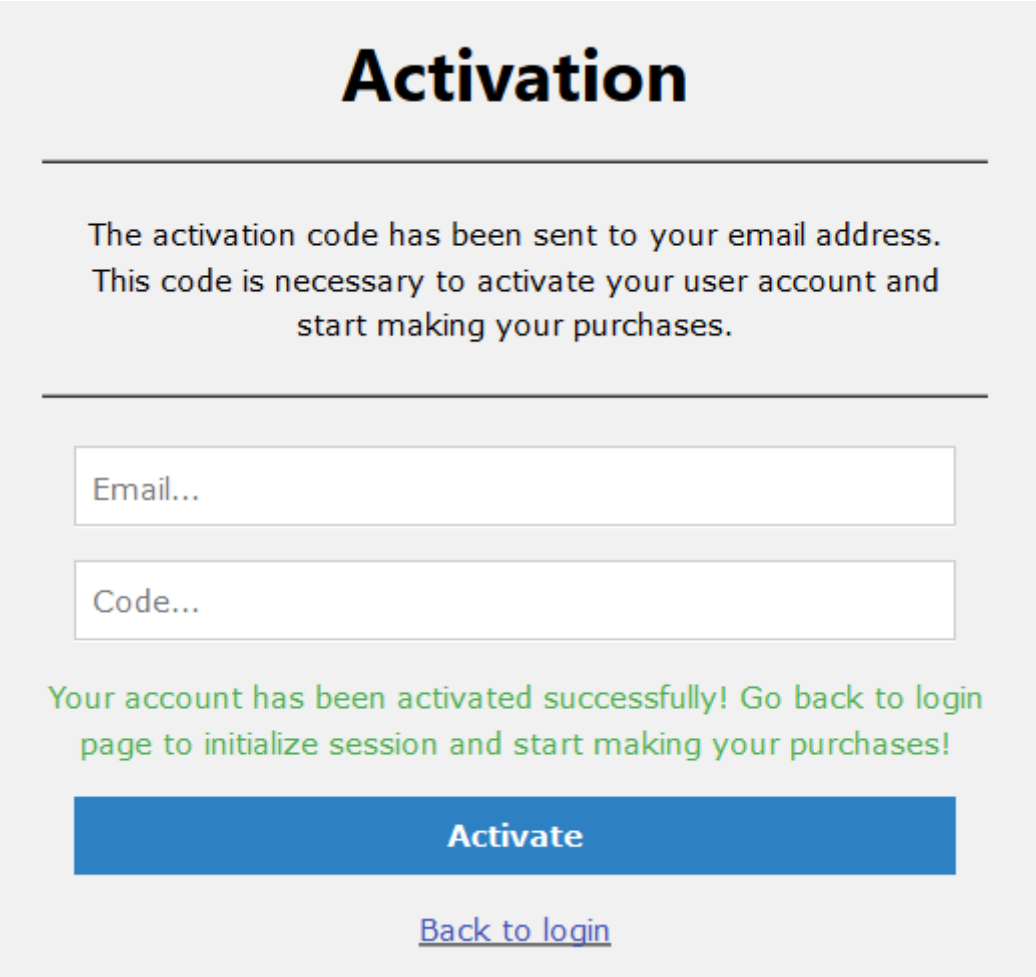


Figura 94: Exemple de confirmació d'activació

5.6.11. Iniciar sessió

Qualsevol client, si no té la sessió iniciada, pot visualitzar, emplenar les dades requerides i enviar el formulari d'inici de sessió.

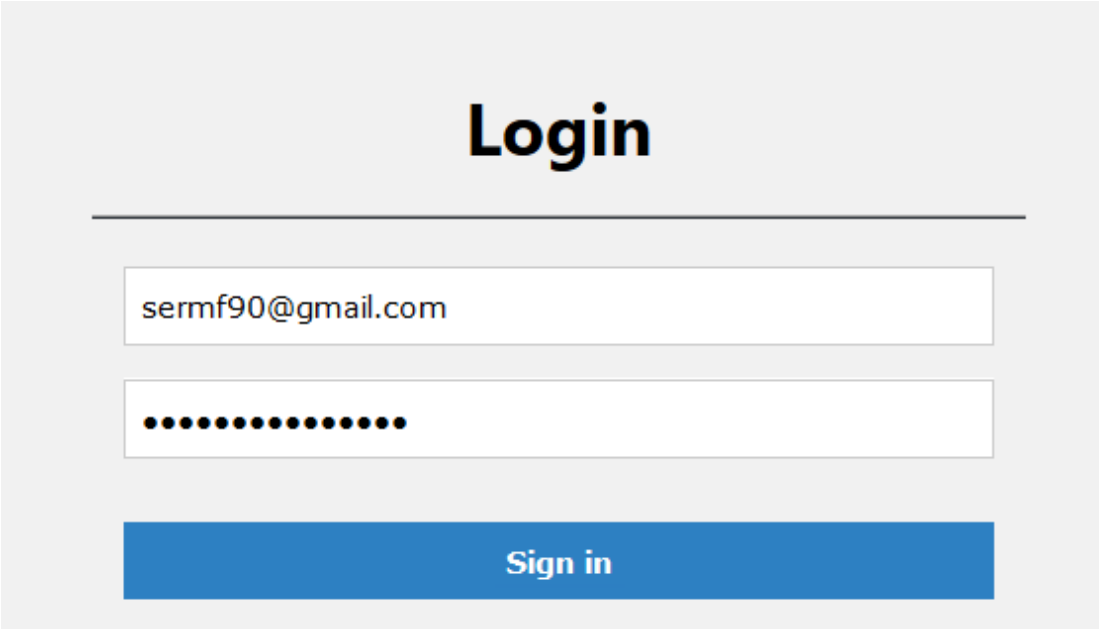


Figura 95: Formulari d'inici de sessió

En aquest formulari, l'usuari ha d'introduir el correu electrònic i la contrasenya que ha registrat en el sistema. Un cop enviada la informació, el sistema realitzarà les validacions corresponents, tant en el client com en el servidor.

```
function validateLogin(email, password){

    //VALIDATE REQUIRED FIELDS
    if(!validateFieldRequired(email) || !validateFieldRequired(password)){
        return "Please fill the information. All the fields are mandatory!";
    }
    //VALIDATE FIELDS LENGTH
    if(!validateFieldLength(email, 255)){
        return "The length of the email cannot be over 255 characters!";
    }
    //VALIDATE EMAIL FORMAT
    if(!validateEmailFormat(email)){
        return "The email has not a valid format!";
    }
    //VALIDATE SECURE PASSWORD
    if(!validatePasswordFormat(password)){
        return "The password has not a valid format!";
    }
    return "";

}
```

Figura 96: Validacions en el client de l'inici de sessió

En cas que alguna de les dades enviades no sigui vàlida, es mostrarà el missatge d'error corresponent en el formulari, com per exemple el següent cas:

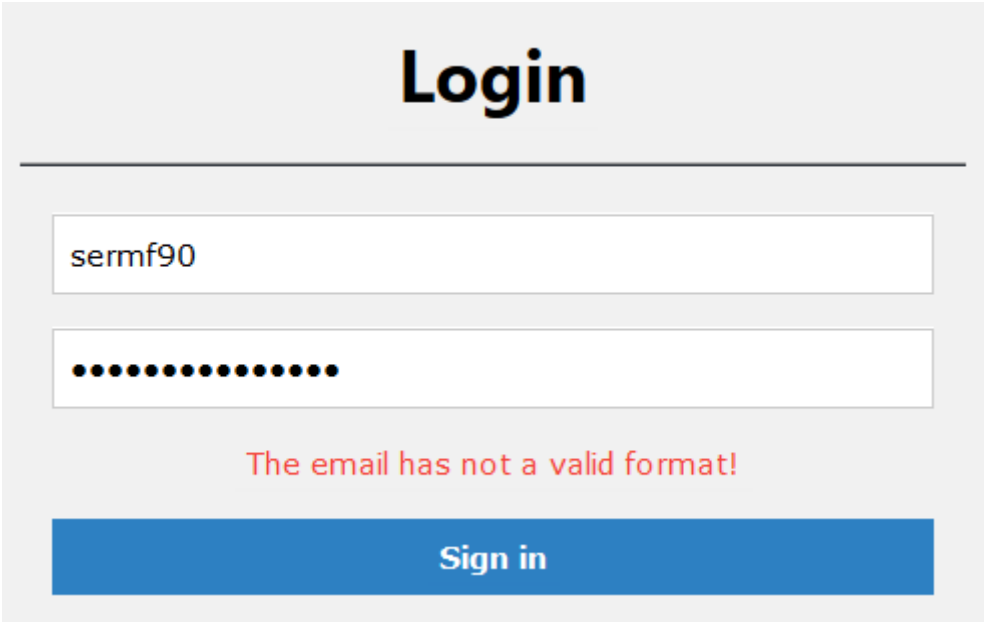


Figura 97: Exemple d'error de validació d'inici de sessió

En cas contrari, el sistema comprova que el correu electrònic hagi sigut enregistrat prèviament, que hagi sigut activat i que no hagi sigut bloquejat pels intents realitzats. Si no és el cas, el sistema suma un intent a l'usuari que vol iniciar sessió i comprova que les credencials introduïdes coincideixin amb les desades en la base de dades, prèviament generant la cadena *hash* a la contrasenya introduïda.

En cas de no coincidir, es mostrarà el missatge d'error corresponent. En cas contrari, es desen el correu electrònic i la cadena *hash*, corresponent a aquest amb un salt afegit, a la sessió i seguidament el sistema redirigeix cap a la pàgina principal.

```php
//ADD USER AND HASH TO SESSION
$_SESSION['login_user'] = $email;
$_SESSION['login_hash'] = hash("md5",$email."61m1n6_k3y$$");
//REDIRECT TO MAIN PAGE
header("location: index.php");
```

Figura 98: Valors d'usuari desats en sessió

Finalment, l'usuari haurà iniciat sessió en el sistema i tindrà accés a les vistes protegides, com per exemple les opcions d'usuari visibles des de la capçalera.

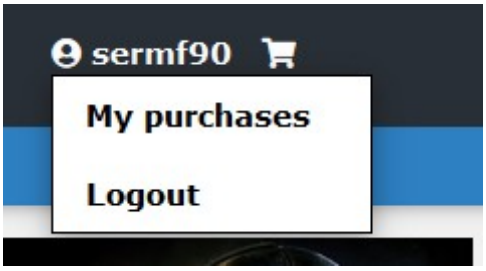


Figura 99: Opcions d'usuari

5.6.12. Modificar contrasenya

Qualsevol client, si no té la sessió iniciada, en cas d'haver oblidat la contrasenya del seu compte d'usuari, pot fer clic en l'enllaç que permet iniciar el procediment per recuperar el compte d'usuari.

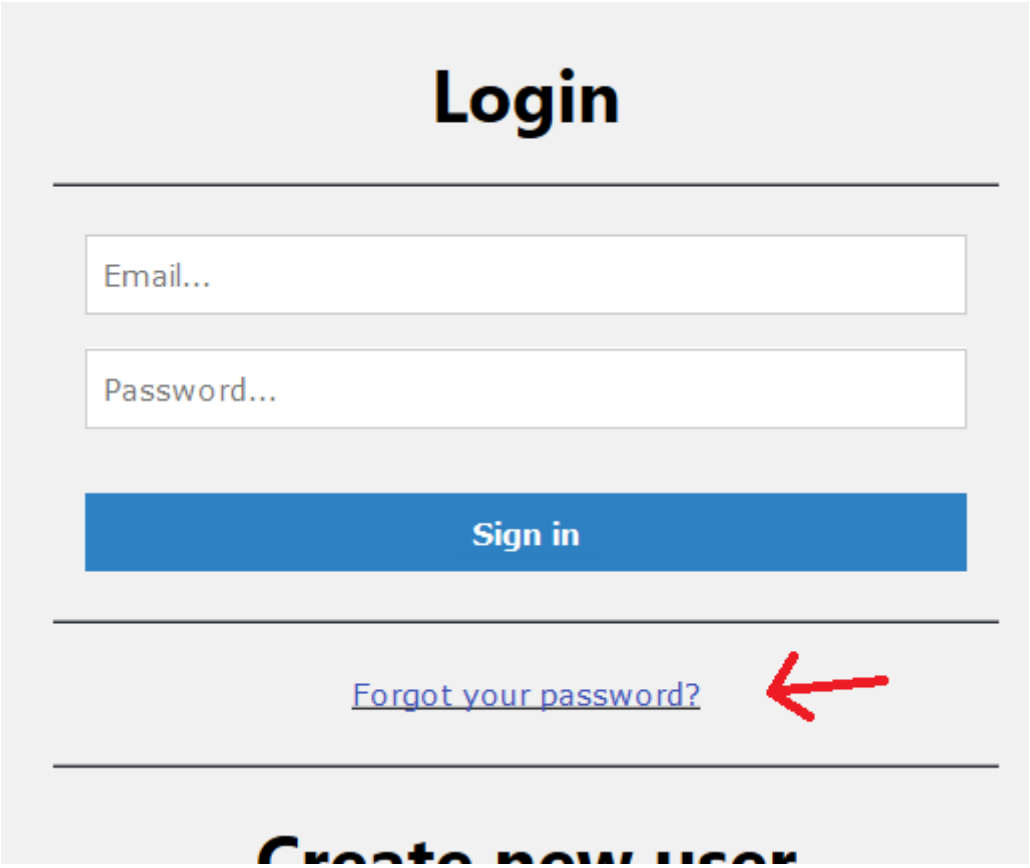


Figura 100: Enllaç per recuperar la contrasenya

Seguidament, el sistema redirigeix cap al formulari de recuperació de comptes, on l'usuari ha d'introduir el correu electrònic amb el qual es va enregistrar en el sistema.

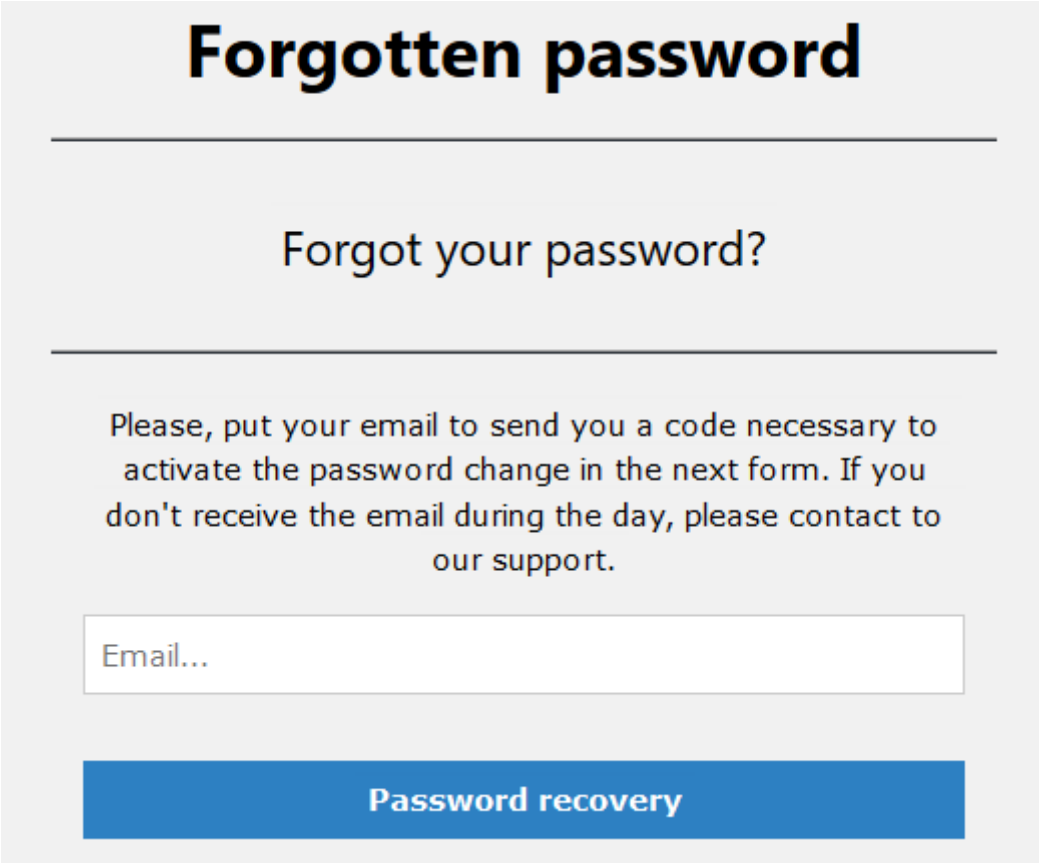


Figura 101: Formulari de recuperació de contrasenya

Un cop enviada la informació, el sistema realitzarà les validacions corresponents, tant en el client com en el servidor.

```
/* PASSWORD RECOVERY */
function validateRecoveryEmail(email){
    //VALIDATE REQUIRED FIELDS
    if(!validateFieldRequired(email)){
        return "Please fill the information. Email is mandatory!";
    }
    //VALIDATE FIELDS LENGTH
    if(!validateFieldLength(email, 255)){
        return "The length of the email cannot be over 255 characters!";
    }
    //VALIDATE EMAIL FORMAT
    if(!validateEmailFormat(email)){
        return "The email has not a valid format!";
    }
    //IF EVERYTHING OK, RETURN EMPTY ERROR
    return "";
}
```

Figura 102: Validacions en el client de la recuperació del compte d'usuari

En cas que el correu electrònic no sigui vàlid, es mostrarà el missatge d'error corresponent en el formulari, com per exemple el següent cas:

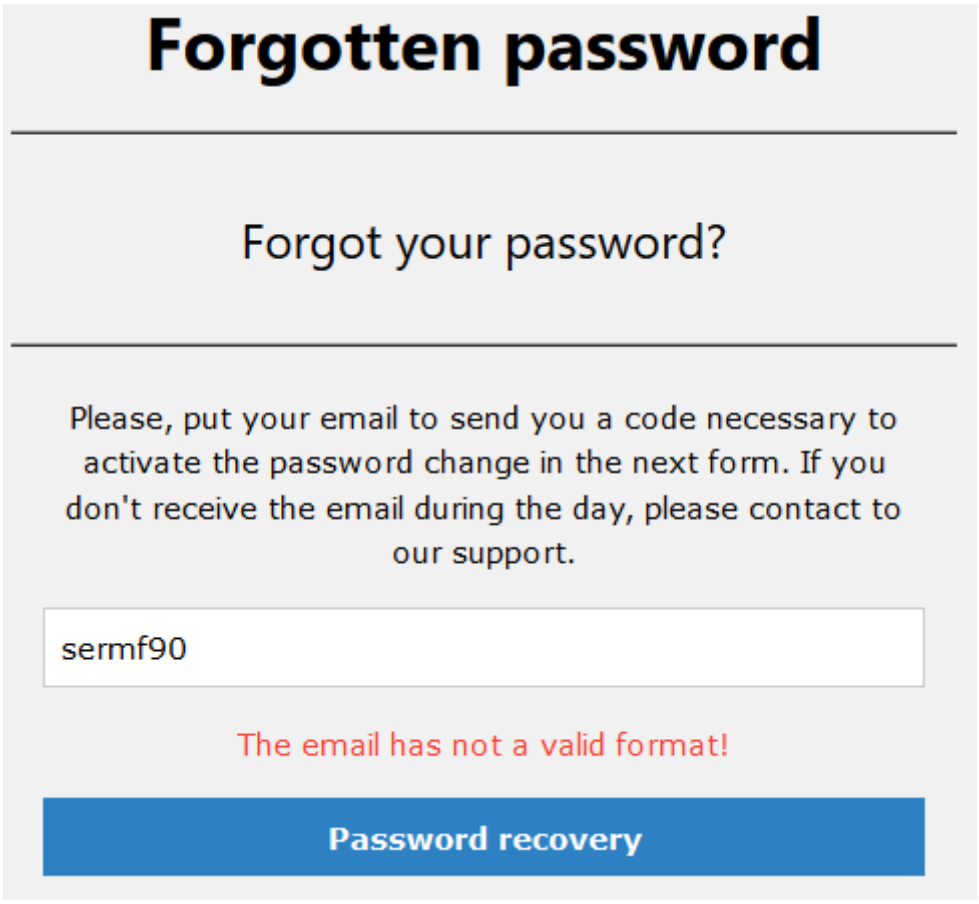


Figura 103: Exemple d'error de validació en el formulari de recuperació

En cas contrari, el sistema comprova que el correu electrònic hagi sigut enregistrat prèviament i, en cas de ser així, el sistema genera un codi únic, que s'utilitzarà posteriorment per recuperar el compte d'usuari, i actualitza l'usuari en la base de dades amb el nou codi de recuperació generat.

```php
//CHECK IF ACCOUNT EXISTS IN DB
$sql = "SELECT * FROM Customer WHERE email = '".$email."'";
$result = mysqli_query($conn,$sql);
$row = mysqli_fetch_array($result,MYSQLI_ASSOC);
$count = mysqli_num_rows($result);
// IF USER EXISTS
if($count == 1) {
    //GENERATE ACTIVATION CODE
    $recoveryCode = uniqid();
    //UPDATE USER RECOVERY CODE
    $sql = "UPDATE Customer SET recovery_code='".$recoveryCode."',
        recovery_att=0 WHERE id='".$row["id"]."'";
    //IF EVERYTHING OK, SEND THE EMAIL AND REDIRECT TO RECOVERY PAGE
    if ($conn->query($sql) === TRUE) {
        $error = Mailer::sendRecoveryPasswordMail($email,$recoveryCode);
        if($error !== NULL && $error !== ''){
            $error = "Error trying to recover the password. Please, try later or
                contact to our support team!";
        }else{
            header("location: modify_password_form.php");
        }
```

Figura 104: Procediment per iniciar la recuperació de comptes d'usuari

Finalment, s'enviarà un correu electrònic al destinatari especificat per enviar a l'usuari el codi generat anteriorment que haurà d'enviar en el següent formulari de modificació de comptes d'usuari.

```php
public static function sendRecoveryPasswordMail($email,$code) {

    /* Include the Composer generated autoload.php file. */
    require("C:/wamp64/composer/vendor/autoload.php");
    require("C:/wamp64/composer/vendor/phpmailer/phpmailer/src/Exception.php");
    require("C:/wamp64/composer/vendor/phpmailer/phpmailer/src/PHPMailer.php");
    require("C:/wamp64/composer/vendor/phpmailer/phpmailer/src/SMTP.php");

    $shopEmail = "gamingkeysshop@hotmail.com";
    $shopEmailPass = "Gaming_Keys_99";
    $shopName = "Gaming Keys";

    $mail = new PHPMailer\PHPMailer\PHPMailer(TRUE);
    try {
        $mail->IsSMTP();
        $mail->CharSet = 'UTF-8';
        $mail->Host = "smtp.live.com";
        $mail->SMTPAuth= true;
        $mail->Port = 587;
        $mail->Username= $shopEmail;
        $mail->Password= $shopEmailPass;
        $mail->SMTPSecure = 'tls';
        $mail->From = $shopEmail;
        $mail->FromName= $shopName;
        $mail->isHTML(true);
        $mail->Subject = 'Password recovery';
        $mail->Body = '<span>Please, copy the code in the activation form to confirm your
            account changes: <strong>'.$code.'</strong></span>';
        $mail->addAddress($email);

        $mail->send();

    } catch (PHPMailer\PHPMailer\Exception $e) {
        return $e->errorMessage();
    }
}
```

Figura 105: Procediment per enviar correus electrònics de recuperació de comptes

Finalment, en cas que tot hagi anat bé, el client haurà rebut el correu electrònic i podrà visualitzar el següent formulari de modificació de comptes d'usuari.

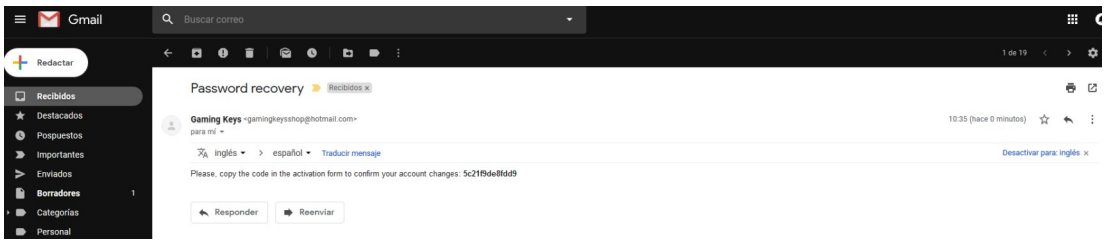


Figura 106: Exemple de correu electrònic enviat amb el codi de recuperació

Qualsevol client, si no té la sessió iniciada, pot visualitzar, emplenar les dades requerides i enviar el formulari de modificació de comptes d'usuari.

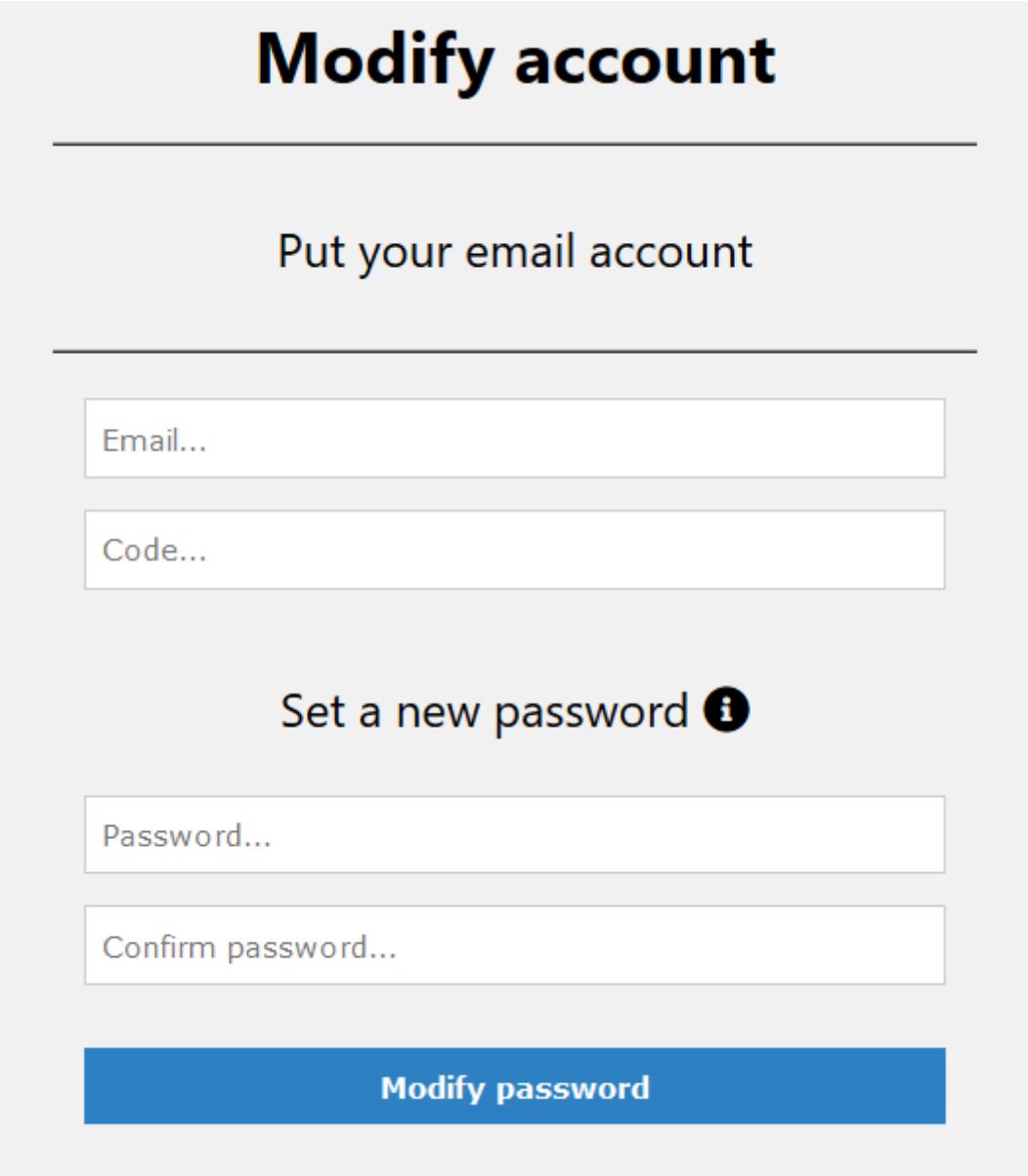


Figura 107: Formulari de modificació de comptes d'usuaris

En aquest formulari, l'usuari ha d'introduir el correu electrònic que ha registrat en el sistema juntament amb el codi de recuperació que li ha arribat a l'adreça indicada, com per exemple el de la figura 106, i la nova contrasenya. Un cop enviada la informació, el sistema realitzarà les validacions corresponents, tant en el client com en el servidor.

```
function validatePassword(password, passwordConfirmation){

    //VALIDATE REQUIRED FIELDS
    if(!validateFieldRequired(password) || !validateFieldRequired(passwordConfirmation)){
        return "Please set the passwords. All the fields are mandatory!";
    }
    //VALIDATE SECURE PASSWORD
    if(!validatePasswordFormat(password)){
        return "Set a secure password. See requirements specified in the info icon!";
    }
    //VALIDATE PASSWORD EQUALITY
    if(!validateFieldEquality(password,passwordConfirmation)){
        return "The value of fields password and password confirmation must be equals!";
    }
    //IF EVERYTHING OK, RETURN EMPTY ERROR
    return "";

}

/* ACTIVATION ACCOUNT */

function validateActivation(email, code){

    //VALIDATE REQUIRED FIELDS
    if(!validateFieldRequired(email) || !validateFieldRequired(code)){
        return "Please fill the information. All the fields are mandatory!";
    }
    //VALIDATE FIELDS LENGTH
    if(!validateFieldLength(email, 255)){
        return "The length of the email cannot be over 255 characters!";
    }else if(!validateFieldLength(code, 255)){
        return "The length of the activation code cannot be over 255 characters!";
    }
    //VALIDATE EMAIL FORMAT
    if(!validateEmailFormat(email)){
        return "The email has not a valid format!";
    }
    //VALIDATE CODE FORMAT
    if(!validateCodeFormat(code)){
        return "The activation code has not a valid format!";
    }
    //IF EVERYTHING OK, RETURN EMPTY ERROR
    return "";

}
```

Figura 108: Validacions en el client de la modificació de comptes d'usuaris

En cas que alguna de les dades enviades no sigui vàlida, es mostrarà el missatge d'error corresponent en el formulari, com per exemple el següent cas:

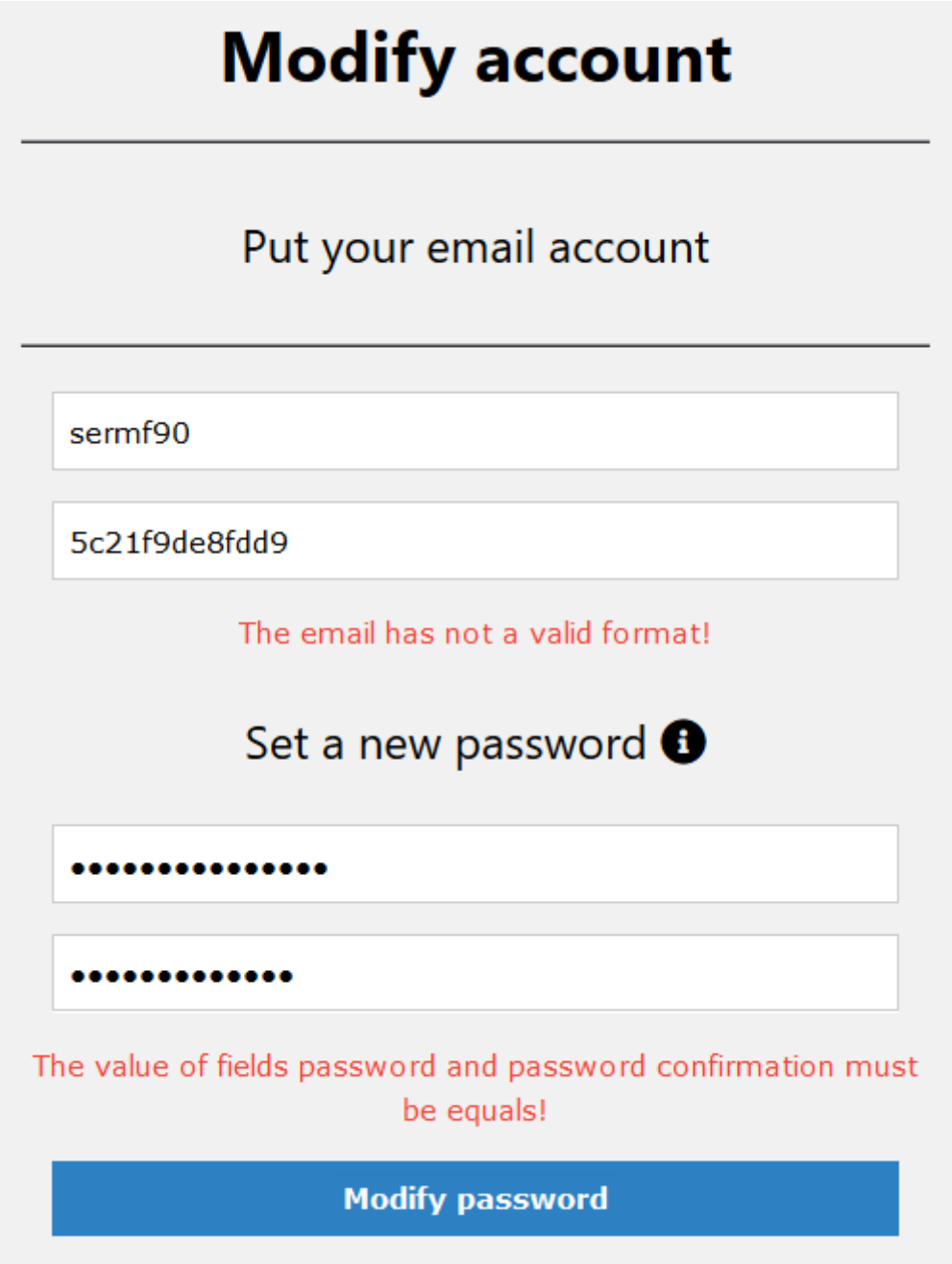


Figura 109: Exemple d'error de validació en el formulari de modificació de compte

En cas contrari, el sistema comprova que el correu electrònic hagi sigut enregistrat prèviament i que no hagi sigut bloquejat pels intents realitzats. Si no és el cas, el sistema suma un intent a l'usuari que vol recuperar el compte i comprova que el codi de recuperació coincideixi amb el generat prèviament.

En cas de no coincidir, es mostrarà el missatge d'error corresponent. En cas contrari, el sistema redirigeix cap a la pàgina principal i l'usuari ja podrà iniciar sessió en el sistema amb les seves noves credencials.

5.6.13. Realitzar pagament

Continuant en el pas de la visualització del resum de la compra, un cop iniciada la sessió d'usuari, pot seleccionar el mètode de pagament en el següent formulari i procedir amb l'operació de cobrament.

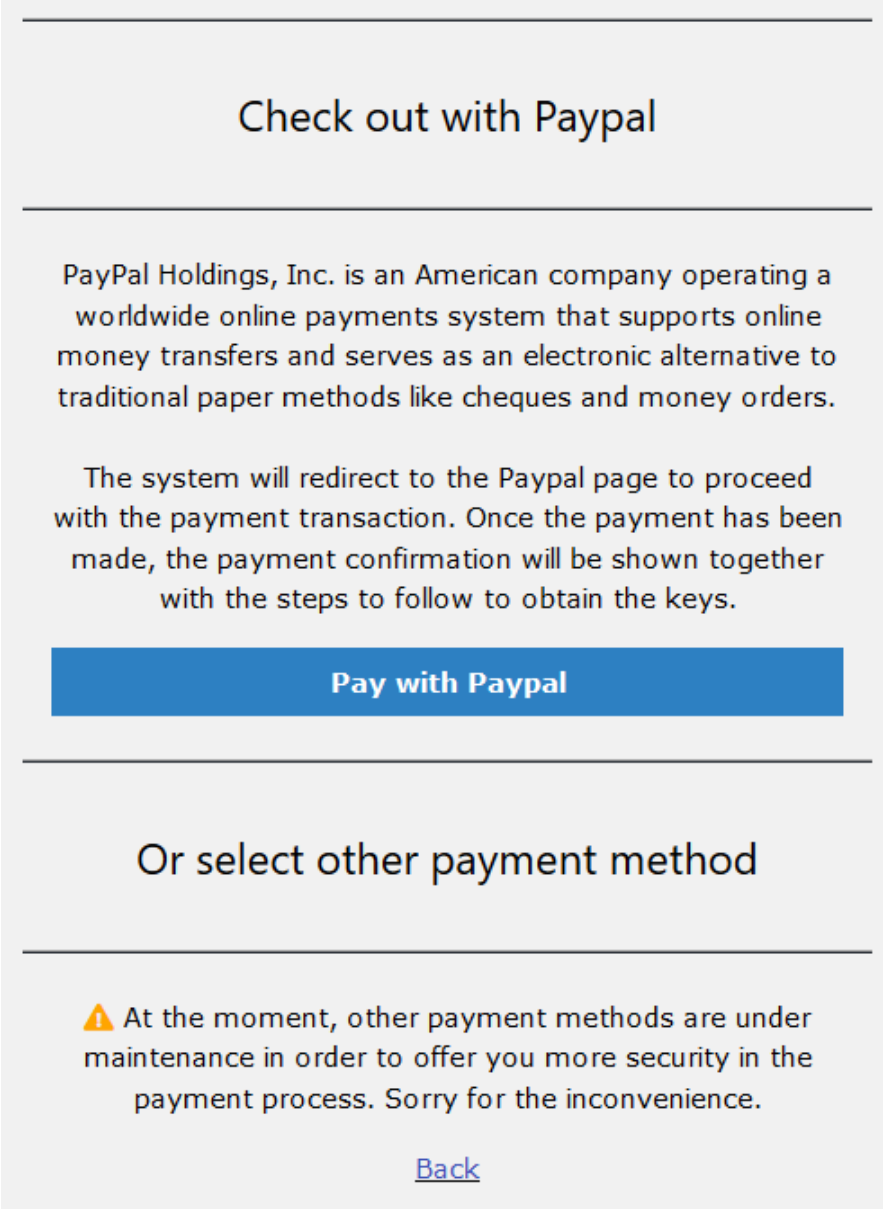


Figura 110: Mètodes de pagament disponibles

Actualment, només es poden realitzar pagaments amb PayPal, però en aquest formulari es podrien afegir tots els mètodes de pagament que hagin sigut integrats.

Seguidament, un cop l'usuari fa clic en el botó de pagament, el sistema recupera els productes de la sessió i realitza les validacions dels identificadors i de les unitats.

En cas de no haver-hi cap error, el sistema procedeix a recuperar els productes de la base de dades i actualitzar el preu final de la compra sumant els preus d'aquests. Seguidament, es desa l'estat actual de la compra en la base de dades, juntament amb els productes del carret.

```
//IF NO ERROR OCURRED, SAVE PAYMENT ORDER TO DATABASE
if($error == ""){
    //SAVE ORDER
    $sql = "INSERT INTO Purchase (customer, total_price, status) VALUES ('".$_
        SESSION['login_user']."', '".$productPrice."', 'Initiated')";
    if ($conn->query($sql) === TRUE) {
        //GET LAST ORDER ID
        $sql = "SELECT * FROM Purchase WHERE customer='".$_SESSION['login_user']."'
            ORDER BY id DESC LIMIT 1";
        $result = mysqli_query($conn,$sql);
        $row = mysqli_fetch_array($result,MYSQLI_ASSOC);
        $count = mysqli_num_rows($result);
        if($count == 1) {
            for($i=0; $i<$countProducts; $i++){
                //SAVE CART
                $sql = "INSERT INTO Cart (purchase_id, product_id, quantity) VALUES
                    ('".$row["id"]."', '".$products[$i]["id"]."', '".$products[$i]["
                    quantity"]."')";
                if ($conn->query($sql) !== TRUE) {
                    $error = "There was a problem during the payment process.
                        Please, try it later or contact to our support.";
                    break;
                }
            }
        } else {
            $error = "There was a problem during the payment process. Please, try
                it later or contact to our support.";
        }
    } else {
        $error = "There was a problem during the payment process. Please, try it
            later or contact to our support.";
    }
}
```

Figura 111: Procediment per desar l'estat incial de la compra en la base de dades

Finalment, el sistema redirigeix a l'usuari cap a la passarel·la externa de cobraments de PayPal enviant la informació següent:

```
//SET PAYPAL DATA
$paypalUrl = "https://www.sandbox.paypal.com/cgi-bin/webscr";
$paypalAccount = "gamingkeys-facilitor@gmail.com";
$productName = "Gaming Keys";
$productUnits = "1";
$productPrice = 0;
$currency = "EUR";
$returnUrl = "http://localhost/gk_shop/payment_success.php";
$returnCancelUrl = "http://localhost/gk_shop/payment_cancel.php";
```

Figura 112: Informació bàsica enviada a PayPal per realitzar una operació de cobrament

Aquesta informació indica, respectivament:

- Adreça del sistema on s'enviarà la petició. En aquest cas es tracta de l'entorn de proves de PayPal, però per fer transaccions reals només caldria modificar aquesta adreça i utilitzar comptes reals.
- Compte de la botiga, creada en apartats anteriors.
- Nom del producte que es comprarà juntament amb la quantitat. En aquest cas, totes les transaccions només contindran un producte anomenat Gaming Keys, simplement per facilitar la integració amb el sistema, però es podria afegir tots els productes del carret.
- El preu total que es cobrarà en la transacció, inicialitzat a 0 però posteriorment actualitzat, com s'ha esmentat anteriorment.
- La moneda en la qual es realitzarà el cobrament, en aquest cas Euros.
- Finalment, els enllaços corresponents al retorn automàtic que realitza el sistema de PayPal, esmentats en la configuració de comptes de PayPal d'aquest document. En cas que l'usuari decideixi cancel·lar el seu pagament abans de realitzar-ho, PayPal retornarà la resposta a la pàgina de cancel·lació. En cas contrari, utilitzarà la configurada en el compte.

