

# WEB SUSTAV ZA PRIKUPLJANJE PODATAKA O IGRAČIMA RAČUNALNIH IGARA SA RAZLIČITIH IGRAČIH PLATFORMI

---

**Mikolić, Marijan**

**Undergraduate thesis / Završni rad**

**2018**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **Algebra University College / Visoko učilište Algebra**

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:225:299012>

*Rights / Prava:* [In copyright](#) / [Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2024-09-01**



*Repository / Repozitorij:*

[Algebra University - Repository of Algebra University](#)



**VISOKO UČILIŠTE ALGEBRA**

ZAVRŠNI RAD

**WEB SUSTAV ZA PRIKUPLJANJE  
PODATAKA O IGRAČIMA RAČUNALNIH  
IGARA SA RAZLIČITIH IGRAČIH  
PLATFORMI**

Marijan Mikolić

Zagreb, 02. 2018.

*„Pod punom odgovornošću pismeno potvrđujem da je ovo moj autorski rad čiji niti jedan dio nije nastao kopiranjem ili plagiranjem tuđeg sadržaja. Prilikom izrade rada koristio sam tuđe materijale navedene u popisu literature, ali nisam kopirao niti jedan njihov dio, osim citata za koje sam naveo autora i izvor, te ih jasno označio znakovima navodnika. U slučaju da se u bilo kojem trenutku dokaže suprotno, spreman sam snositi sve posljedice uključivo i poništenje javne isprave stečene dijelom i na temelju ovoga rada“.*

*U Zagrebu, 22.2.2018.*

# Predgovor

Zahvaljujem mentoru Ivanu Božajiću, struč. spec. ing. sec. koji je nadgledao izradu završnog rada i preporučio tehnologije i metode koje su omogućile realizaciju ovog projekta te bez čijih predavanja nikada ne bih postigao znanja potrebna da shvatim da je ovakav sustav moguć.

**Prilikom uvezivanja rada, Umjesto ove stranice ne zaboravite umetnuti original potvrde o prihvaćanju teme završnog rada kojeg ste preuzeli u studentskoj referadi**

## Sažetak

Video igre danas su jedan od najraširenijih oblika zabave u kojem uživaju ljudi razne dobi diljem svijeta. Zbog razvoja tehnologije, video igre proširile su se na razne svakodnevne uređaje kao što su pametni telefoni te je zbog toga danas broj ljudi koji redovno igraju video igre velik. Video igre igračima predstavljaju zabavu, bijeg od stvarnosti, priliku za socijalizaciju te ponekad umjetnički izražaj.

U današnje digitalno doba, većina igara povezuje se na online platforme putem kojih se prate statistike o igranju. No, svaki veći izdavač igara ima svoju platformu, što znači da je ukupna statistika nekog igrača razbacana po Internetu. Mnogim igračima je stalo do tih statistika te ih smeta što su tako fragmentirane. Zbog toga se rodila ideja da se izradi usluga koja će te podatke sakupiti i obraditi kako bi se prikazali na jednom mjestu smatrajući da je to proizvod za koji će ljudi biti zainteresirani.

Playtracker je web sustav za ljubitelje videoigara koji skuplja podatke sa raznih igračih platformi i objedinjuje ih na jedinstveni korisnički profil. Rad obuhvaća istraživanje i dizajn, podešavanje infrastrukture, te kodiranje cijelog sustava od PHP back-enda do HTML/CSS/JS front-enda. Prije izrade rada autor je bio upoznat s korištenim metodama i tehnologijama, no kako bi konačni proizvod bio kompletan i spreman za javnost morao je proširiti znanje u tim poljima dodatnim tehnikama kao što je korištenje službenih metoda za pristup podacima neke usluge, znanih kao „API“, bez kojih projekt ne bi bio moguć.

Rad demonstrira proces izrade samostalnog proizvoda za tržište weba i na što je bitno paziti pri tom procesu, objašnjava odluke i opisuje tehnička ograničenja. Također opisuje korištenje API-eva i kako strugati podatke s web stranica platformi koje nemaju API.