Un cop en la passarel·la externa, l'usuari pot introduir les credencials del seu compte de PayPal, en aquest cas el compte personal de l'entorn de proves de PayPal, creat també anteriorment en la configuració de comptes de PayPal d'aquest document.

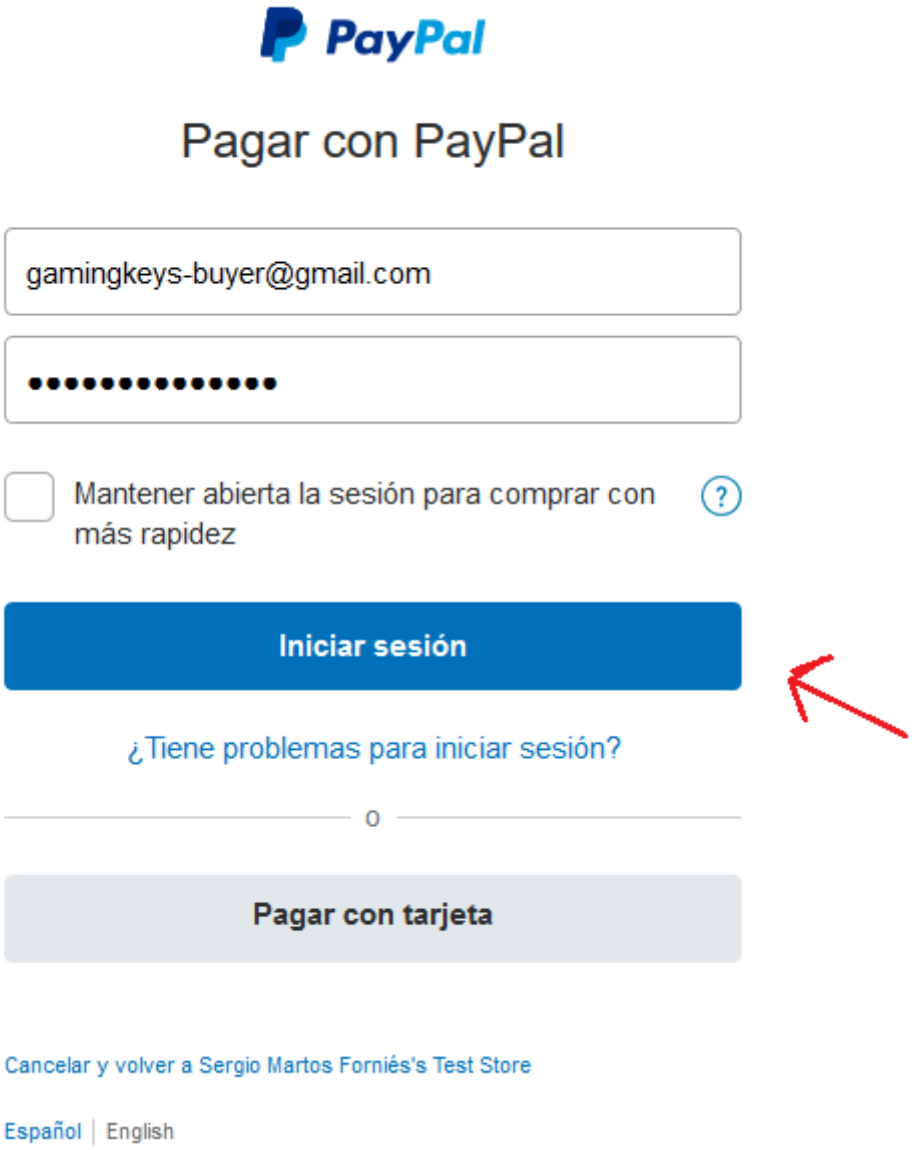


Figura 113: Inici de pagament amb PayPal

Un cop l'usuari inicia sessió en PayPal, es mostra el resum de la compra de la passarel·la.

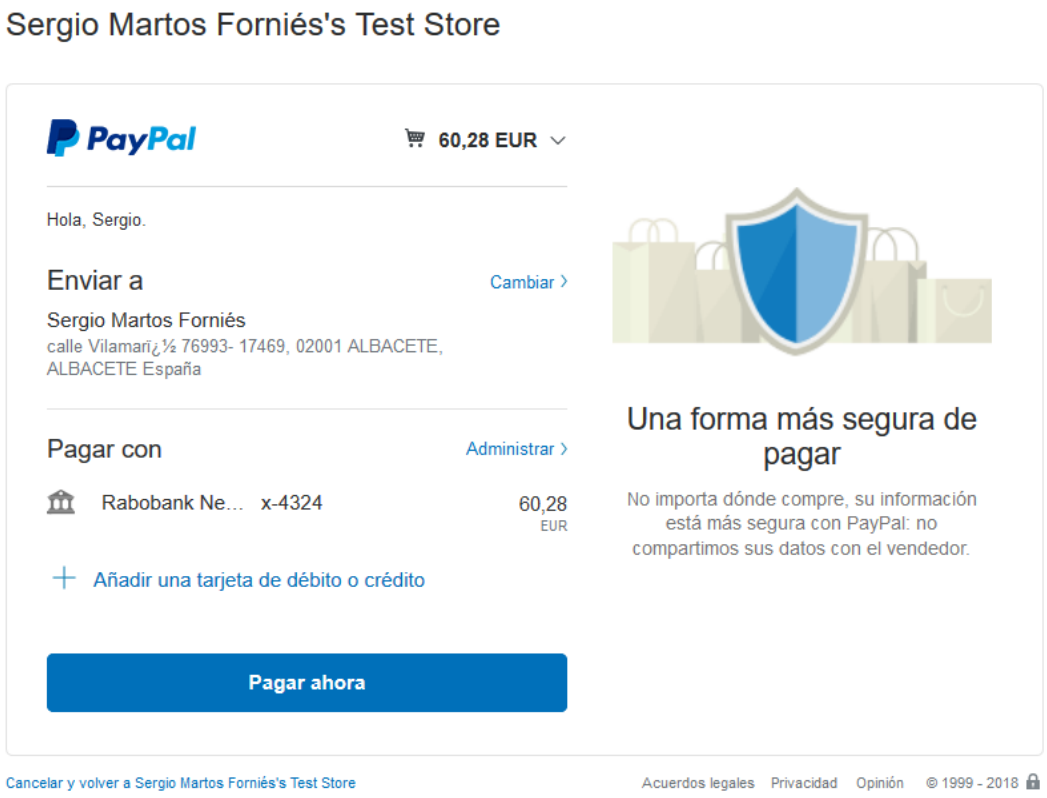


Figura 114: Formulari de confirmació de pagament amb PayPal

En aquest resum, es pot visualitzar molta informació de la compra, com per exemple el nom de la botiga i de la persona, adreça, mètode de pagament o el carret de la compra.

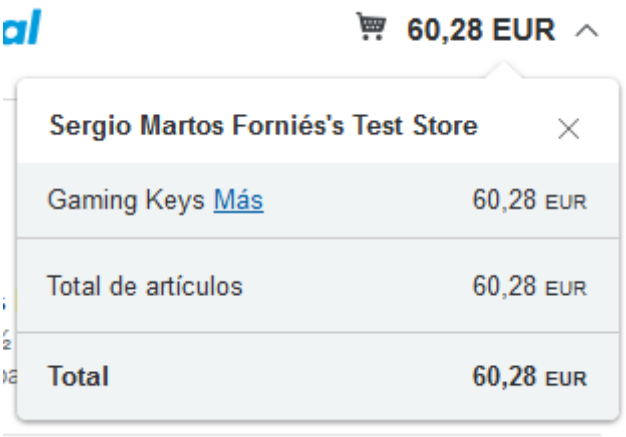


Figura 115: Carret de la compra de PayPal

En aquest cas, com s'ha esmentat anteriorment, totes les transaccions només contindran el producte anomenat *Gaming Keys*, però aquest fet no afectarà l'assignació de les claus dels productes al client. Seguidament, un cop confirma que vol realitzar el pagament, si tot va bé, el sistema de PayPal mostra a l'usuari la confirmació d'èxit en l'operació i redirigirà automàticament cap a la pàgina d'èxit de pagament configurada anteriorment.

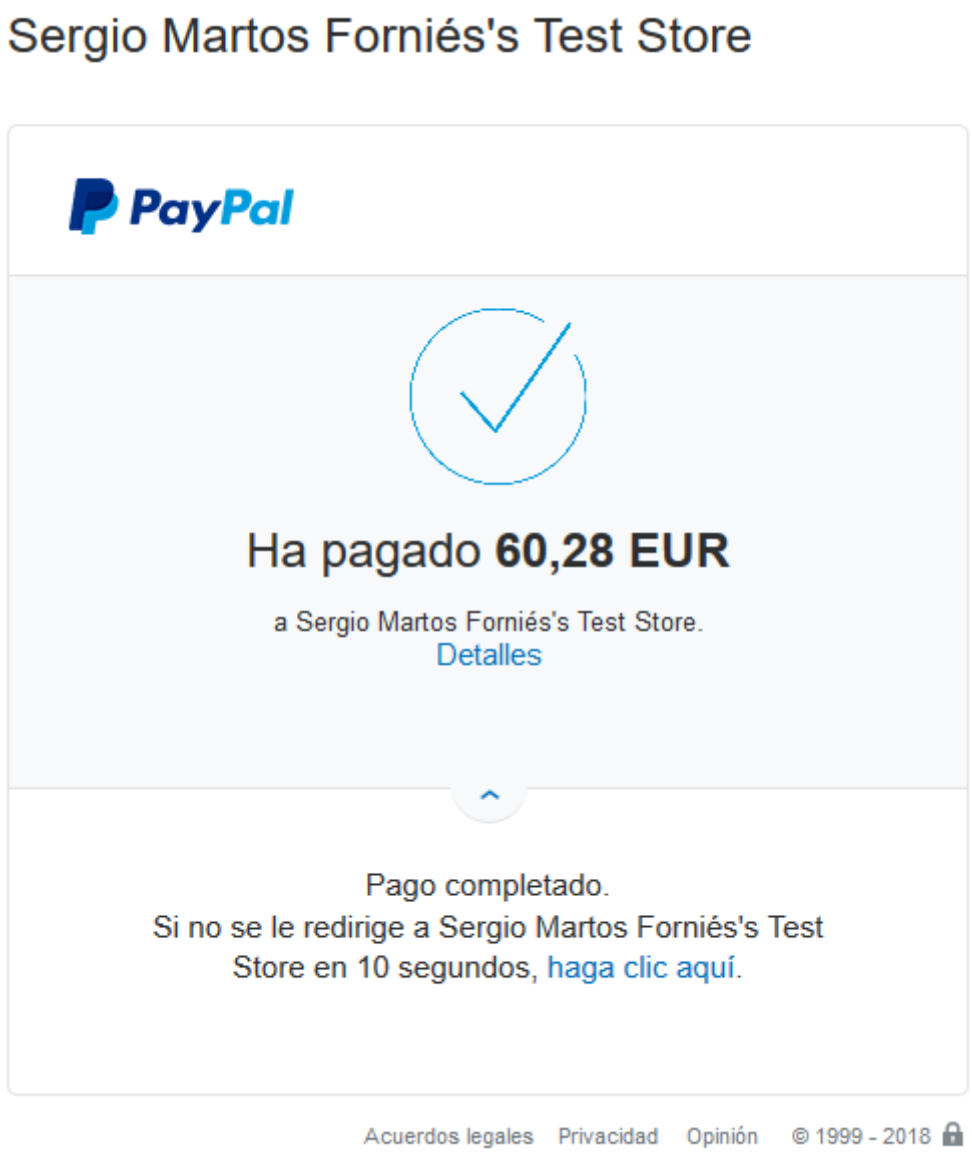


Figura 116: Confirmació del pagament amb PayPal

Un cop de nou en el nostre sistema, aquest recupera les dades que PayPal retorna sobre la transacció i actualitza l'estat de l'intent de compra desat en la base de dades prèviament abans d'abandonar la botiga, juntament amb la data d'actualització i l'identificador de la transacció obtingut de PayPal. Seguidament es verifica que l'import cobrat i l'import del carret siguin el mateix i, en cas de ser així, es procedeix a assignar les claus dels productes a l'usuari.

```
//ASSIGN KEYS
$sql = "SELECT * FROM Purchase WHERE customer='".$_SESSION['
    login_user']."' AND status='Completed' ORDER BY id DESC LIMIT 1"
    ;
$result = mysqli_query($conn,$sql);
$row = mysqli_fetch_array($result,MYSQLI_ASSOC);
$count = mysqli_num_rows($result);
if($count > 0){
    //FIND CART PRODUCTS BY PURCHASE
    $sqlCart = "SELECT * FROM Cart WHERE purchase_id='".$row["id"]."
        '";
    $resultCart = mysqli_query($conn,$sqlCart);
    $countCart = mysqli_num_rows($resultCart);
    if($countCart > 0){
        //FIND PRODUCTS KEYS BY ID
        while ($rowCart = mysqli_fetch_array($resultCart,
            MYSQLI_ASSOC)){
            $sqlKey = "SELECT * FROM Productkey WHERE product_id='".
                $rowCart["product_id"]."' AND customer is null
                LIMIT ".$rowCart["quantity"];
            $resultKey = mysqli_query($conn,$sqlKey);
            $countKey = mysqli_num_rows($resultKey);
            if($countKey > 0 && $countKey == $rowCart["quantity"]){
                while ($rowKey = mysqli_fetch_array($resultKey,
                    MYSQLI_ASSOC)){
                    //ASSIGN KEY
                    $sqlUpdateKey = "UPDATE Productkey SET customer=
                        '".$_SESSION['login_user']."' WHERE id=".$
                        rowKey["id"];
                    if ($conn->query($sqlUpdateKey) !== TRUE) {
                        $error = $errorMessageKeys;
                        break;
                    }
                }
            }else{
                $error = $errorMessageNoKeys;
                break;
            }
        }
    }
```

Figura 117: Procediment per assignar claus de productes

Qualsevol problema que hi hagi en l'assignació, com per exemple que s'acabin les existències, el sistema mostrarà el missatge d'error corresponent per avisar a l'usuari que s'ha de posar en contacte amb la botiga. En cas contrari, es mostrarà un missatge de confirmació, avisant a l'usuari que ja pot visualitzar les claus comprades en la pàgina corresponent.

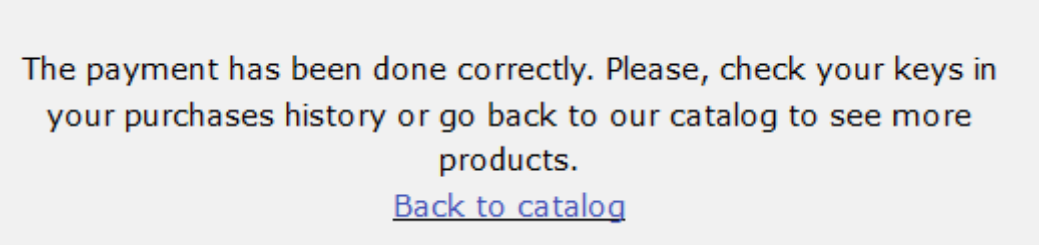


Figura 118: Missatge de confirmació de pagament amb èxit

No obstant això, en cas que l'usuari hagi cancel·lat el pagament des de l'entorn de PayPal, aquest enviarà la resposta cap a la pàgina de cancel·lació informada prèviament en la sol·licitud de compra i el sistema actualitzarà l'estat de la petició en la base de dades i mostrarà el missatge de confirmació de cancel·lació.

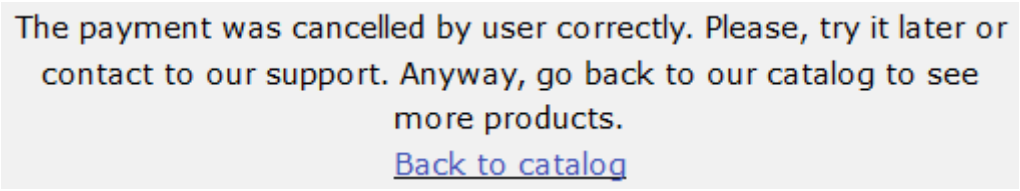


Figura 119: Missatge de confirmació de la cancel·lació amb èxit del pagament

Per verificar que s'han realitzat correctament els pagaments, es pot comprovar en els comptes de PayPal creats, tant en el del client com el de la botiga. Per exemple, en el compte business de la botiga es poden veure el llistat de transaccions i els seus detalls, inclosa la realitzada en l'exemple.

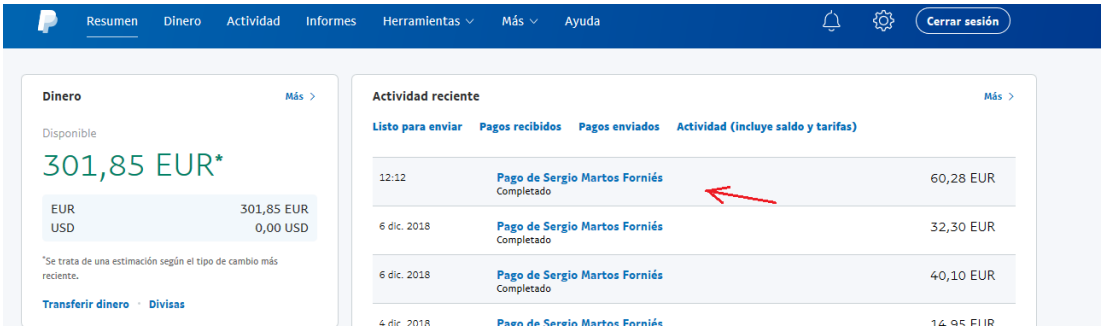
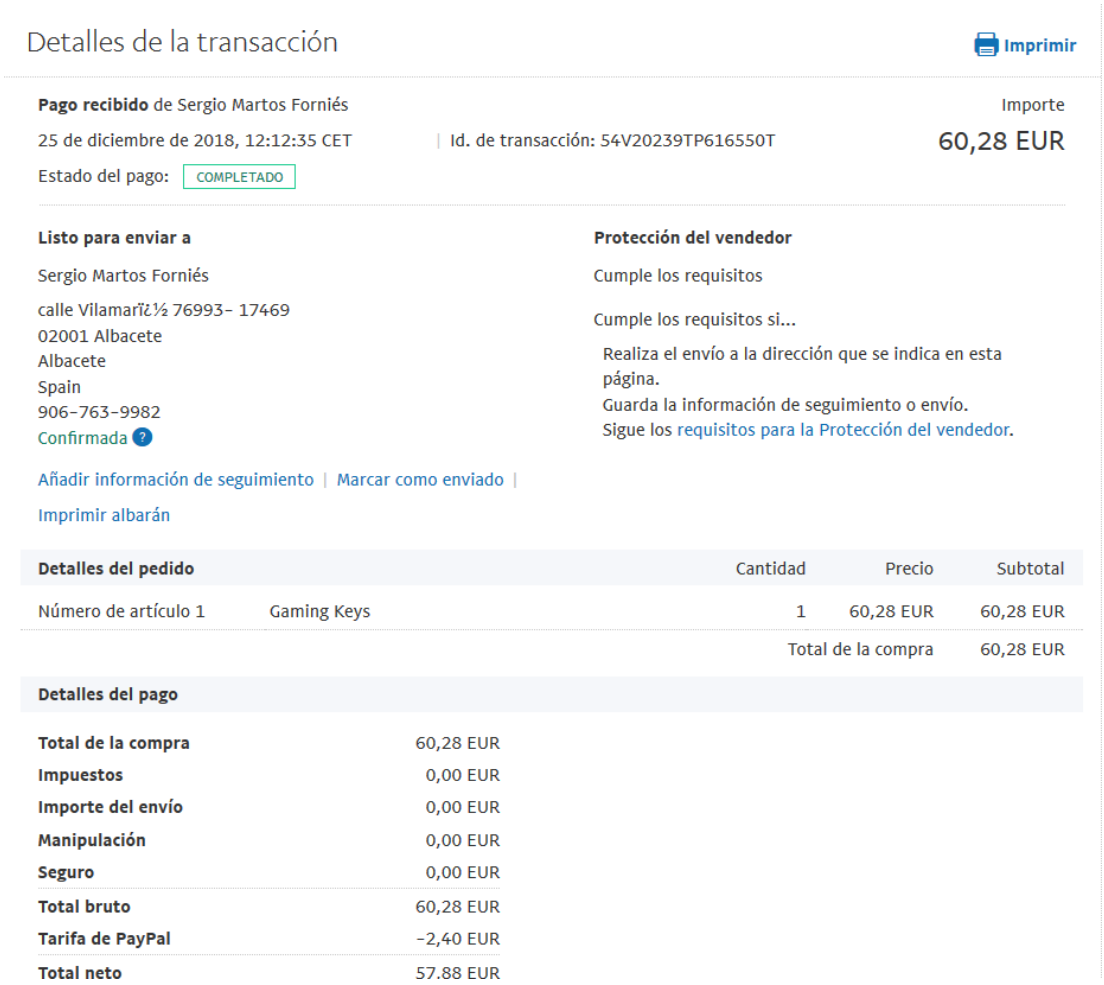


Figura 120: Exemple de transacció bancària en PayPal



Detalles de la transacción

Imprimir

Pago recibido de Sergio Martos Forniés

25 de diciembre de 2018, 12:12:35 CET | Id. de transacción: 54V20239TP616550T

Importe

60,28 EUR

Estado del pago: COMPLETADO

Listo para enviar a

Sergio Martos Forniés

calle Vilamarï¿½ 76993- 17469
02001 Albacete
Albacete
Spain
906-763-9982
Confirmada

Añadir información de seguimiento | Marcar como enviado |

Imprimir albarán

Protección del vendedor

Cumple los requisitos

Cumple los requisitos si...

Realiza el envío a la dirección que se indica en esta página.
Guarda la información de seguimiento o envío.
Sigue los requisitos para la Protección del vendedor.

Detalles del pedido		Cantidad	Precio	Subtotal
Número de artículo 1	Gaming Keys	1	60,28 EUR	60,28 EUR
			Total de la compra	60,28 EUR

Detalles del pago

Total de la compra	60,28 EUR
Impuestos	0,00 EUR
Importe del envío	0,00 EUR
Manipulación	0,00 EUR
Seguro	0,00 EUR
Total bruto	60,28 EUR
Tarifa de PayPal	-2,40 EUR
Total neto	57.88 EUR

Figura 121: Detalls d'una transacció bancària en PayPal

Un cop finalitzat amb èxit el procés de compra en la botiga, les claus apareixen disponibles en l'historial de compres de l'usuari.

5.6.14. Visualitzar l'historial de compres

Qualsevol client, si té la sessió iniciada, pot visualitzar l'historial de compres realitzades fent clic en l'enllaç corresponent de les opcions d'usuari de la capçalera.

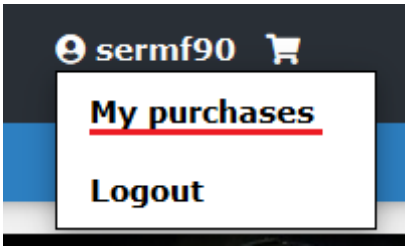


Figura 122: Opció per visualitzar l'historial de compres

El sistema comprova en base de dades quines compres ha realitzat l'usuari. En cas que l'usuari encara no hagi realitzat cap compra, el sistema mostra el missatge d'error corresponent, juntament amb un enllaç per redirigir a l'usuari cap al catàleg.

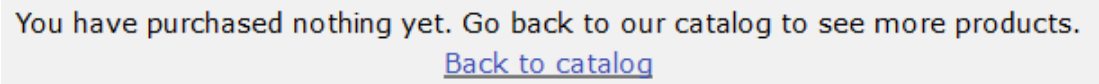


Figura 123: Historial de compres buit

En cas contrari, seguint l'exemple del cas anterior, el sistema cerca tots els productes assignats a l'usuari en la base de dades i els mostra en un llistat com el següent:

Figura 124: Exemple d'historial de compres

A primera vista, no es poden visualitzar les claus dels productes per motius de seguretat que s'explicaran més endavant. Per tant, per visualitzar-les, cal fer clic en la icona de l'ull i poder-les obtenir.

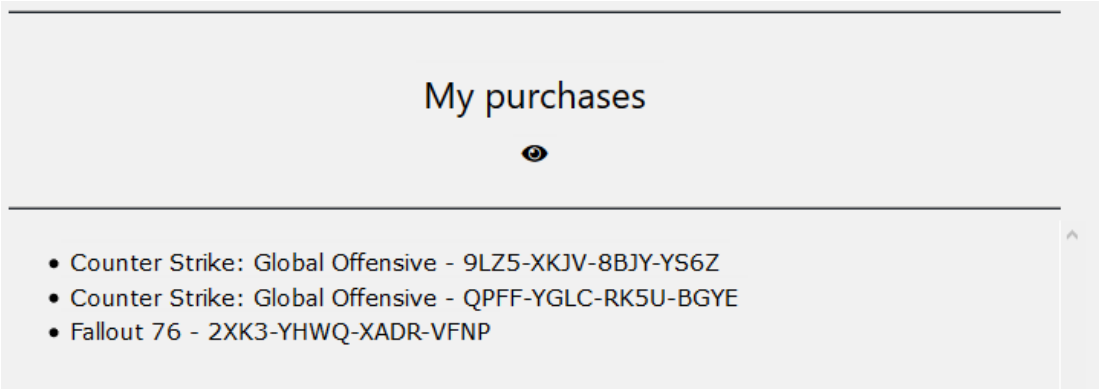


Figura 125: Exemple d'historial de compres mostrant les claus dels productes

I si es desitja tornar-les a amagar, només cal tornar a fer clic a la icona. Cal esmentar que aquestes claus han sigut generades aleatòriament. Per tant, no es poden activar els productes en les plataformes corresponents. En cas que ho fossin, només caldria aplicar la clau seguint el mètode de la plataforma corresponent. Per exemple en la plataforma Steam:

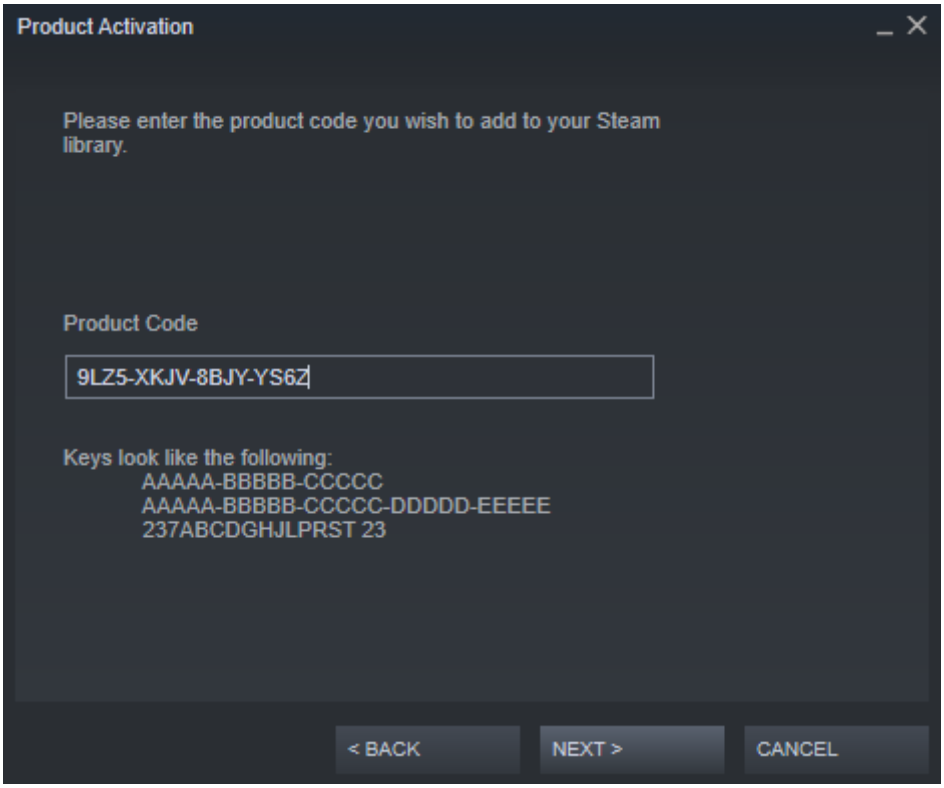


Figura 126: Exemple d'activació de producte en la plataforma Steam

5.6.15. Tancar sessió

Finalment en qualsevol moment del procés de compra, excepte durant la realització del pagament, sempre que el client hagi iniciat sessió d'usuari prèviament pot tancar-la fent clic en el següent botó:

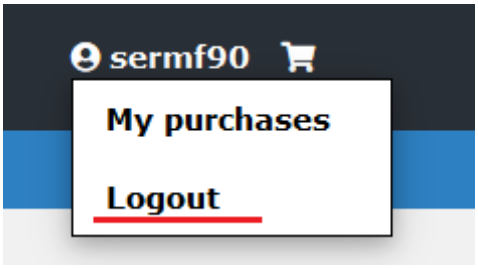


Figura 127: Opció de tancar sessió

Un cop s'ha fet clic, el sistema elimina tota la informació desada en la sessió (dades d'inici de sessió i carret de la compra) i realitza una redirecció cap a la pàgina principal, on es podrà visualitzar de nou l'opció d'inici de sessió i el carret de la compra completament buit:

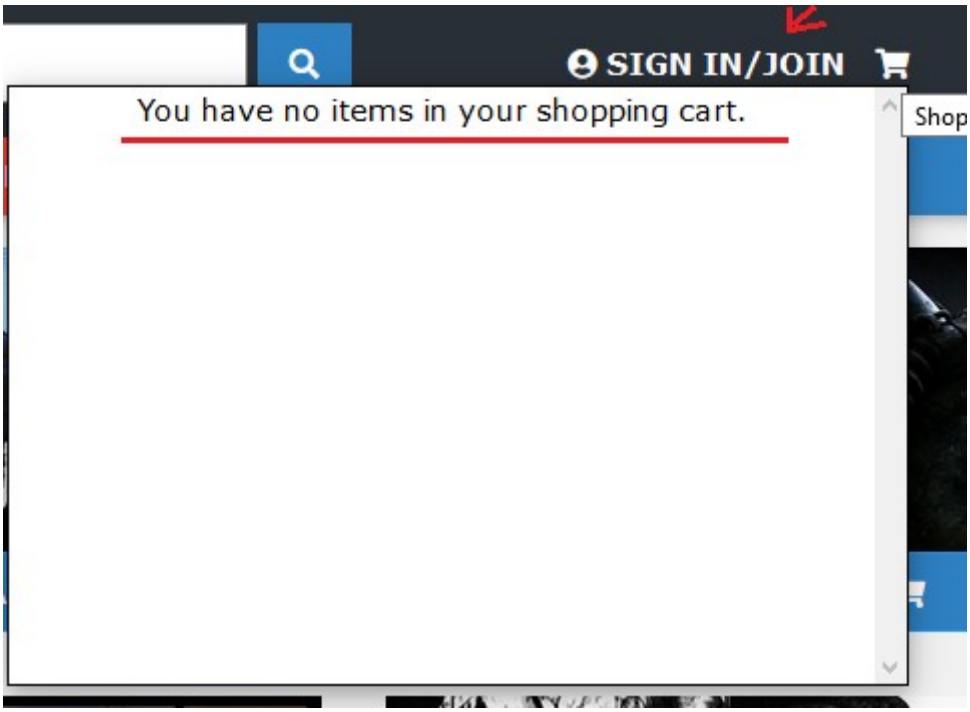


Figura 128: Resultat del tancament de sessió

5.7. Implementació de seguretat

Un cop vista l'implementació funcional en aquest document, en aquest subapartat s'explicaran les parts més rellevants de tot el desenvolupament que comporten les mesures de seguretat de *Gaming Keys*, tant les mesures que afecten directament als diferents casos d'ús que pot realitzar l'usuari com les operacions més rellevants que es realitzen en el servidor per la protecció de la botiga.

Cal esmentar que les mesures aplicades en aquest projecte no seran les més segures ni òptimes degut que es tracta d'un entorn de desenvolupament i l'objectiu ha sigut anar provant diferents conceptes.

5.7.1. Informació de sessió d'usuari

Cada cop que l'usuari inicia sessió es desen les dades que posteriorment seran utilitzades per identificar-lo durant la navegació per la botiga. Concretament es desen dos valors:

```
//ADD USER AND HASH TO SESSION
$_SESSION['login_user'] = $email;
$_SESSION['login_hash'] = hash("md5",$email."61m1n6_k3y$$");
//REDIRECT TO MAIN PAGE
header("location: index.php");
```

Figura 129: Dades que es desen a l'inciar sessió d'usuari

Primerament, es desa el correu electrònic directament i, seguidament, es desa una cadena *hash* generada en *MD5* amb el correu electrònic juntament amb un salt amb valor "61m1n6_k3y$$", que s'utilitzarà conjuntament amb el correu electrònic en text pla per identificar la sessió d'usuari.

El salt és un valor clau només conegut pel servidor que serà utilitzat posteriorment , com es podrà veure en el següent apartat, per garantir que la sessió d'usuari no ha sigut suplantada i garantint la protecció d'accés a les diferents vistes de la botiga.

Cal esmentar que per generar la cadena *hash* s'ha utilitzat l'algoritme *MD5* perquè no se li ha donat molta importància a aquest fet, degut que es tracta d'un entorn de desenvolupament. La recomanació, però, és que s'utilitzin algoritmes més segurs com pot ser *SHA-512* o més noves.

5.7.2. Control d'accés a les vistes

Com bé s'ha pogut veure en el diagrama de flux de la figura 1 d'aquest document, algunes de les vistes requereixen haver iniciat, o no, sessió d'usuari. Cada cop que

s'accedeix a alguna d'aquestes s'aplica un control per verificar l'estat de la sessió d'usuari.

Per tant, en cas que la vista requereixi que l'usuari hagi iniciat sessió prèviament, s'aplicarà el procediment següent:

```php
<?php
//IF USER IS LOGGED, REDIRECT TO MAIN PAGE
session_start();
if(!isset($_SESSION['login_user']) || !isset($_SESSION['login_hash'])){
   header("location: login_form.php");
}else{
    $salt = "61m1n6_k3y$$";
    $hashedLogin = hash("md5",$_SESSION['login_user'].$salt);
    if($hashedLogin != $_SESSION['login_hash']){
        header("location: login_form.php");
    }
}
?>
```

Figura 130: Control de sessió d'usuari identificat

El sistema comprova que si cap dels dos valors que es desen en sessió en iniciar sessió d'usuari, com s'ha vist anteriorment, tenen contingut, automàticament es mostra el formulari d'inici de sessió. En cas contrari, es genera la cadena hash amb el valor del correu electrònic en sessió i es verifica que el corresponent hash coincideix amb el que existia en sessió. Només en cas de ser cert, el sistema continua amb el procediment corresponent. En cas contrari, automàticament es mostra el formulari d'inici de sessió.

D'aquesta manera, es garanteix que només l'usuari enregistrat que hagi iniciat sessió prèviament, tingui accés a la vista corresponent.

Contràriament, en el cas invers, el sistema comprova que cap dels dos valors estan buits. De ser així, el sistema automàticament redirigeix a l'usuari cap al catàleg, mentre que en cas contrari el sistema continua amb el procediment corresponent.

```php
<?php
//IF USER IS LOGGED, REDIRECT TO MAIN PAGE
session_start();
if(isset($_SESSION['login_user']) || isset($_SESSION['login_hash'])){
   header("location: index.php");
}
?>
```

Figura 131: Control de sessió d'usuari sense identificar

5.7.3. Validacions de formularis

Cada cop que s'envia un formulari, sigui mitjançant una petició GET o una petició POST, es realitzen les validacions corresponents als valors que es reben juntament amb la petició.

Les validacions principals que es realitzen sobre un valor són: validacions de contingut, de longitud, d'igualtat i de format.

Validacions de contingut

Aquest tipus de validacions verifiquen que un valor que es requerit contingui valor. En aquest cas, tots els valors dels formularis són requerits i, per tant, tots han de tenir contingut.

```php
static function validateFieldRequired($field){
    if($field == ""){
        return false;
    }
    return true;
}
```

Figura 132: Procediment per verificar el contingut d'un valor

Validacions de longitud

Aquest tipus de validacions comproven que un valor no superi una longitud màxima, que és determinada per la longitud del camp corresponent de la taula de la base de dades. D'aquesta manera s'eviten errors corresponents a l'intent de desar informació que excedeix dels límits.

```php
static function validateFieldLength($field, $length){
    if(strlen($field) > $length){
        return false;
    }
    return true;
}
```

Figura 133: Procediment per verificar la longitud d'un valor

Validacions d'igualtat

Aquest tipus de validacions es realitzen quan dos valors han de tenir el mateix

contingut. En aquest cas, els únics dos valors del sistema que requereixen una confirmació són el correu electrònic i la contrasenya, com ja s'ha vist en apartats anteriors.

```
static function validateFieldEquality($field, $equalField){
    if($field != $equalField){
        return false;
    }
    return true;
}
```

Figura 134: Procediment per verificar l'igualtat de dos valors

Validacions de format

Aquest tipus de validacions es realitzen per controlar el format dels valors que arriben, evitant així qualsevol tipus d'atac d'injecció. Per controlar el format d'aquests s'aplica una expressió regular.

Una expressió regular és una seqüència de caràcters que formen un patró que normalment s'utilitza per cercar patrons dins d'un text. D'aquesta manera, el sistema comprova que el valor contingui el patró indicat. Un exemple molt clar és l'aplicació d'aquestes expressions per validar el correu electrònic:

```
static function validateEmailFormat($email){

    if(preg_match("/^([a-zA-Z0-9_\-\.]+)@([a-zA-Z0-9_\-\.]+)\.([a-zA-Z]{2,5})$/", $email) ==
        null){
        return false;
    }
    return true;
}
```

Figura 135: Validació de format del correu electrònic

Resumidament, les validacions de format que s'apliquen als següents valors són:

- Identificador del producte: ha de contenir només un nombre sencer. Per exemple: *0, 1, 2, 3*.
- Unitats del producte: ha de contenir només un nombre sencer i superior a 0. Per exemple: *1, 2, 3, 4*.
- Filtres del producte (plataforma i categoria): ha de contenir només lletres. Per exemple: *plataforma, categoria, MMO, Steam*.
- Nom del producte: pot contenir lletres, nombres i espais. Per exemple:

Fallout78, Fallout 78, Producte7 Fallout8.

- Nom de l'usuari: pot ser un nom simple o compost, només format per lletres i, en casos especials, pot haver-hi apòstrofs. Per exemple: *Joan, Juan José, D'jongo García*.

- Cognom d'usuari: pot ser només un cognom simple i, com en el cas anterior, només format per lletres i, si és el cas, pot haver-hi apòstrofs. Per exemple: *García, D'jongo, Mbappé*.

- Correu electrònic. Ha d'estar format per tres parts: nom, domini i extensió. Tant la primera com la segona part poden estar formades per nombres i lletres i, en alguns casos, pels caràcters especials '_', '-' o '.'.

  Entre el nom i el domini ha d'existir el caràcter '@' que s'utilitza per indicar el final d'un valor i l'inici de l'altre. Finalment entre la segona i la tercera part s'han de separar per un punt, on l'última part només pot contenir lletres i tenir una longitud entre 2 i 5 caràcters.

  Per exemple: *ser@fmail.es*, *ser_90@hmail.commm*, *ser90@Gmail.COM*

- Contrasenya: aquesta validació és la més complexa de totes. El format que ha de contenir es pot veure en les recomanacions que s'indiquen a l'usuari a l'hora d'establir una contrasenya, com es pot veure en la figura 85 en l'apartat d'implementacions funcionals de creació de comptes d'usuari. Per exemple: *Gamin@Keys10*, *Gaming@Eys1!*.

- Codi d'activació o recuperació: només ha de contenir lletres minúscules i nombres. Per exemple: *5c21ea542a62f, 5c21f9de8fdd9*.

En cas que qualsevol d'aquestes validacions no es compleixin, el sistema no continua amb el procediment i, en cas de necessitat, mostra els missatges d'error corresponents. Per exemple, si es realitza la següent cerca en el catàleg:

Figura 136: Cerca malintencionada de productes amb sessió iniciada

El sistema comprova que el nom del producte que es desitja cercar no té el format adient i mostra a l'usuari el missatge corresponent indicant que no existeix cap coincidència:

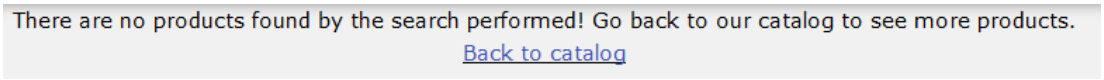


Figura 137: Resultat de la cerca malintencionada

En cas que arribés qualsevol altre valor que no està contemplat com a possible entrada en el sistema, aquest simplement l'obviarà i seguirà amb el corresponent procediment.

5.7.4. Compte d'usuari

En el apartat d'implementacions funcionals s'ha vist els aspectes funcionals més rellevants de la gestió d'usuaris que realitza el sistema.

En els següents subapartats es presenten les mesures de seguretat més rellevants aplicades per la protecció i verificació dels comptes d'usuari.

5.7.4.1. Bloqueig

En la botiga, un compte d'usuari pot ser bloquejat per seguretat segons els intents fallits de les operacions més rellevants en la gestió d'usuaris, com són l'inici de sessió, l'activació del compte i la recuperació de la contrasenya.

Aquesta mesura de seguretat és aplicada degut que existeix persones malintencionades que, mitjançant atacs de força bruta o mitjançant diccionaris, intenten aconseguir l'accés en qualsevol sistema.

Per tant, el sistema va comptant els intents fallits que ha realitzat l'usuari i, en cas de ser superior a 5 vegades, bloqueja completament les operacions que es puguin realitzar amb aquell correu electrònic fins que es posi en contacte amb la botiga.

Iniciar sessió d'usuari

Aquest cas és el més simple. En aquest cas, la persona malintencionada pot realitzar intents fallits intentant esbrinar la contrasenya associada al correu electrònic de la víctima. Per tant, el sistema un cop determina que ha fallat massa vegades, bloqueja a l'usuari corresponent.

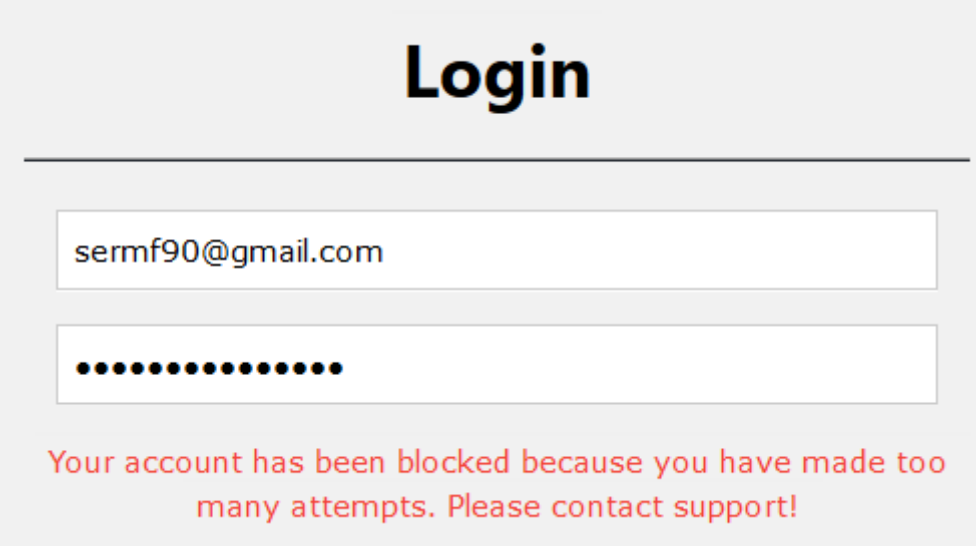


Figura 138: Bloqueig de compte d'usuari a l'iniciar sessió

Activar compte d'usuari

En aquest cas, la persona malintencionada pot haver utilitzat, per exemple, un compte temporal de correu electrònic per enregistrar-se en el sistema. D'aquesta manera, pot realitzar intents fallits intentant esbrinar el codi d'activació i aconseguir tenir usuari en la botiga amb un correu electrònic no existent.

No obstant això, el sistema un cop determina que ha fallat massa vegades, bloqueja l'activació del compte.

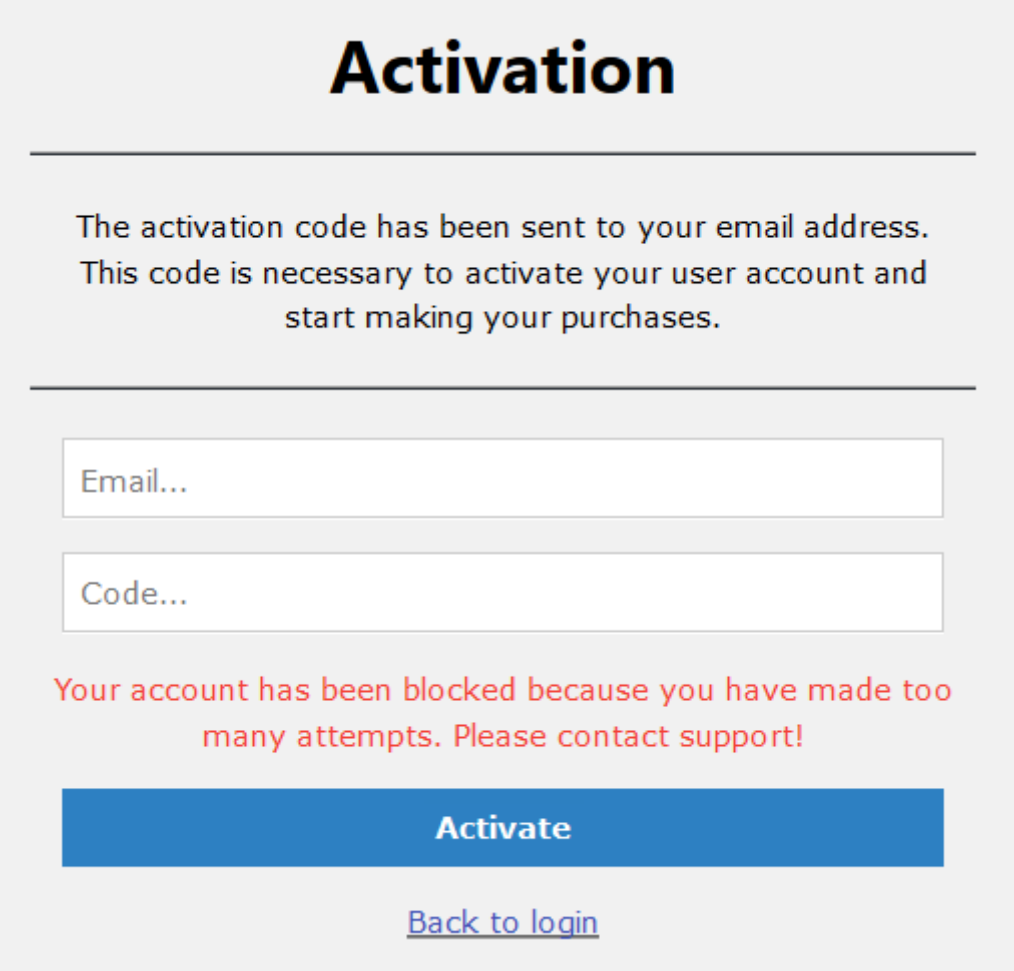


Figura 139: Bloqueig de compte d'usuari en l'activació

Recuperar compte d'usuari

Semblant al cas anterior, una persona malintencionada pot saber de l'existència d'un correu electrònic desat en el sistema i intentar recuperar la seva contrasenya. D'aquesta manera, pot realitzar intents fallits intentant esbrinar el codi de recuperació per aconseguir establir la contrasenya desitjada i obtenir el compte d'usuari per sempre.

No obstant això, el sistema un cop determina que ha fallat massa vegades, bloqueja la recuperació del compte.

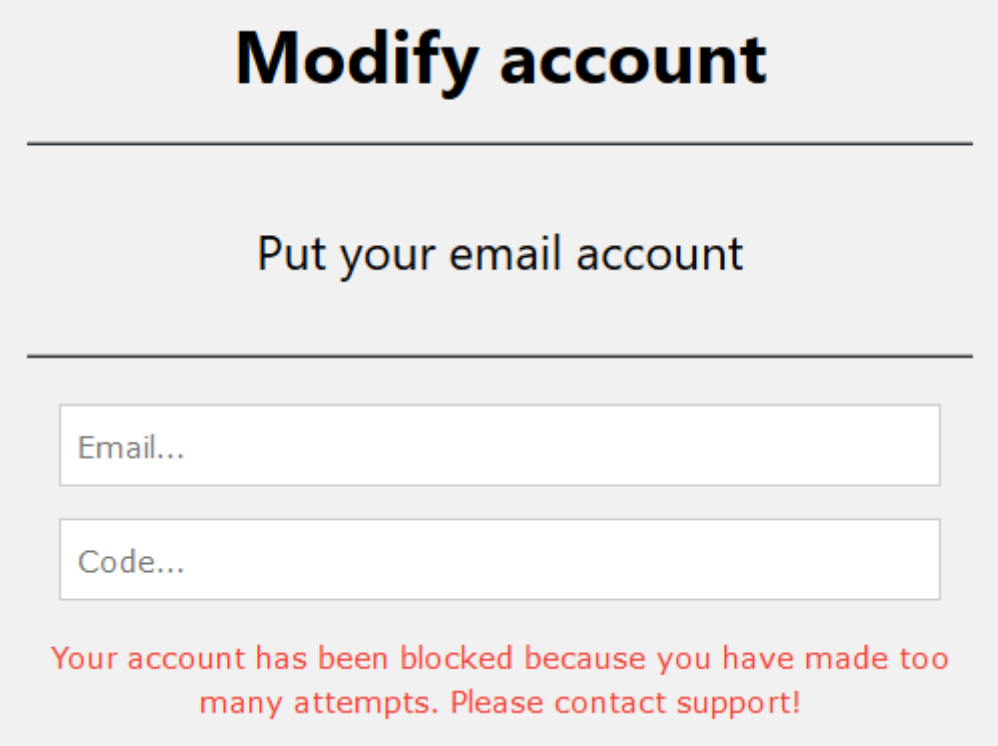


Figura 140: Bloqueig de compte d'usuari en la recuperació

Cal esmentar, que aquest la gestió que comporta el bloqueig permanent d'un usuari podria haver sigut implementat d'una manera més automatitzada com, per exemple, afegint un bloqueig temporal o avisant per correu electrònic a l'usuari dels intents realitzats en el seu compte, però no entrava dins de l'abast del projecte.

Per tant, s'ha aplicat la solució més simple per afegir, conceptualment, aquesta mesura de seguretat.

5.7.4.2 Codi de verificació

Per assegurar que l'usuari correspon a l'adreça del correu electrònic que especifica en el formulari de creació del compte d'usuari, el sistema envia un correu amb un codi únic a l'adreça corresponent que l'usuari haurà d'utilitzar en el següent formulari, tant en l'activació del compte com en la seva recuperació.

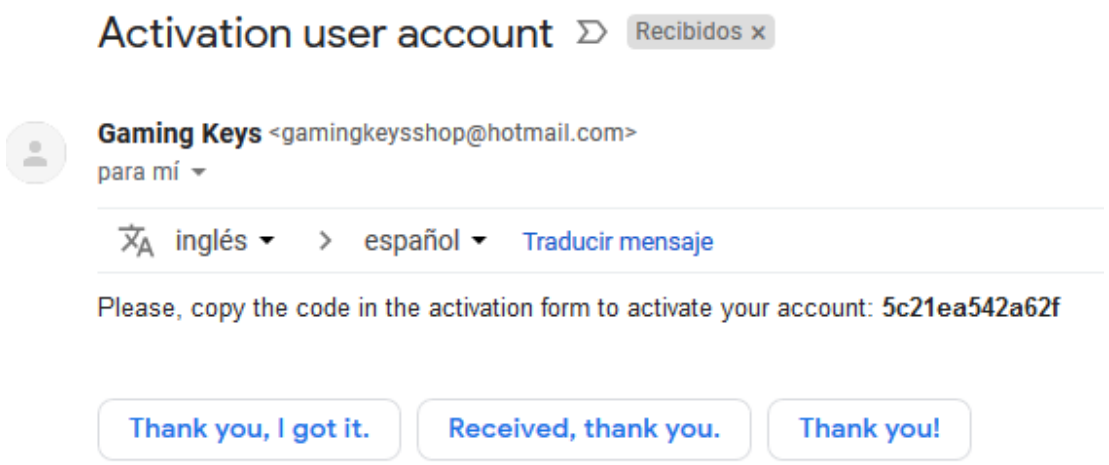


Figura 141: Exemple de codi de verificació

Aquesta mesura de seguretat és aplicada per dificultar l'activació o la recuperació del compte d'usuari a la persona malintencionada, fent que, per aconseguir els seus propòsits, hagi de tenir accés al compte de correu electrònic esmentat.

Aquest codi de verificació és un codi únic, format per un conjunt de lletres i números, d'una mida mínimament llarga com perquè el temps sigui un impediment a l'hora d'intentar esbrinar-la. Aquest és generat amb el mètode de PHP següent:

```
//GENERATE ACTIVATION CODE
$activationCode = uniqid();
```

Figura 142: Mètode que genera un codi únic de verificació

D'aquesta manera, juntament amb el cas anterior, s'aconsegueix la protecció ideal contra els atacs de força bruta.

5.7.5. Correu electrònic

És molt important que l'enviament de dades mitjançant correus electrònics des de la botiga cap a l'usuari es faci d'una manera segura, per protegir l'accés a l'informació en cas que un espia estigués analitzant la informació que viatja en la xarxa. Per tant, cal implementar les opcions adients perquè la transmissió d'informació sigui segura.

```php
$shopEmail = "gamingkeysshop@hotmail.com";
$shopEmailPass = "Gaming_Keys_99";
$shopName = "Gaming Keys";

$mail = new PHPMailer\PHPMailer\PHPMailer(TRUE);
try {
    $mail->IsSMTP();
    $mail->CharSet = 'UTF-8';
    $mail->Host = "smtp.live.com";
    $mail->SMTPAuth= true;
    $mail->Port = 587;
    $mail->Username= $shopEmail;
    $mail->Password= $shopEmailPass;
    $mail->SMTPSecure = 'tls';
    $mail->From = $shopEmail;
    $mail->FromName= $shopName;
    $mail->isHTML(true);
    $mail->Subject = 'Activation user account';
    $mail->Body = '<span>Please, copy the code in the activation form to activate your
        account: <strong>'.$code.'</strong></span>';
    $mail->addAddress($email);
```

Figura 143: Exemple de característiques en correu electrònic

Entre les configuracions més bàsiques de seguretat, es troba l'especificació del servidor de correus juntament amb el port, les credencials i l'especificació del TLS[10].

Servidor de correu electrònic

Fa referència a un dels servidors de correus electrònics de Outlook, el qual permet enviar i rebre correus electrònics de manera segura. Aquest servidor, entre altres mesures aplicades en el seu sistema, permet la transmissió d'informació de manera segura, analitzant el contingut del correu i verificant que el seu origen sigui d'entorns confiables, evitant així la transmissió d'informació no segura o no desitjada.

Per tant, és possible que realitzant proves en un entorn de desenvolupament, el servidor acabi per denegar la transmissió de més correus electrònics fins a realitzar la verificació corresponent mitjançant l'enviament d'un codi numèric al telèfon mòbil. Un cop verificada, es pot tornar a realitzar les proves corresponents.

Credencials

Són els valors que corresponen a l'identificació del correu electrònic de la botiga. En aquest cas, l'usuari i la contrasenya estan desats en el mateix fitxer, però s'hagués pogut implementar una solució per no deixar en text pla aquests valors.

Transport Layer Security

Transport Layer Security tenen per objectiu la seguretat de les transaccions en

línia xifrant la informació corresponent fent que les dades interceptades per tercers, com per exemple dades bancàries, no es puguin desxifrar.

Per tant, aplicant aquesta implementació es pot garantir que les dades que arriben a l'usuari ho fan de manera segura.

5.7.6. Pagament

El procés de pagament és un dels més importants en la botiga. En aquest procés és on es transferirà els diners de l'usuari cap al compte de *Gaming Keys*. Per tant, cal aplicar les mesures necessàries per realitzar aquesta operació d'una manera segura i implementar les validacions adients per comprovar la seva validesa.

Com s'ha vist en la implementació funcional, els productes del carret de la compra juntament amb l'import total d'aquesta es desen en la sessió. Aleshores, prèviament al procés de pagament, el sistema recupera els identificadors dels productes i els cerca en la base de dades per extreure la informació, incloent el preu de l'article. Seguidament, de no haver-hi cap error, estableix l'import final amb la suma de tots els preus corresponents als productes. D'aquesta manera s'evita que, en algún moment, un usuari malintencionat hagi modificat els preus en el client pels seus interessos.

```
//FIND CART PRODUCTS IN DATABASE
$products = $_SESSION["shopping-cart"];
$countProducts = count($products);
for($i=0; $i<$countProducts; $i++){
    $error = Validation::validateProductIdentifier($products[$i]["id"]);
    if($error == ""){
        $error = Validation::validateProductQuantity($products[$i]["quantity"]);
        if($error == ""){
            //SEARCH PRODUCT BY ID
            $sql = "SELECT * FROM Product WHERE id='".$products[$i]["id"]."'";
            $result = mysqli_query($conn,$sql);
            $row = mysqli_fetch_array($result,MYSQLI_ASSOC);
            $count = mysqli_num_rows($result);
            //IF PRODUCT EXISTS, PLUS PRICE TO TOTAL
            if($count == 1){
                $productPrice += $products[$i]["quantity"]*$row["price"];
            }else{
                //IF PRODUCT DOESN'T EXIST, SHOW ERROR
                $error = "There was a problem during the payment process. Please,
                    try it later or contact to our support.";
                break;
            }
```

Figura 144: Procediment per calcular l'import final de la compra

Existeixen diferents integracions amb la passarel·la de PayPal però, en aquest cas, s'ha implementat la més simple. Per tant, l'aplicació de les mesures de seguretat

recauen directament a PayPal, que ofereix un sistema segur perquè el client pugui realitzar el pagament totalment protegit.

No obstant això, el sistema ha de comprovar que la transacció en la passarel·la externa ha anat correctament i que s'ha rebut els diners. Per tant, el sistema comprova que l'estat de la transacció té valor "completed" i, a més a més, que el preu que s'ha cobrat sigui el mateix que el preu total del carret de la compra, per evitar que s'hagi modificat l'import pel camí.

```
//IF PAYMENT SUCCESS AND THE PRICE HAS NOT BEEN MODIFIED
if($payment_status == 'Completed' && $orderPrice == $payment_gross){
    $confirmation = "The payment has been done correctly. Please, check
        your keys in your purchases history or go back to our catalog
        to see more products.";
```

Figura 145: Verificació de la transacció

Com es pot veure, no s'han implementat gaires comprovacions degut que es tracta d'una integració senzilla. En cas d'una integració completa, i segons les necessitats del sistema, s'haurien d'aplicar més mesures analitzant els resultats de les transaccions, com per exemple que el compte del client no tingui diners, que s'intenti fer devolucions, etc.

5.7.7. Protecció de visibilitat de claus de productes

Una altra mesura de seguretat aplicada al sistema, no menys important, és la protecció en la visualització de claus en l'històric de compres.

S'ha vist anteriorment que quan l'usuari accedeix a la vista de l'històric, es mostra el llistat de productes però no les claus corresponents, com es pot veure en la figura 124 d'aquest document.

Aquesta mesura és degut perquè, en cas que l'usuari no estigui sol en el moment de la visualització, ja sigui perquè està en un lloc públic o simplement té gent al voltant, una persona malintencionadament pot copiar les claus i activar els productes en les plataformes corresponents, fent que l'usuari ja no pogués activar el producte, degut que es tracten de claus úniques.

Per tant, l'usuari disposa de la icona de l'ull, com es pot veure en la figura 125, per mostrar o amagar quan vulgui les claus dels productes.

6. Proves

En aquesta secció del document es realitzaran les proves de concepte de la implementació de *Gaming Keys*. En els següents subapartats s'exposaran els casos més rellevants i les respostes que ofereix el sistema mitjançant captures, mostrant així tots els resultats del seu funcionament i la seguretat implementada.

6.1. Catàleg

Cas 1: visualització del catàleg des de la pàgina principal

Resultat: visualització del catàleg

Cas 2: visualització del catàleg fent clic al logo de la botiga

Resultat: visualització del catàleg

Cas 3: visualització del catàleg fent clic al logo de la botiga des de qualsevol altra vista

Resultat: visualització del catàleg

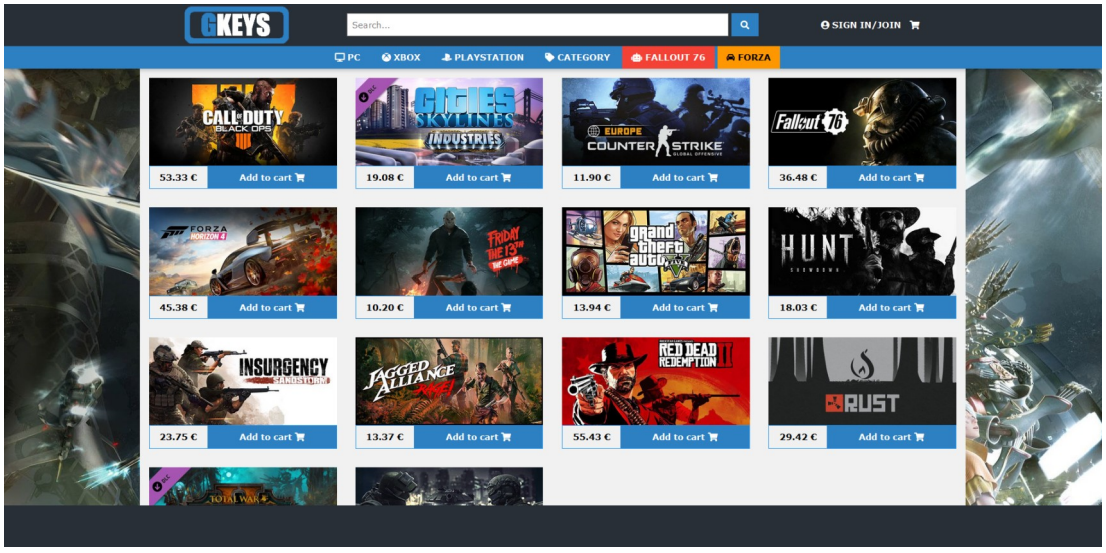


Figura 146: Resultat de les proves del catàleg

6.2. Cerca de productes

Cas 1: cerca per text de productes existents

Entrada: *Ge*

Resultat: visualització del catàleg amb els productes corresponents

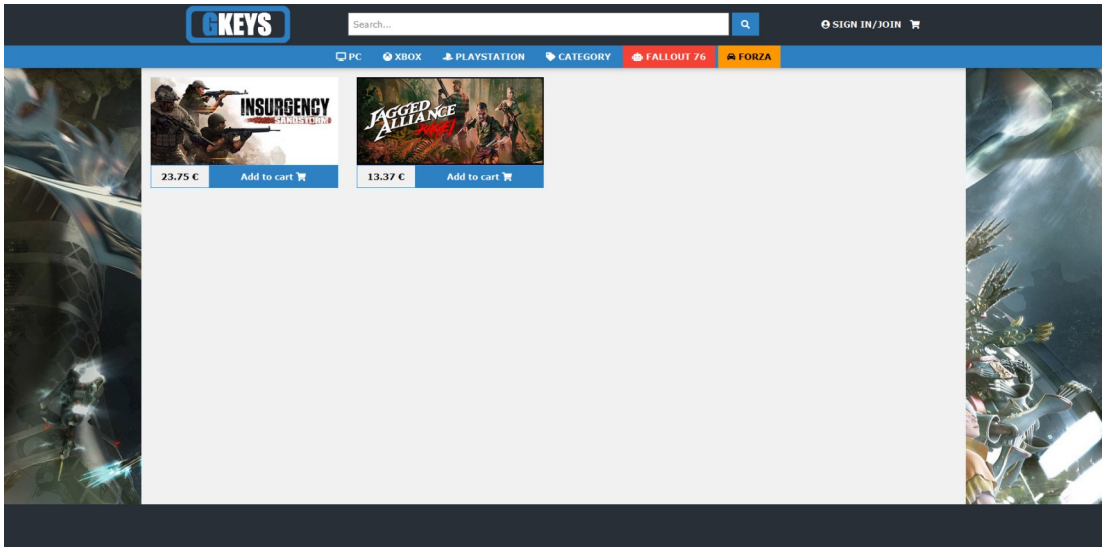


Figura 147: Resultat del cas 1 de la cerca per text de productes

Cas 2: cerca per text de productes no existents

Entrada: *Fat 66*

Resultat: visualització del missatge d'error corresponent

Cas 3: cerca per text de productes (XSS)

Entrada: *<script>alert("XSS injection!")</script>*

Resultat: visualització del missatge d'error corresponent

Cas 4: cerca per text de productes (SQL Injection)

Entrada: *OR 1=1;--*

Resultat: visualització del missatge d'error corresponent

Cas 5: cerca per plataforma de productes existents

Entrada: *Steam*

Resultat: visualització del catàleg amb els productes corresponents

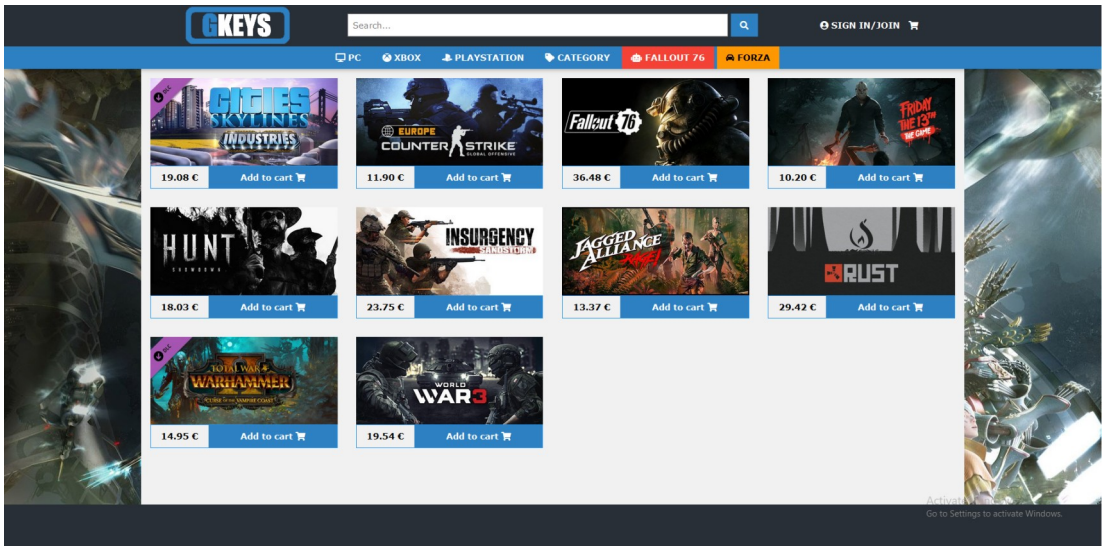


Figura 148: Resultat del cas 5 de la cerca per plataforma de productes

Cas 6: cerca per plataforma de productes no existents

Entrada: *Ge*

Resultat: visualització del missatge d'error corresponent

Cas 7: cerca per plataforma de productes (XSS)

Entrada: *<script>alert("XSS injection!")</script>*

Resultat: visualització del missatge d'error corresponent

Cas 8: cerca per plataforma de productes (SQL Injection)

Entrada: *OR 1=1;--*

Resultat: visualització del missatge d'error corresponent

Cas 9: cerca per categoria de productes existents

Entrada: *Action*

Resultat: visualització del catàleg amb els productes corresponents

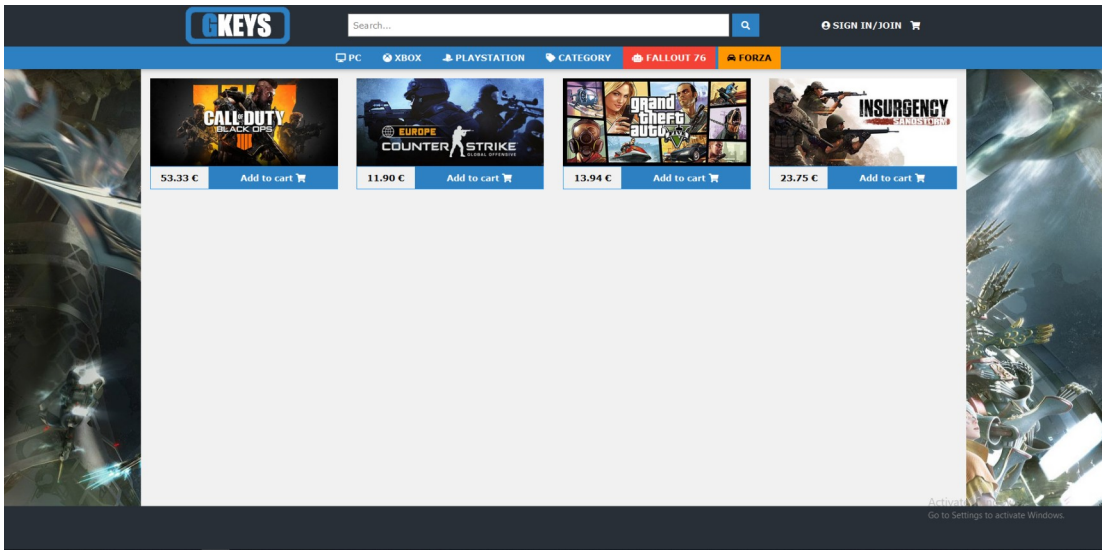


Figura 149: Resultat del cas 9 de la cerca per categoria de productes

Cas 10: cerca per categoria de productes no existents

Entrada: *Ge*

Resultat: visualització del missatge d'error corresponent

Cas 11: cerca per categoria de productes (XSS)

Entrada: *<script>alert("XSS injection!")</script>*

Resultat: visualització del missatge d'error corresponent

Cas 12: cerca per categoria de productes (SQL Injection)

Entrada: *OR 1=1;--*

Resultat: visualització del missatge d'error corresponent

Cas 13: cerca per identificador del producte existent

Entrada: *4*

Resultat: visualització del catàleg amb el producte corresponent

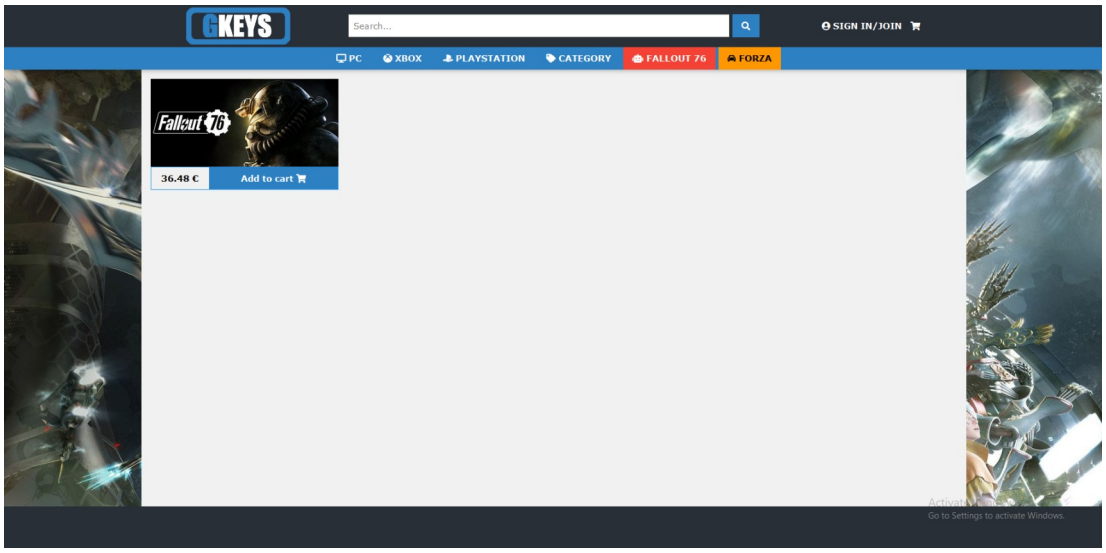


Figura 150: Resultat del cas 13 de la cerca per identificador de producte

Cas 14: cerca per identificador de producte no existent

Entrada: *X*

Resultat: visualització del missatge d'error corresponent

Cas 15: cerca per producte destacat (XSS)

Entrada: *<script>alert("XSS injection!")</script>*

Resultat: visualització del missatge d'error corresponent

Cas 16: cerca per producte destacat (SQL Injection)

Entrada: *OR 1=1;--*

Resultat: visualització del missatge d'error corresponent

Cas 17: cerca per qualsevol altre valor GET

Entrada: *camp 'camp' amb valor 'valor'*

Resultat: visualització del catàleg, com la figura 146, del subapartat anterior.

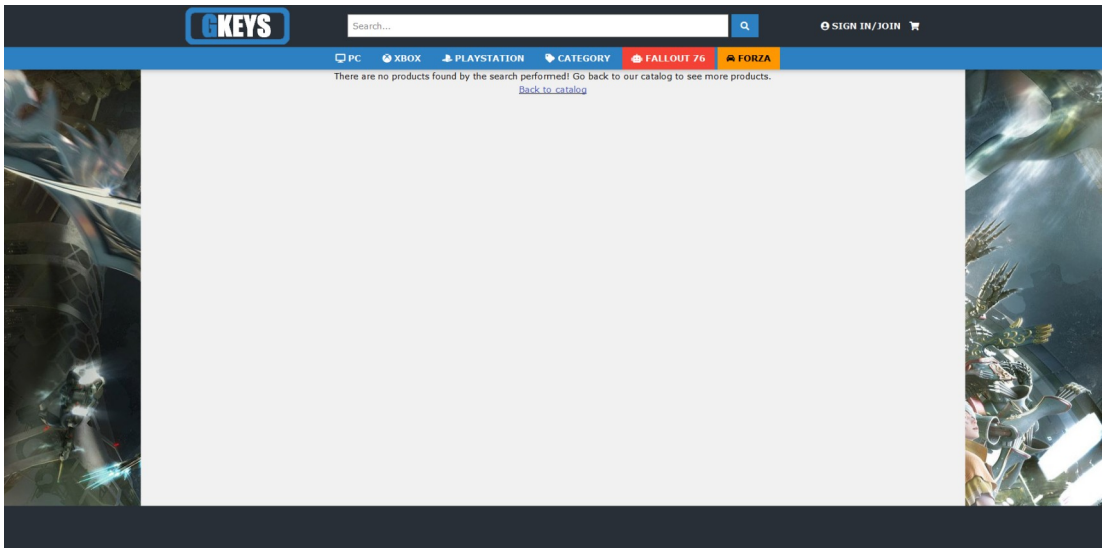


Figura 151: Resultat dels casos de la cerca de productes amb missatge d'error

6.3. Visualització del producte

Cas 1: visualització del producte existent

Entrada: *4*

Resultat: visualització del producte corresponent

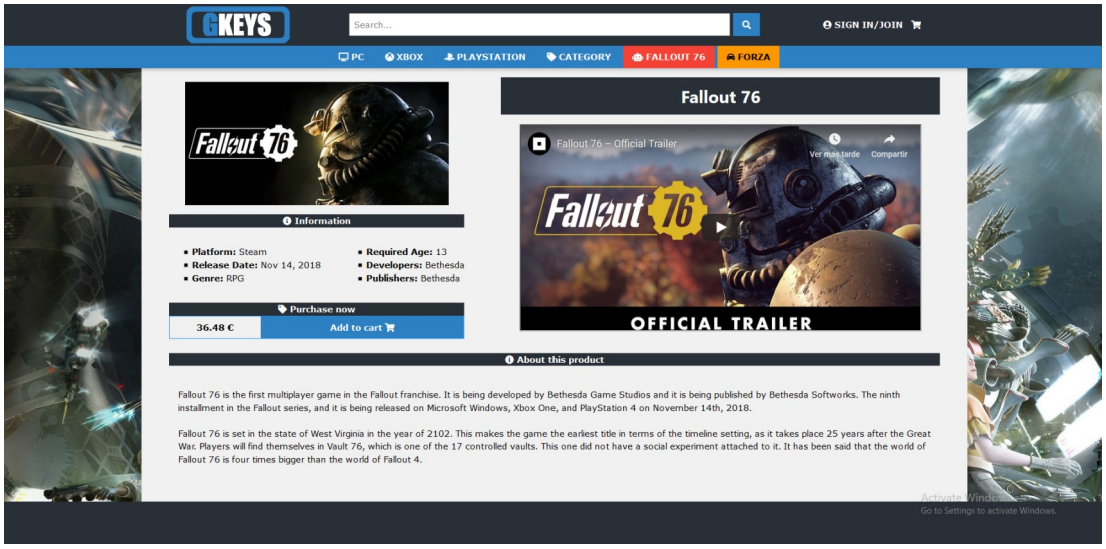


Figura 152: Resultat del cas 1 de la visualització del producte

Cas 2: visualització del producte no existent

Entrada: *X*

Resultat: visualització del missatge d'error corresponent

Cas 3: visualització del producte (XSS)

Entrada: *<script>alert("XSS injection!")</script>*

Resultat: visualització del missatge d'error corresponent

Cas 4: visualització del producte (SQL Injection)

Entrada: *OR 1=1;--*

Resultat: visualització del missatge d'error corresponent

Cas 5: visualització del producte per qualsevol altre valor GET

Entrada: *camp 'camp' amb valor 'valor'*

Resultat: visualització del missatge d'error corresponent

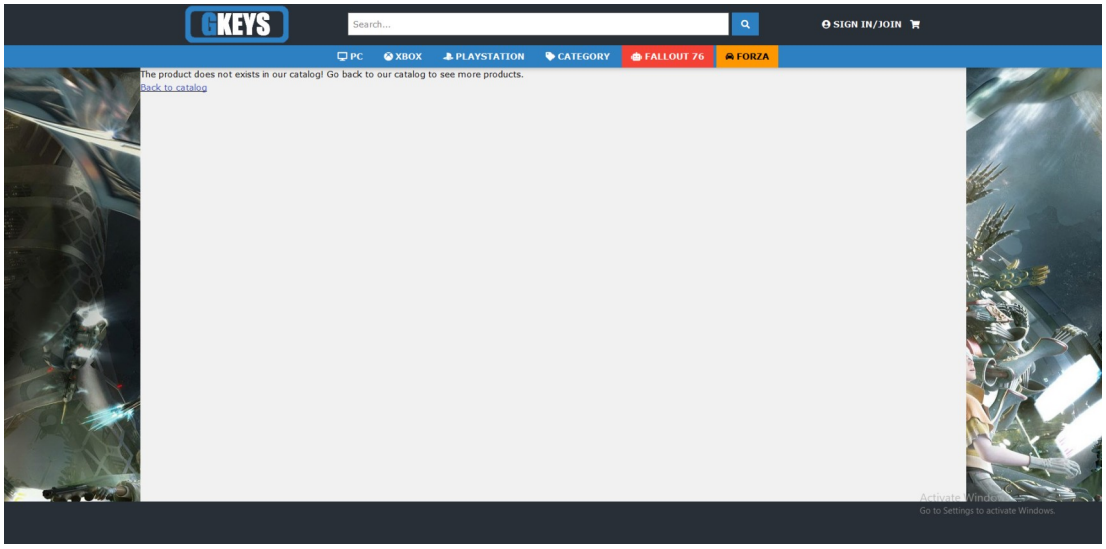


Figura 153: Resultat dels casos de la visualització del producte amb missatge d'error

6.4. Visualització del carret

Cas 1: visualització del carret sense productes afegits

Resultat: visualització del contingut del carret amb el missatge corresponent

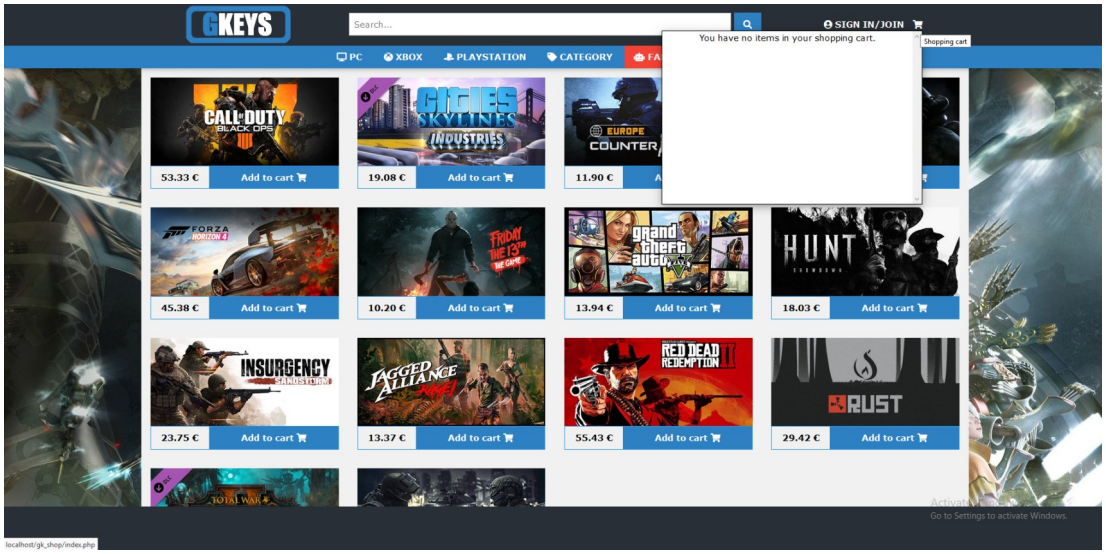


Figura 154: Resultat del cas 1 de la visualització del carret

Cas 2: visualització del carret amb productes afegits

Resultat: visualització dels productes afegits al carret

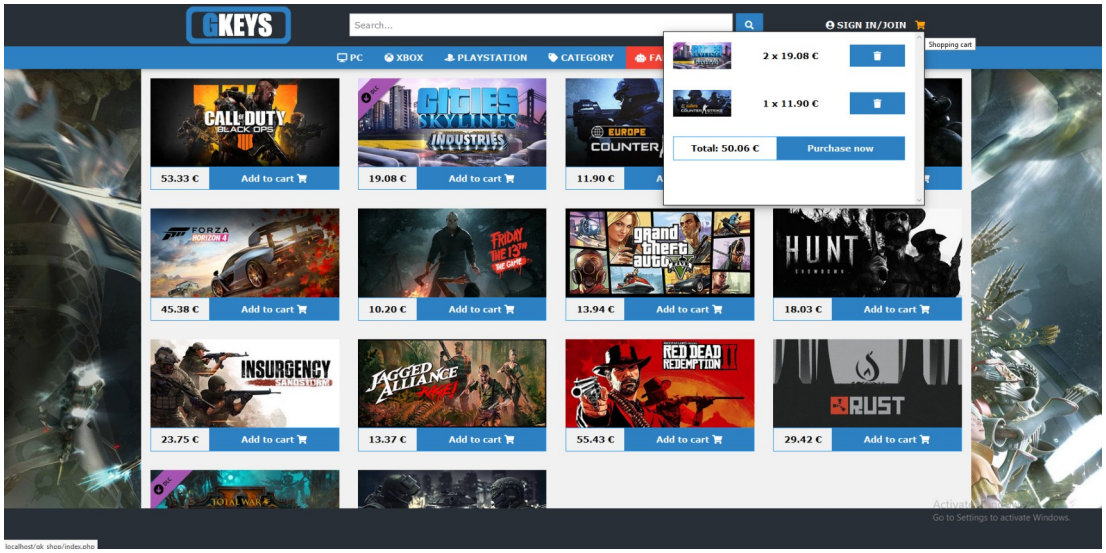


Figura 155: Resultat del cas 2 de la visualització del carret

6.5. Afegir productes

Cas 1: afegir productes existents des del catàleg

Entrada: *1*

Resultat: visualització del producte afegit en el carret

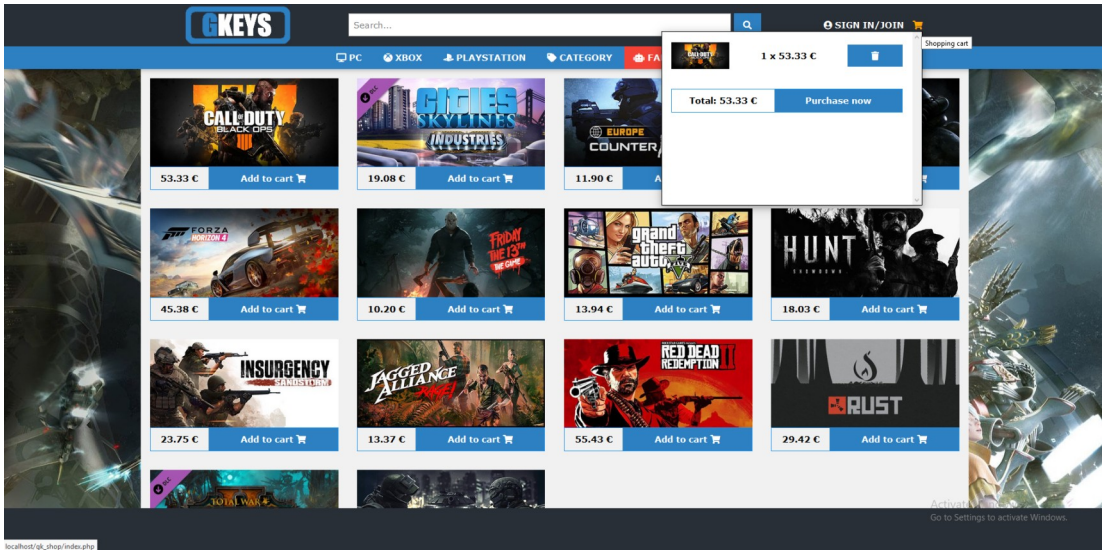


Figura 156: Resultat del cas 1 d'afegir productes

Cas 2: afegir productes existents des de la visualització del producte

Entrada: *2*

Resultat: visualització del producte afegit en el carret i redirecció cap al catàleg

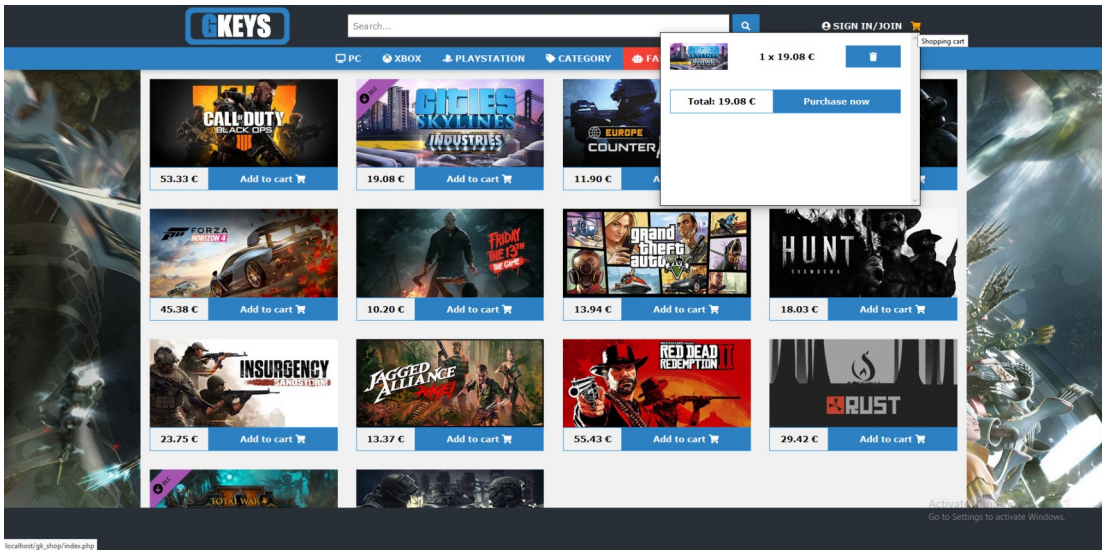


Figura 157: Resultat del cas 2 d'afegir productes

Cas 3: afegir productes ja existents en el carret

Entrada: *1*

Resultat: visualització de l'increment d'una unitat del producte corresponent

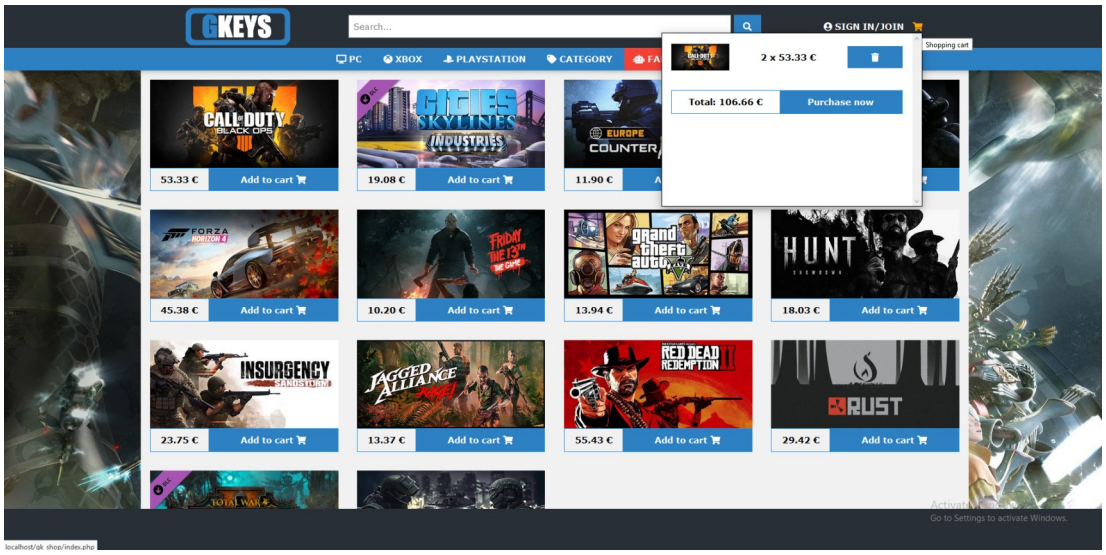


Figura 158: Resultat del cas 3 d'afegir productes

Cas 4: afegir productes no existents des del catàleg

Entrada: *X*

Resultat: visualització del missatge d'error corresponent, com la figura 151 d'apartats anteriors.

Cas 5: afegir productes no existents des de la visualització del producte

Entrada: *X*

Resultat: visualització del missatge d'error corresponent, com la figura 153 d'apartats anteriors.

Cas 6: afegir productes malintencionadament (XSS)

Entrada: *<script>alert("XSS injection!")</script>*

Resultat: visualització del missatge d'error corresponent, com les figures 151 o 153 d'apartats anteriors.

Cas 7: afegir productes malintencionadament (SQL Injection)

Entrada: *OR 1=1;--*

Resultat: visualització del missatge d'error corresponent, com les figures 151 o 153 d'apartats anteriors.

Cas 8: afegir productes des d'una altra vista

Entrada: *1*

Resultat: el sistema ignora les peticions

6.6. Eliminar productes

Cas 1: eliminar productes existents en el carret des del catàleg

Entrada: *1*

Resultat: visualització de la desaparició del producte eliminat en el carret

Cas 2: eliminar productes existents en el carret des de la visualització del producte

Entrada: *2*

Resultat: visualització de la desaparició del producte eliminat en el carret

Cas 3: eliminar productes existents en el carret amb unitats superiors a 1

Entrada: *2*

Resultat: visualització de la resta de l'unitat del producte corresponent

Cas 4: eliminar productes no existents en el carret des del catàleg

Entrada: *X*

Resultat: visualització del missatge d'error corresponent, com la figura 151 d'apartats anteriors.

Cas 5: eliminar productes no existents en el carret des de la visualització del producte

Entrada: *X*

Resultat: visualització del missatge d'error corresponent, com la figura 153 d'apartats anteriors.

Cas 6: eliminar productes malintencionadament (XSS)

Entrada: *<script>alert("XSS injection!")</script>*

Resultat: visualització del missatge d'error corresponent, com les figures 151 o 153 d'apartats anteriors.

Cas 7: eliminar productes malintencionadament (SQL Injection)

Entrada: *OR 1=1;--*

Resultat: visualització del missatge d'error corresponent, com les figures 151 o 153 d'apartats anteriors.

Cas 8: eliminar productes des d'una altra vista

Entrada: *1*

Resultat: el sistema ignora les peticions

6.7. Resum de la compra

Cas 1: visualització del resum de compra amb productes

Resultat: visualització del contingut del resum de la compra

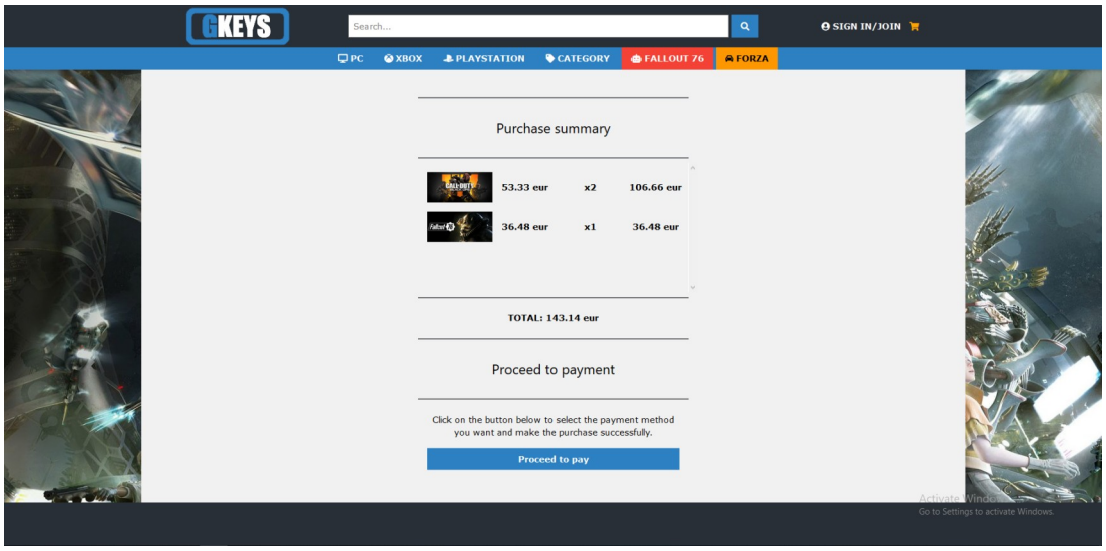


Figura 159: Resultat del cas 1 del resum de la compra

Cas 2: visualització del resum de compra sense productes

Resultat: visualització del missatge d'error corresponent

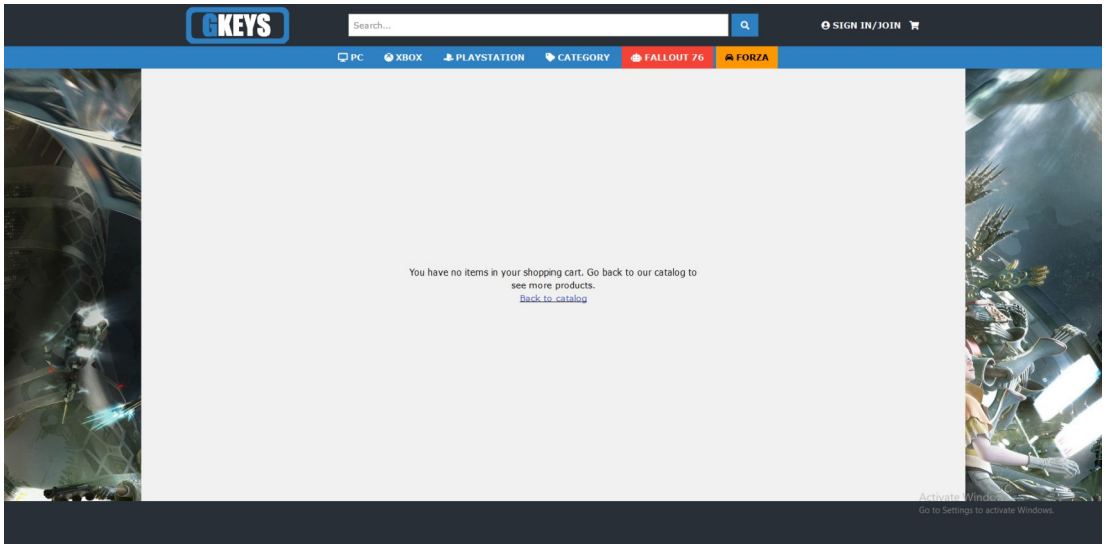


Figura 160: Resultat del cas 2 del resum de la compra

6.8. Creació de compte d'usuari

Cas 1: creació de compte d'usuari amb dades vàlides

Entrada: *Sergio, Martos, sermf90@gmail.com, sermf90@gmail.com, Keys@Gaming10, Keys@Gaming10* respectivament a cada camp

Resultat: enviament del codi d'activació al correu especificat i redirecció cap al formulari d'activació

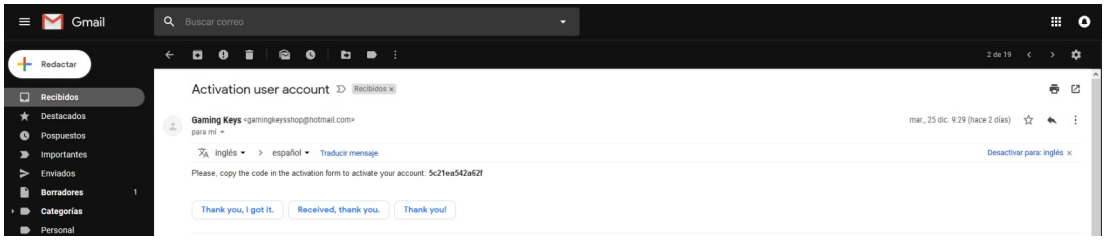


Figura 161: Resultat del cas 1 de la creació del compte d'usuari (correu electrònic)

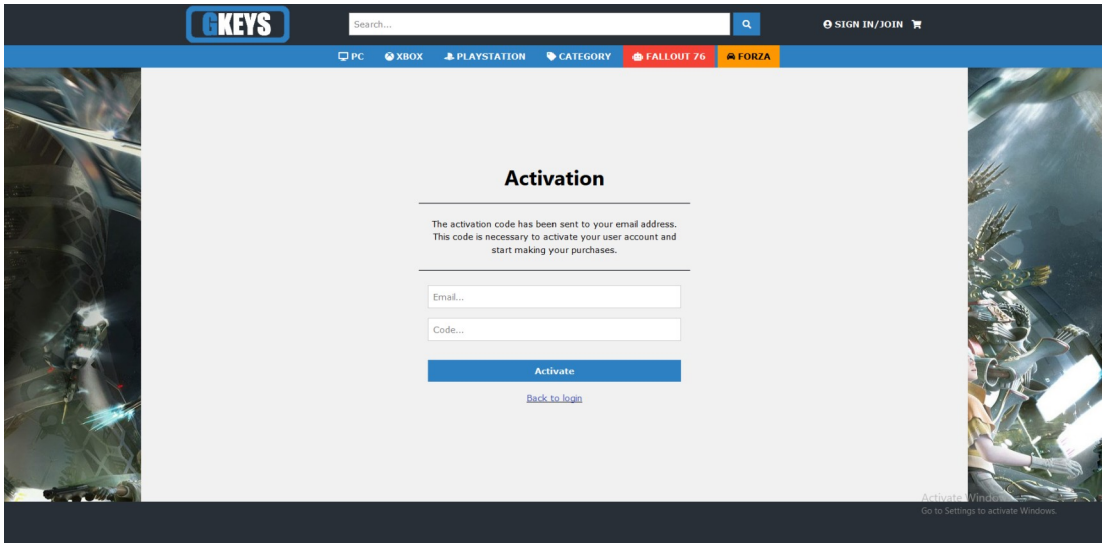


Figura 162: Resultat del cas 1 de la creació del compte d'usuari (redirecció)

Cas 2: creació de compte d'usuari amb dades no vàlides

Entrada: *S_ergio, M_artos, sermf90, sermf90, xxxx, xxxx* respectivament a cada camp

Resultat: visualització del missatge d'error corresponent

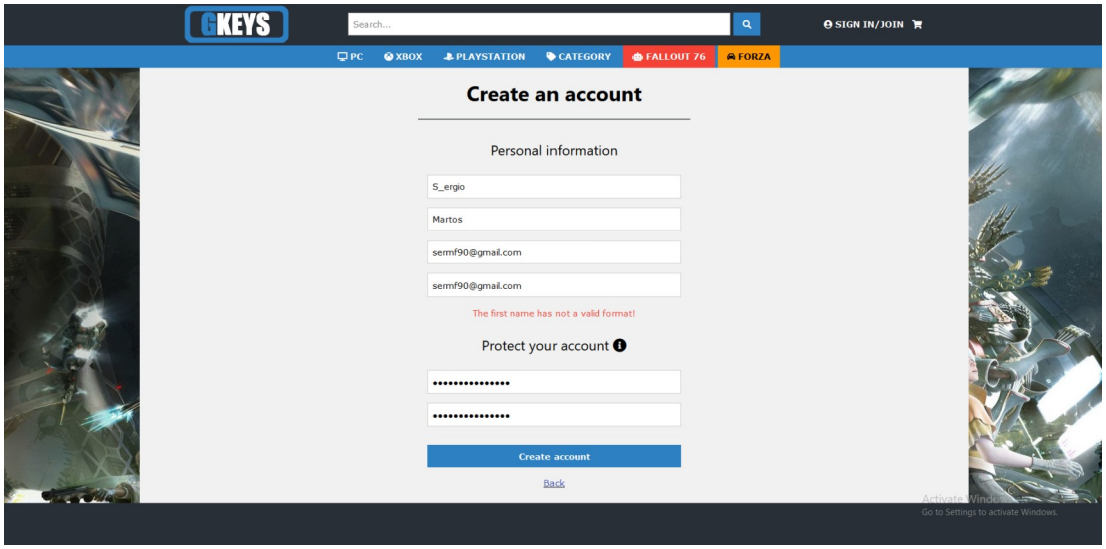


Figura 163: Resultat del cas 2 de la creació del compte d'usuari (nom no vàlid)

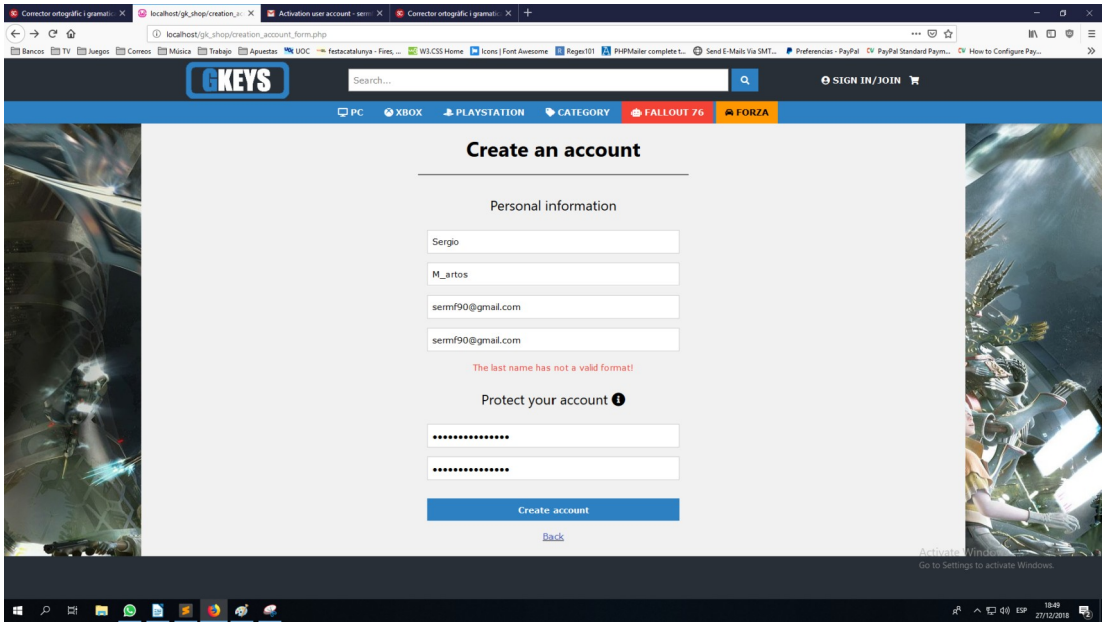


Figura 164: Resultat del cas 2 de la creació del compte d'usuari (cognom no vàlid)

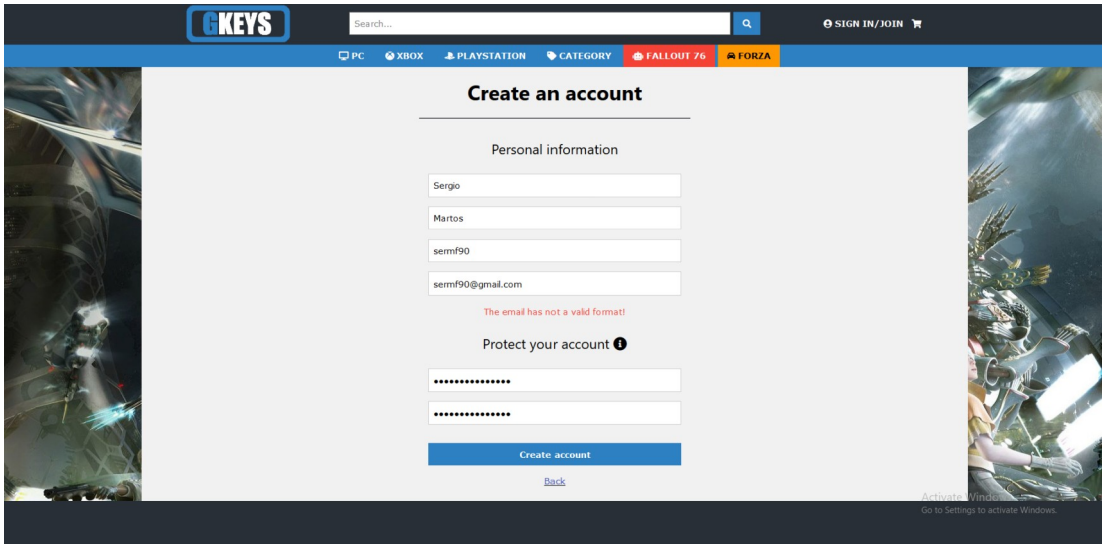


Figura 165: Resultat del cas 2 de la creació del compte d'usuari (correu no vàlid)

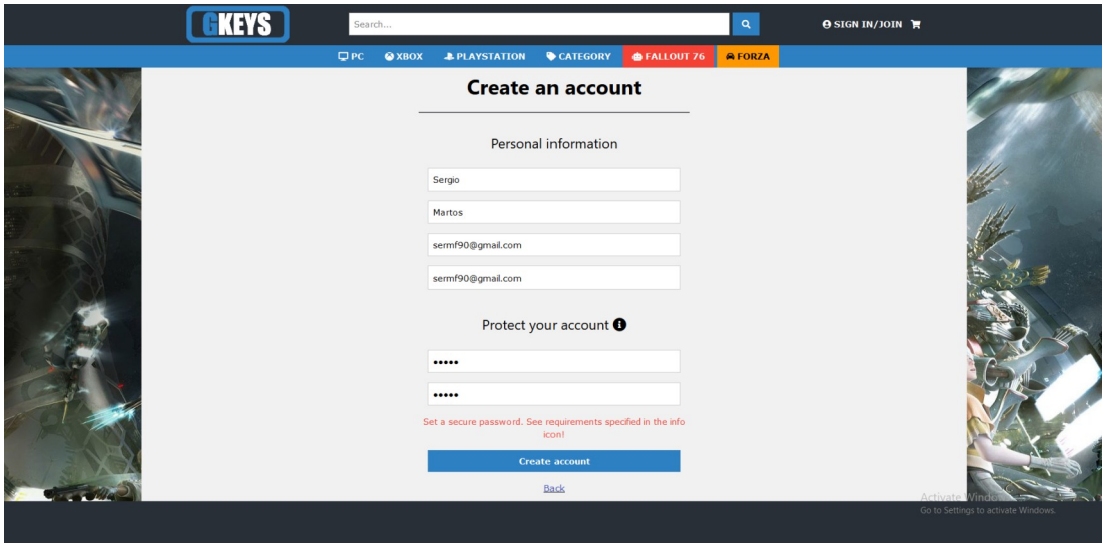


Figura 166: Resultat del cas 2 de la creació del compte d'usuari (contrasenya no vàlida)

Cas 3: creació de compte d'usuari amb confirmacions de correu i contrasenya diferents als corresponents

Entrada: *sermf90@gmail.com*, *sermf90@hotmail.es*, *Keys@Gaming10*, *Keys@10Gaming* respectivament a cada camp

Resultat: visualització del missatge d'error corresponent

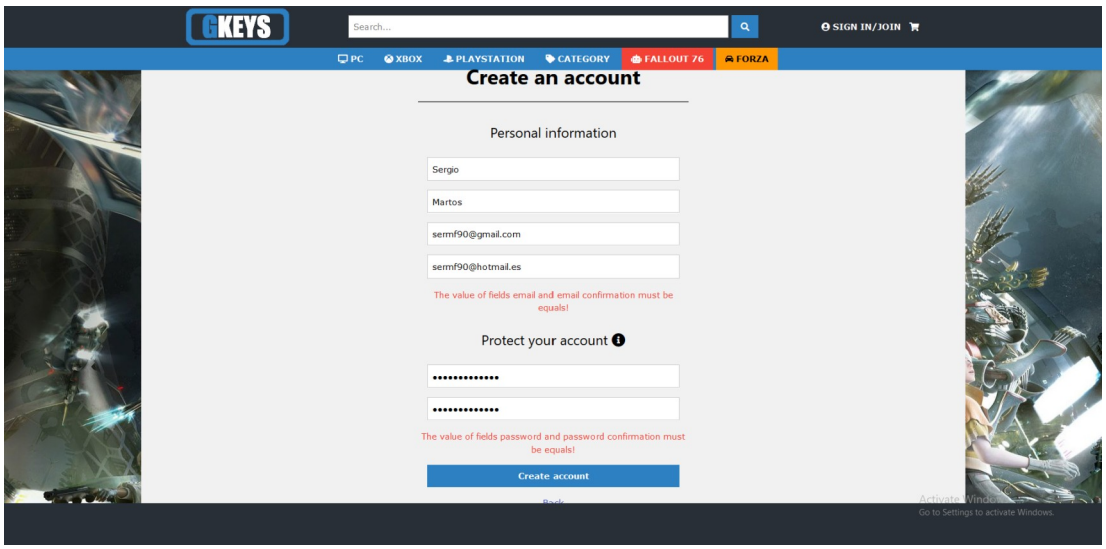


Figura 167: Resultat del cas 3 de la creació del compte d'usuari

Cas 4: creació de compte d'usuari sense dades

Resultat: visualització del missatge d'error corresponent

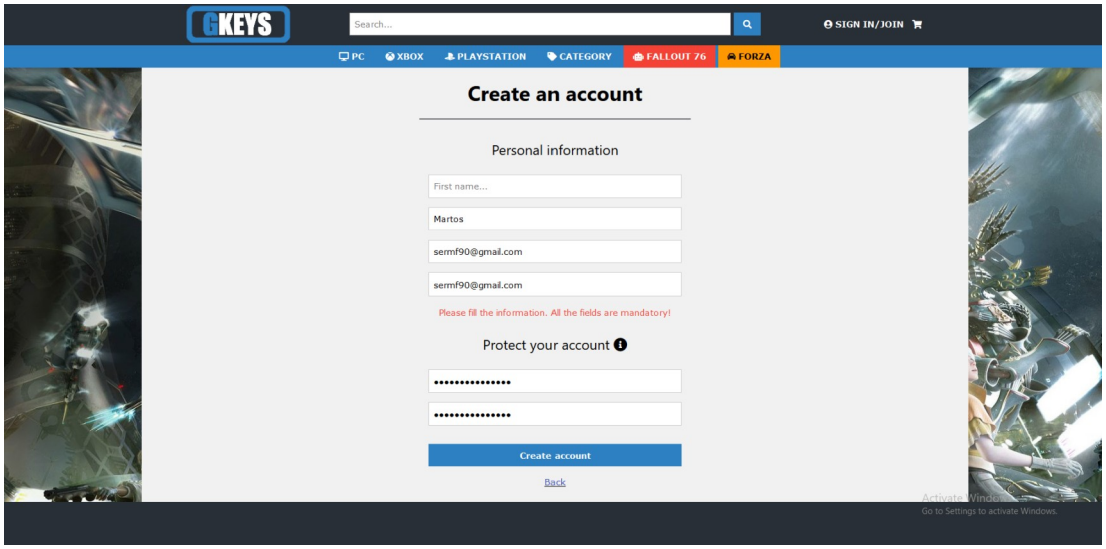


Figura 168: Resultat del cas 4 de la creació del compte d'usuari (nom buit)

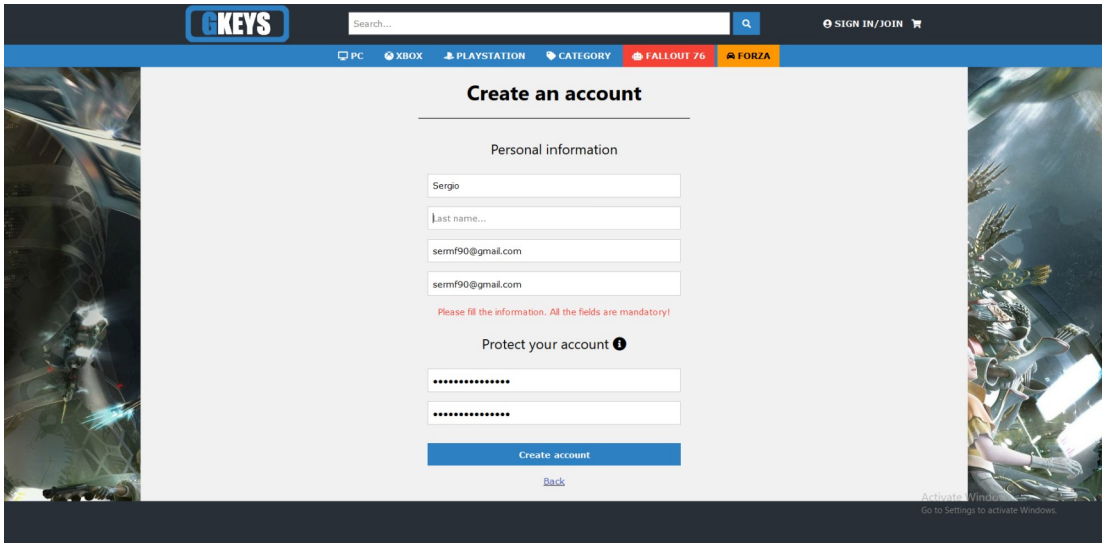


Figura 169: Resultat del cas 4 de la creació del compte d'usuari (cognom buit)

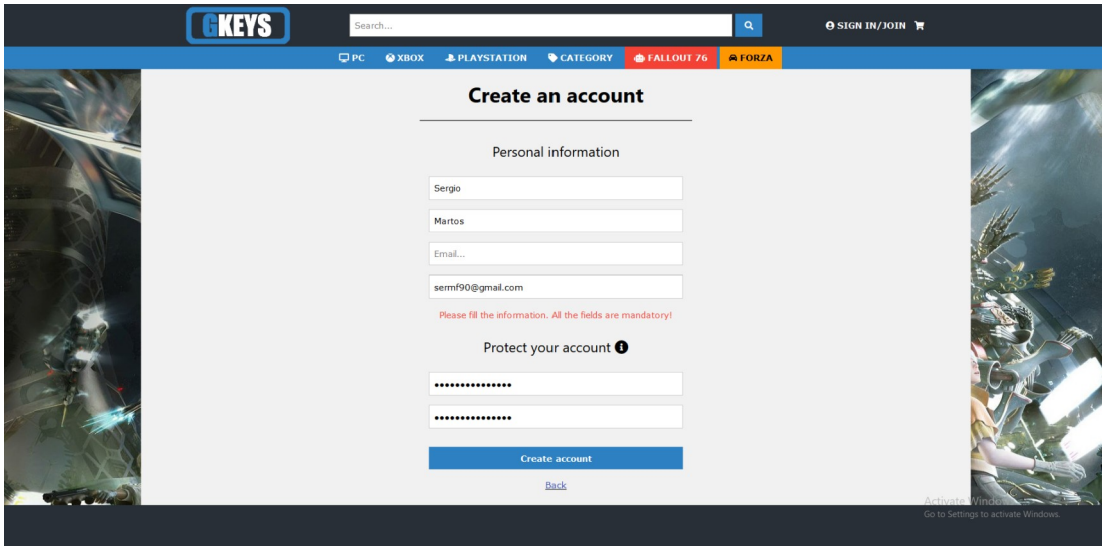


Figura 170: Resultat del cas 4 de la creació del compte d'usuari (correu buit)

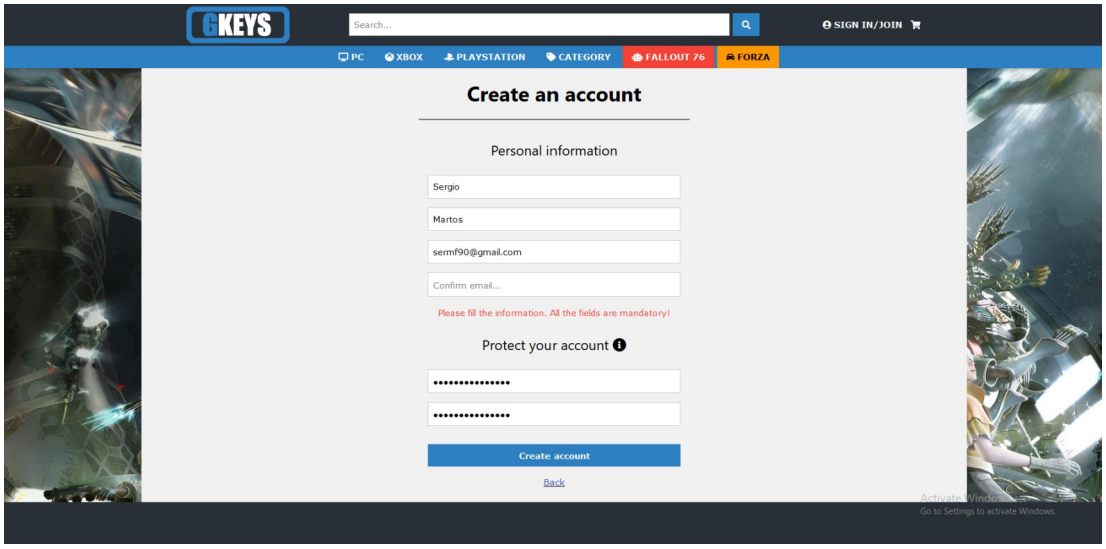


Figura 171: Resultat del cas 4 de la creació del compte d'usuari (confirmació correu buida)

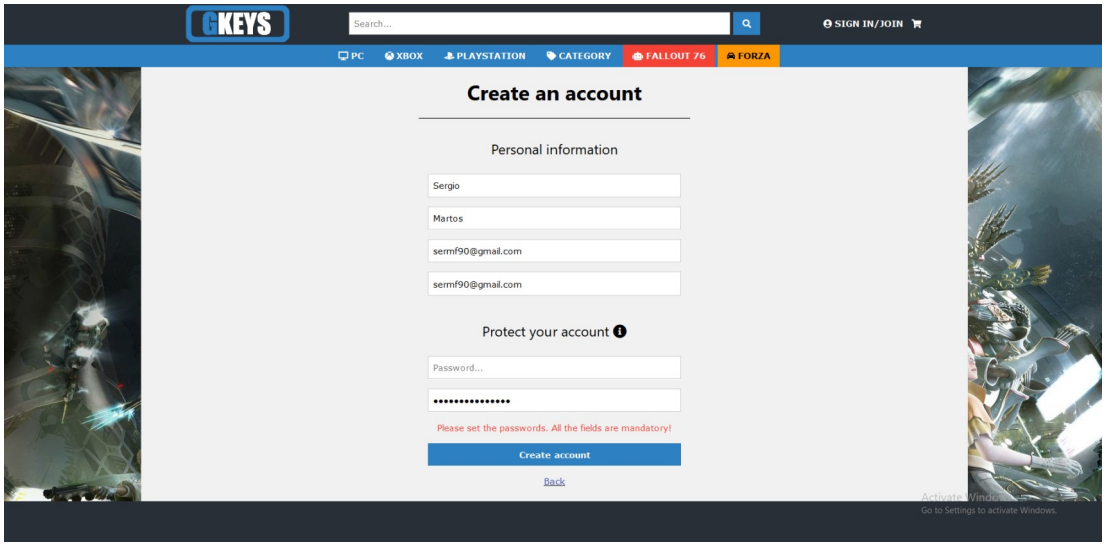


Figura 172: Resultat del cas 4 de la creació del compte d'usuari (contrasenya buida)

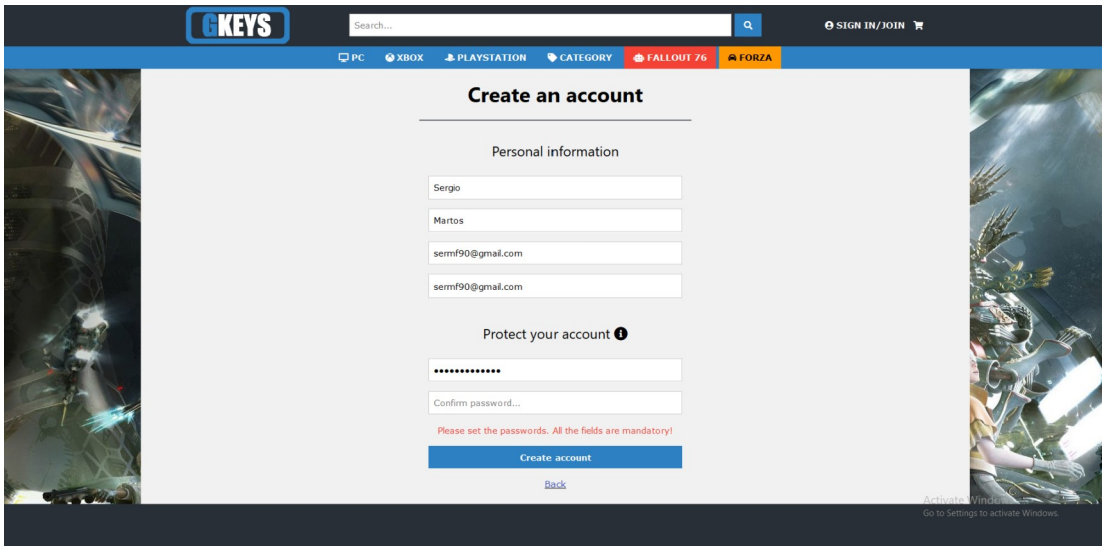


Figura 173: Resultat del cas 4 de la creació del compte d'usuari (confirmació contrasenya buida)

Cas 5: creació de compte d'usuari amb dades no vàlides (XSS)

Entrada: *<script>alert("XSS injection!")</script>* respectivament a cada camp

Resultat: visualització del missatge d'error corresponent

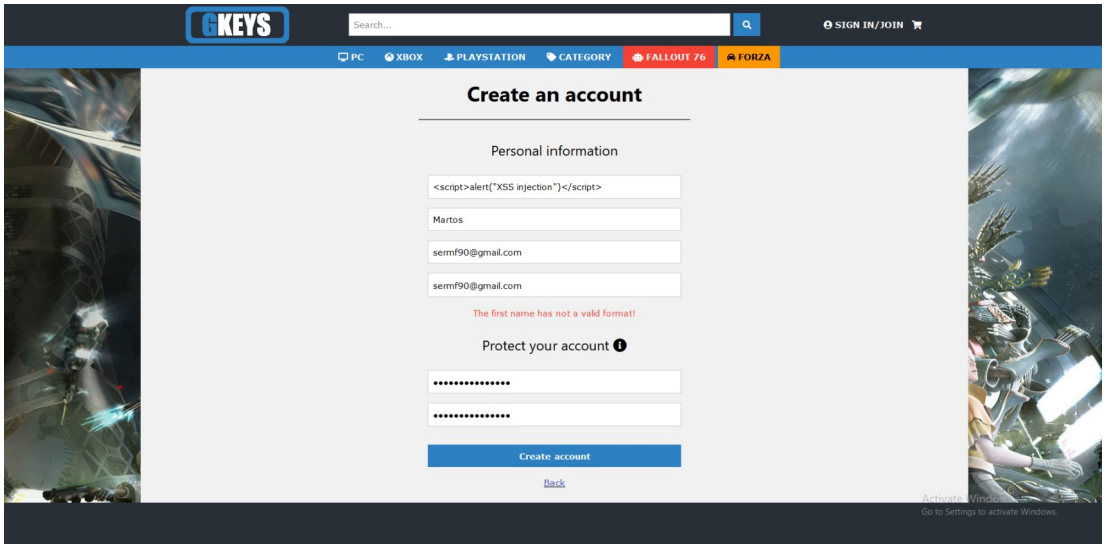


Figura 174: Resultat del cas 5 de la creació del compte d'usuari (nom malintencionat)

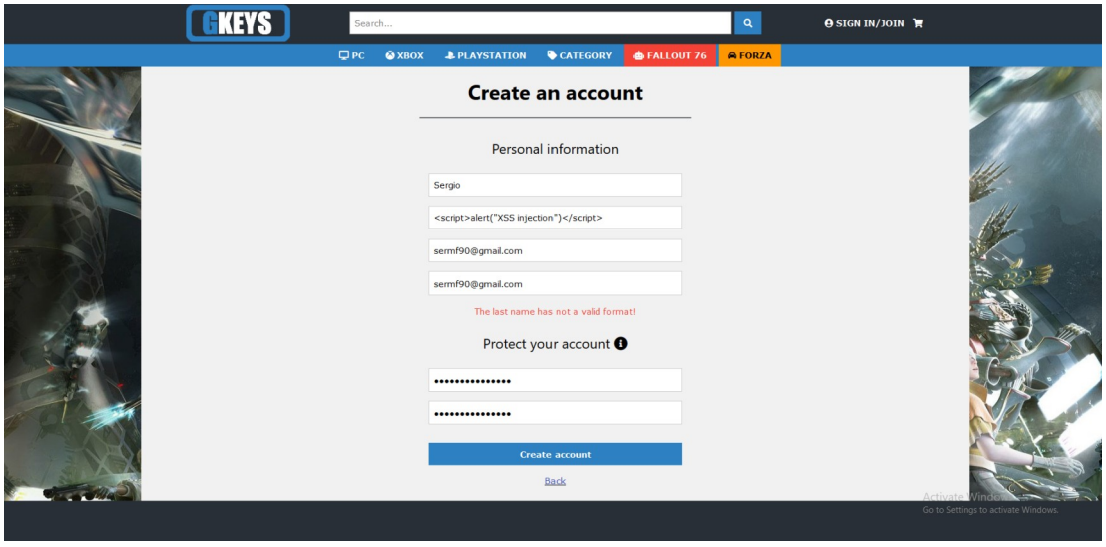


Figura 175: Resultat del cas 5 de la creació del compte d'usuari (cognom malintencionat)

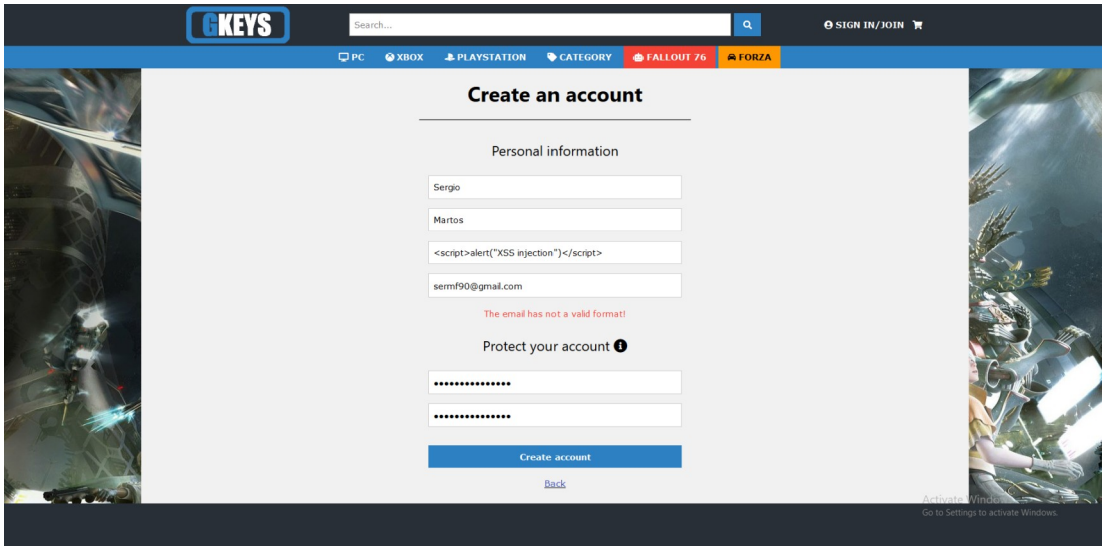


Figura 176: Resultat del cas 5 de la creació del compte d'usuari (correu malintencionat)

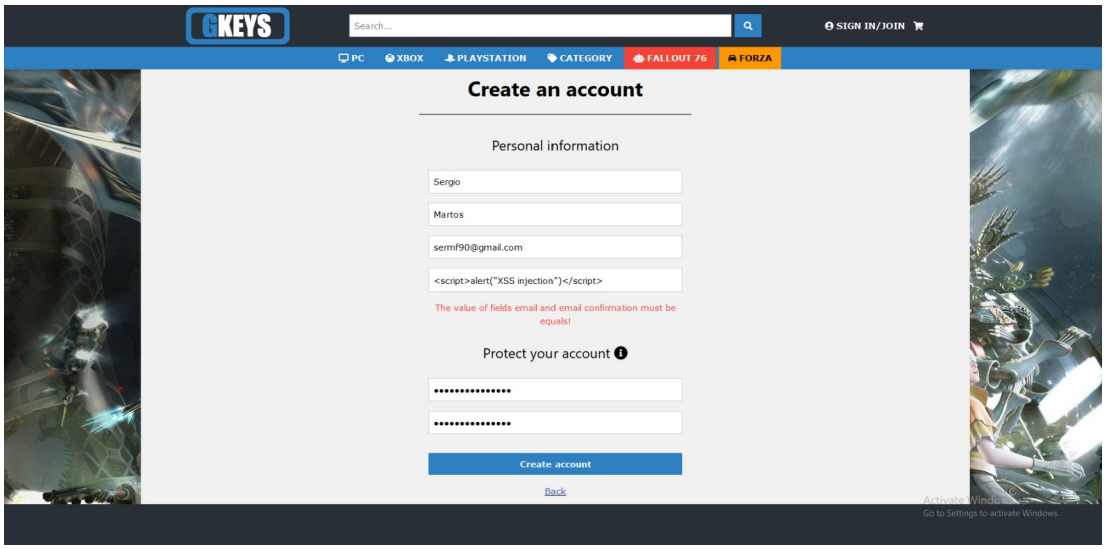


Figura 177: Resultat del cas 5 de la creació del compte d'usuari (confirmació del correu malintencionat)

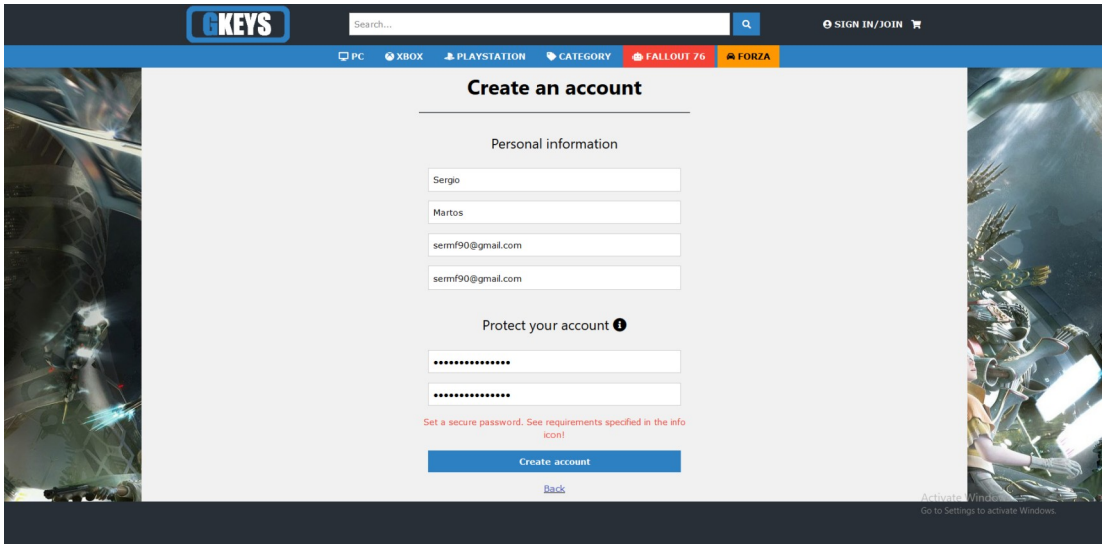


Figura 178: Resultat del cas 5 de la creació del compte d'usuari (contrasenya malintencionada)

Cas 6: creació de compte d'usuari amb dades no vàlides (SQL Injection)

Entrada: *OR 1=1;--* respectivament a cada camp

Resultat: visualització del missatge d'error corresponent

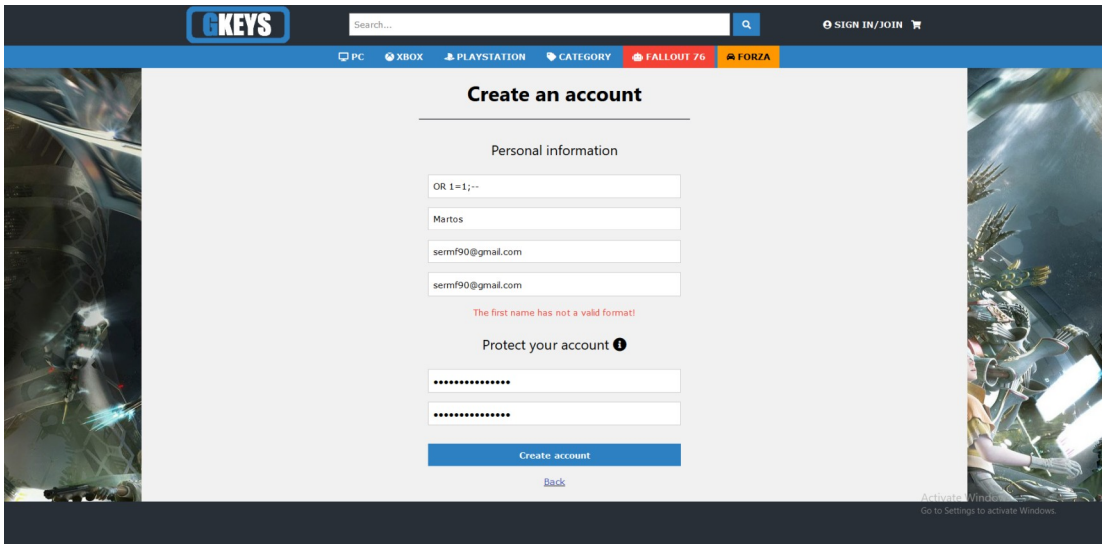


Figura 179: Resultat del cas 6 de la creació del compte d'usuari (nom malintencionat)

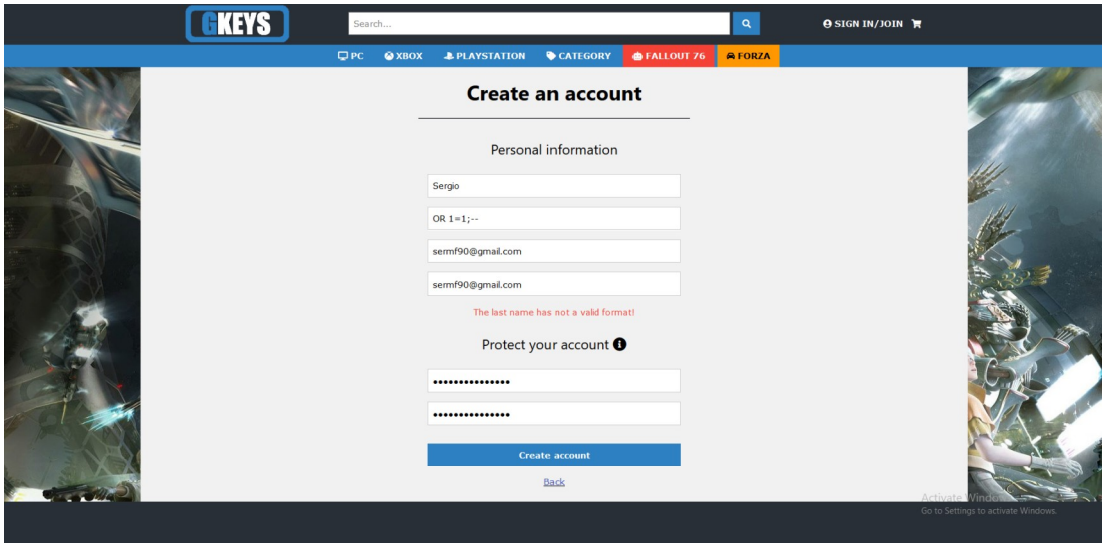


Figura 180: Resultat del cas 6 de la creació del compte d'usuari (cognom malintencionat)

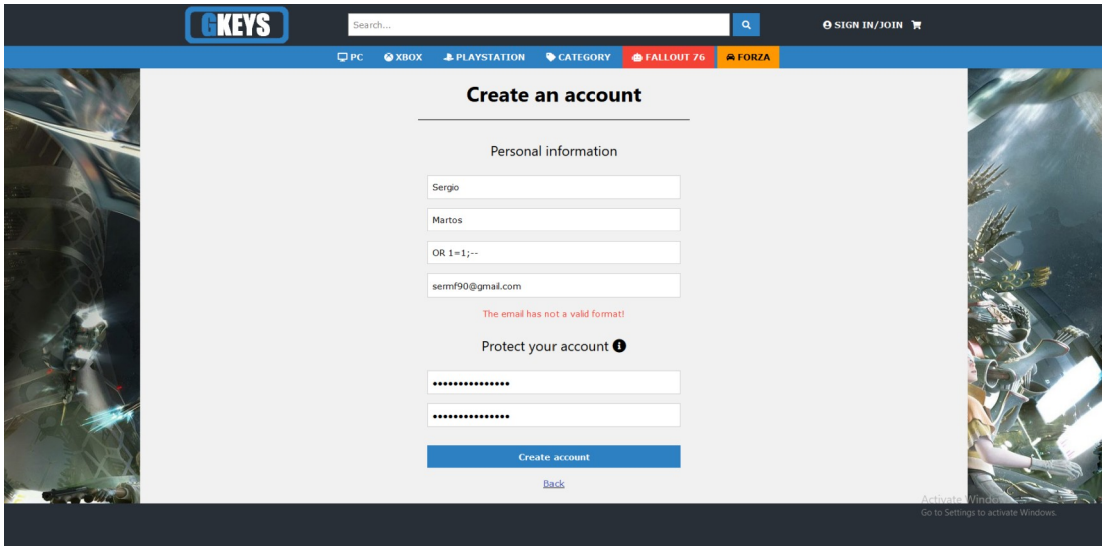


Figura 181: Resultat del cas 6 de la creació del compte d'usuari (correu malintencionat)

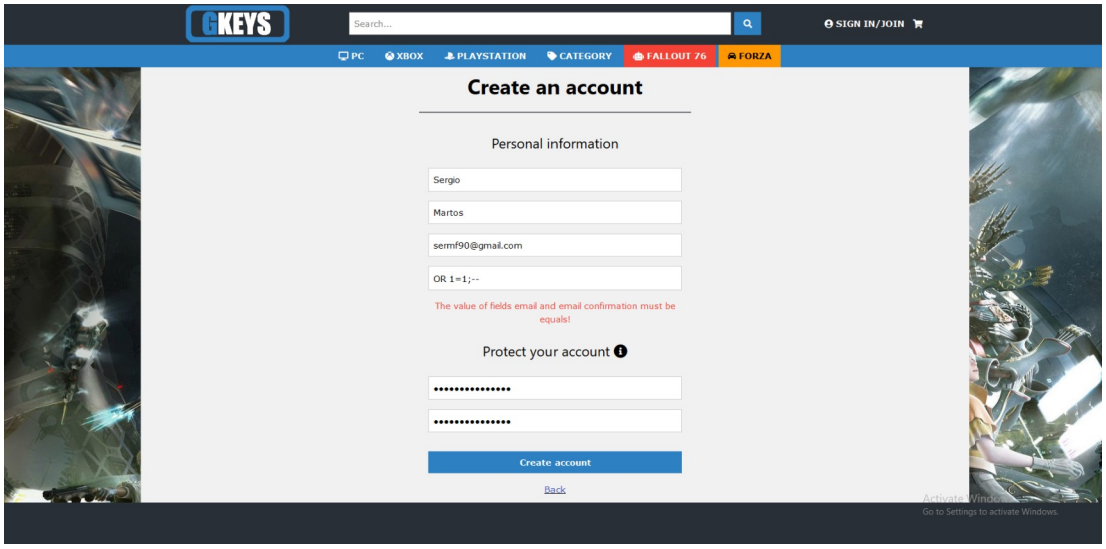


Figura 182: Resultat del cas 6 de la creació del compte d'usuari (confirmació del correu malintencionada)

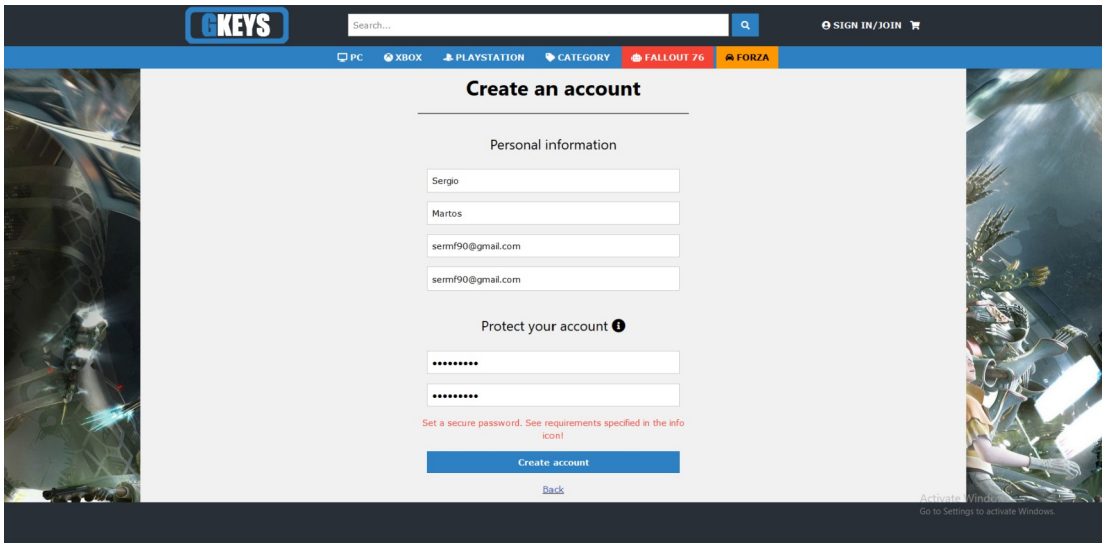


Figura 183: Resultat del cas 6 de la creació del compte d'usuari (contrasenya malintencionada)

Cas 7: creació de compte d'usuari ja existent

Entrada: qualsevol correu electrònic que ja existeixi en el sistema

Resultat: visualització del missatge d'error corresponent

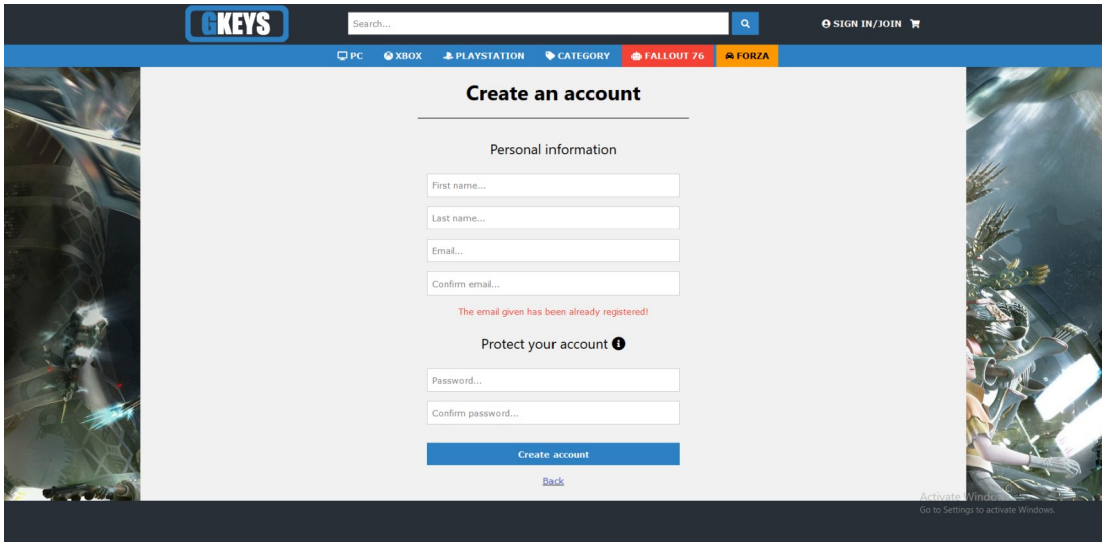


Figura 184: Resultat del cas 7 de la creació del compte d'usuari

6.9. Activació de compte d'usuari

Cas 1: activació de compte d'usuari amb dades correctes

Entrada: *sermf90@gmail.com*, *5c21ea542a62f* respectivament a cada camp

Resultat: Visualització del missatge de confirmació

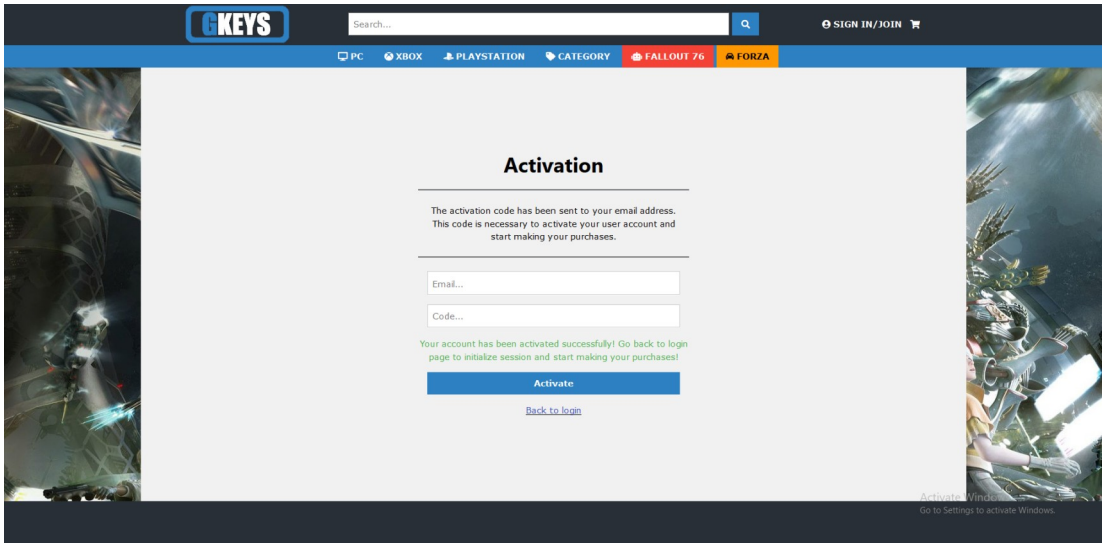


Figura 185: Resultat del cas 1 d'activació del compte d'usuari

Cas 2: activació de compte d'usuari amb dades no vàlides

Entrada: sermf90, *X_X* respectivament a cada camp

Resultat: visualització del missatge d'error corresponent

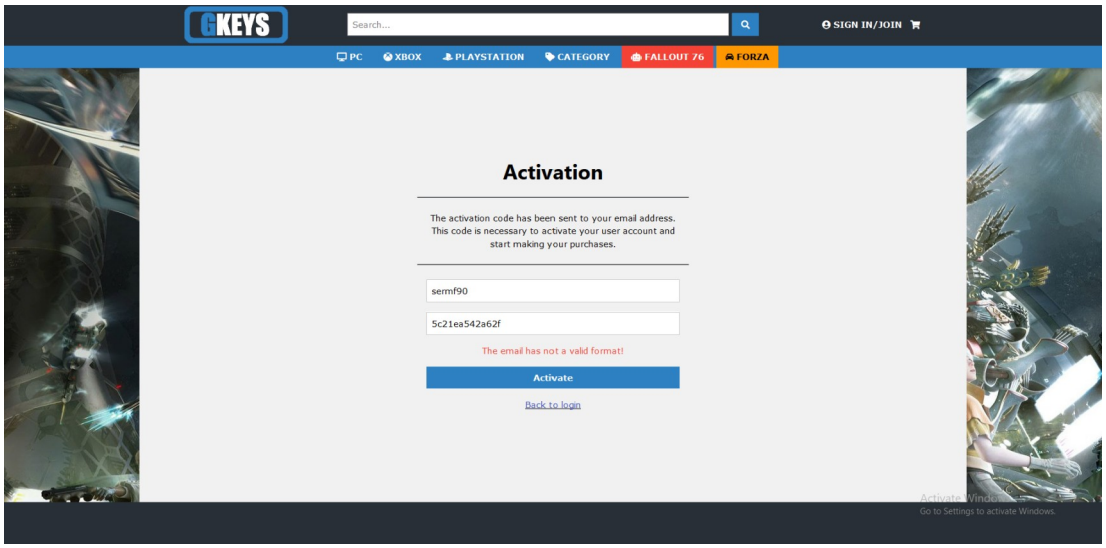


Figura 186: Resultat del cas 2 d'activació del compte d'usuari (correu no vàlid)

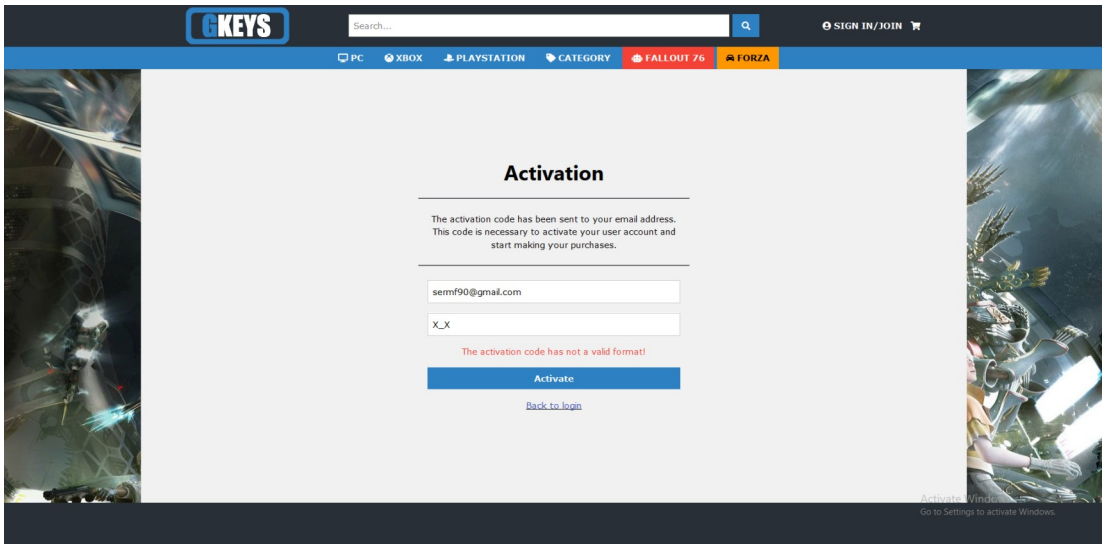


Figura 187: Resultat del cas 2 d'activació del compte d'usuari (codi no vàlid)

Cas 3: activació de compte d'usuari sense dades

Resultat: visualització del missatge d'error corresponent

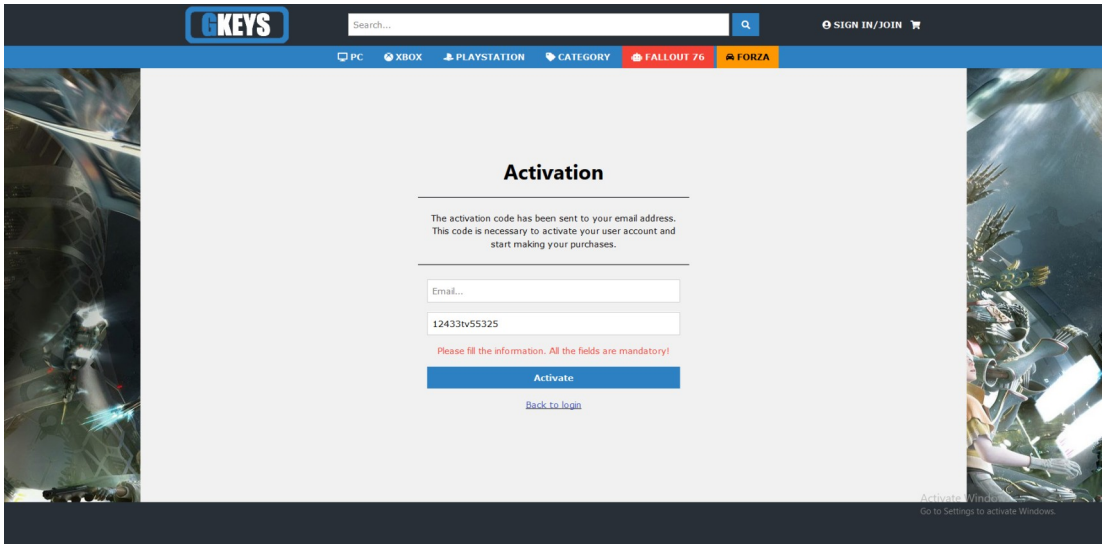


Figura 188: Resultat del cas 3 d'activació del compte d'usuari (correu buit)

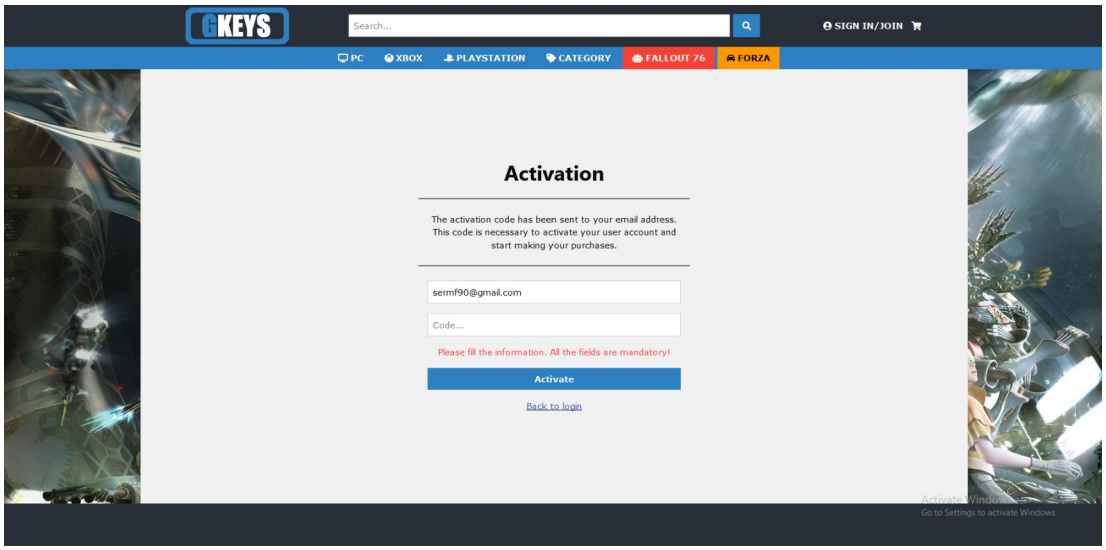


Figura 189: Resultat del cas 3 d'activació del compte d'usuari (codi buit)

Cas 4: activació de compte d'usuari amb dades no vàlides (XSS)

Entrada: *<script>alert("XSS injection!")</script>* respectivament a cada camp

Resultat: visualització del missatge d'error corresponent

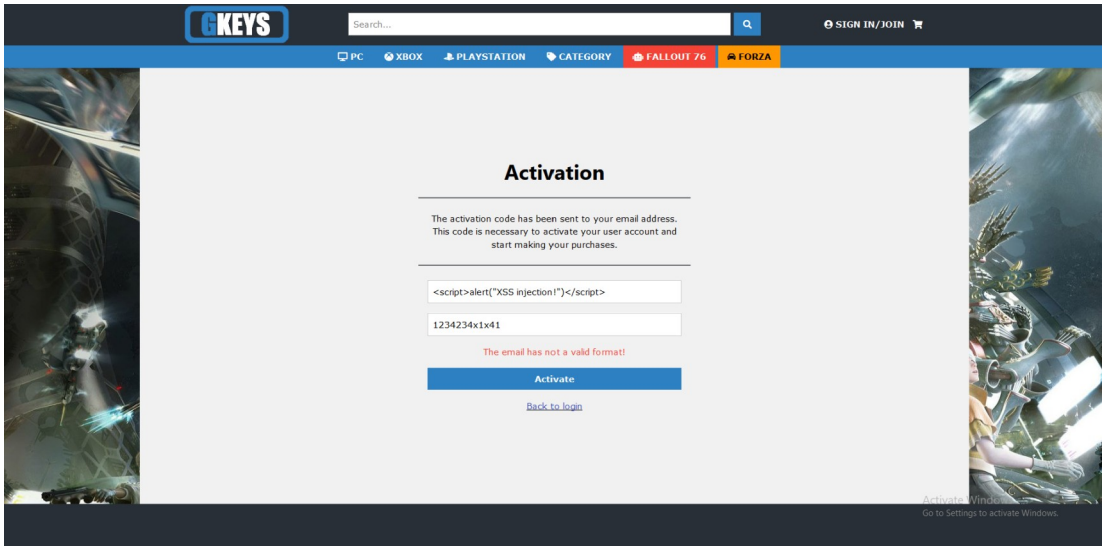


Figura 190: Resultat del cas 4 d'activació del compte d'usuari (correu malintencionat)

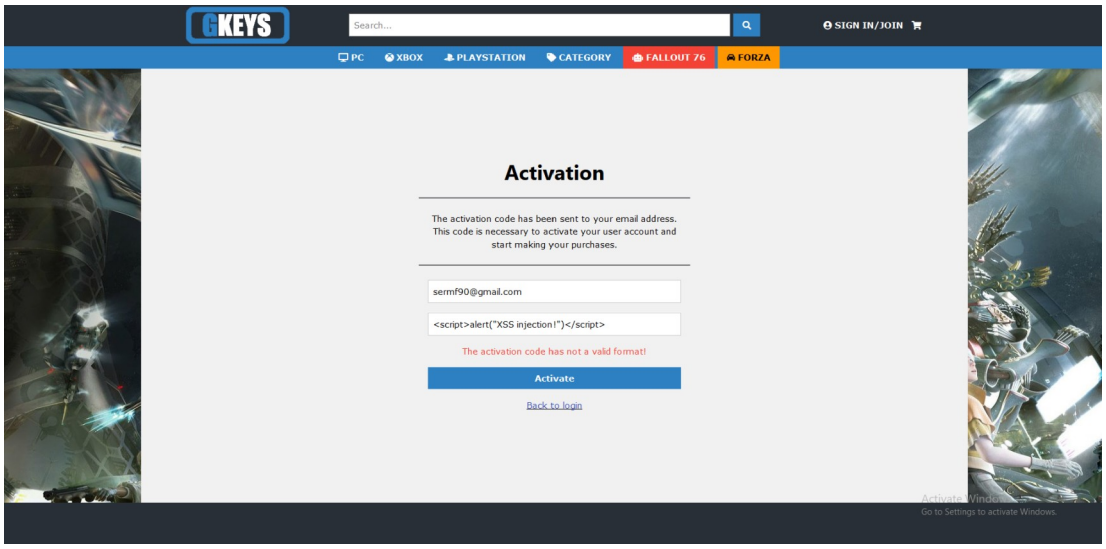


Figura 191: Resultat del cas 4 d'activació del compte d'usuari (codi malintencionat)

Cas 5: activació de compte d'usuari amb dades no vàlides (SQL Injection)

Entrada: *OR 1=1;--* respectivament a cada camp

Resultat: visualització del missatge d'error corresponent

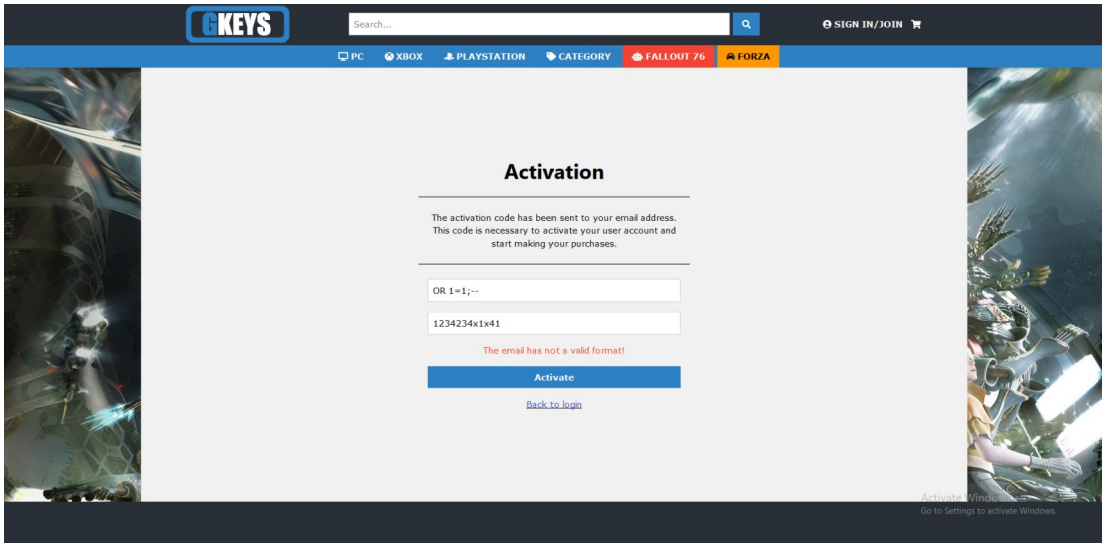


Figura 192: Resultat del cas 5 d'activació del compte d'usuari (correu malintencionat)

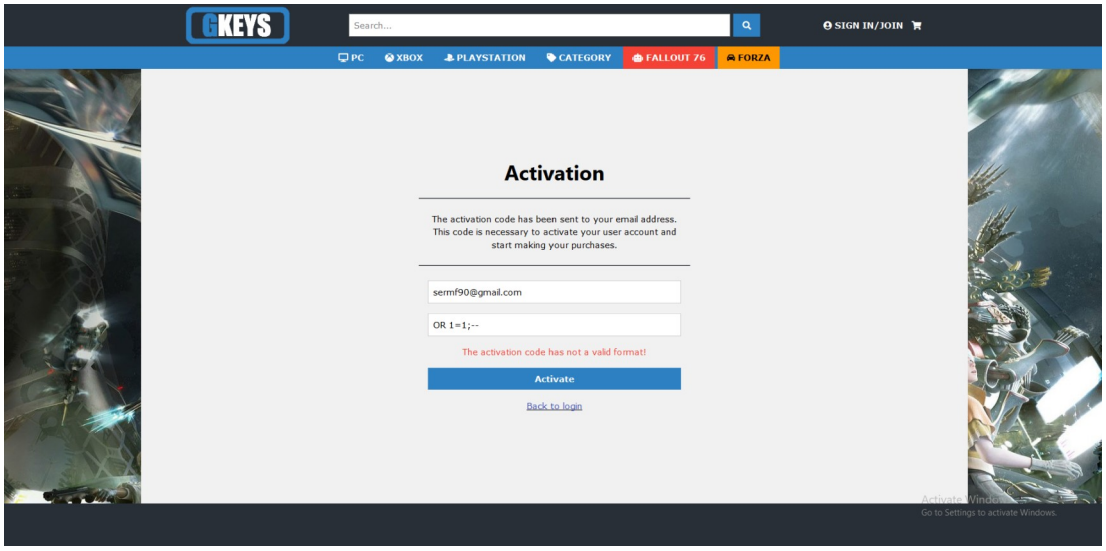


Figura 193: Resultat del cas 5 d'activació del compte d'usuari (codi malintencionat)

Cas 6: activació de compte d'usuari prèviament activat

Entrada: qualsevol compte d'usuari ja activat en el sistema

Resultat: visualització del missatge d'error corresponent

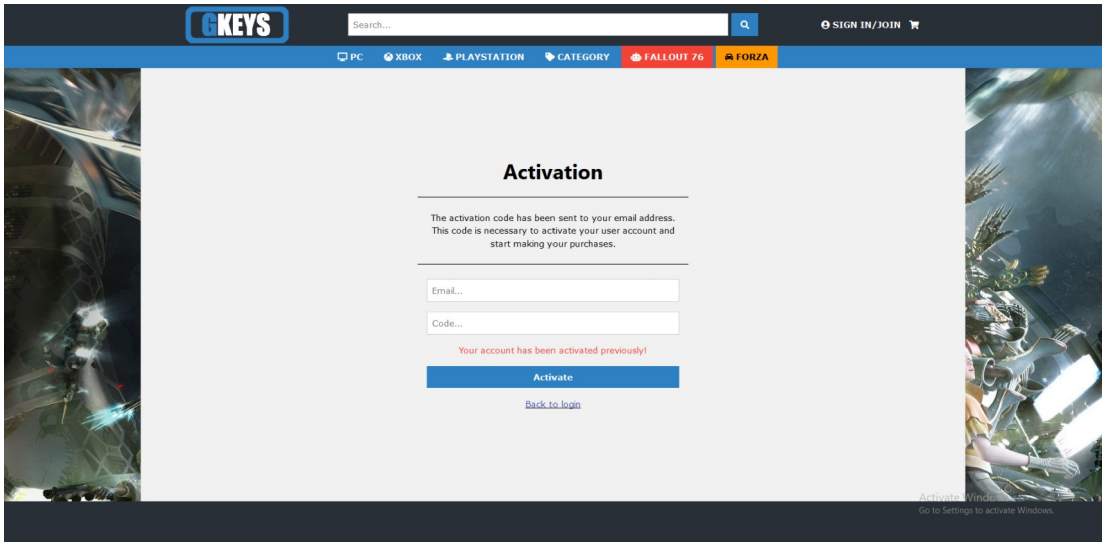


Figura 194: Resultat del cas 6 d'activació del compte d'usuari

Cas 7: activació de compte d'usuari no existent

Entrada: qualsevol correu electrònic que no existeixi en el sistema

Resultat: visualització del missatge d'error corresponent

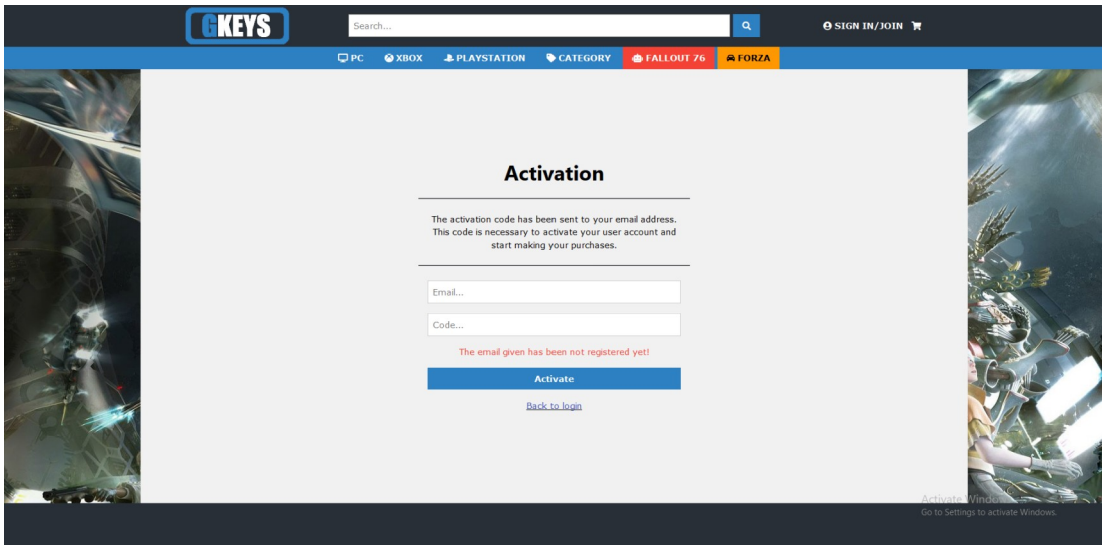


Figura 195: Resultat del cas 7 d'activació del compte d'usuari

Cas 8: activació de compte d'usuari amb codi vàlid però incorrecte

Entrada: correu electrònic que s'hagi d'activar en el sistema i un codi d'activació amb format vàlid però incorrecte.

Resultat: visualització del missatge d'error corresponent

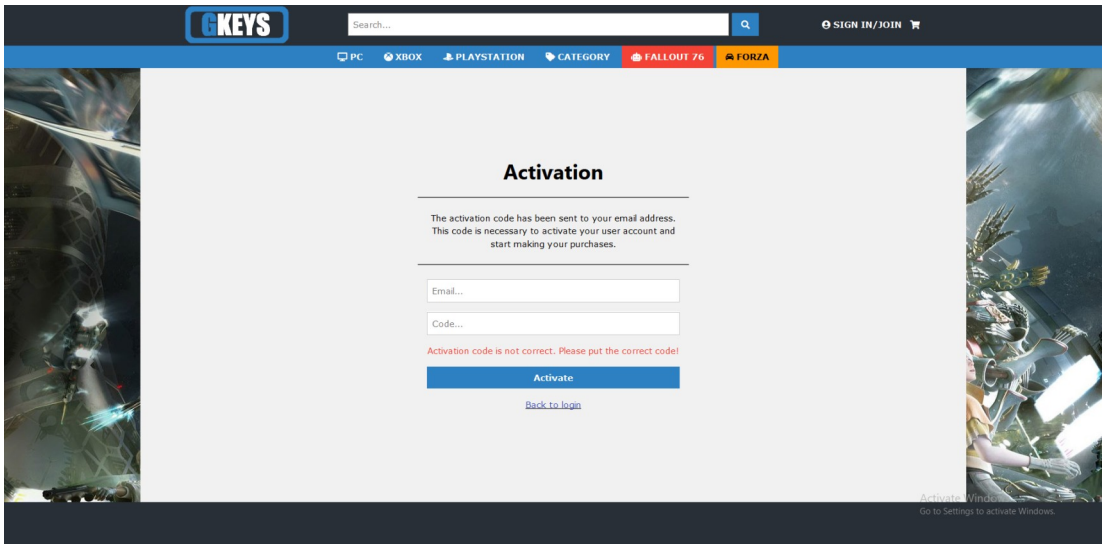


Figura 196: Resultat del cas 8 d'activació del compte d'usuari

Cas 9: Bloqueig d'activació del compte d'usuari

Entrada: correu electrònic existent en el sistema i codi no correcte, durant 6 intents

Resultat: visualització del missatge d'error corresponent

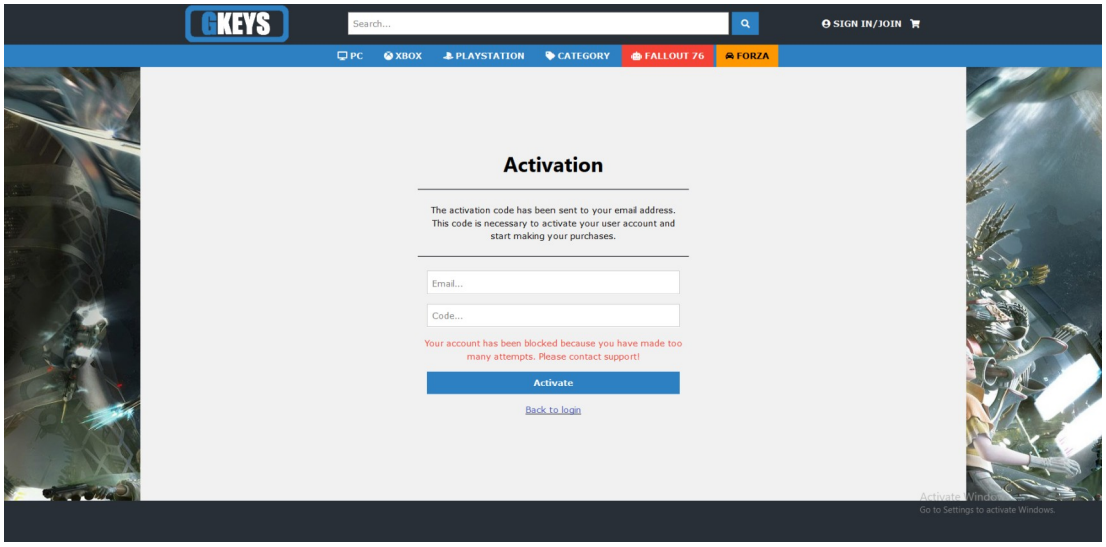


Figura 197: Resultat del cas 9 d'activació del compte d'usuari

6.10. Inici de sessió

Cas 1: inici de sessió d'usuari amb usuari existent

Entrada: *sermf90@gmail.com i la contrasenya corresponent*

Resultat: El sistema inicia la sessió i realitza una redirecció cap a la pàgina principal

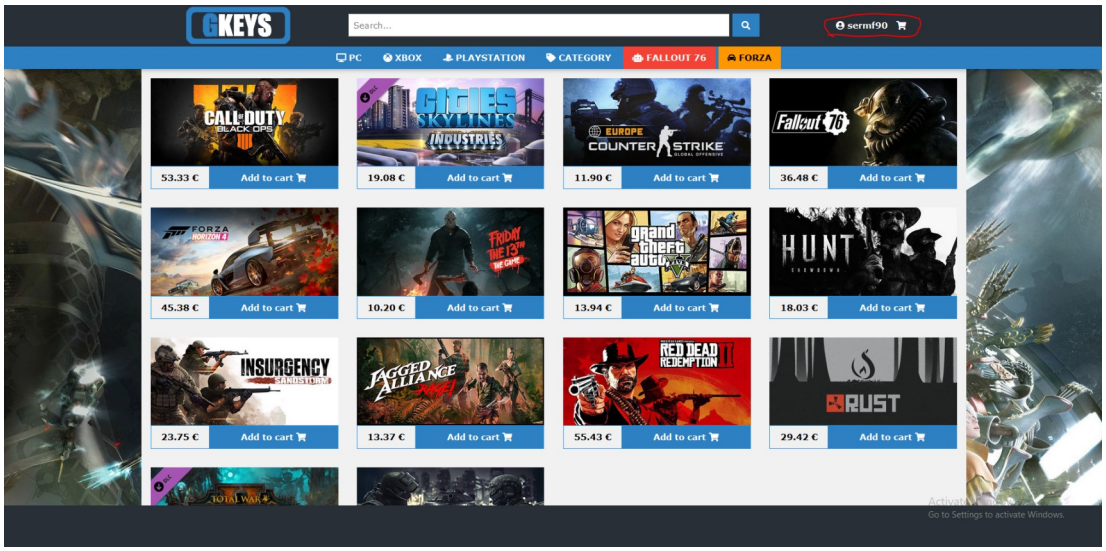


Figura 198: Resultat del cas 1 d'inici de sessió d'usuari

Cas 2: inici de sessió d'usuari amb dades no vàlides

Entrada: sermf90, *1234* respectivament a cada camp

Resultat: visualització del missatge d'error corresponent

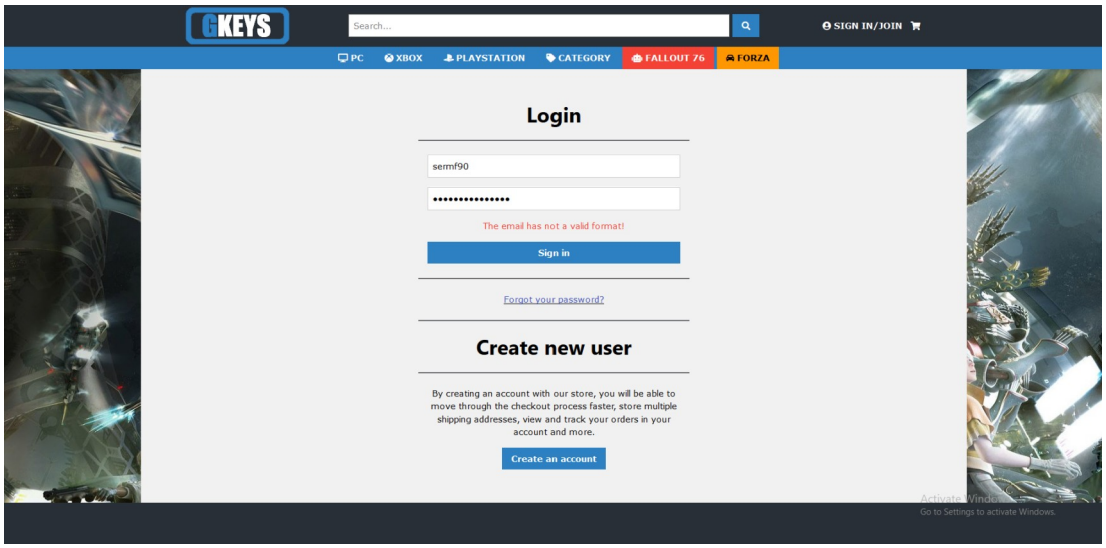


Figura 199: Resultat del cas 2 d'inici de sessió d'usuari (correu no vàlid)

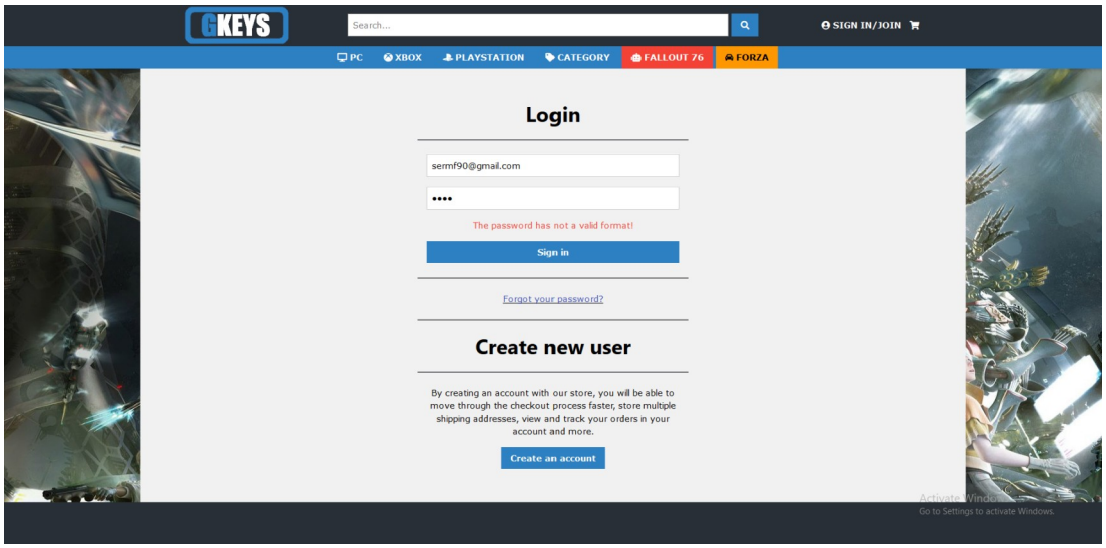


Figura 200: Resultat del cas 2 d'inici de sessió d'usuari (contrasenya no vàlida)

Cas 3: inici de sessió d'usuari sense dades

Resultat: visualització del missatge d'error corresponent

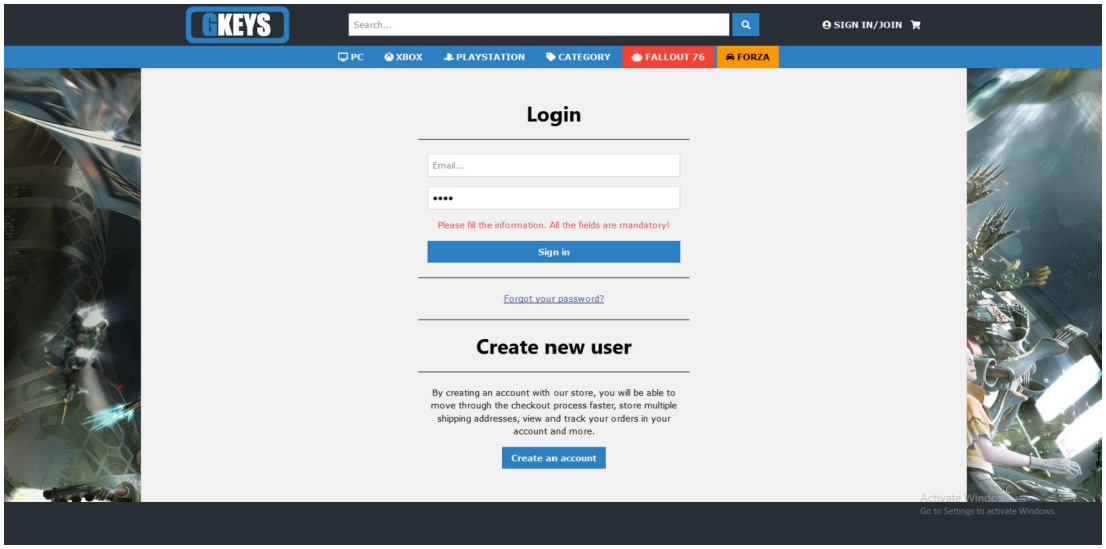


Figura 201: Resultat del cas 3 d'inici de sessió d'usuari (correu buit)

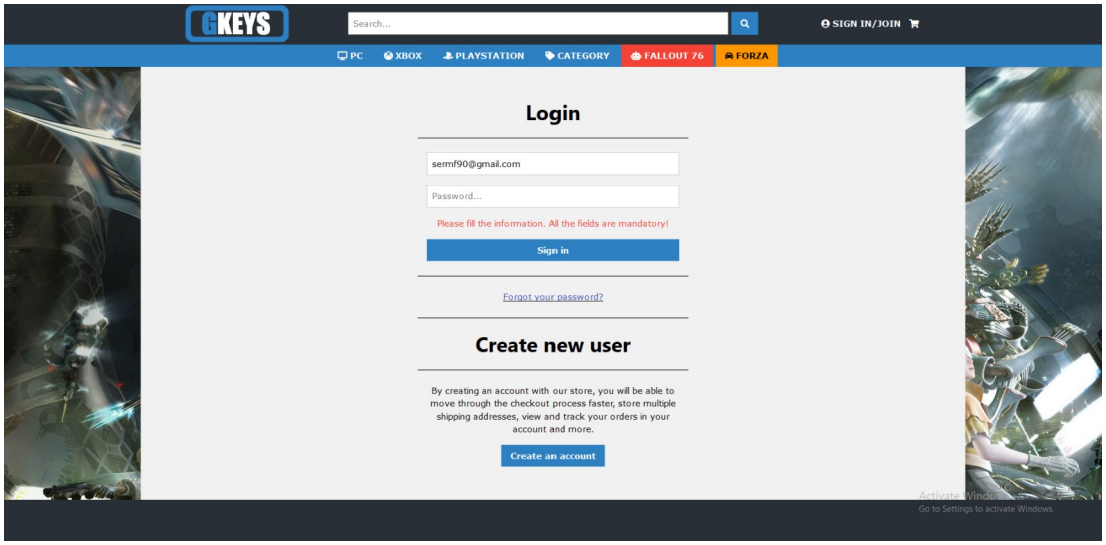


Figura 202: Resultat del cas 3 d'inici de sessió d'usuari (contrasenya buida)

Cas 4: inici de sessió d'usuari amb dades no vàlides (XSS)

Entrada: *<script>alert("XSS injection!")</script>* respectivament a cada camp

Resultat: visualització del missatge d'error corresponent

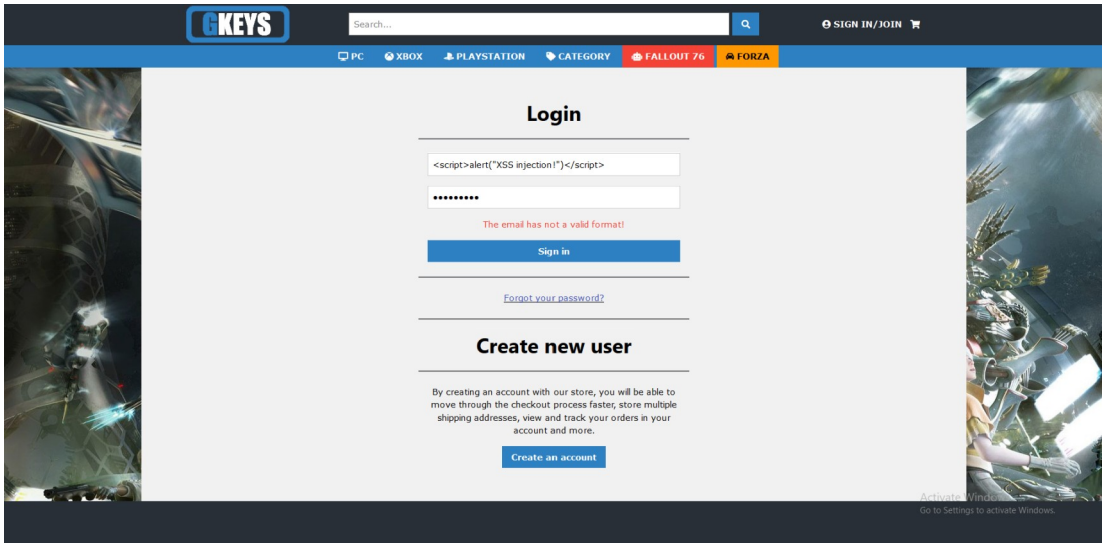


Figura 203: Resultat del cas 4 d'inici de sessió d'usuari (correu malintencionat)

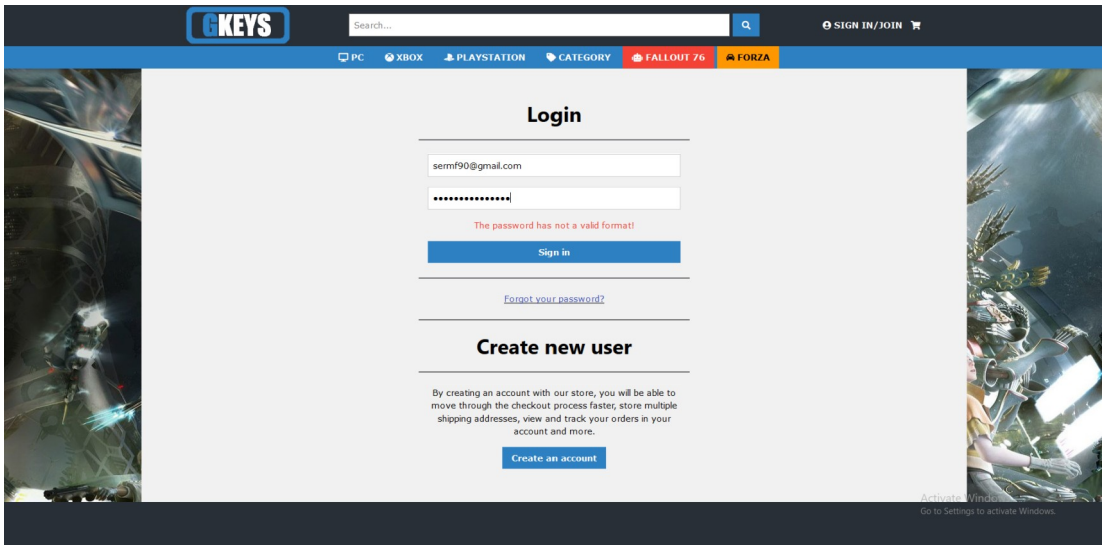


Figura 204: Resultat del cas 4 d'inici de sessió d'usuari (contrasenya malintencionada)

Cas 5: inici de sessió d'usuari amb dades no vàlides (SQL Injection)

Entrada: *OR 1=1;*-- respectivament a cada camp

Resultat: visualització del missatge d'error corresponent

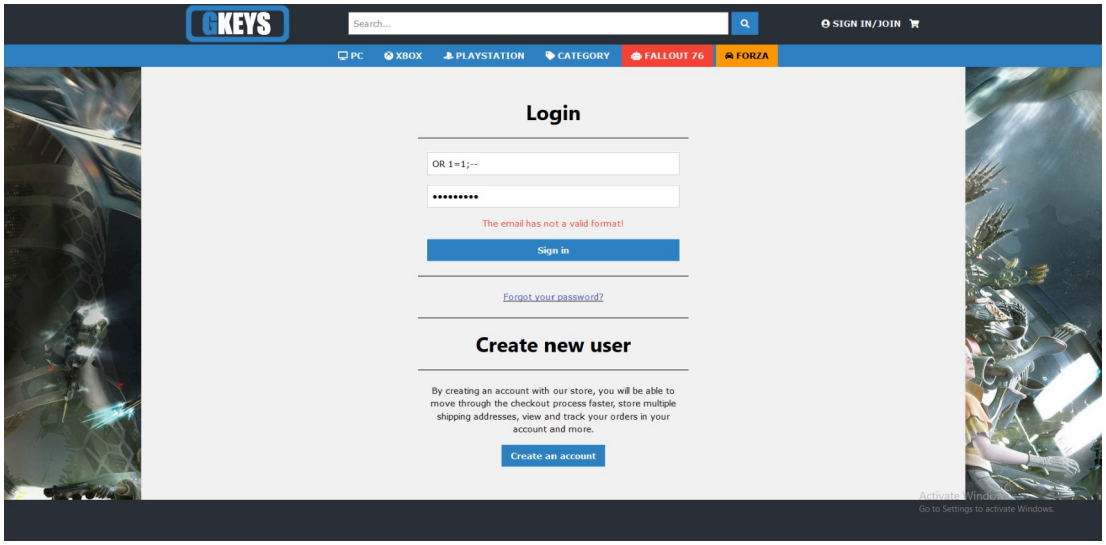


Figura 205: Resultat del cas 5 d'inici de sessió d'usuari (correu malintencionat)

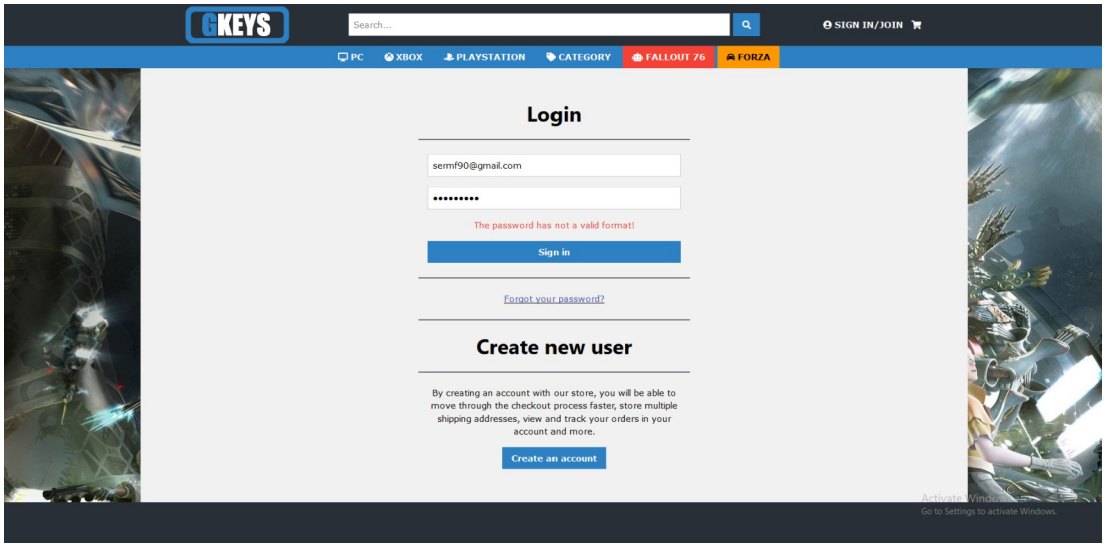


Figura 206: Resultat del cas 5 d'inici de sessió d'usuari (contrasenya malintencionada)

Cas 6: inici de sessió d'usuari no existent

Entrada: qualsevol correu electrònic que no existeixi en el sistema

Resultat: visualització del missatge d'error corresponent

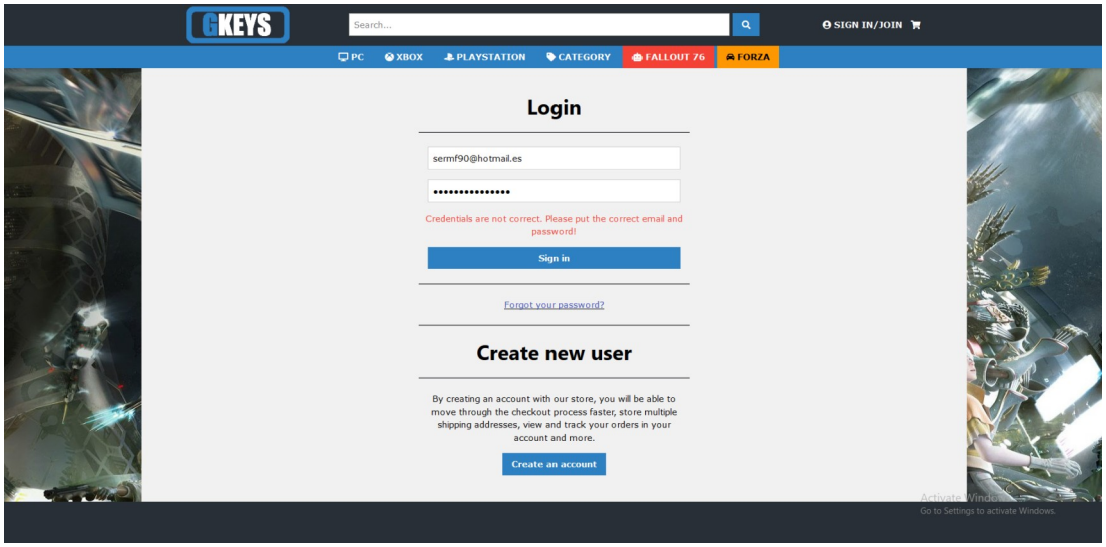


Figure 207: Resultat del cas 6 d'inici de sessió d'usuari

Cas 7: inici de sessió d'usuari amb contrasenya incorrecte

Entrada: qualsevol correu electrònic que existeixi en el sistema i una contrasenya vàlida però incorrecte

Resultat: visualització del missatge d'error corresponent

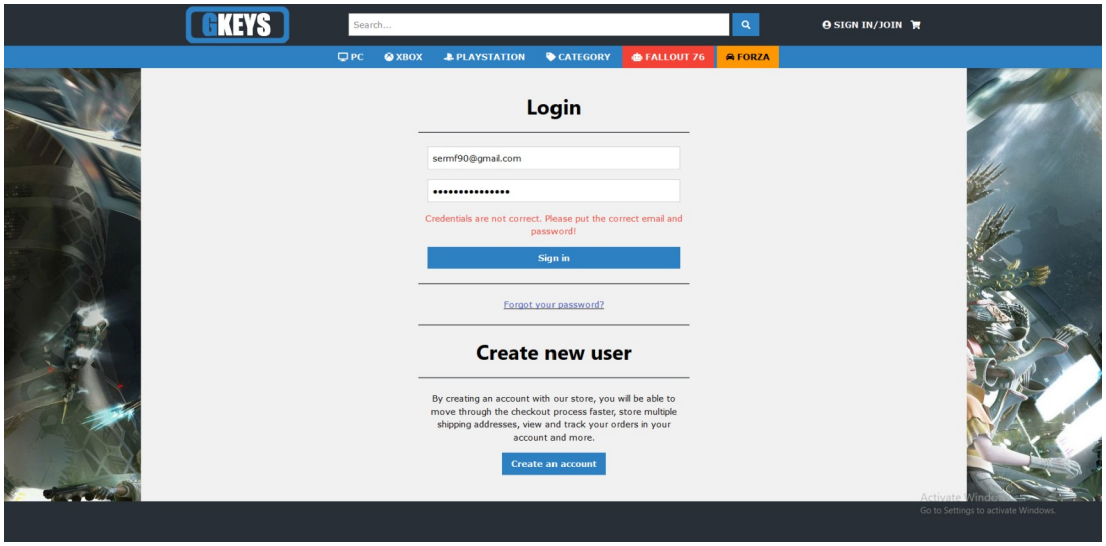


Figura 208: Resultat del cas 7 d'inici de sessió d'usuari

Cas 8: inici de sessió d'usuari bloquejat

Entrada: qualsevol compte d'usuari bloquejat en el sistema

Resultat: visualització del missatge d'error corresponent

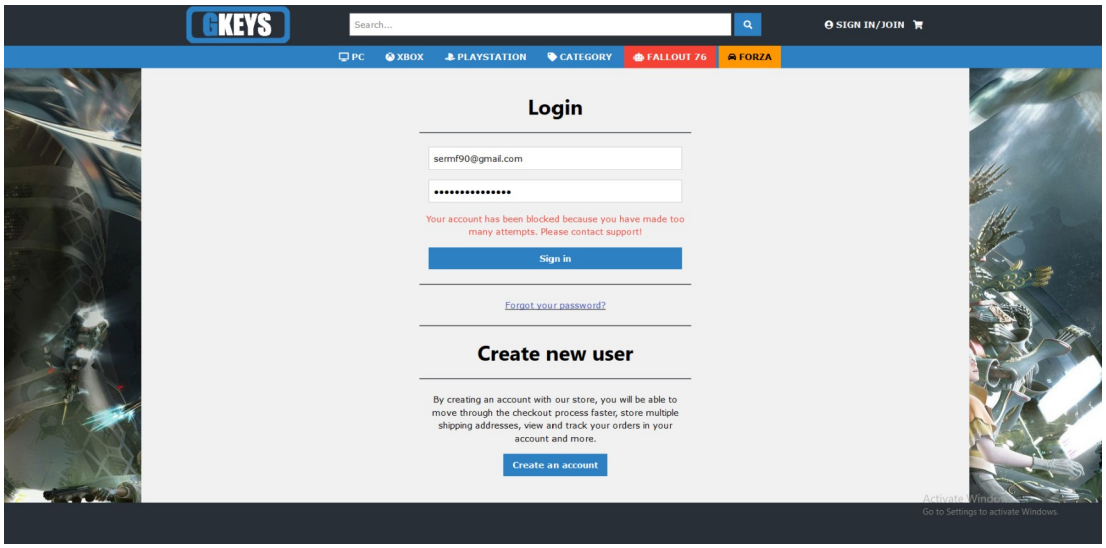


Figura 209: Resultat del cas 8 d'inici de sessió d'usuari

Cas 9: inici de sessió d'usuari no activat

Entrada: qualsevol compte d'usuari bloquejat en el sistema

Resultat: visualització del missatge d'error corresponent

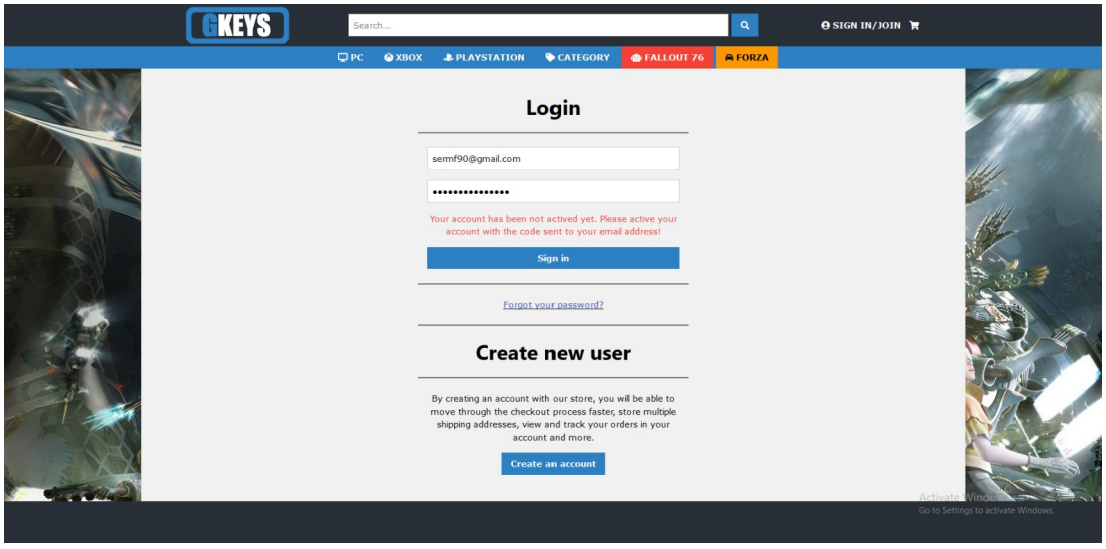


Figura 210: Resultat del cas 9 d'inici de sessió d'usuari

6.11. Recuperació de compte d'usuari

Cas 1: sol·licitar la recuperació del compte amb usuari existent

Entrada: *sermf90@gmail.com*

Resultat: El sistema envia el codi de recuperació al correu indicat i realitza una redirecció cap al formulari de modificació de contrasenyes.

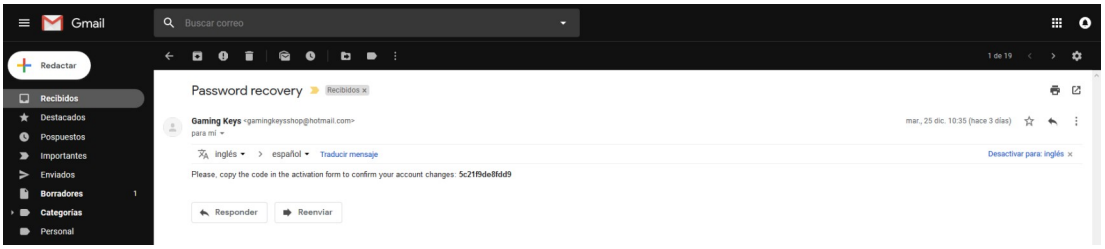


Figura 211: Resultat del cas 1 de la recuperació del compte d'usuari (correu electrònic)

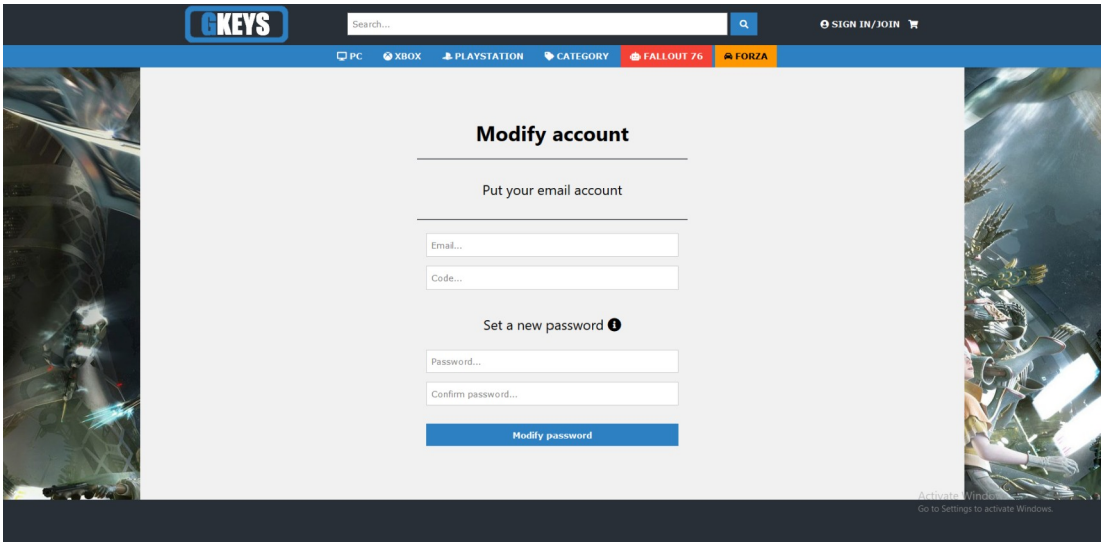


Figura 212: Resultat del cas 1 de la recuperació del compte d'usuari (formulari de recuperació)

Cas 2: sol·licitar la recuperació del compte amb dades no vàlides

Entrada: *sermf90*

Resultat: visualització del missatge d'error corresponent

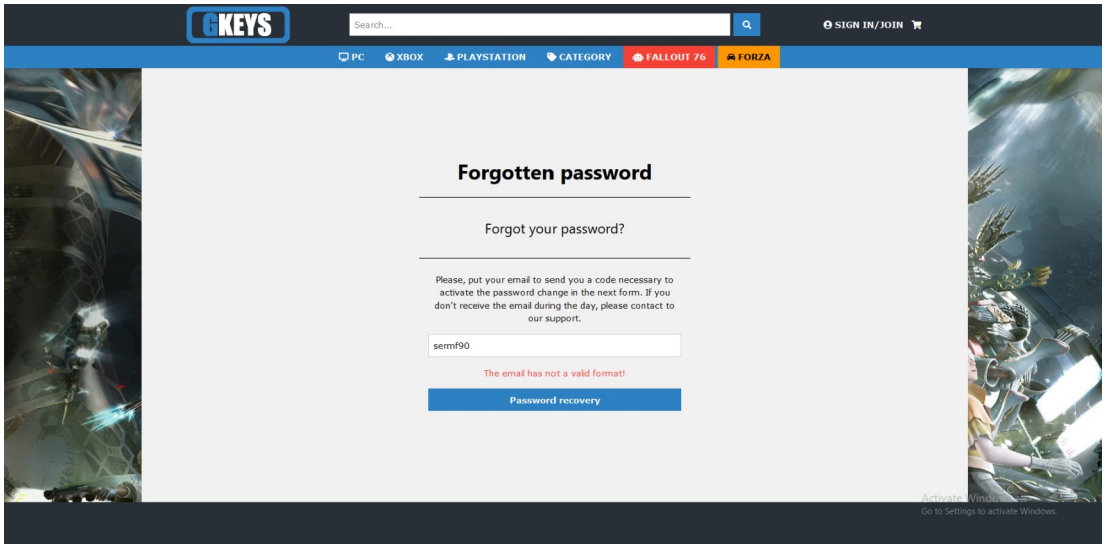


Figura 213: Resultat del cas 2 de la recuperació del compte d'usuari

Cas 3: sol·licitar la recuperació del compte sense dades

Resultat: visualització del missatge d'error corresponent

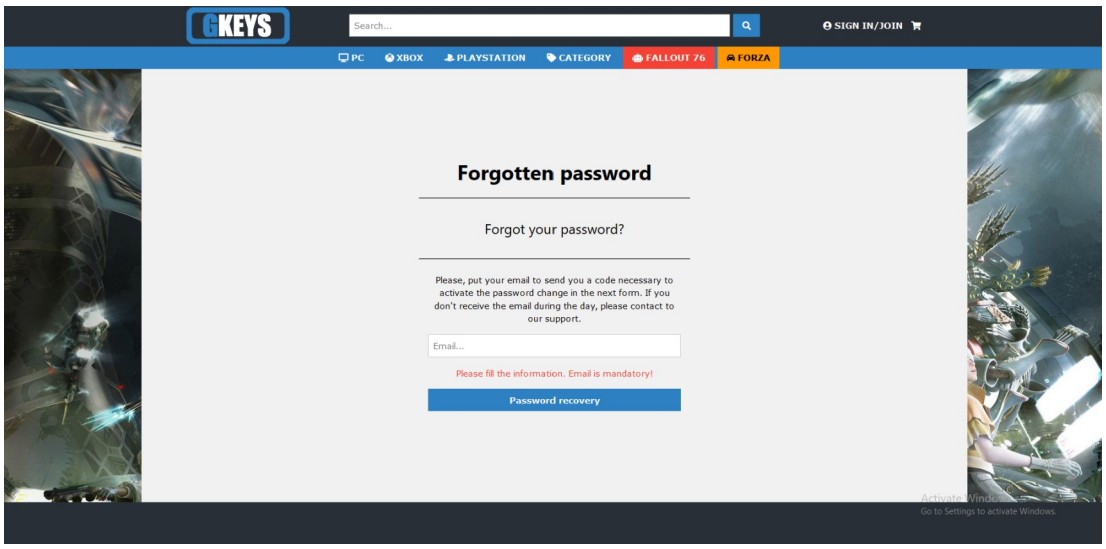


Figura 214: Resultat del cas 3 de la recuperació del compte d'usuari

Cas 4: sol·licitar la recuperació del compte amb dades no vàlides (XSS)

Entrada: *<script>alert("XSS injection!")</script>*

Resultat: visualització del missatge d'error corresponent

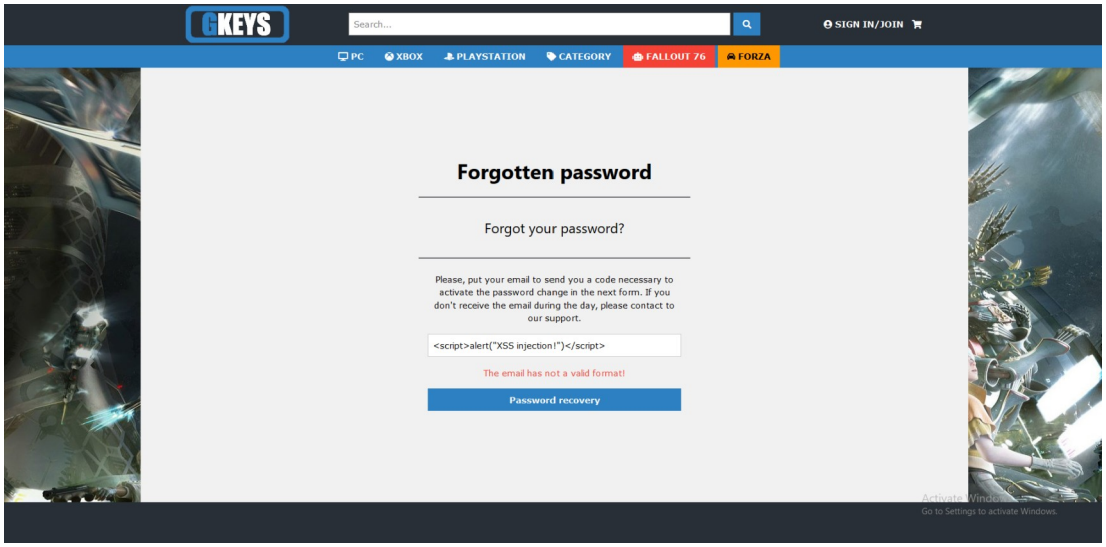


Figura 215: Resultat del cas 4 de la recuperació del compte d'usuari

Cas 5: sol·licitar la recuperació del compte amb dades no vàlides (SQL Injection)

Entrada: *OR 1=1;--*

Resultat: visualització del missatge d'error corresponent

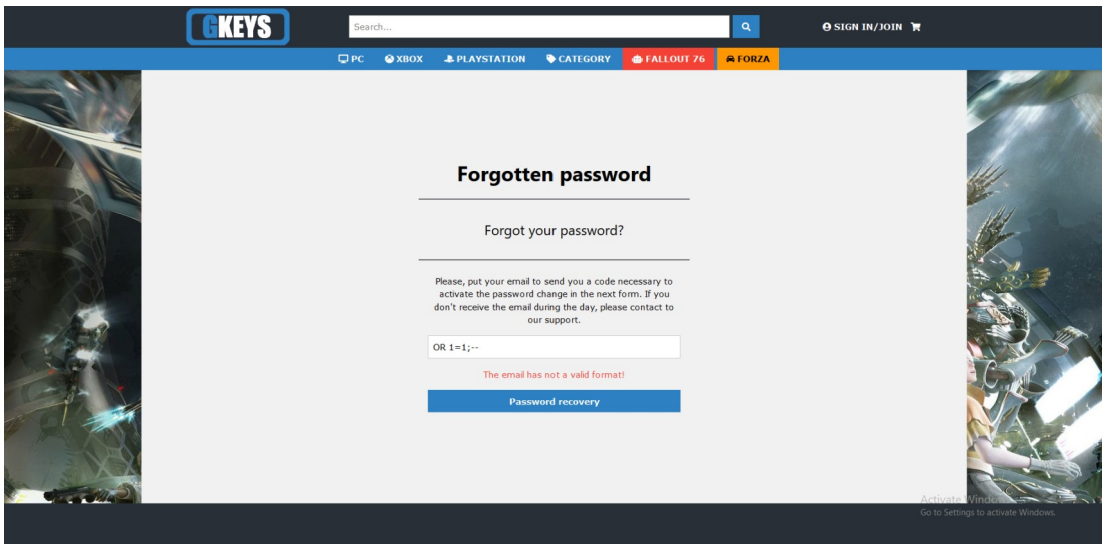


Figura 216: Resultat del cas 5 de la recuperació del compte d'usuari

Cas 6: sol·licitar la recuperació d'un compte no existent

Entrada: qualsevol correu electrònic que no existeixi en el sistema

Resultat: visualització del missatge d'error corresponent

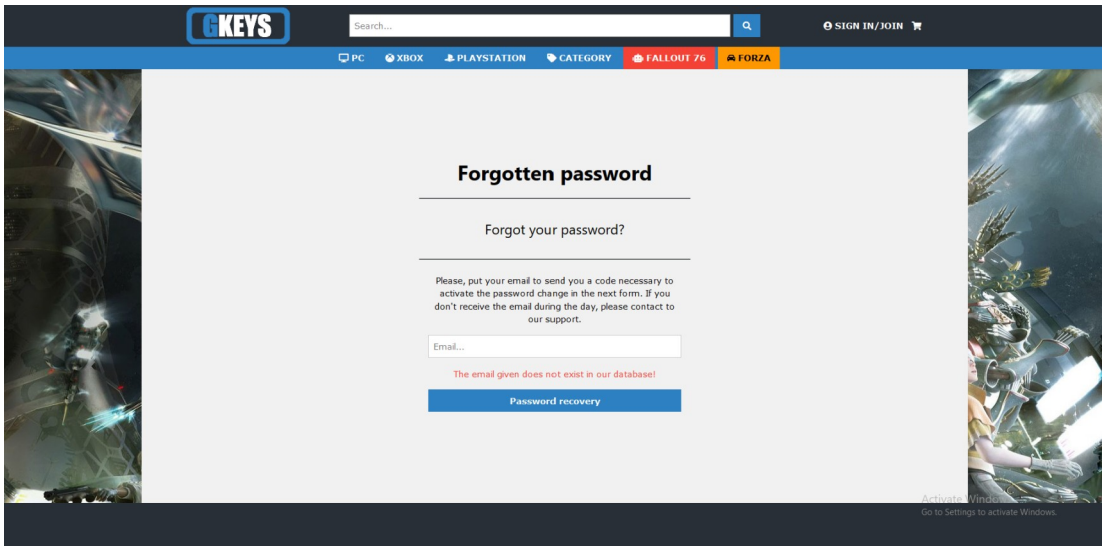


Figura 217: Resultat del cas 6 de la recuperació del compte d'usuari

6.12. Modificació de contrasenya

Cas 1: modificació de contrasenya amb dades vàlides

Entrada: *codi corresponent*, *sermf90@gmail.com*, *Gamin@Keys1@*, *Gamin@Keys1@* respectivament a cada camp.

Resultat: El sistema modifica la contrasenya de l'usuari i realitza una redirecció cap a la pàgina principal

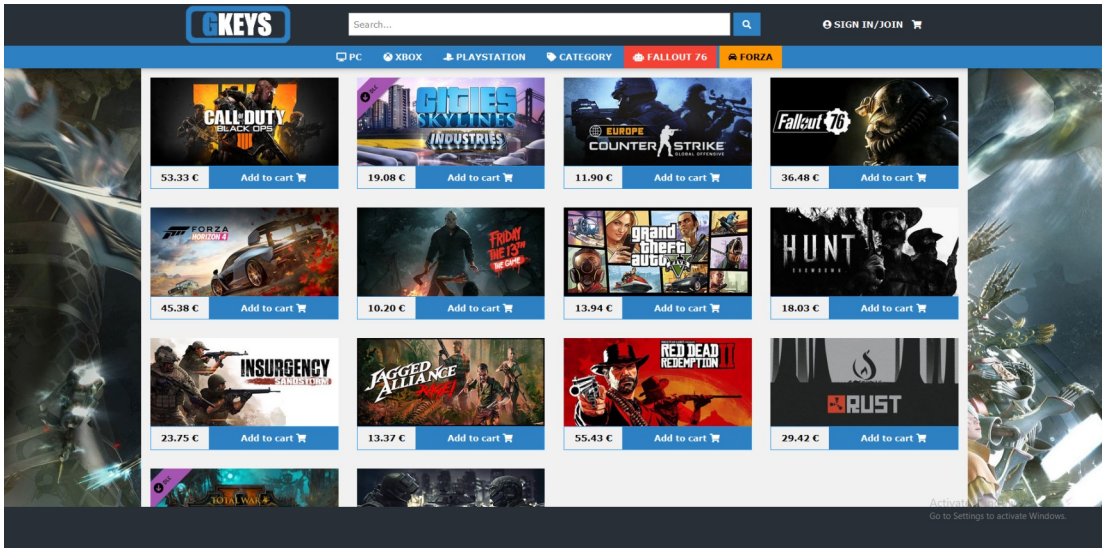


Figura 218: Resultat del cas 1 de la modificació de la contrasenya

Cas 2: modificació de contrasenya amb dades no vàlides

Entrada: *sermf90, X_X, 1234, 1234* respectivament a cada camp

Resultat: visualització del missatge d'error corresponent

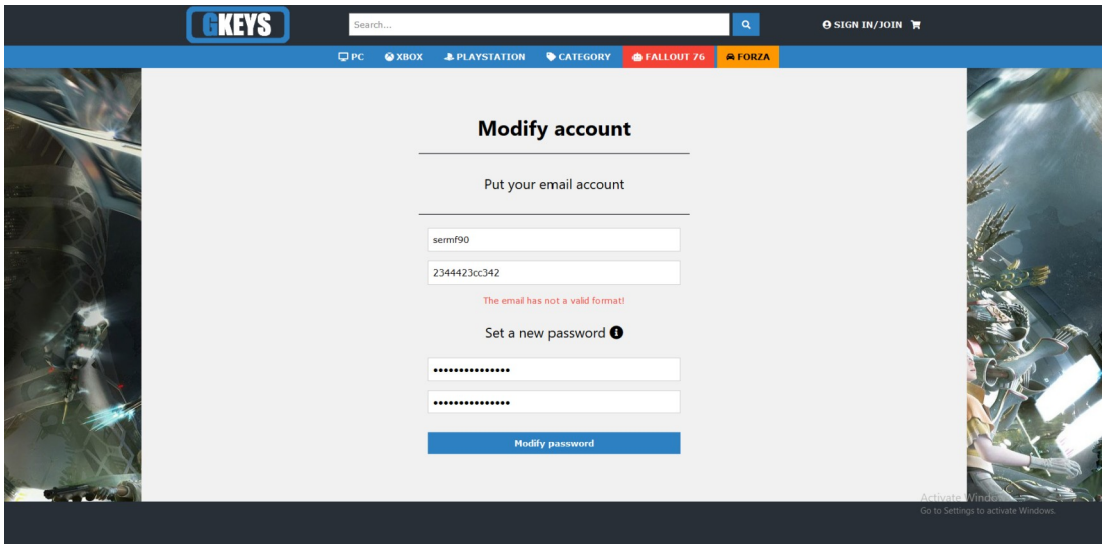


Figura 219: Resultat del cas 2 de la modificació de la contrasenya (correu no vàlid)

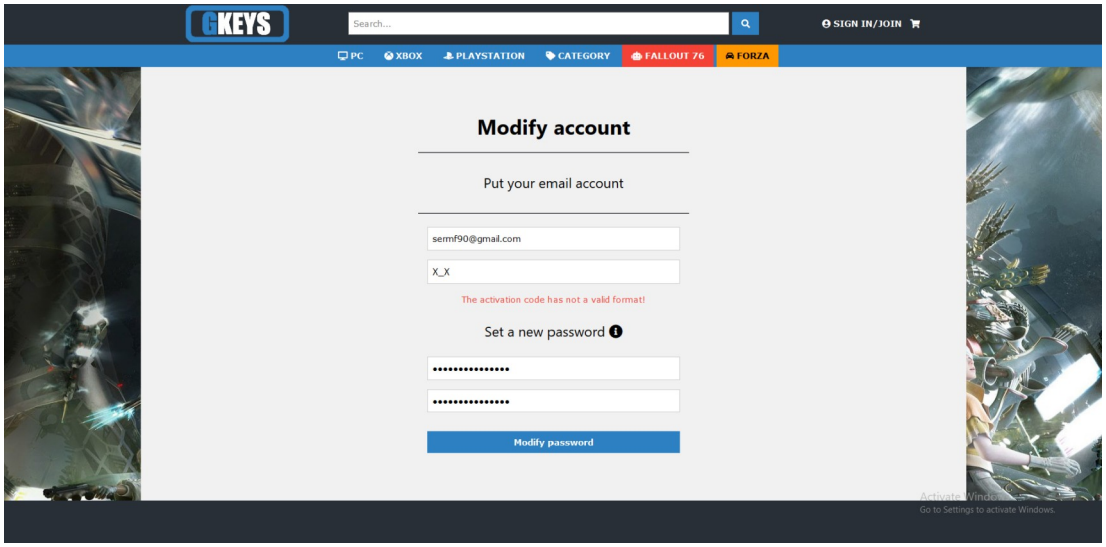


Figura 220: Resultat del cas 2 de la modificació de la contrasenya (codi no vàlid)

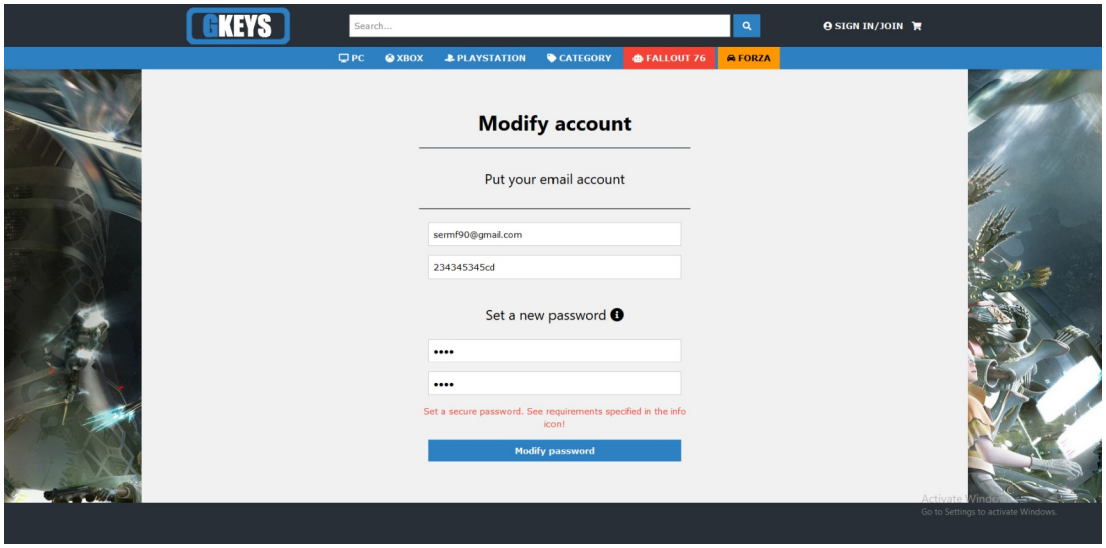


Figura 221: Resultat del cas 2 de la modificació de la contrasenya (contrasenya no vàlida)

Cas 3: modificació de contrasenya amb confirmació de contrasenya diferent a la contrasenya corresponent

Entrada: *Keys@Gaming10*, *Keys@10Gaming* respectivament a cada camp

Resultat: visualització del missatge d'error corresponent

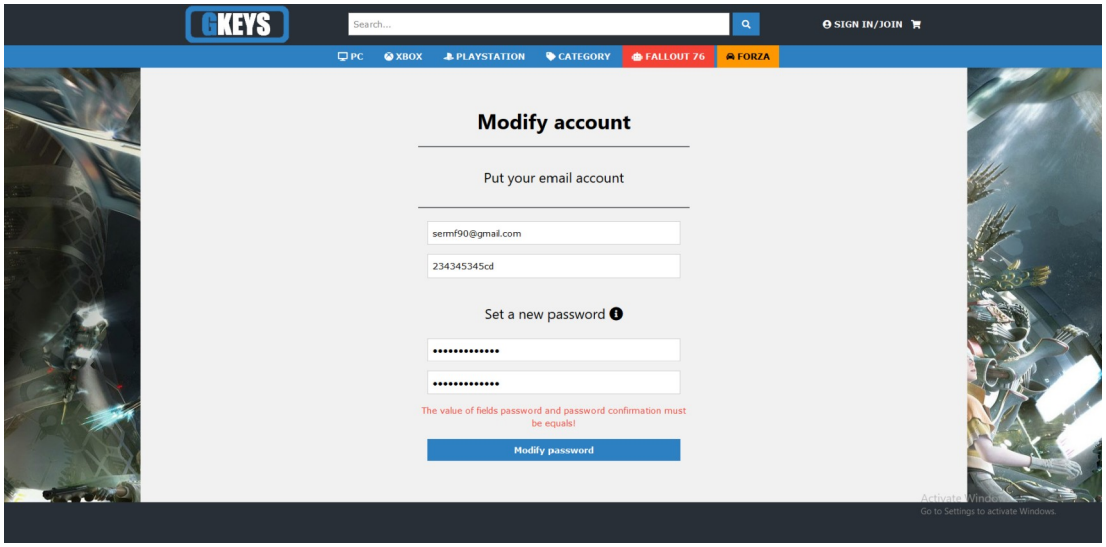


Figura 222: Resultat del cas 3 de la modificació de la contrasenya

Cas 4: inici de sessió d'usuari sense dades

Resultat: visualització del missatge d'error corresponent

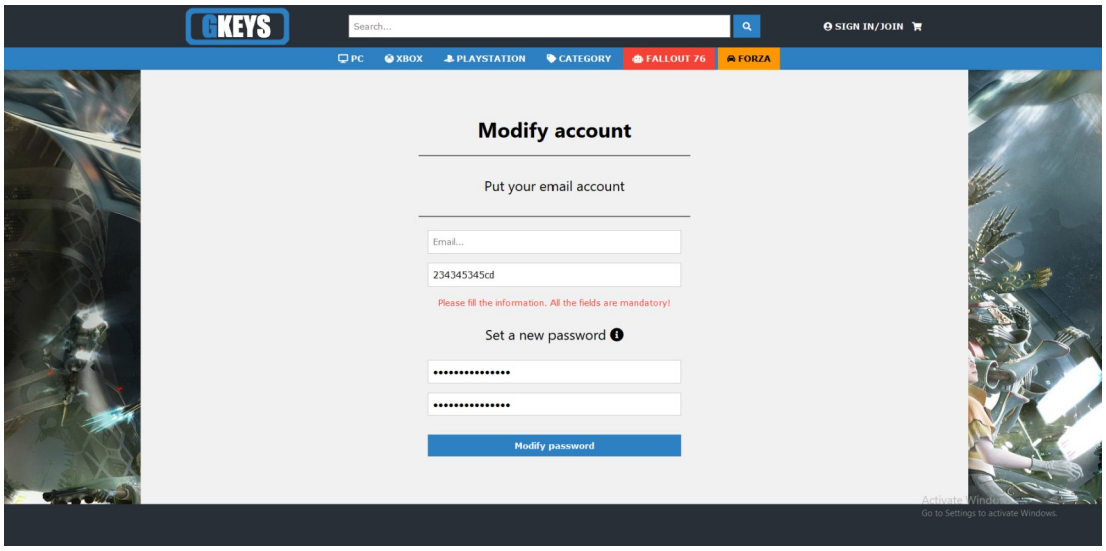


Figura 223: Resultat del cas 4 de la modificació de la contrasenya (correu buit)

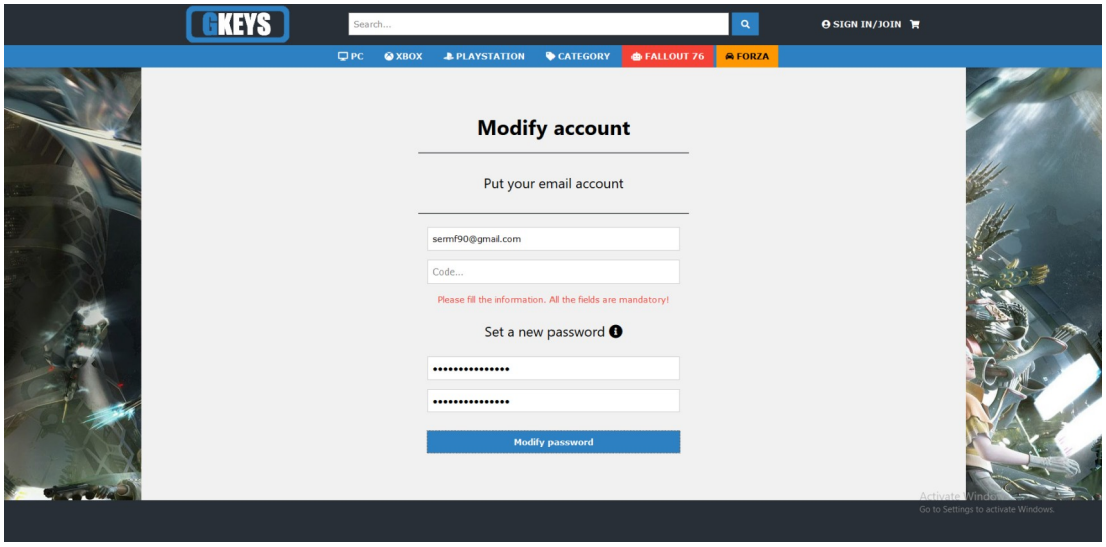


Figura 224: Resultat del cas 4 de la modificació de la contrasenya (codi buit)

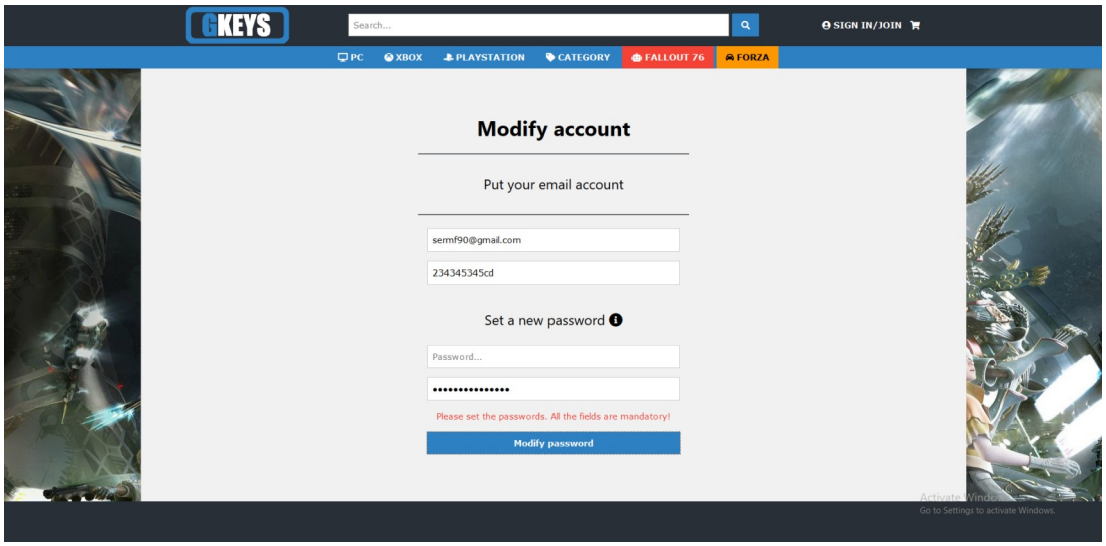


Figura 225: Resultat del cas 4 de la modificació de la contrasenya (contrasenya buida)

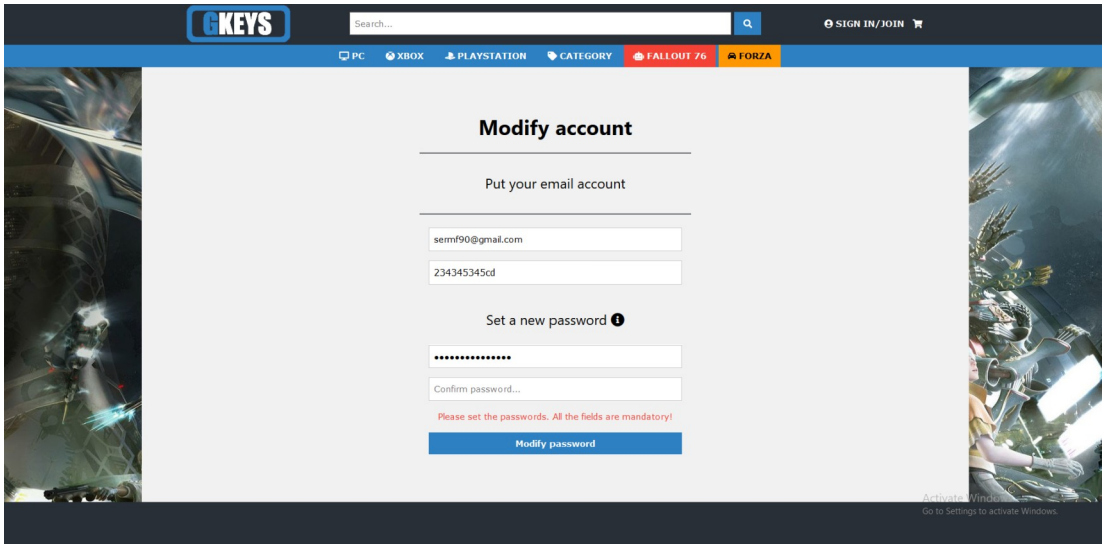


Figura 226: Resultat del cas 4 de la modificació de la contrasenya (confirmació de contrasenya buida)

Cas 5: modificació de contrasenya amb dades no vàlides (XSS)

Entrada: *<script>alert("XSS injection!")</script>* respectivament a cada camp

Resultat: visualització del missatge d'error corresponent

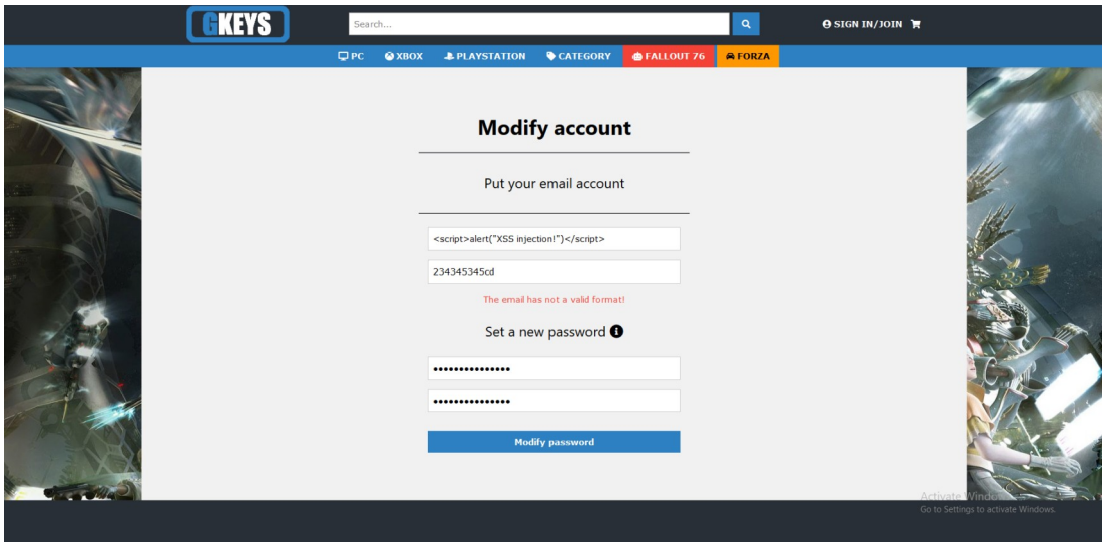


Figura 227: Resultat del cas 5 de la modificació de la contrasenya (correu malintencionat)

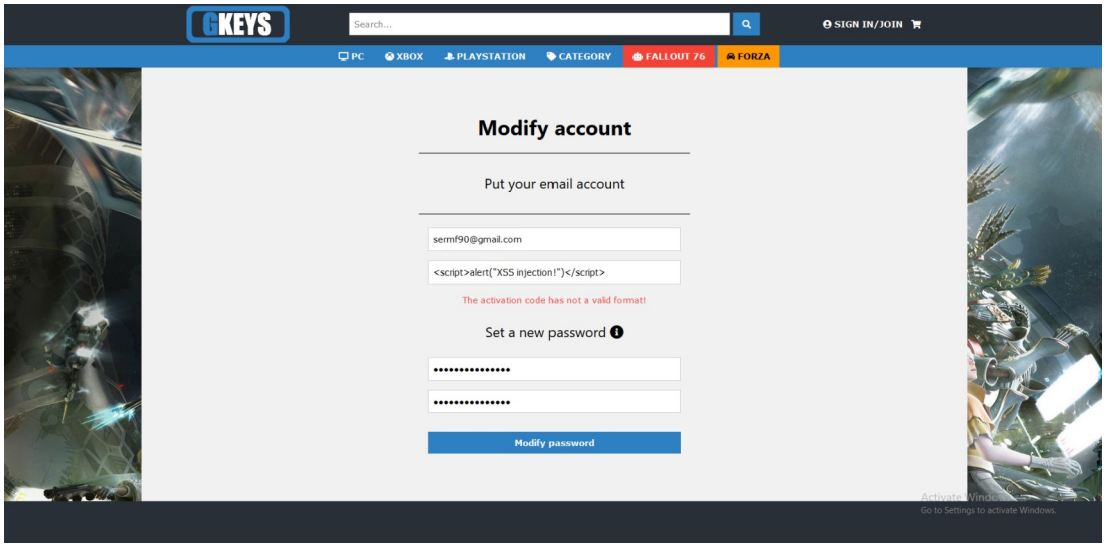


Figura 228: Resultat del cas 5 de la modificació de la contrasenya (codi malintencionat)

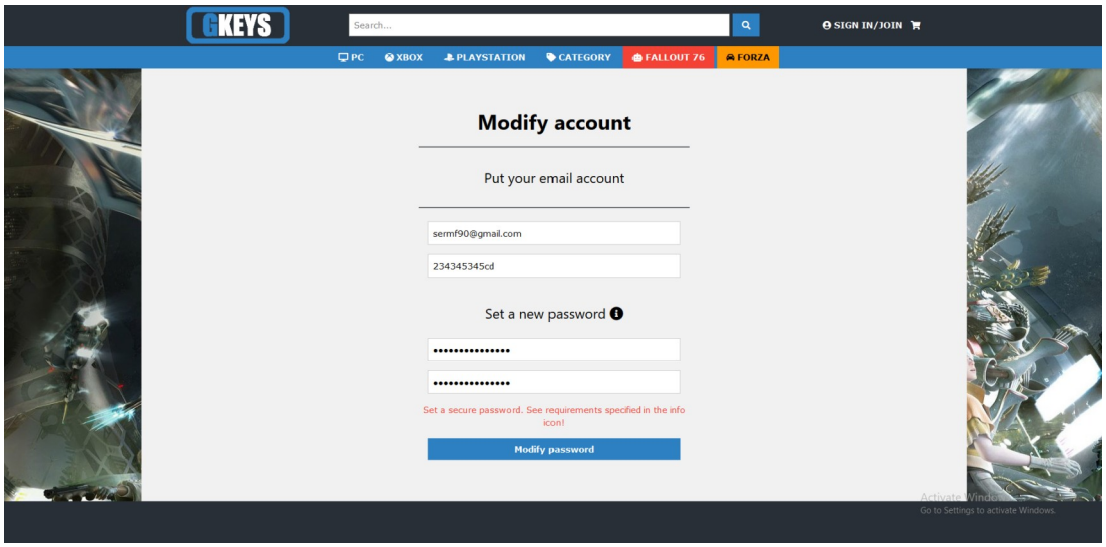


Figura 229: Resultat del cas 5 de la modificació de la contrasenya (contrasenya malintencionada)

Cas 6: modificació de contrasenya amb dades no vàlides (SQL Injection)

Entrada: *OR 1=1;*-- respectivament a cada camp

Resultat: visualització del missatge d'error corresponent

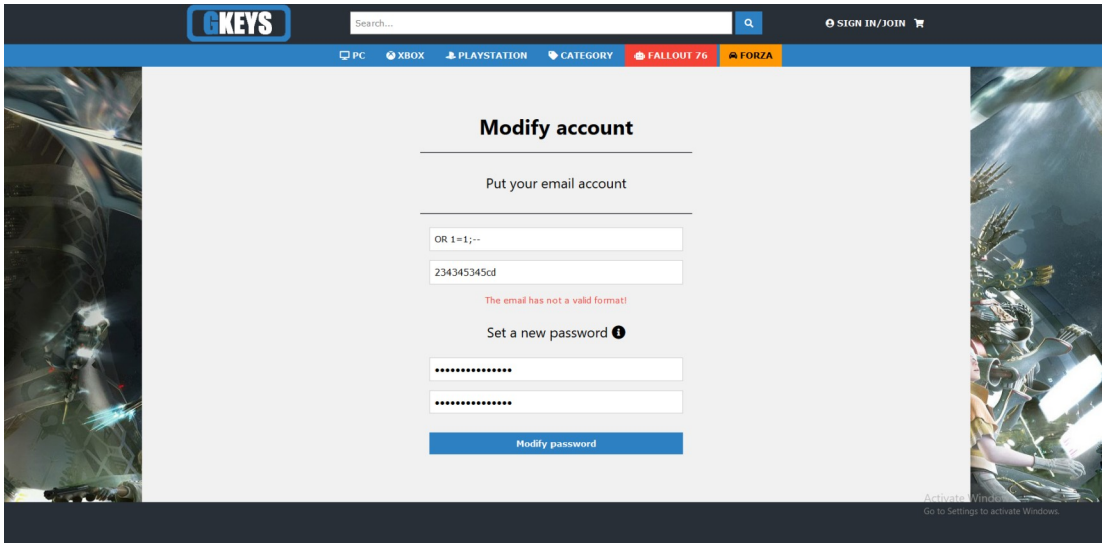


Figura 230: Resultat del cas 6 de la modificació de la contrasenya (correu malintencionat)

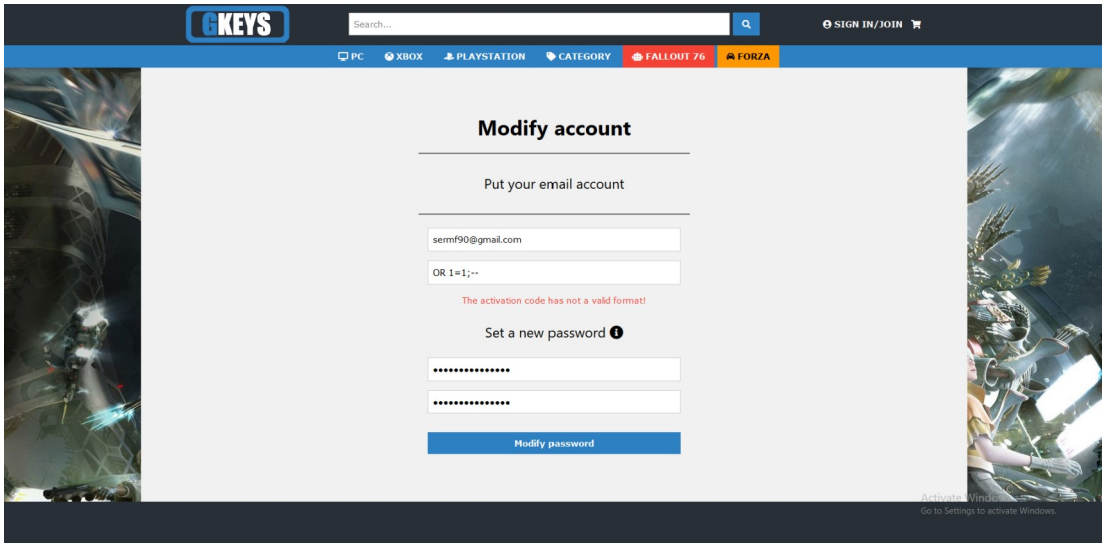


Figura 231: Resultat del cas 6 de la modificació de la contrasenya (codi malintencionat)

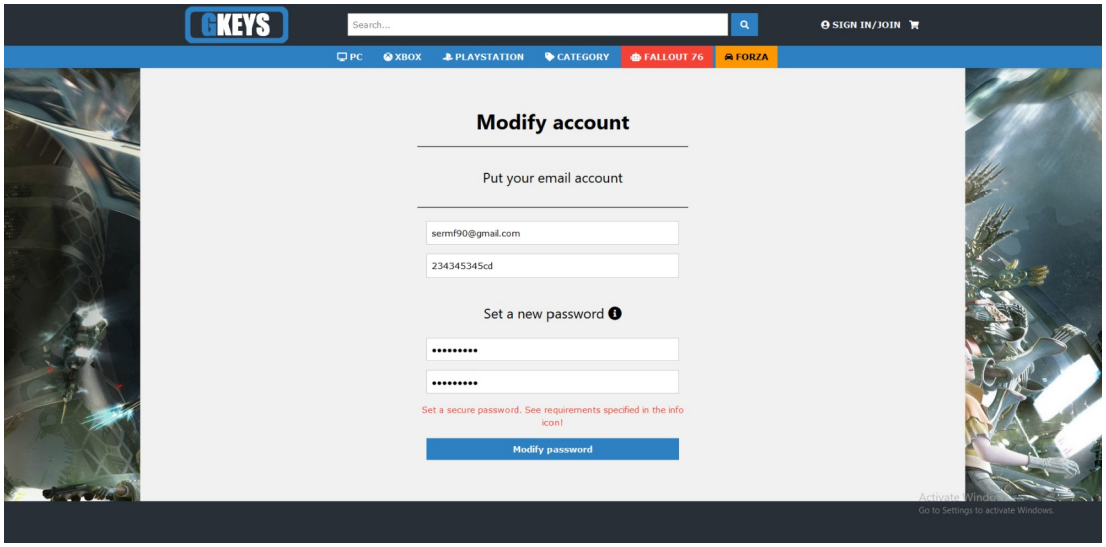


Figura 232: Resultat del cas 6 de la modificació de la contrasenya (contrasenya malintencionada)

Cas 7: modificació de contrasenya d'un usuari bloquejat

Entrada: qualsevol compte d'usuari bloquejat en el sistema

Resultat: visualització del missatge d'error corresponent

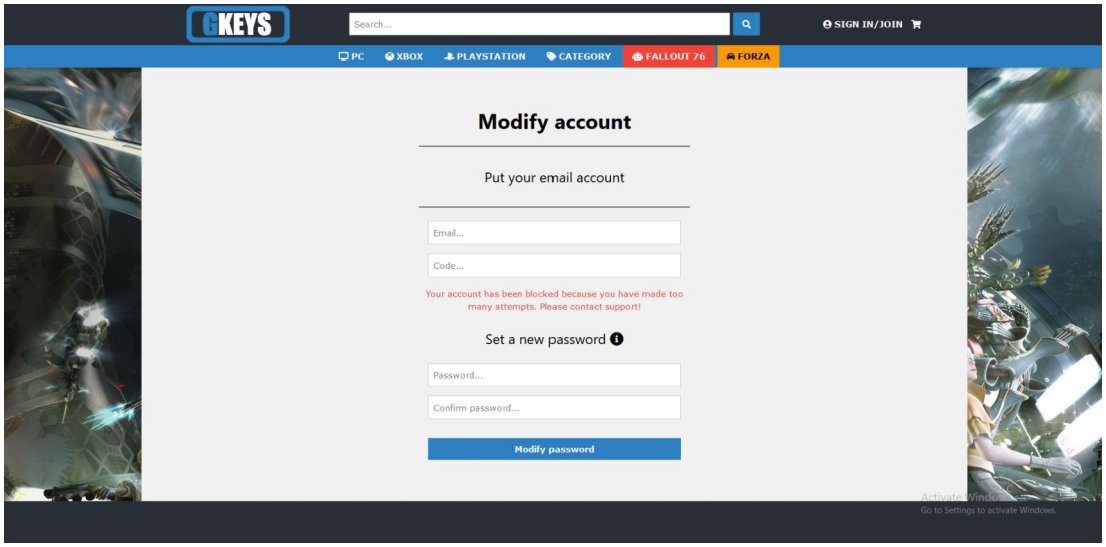


Figura 233: Resultat del cas 7 de la modificació de la contrasenya

Cas 8: modificació de contrasenya d'un usuari no existent

Entrada: qualsevol correu electrònic que no existeixi en el sistema

Resultat: visualització del missatge d'error corresponent

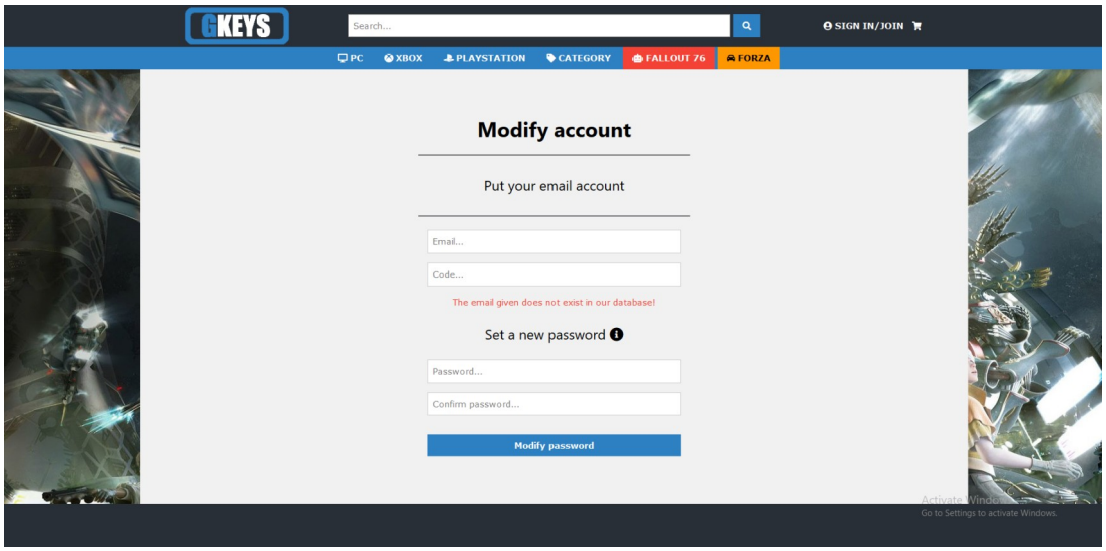


Figura 234: Resultat del cas 8 de la modificació de la contrasenya

Cas 9: modificació de contrasenya amb codi de recuperació incorrecte

Entrada: qualsevol correu electrònic que existeixi en el sistema i amb el codi recuperació amb format vàlid però incorrecte

Resultat: visualització del missatge d'error corresponent

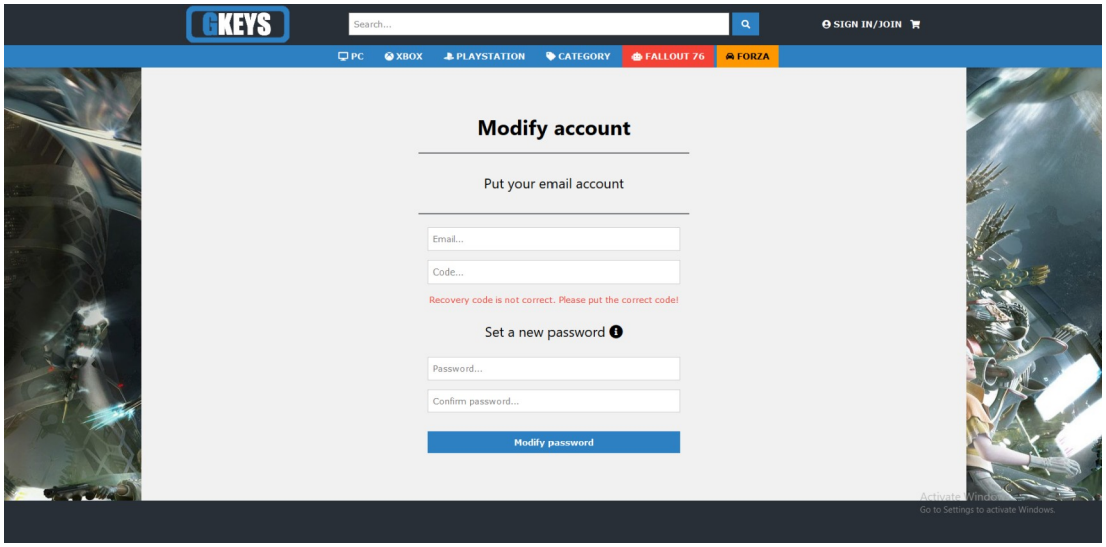


Figura 235: Resultat del cas 9 de la modificació de la contrasenya

6.13. Procés de pagament

Cas 1: inici del procés de pagament sense productes en el carret de la compra

Resultat: El sistema comprova el contingut del carret i realitza una redirecció cap a la pàgina principal

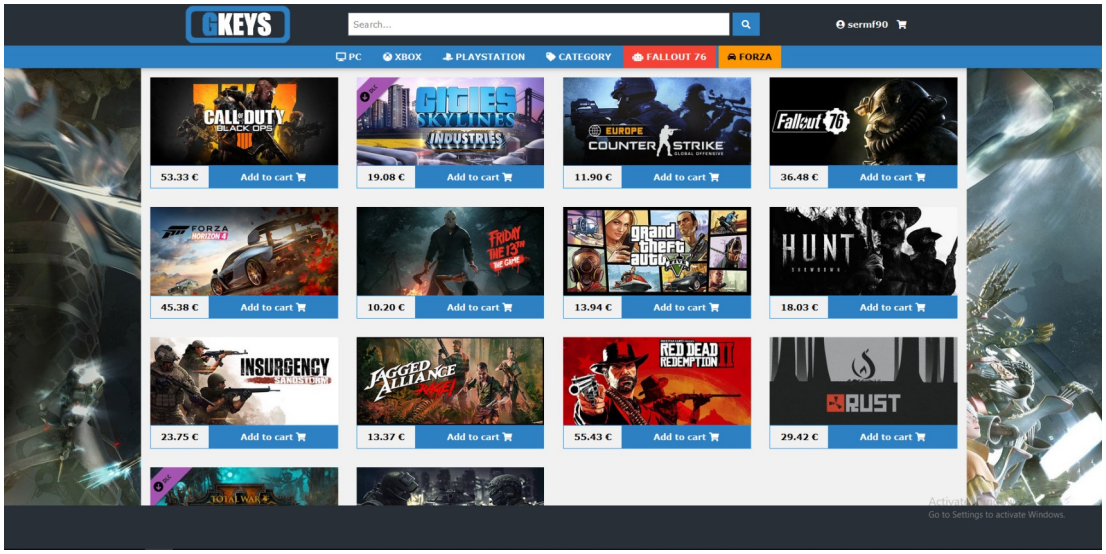


Figura 236: Resultat del cas 1 de l'inici del procés de pagament

Cas 2: inici del procés de pagament amb productes en el carret de la compra

Resultat: visualització del formulari de selecció del mètode de pagament

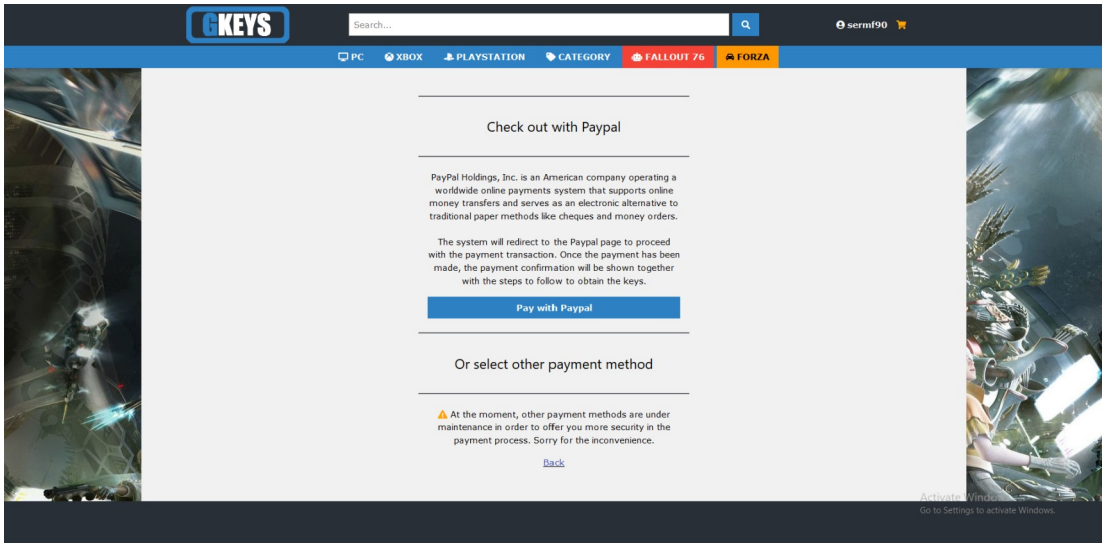


Figura 237: Resultat del cas 2 de l'inici del procés de pagament

Cas 3: procés de pagament realitzat amb èxit amb PayPal

Resultat: el sistema otorga les claus dels productes corresponents a l'usuari, actualitza l'estat del procés de compra en la base de dades i mostra el missatge de confirmació corresponent

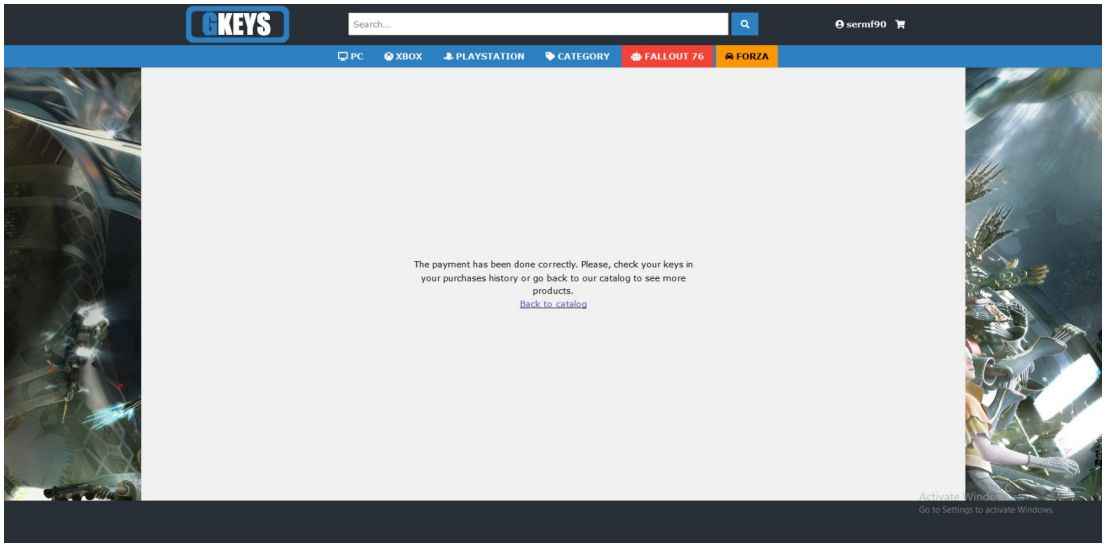


Figura 238: Resultat del cas 3 del procés de pagament

Cas 4: procés de pagament cancel·lat amb èxit amb PayPal

Resultat: el sistema actualitza l'estat del procés de compra en la base de dades i mostra el missatge de confirmació corresponent

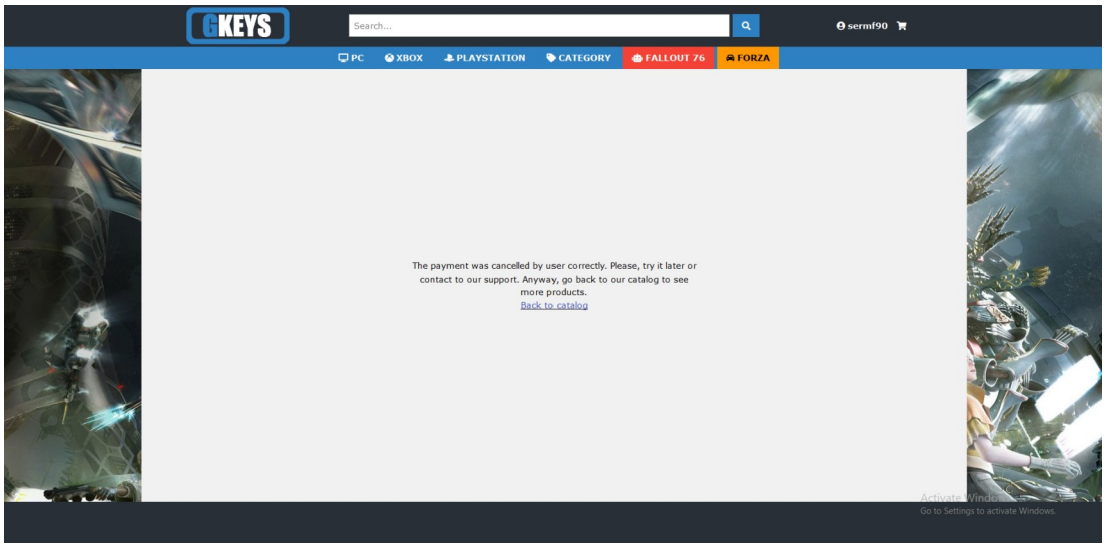


Figura 239: Resultat del cas 4 del procés de pagament

Cas 5: procés de pagament realitzat amb èxit amb PayPal però sense existències de claus pels productes comprats

Resultat: el sistema actualitza l'estat del procés de compra en la base de dades i mostra el missatge d'error corresponent

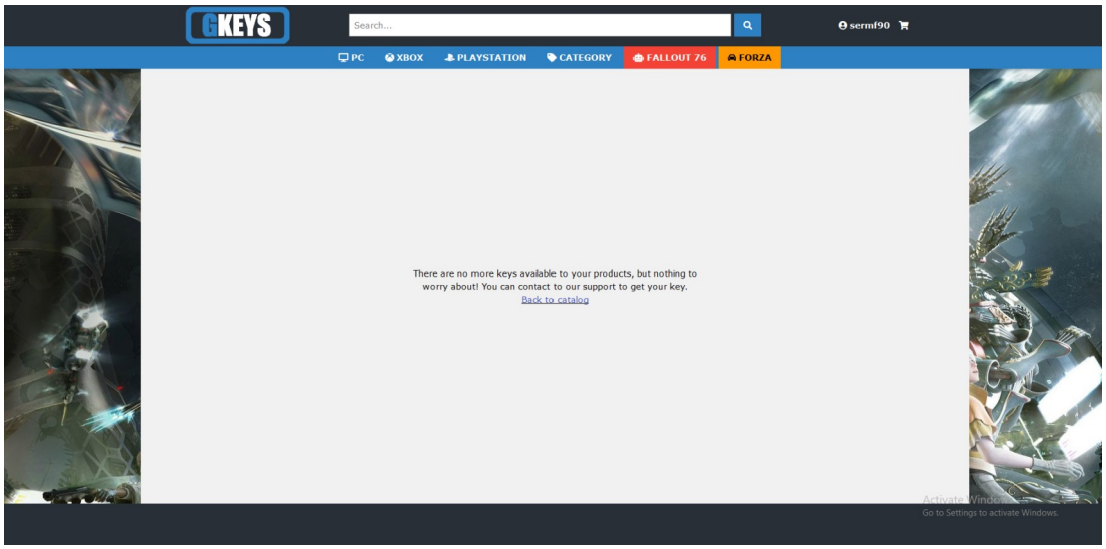


Figura 240: Resultat del cas 5 del procés de pagament

Cas 6: procés de pagament realitzat erròniament

Resultat: el sistema actualitza l'estat del procés de compra en la base de dades i mostra el missatge d'error corresponent

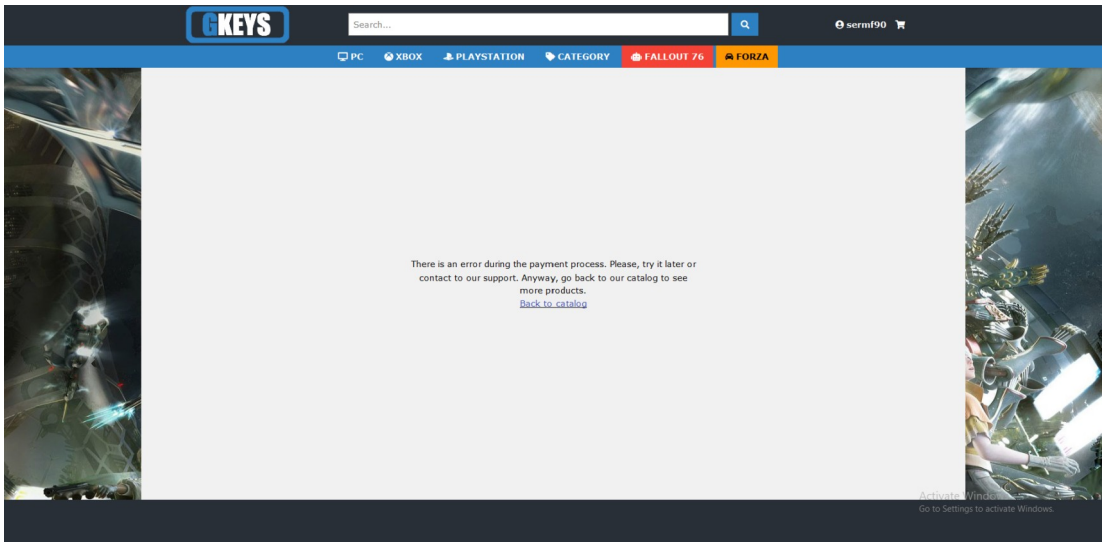


Figura 241: Resultat del cas 6 del procés de pagament

6.14. Historial de compres

Cas 1: visualització de l'historial de compres sense haver realitzat compres prèviament

Resultat: visualització del missatge d'error corresponent

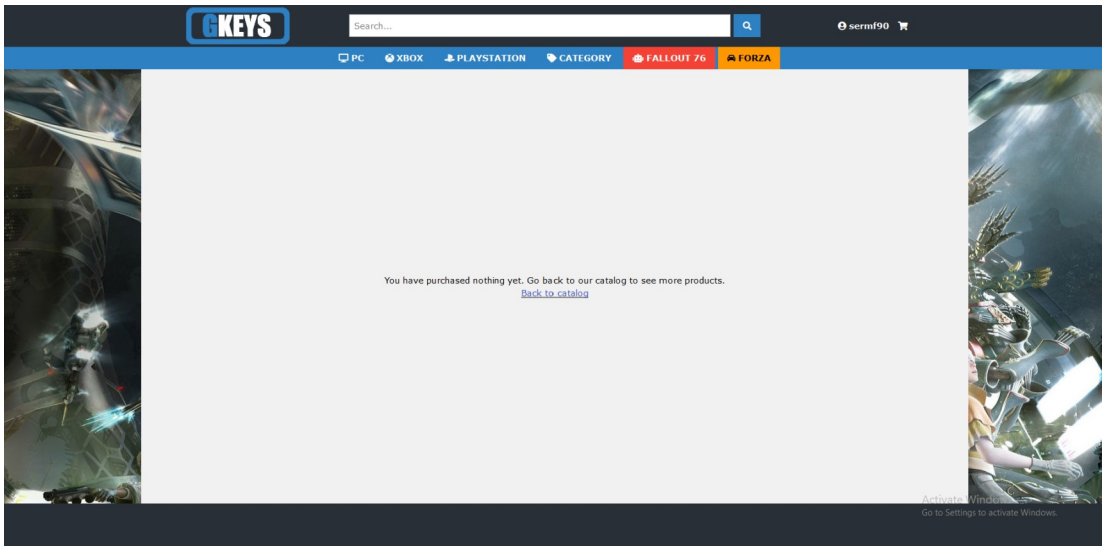


Figura 242: Resultat del cas 1 de l'historial de compres

Cas 2: visualització de l'historial de compres havent realitzat compres prèviament

Resultat: visualització del llistat dels productes amb les claus corresponents no visibles

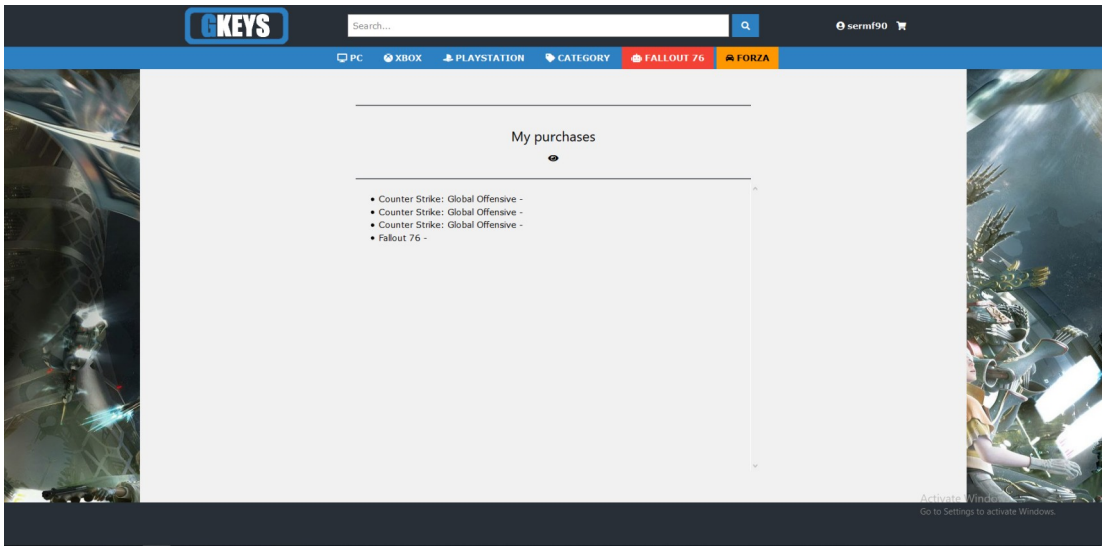


Figura 243: Resultat del cas 2 de l'historial de compres

Cas 3: visualització de les claus dels productes de l'historial de compres fent clic a la icona

Resultat: visualització del llistat dels productes amb les claus corresponents visibles

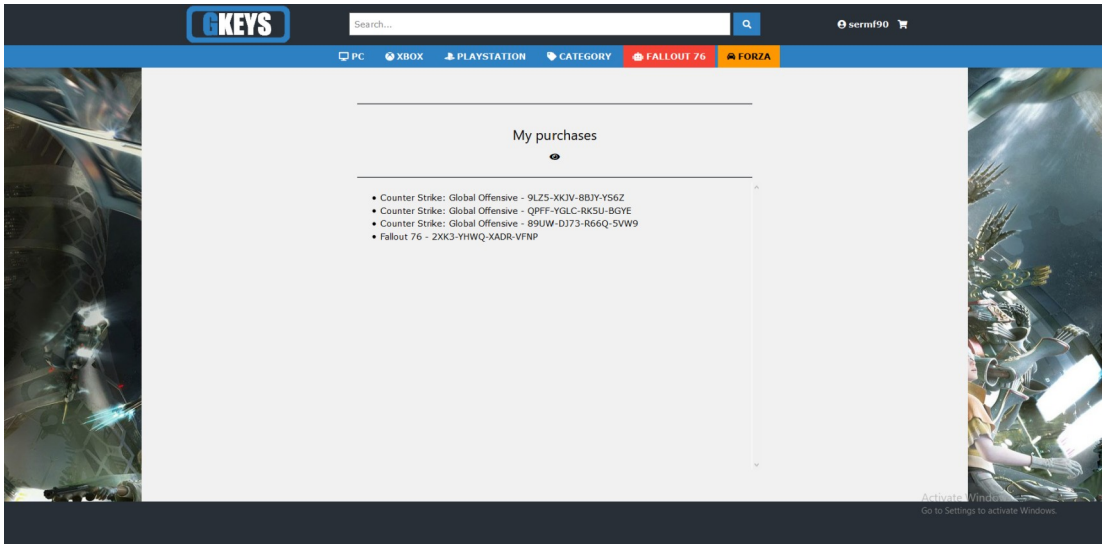


Figura 244: Resultat del cas 3 de l'historial de compres

Cas 4: ocultació de les claus dels productes de l'historial de compres, un cop visibles, fent clic a la icona

Resultat: exactament com el cas 2 d'aquest apartat

6.15. Tancar sessió

Cas 1: tancar sessió d'usuari amb sessió iniciada

Resultat: el sistema elimina tota la informació desada en la sessió i redirecciona cap a la pàgina principal

Cas 2: tancar sessió d'usuari amb sessió no iniciada

Resultat: el sistema elimina tota la informació desada en la sessió i redirecciona cap a la pàgina principal

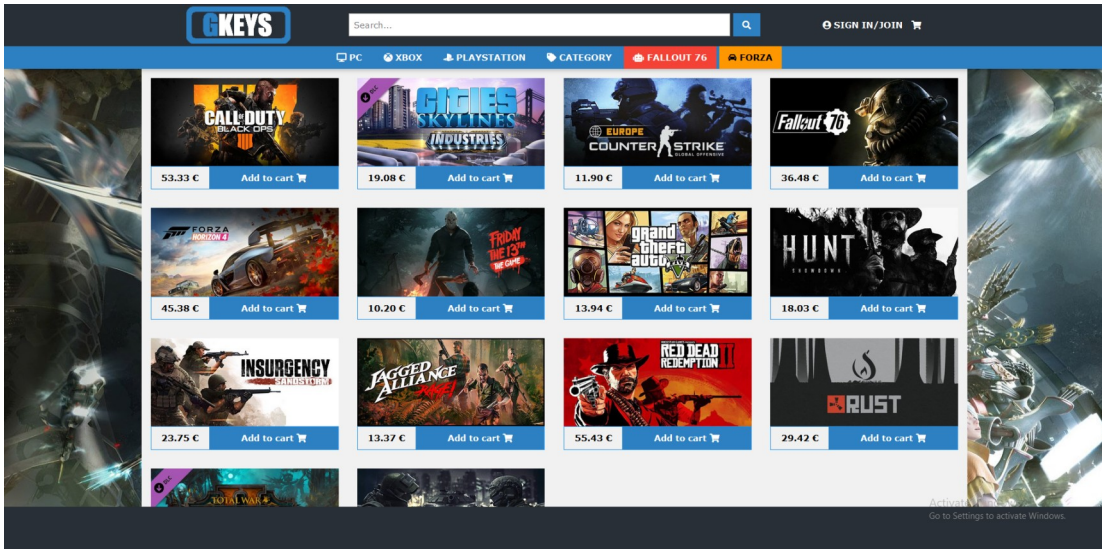


Figura 245: Resultat dels casos de tancar sessió d'usuari

6.16. Control de sessió

Cas 1: accés a una vista amb sessió d'usuari que requereix iniciada la sessió d'usuari

Resultat: el sistema continua amb el procediment i mostra el contingut de la vista

Cas 2: accés a una vista sense sessió d'usuari que requereix iniciada la sessió d'usuari

Resultat: el sistema realitza una redirecció cap al formulari d'inici de sessió

Cas 3: accés a una vista amb sessió d'usuari que requereix que no hagi cap sessió iniciada

Resultat: el sistema continua amb el procediment i mostra el contingut de la vista

Cas 4: accés a una vista amb sessió d'usuari que requereix que no hagi cap sessió iniciada

Resultat: el sistema mostra el contingut de la vista

7. Extensions

En aquest sistema es poden implementar noves funcionalitats i millores que, de moment, queden fora de l'abast del projecte. Aquestes propostes es subdivideixen en dos subapartats diferents, separant els evolutius funcionals i les implementacions de seguretat.

7.1. Extensions funcionals

En aquest apartat s'esmenten, molt resumidament, algunes de les propostes que es poden implementar en el sistema en l'àmbit funcional. Aquestes propostes són les següents:

- **Gestió de rols d'usuaris:** Es podria donar diferents rols als usuaris del sistema per poder duu a terme l'administració de la pàgina i del contingut, així com la revisió i la vigilància dels usuaris.

- **Administrador de contingut (CMS):** Es podria implementar un sistema d'administració de continguts per gestionar dinàmicament i en un entorn més adequat tot el que fa referència a la gestió dels productes (afegir, modificar o eliminar productes, aplicar descomptes, determinar productes destacats, etc).

- **Sistema d'anàlisis de negoci:** Es podria implementar un sistema d'anàlisi d'informació de negoci amb l'objectiu de mostrar-la gràficament i poder prendre les decisions corresponents per la millora de la botiga i del negoci.

- **Venta de caixes amb jocs aleatoris:** Es podria vendre, com un producte més de la botiga, unes caixes les quals conté un joc d'un llistat de jocs amb diferents probabilitats. Es tractaria com una espècie de joc d'atzar, on les probabilitats implementades i la sort determina el premi del client.

- **Moneda virtual interna:** Es podria implementar un sistema de pagament virtual propi de la botiga, on l'usuari podria comprar monedes amb diners reals i utilitzar-les per realitzar les compres corresponents.

- **Sistema de puntuació:** El client, un cop finalitzada la compra, podria puntuar i realitzar comentaris sobre els productes adquirits. D'aquesta manera s'ofereix a l'usuari una manera d'expresar la seva opinió sobre els productes o la botiga i, al mateix temps, mostrar fiabilitat de cara a nous clients.

- **Informació de l'empresa i contacte**: Tot i ser un dels aspectes més bàsics de qualsevol empresa, no ha sigut possible, concretament, en aquest projecte. S'ha de mostrar al client tota la informació necessària sobre l'empresa de Gaming Keys, juntament amb els mètodes i dades de contacte per solucionar qualsevol tipus de dubte o incidència.

- **Mètodes de pagament més utilitzats:** Cal oferir a l'usuari els mètodes de pagament més utilitzats per captar el màxim de clients possibles. En aquest cas, s'està oferint només Paypal.

7.2. Extensions de seguretat

En aquest apartat s'esmenten, molt resumidament, algunes de les propostes que es poden implementar en la botiga per incrementar la seguretat en tots els aspectes possibles. Aquestes propostes són les següents:

- **Xifrat de la base de dades:** És un requisit que la informació desada estigui protegida contra l'accés no desitjat per part d'altres usuaris. En aquest cas, es protegeix la informació com la contrasenya de l'usuari, però es pot implementar una funcionalitat per protegir tota la informació necessària.

- **Còpies de seguretat:** Aquest aspecte és fonamental per preservar tota la informació que fa referència a *Gaming Keys*. En cas que el servidor deixi de funcionar, que un atacant modifica o elimina la informació de la base de dades o qualsevol altra incidència, el sistema ha d'estar preparat per oferir una alternativa al client i assegurar la integritat de la informació.

- **Ús de certificats SSL/TLS**: L'ús d'entitats certificadores per l'obtenció d'un certificat oferirà als clients la confiança necessària per realitzar les operacions necessàries amb total seguretat.

  Cal esmentar que cal tenir-los actualitzats a les últimes versions, degut que ja s'han conegut casos d'intercepcions d'informació exitoses, en versions més antigues, mitjançant una eina coneguda amb el nom de BEAST (*Browser Exploit Against SSL/TLS*).

- **Monitoratge:** Cal implementar un sistema de monitoratge que permeti recollir tota la informació possible sobre el comportament dels usuaris per intentar evitar qualsevol comportament no desitjat dins de la botiga i dur a terme un registre dels esdeveniments dins d'aquesta.

- **Bloqueig per ip**: En cas que una adreça ip estigués realitzant moltes peticions seguides pot indicar que és possible que estigui intentant realitzar una acció malintencionadament, com per exemple atacs DoS o DDoS. Per tant, s'hauria de configurar el servidor de tal manera que realitzés un bloqueig segons la ip d'origen.

- **Auditories:** Cal realitzar periòdicament auditories, sigui internament o externament, com per exemple una entitat certificadora, per garantir la seguretat en tot el sistema i poder oferir una experiència de qualitat als clients.

- **Política de privacitat:** és un document legal que planteja com una organització processa les dades de l'usuari. Aquest document comporta un contracte en el qual l'empresa garanteix mantenir la informació personal de l'usuari amb motius funcionals. Més concretament: informació recollida, *cookies*, enllaç a tercers i control de la informació.

  Per tant, la botiga ha de fer visible la informació referent a aquest àmbit per l'usuari, on aquest ha de comprendre i acceptar les condicions d'ús, degut que, en cas contrari, l'ús d'aquesta informació en el sistema es consideraria una violació de privacitat.

- **Avís de l'ús de cookies**: Segons la llei vigent, es determina que hi ha dos tipus de cookies, una d'elles són les que requereixen autorització per part de l'usuari i les altres no. Per tant, cal mostrar una pantalla a l'usuari on s'informi que el sistema farà ús de les cookies, quina serà la informació que es desarà i com s'utilitzarà. En cas contrari, les cookies que no requereixen cap mena d'autorització són les que són estrictament necessàries per al funcionament del sistema web en l'àmbit tècnic, com per exemple la informació referent al carret de la compra.

- **Implantació del Captcha[11]:** Aquests tipus de proves consisteixen en la introducció per part de l'usuari d'un conjunt de caràcters que es mostren en una imatge distorsionada per pantalla, amb l'objectiu de determinar quan un usuari és humà i quan no. Aquesta prova es pot implementar, per exemple, per evitar les possibles operacions per part de bots.

- **Compliment de la normativa PCI DSS[12]:** En cas d'integrar mètodes de pagament que comportin l'ús de targetes de pagament (dèbit o crèdit), és estrictament necessari el compliment de totes les especificacions que determina l'estàndard i la seva validació de manera periòdica. Concretament, de manera resumida:

  - Desenvolupar i mantenir una xarxa segura.

  - Protegir les dades dels propietaris de les targetes.

  - Mantenir un programa de gestió de vulnerabilitats.

  - Aplicar mesures sòlides de control d'accés.

  - Monitorar i provar regularment les xarxes.

  - Mantenir una política de seguretat de la informació.

- **Gestió del frau**: En cas d'oferir en *Gaming Keys* diferents mètodes de pagament, cal integrar un sistema que gestioni i protegeixi la botiga contra les transaccions fraudulentes. Actualment, existeixen diferents entitats que ofereixen els seus serveis, com per exemple CyberSource[13] o Riskified[14].

8. Bibliografia

- *[1] Sublime text, website of the project. URL: https://www.sublimetext.com/*
- *[2] Apache HTTP Server, website of the project. URL: https://httpd.apache.org/*
- *[3] PHP Hypertext Preprocessor, website of the project. URL: http://php.net/*
- *[4] MySQL, website of the project. URL: https://www.mysql.com/*
- *[5] Composer, website of the project. URL: https://getcomposer.org/*
- *[6] PHPMailer, website of the information and source code of the project. URL: https://github.com/PHPMailer/PHPMailer*
- *[7] PayPal, website of the company. URL: https://www.paypal.com/ni/home*
- *[8] W3.CSS, website of the framework. URL: https://www.w3schools.com/w3css/*
- *[9] Font Awesome, website of the project. URL: https://fontawesome.com/*
- *[10] Transport Layer Security, website of information about it. URL: https://es.wikipedia.org/wiki/Transport_Layer_Security*
- *[11] Captcha, website of information about it. URL: https://es.wikipedia.org/wiki/Captcha*
- *[12] Payment Card Industry Data Security Standard, website of the standard. URL: https://www.pcisecuritystandards.org/*
- *[13] CyberSource, website of the company. URL: https://www.cybersource.com/es-ES/*
- *[14] Riskified, website of the company. URL: https://www.riskified.com/*