**Ključne riječi:** web sustav, API, struganje podataka, videoigre, korisnički profil.

# Sadržaj

1.	Uvod .....	1
2.	Projekt.....	2
2.1.	Osnovni opis projekta.....	2
2.2.	Kome je projekt namijenjen i zašto je potreban .....	2
2.3.	Pregled izvora podataka.....	3
2.4.	Pregled programskih jezika i alata koji se koriste u izradi .....	4
2.5.	Planiranje projekta – obuhvat i organizacija .....	4
2.6.	Infrastruktura projekta .....	5
3.	Front-end i dizajn aplikacije .....	6
3.1.	Programska struktura.....	6
3.2.	Vizualni identitet i brend .....	7
3.2.1.	Ime i logo.....	7
3.2.2.	Vizualni identitet .....	8
3.3.	Struktura i osnovni raspored i stil stranice .....	9
3.3.1.	Naslovnica .....	9
3.3.2.	Kontrolna ploča korisnika .....	9
3.3.3.	Profil korisnika .....	11
3.4.	Korisničko sučelje .....	13
3.5.	Testiranje korisničkog sučelja .....	14
3.6.	Praćenje i analitika.....	15
4.	Back-end aplikacije .....	17
4.1.	Programska struktura.....	17
4.2.	Asinkroni prikaz i dohvaćanje podataka .....	17

4.3.	Baza podataka.....	18
4.3.1.	Odabir i uloga baze podataka .....	18
4.3.2.	Tablice i organizacija.....	18
4.4.	Application Interface – API.....	19
4.4.1.	Uloga API-a u projektu .....	20
4.4.2.	Pregled korištenih API-a .....	20
4.5.	Struganje podataka s web stranice.....	21
4.6.	Obrada podataka.....	22
	Zaključak .....	24
	Popis kratica .....	25
	Popis slika.....	26
	Literatura .....	27



# 1. Uvod

Od malih nogu autor ovog rada volio je video igre, počevši od svoje prve igraće konzole – originalnog PlayStationa. I danas ih igra svakodnevno, prati najnovije vijesti u industriji videoigara te sudjeluje u online raspravama na tu temu. Jedna od stvari koje je primijećena je nezadovoljstvo mnogih igrača zbog fragmentacije tržišta digitalnih platformi koje prodaju igre i prate statistiku o igranju. Nije im se svidjelo kako ne mogu dobiti potpuni pregled svojih statistika preko svih platformi te kako moraju instalirati posebne aplikacije kako bi pokrenuli jednu specifičnu igru nekog izdavača koja ih zanima. Tako je došla ideja da se izradi neko rješenje za taj problem – web sustav koji bi skupio sve te podatke na jedno mjesto. Web je savršen medij za takvo rješenje jer pokriva sve uređaje popularne među igračima (računala, mobilni uređaji, igraće konzole) te koristi web preglednik umjesto da tjera igrače da instaliraju dodatne aplikacije.

Web sustav se zove PlayTracker i njegova svrha je prikupljati podatke o igračima računalnih igara s raznih dostupnih igračkih platformi te zatim te podatke obraditi i prikazati na jednom urednom i atraktivnom profilu, što je privlačna mogućnost mnogim igračima koji vole svoj napredak kroz igre dijeliti s drugima.

Ovaj završni rad za cilj je imao stvaranje potpunog proizvoda od početka do kraja što znači da pokriva potpunu tematiku izrade web stranica od planiranja i dizajna, i back-end i front-end programiranje, kao i podešavanje poslužitelja da obavlja posao služenja web sustava.

Opseg projekta je pozamašan i obuhvaća vještine iz mnogo polja te poznavanje više programskih jezika. Već sam poznavao jezike – HTML, CSS, JS, PHP, SQL, no moja znanja nisu bila dovoljna za izradu potpunog proizvoda spremnog za tržište te ih je bilo potrebno proširiti svladavanjem novih tehnika, poglavito unutar PHP jezika pomoću kojega se dohvaćaju i obrađuju podaci.

## 2. Projekt

### 2.1. Osnovni opis projekta

PlayTracker je web sustav za ljubitelje videoigara koji putem API-ja skuplja podatke sa raznih igračih platformi i objedinjuje ih na jedinstveni profil time rješavajući problem fragmentiranog tržišta igračih platformi. Sustav je javno dostupan i krajnjem korisniku omogućuje pretraživanje baze igrača te stvaranje i podešavanja vlastitog profila.

Korisnički profil prikazuje statistiku o korisniku koja je zbrojena sa svih povezanih igračih platformi te time predstavlja njegovo ukupno igračko iskustvo neovisno o platformi. Profil također prikazuje korisnikova korisnička imena na svim povezanim platformama što olakšava stvaranje novih prijateljstava jer se korisnici moraju pronaći samo jednom na PlayTrackeru, umjesto pojedinačno na svakoj platformi.

Korisnik registracijom na servis i verifikacijom email adrese dobiva pristup svojoj kontrolnoj ploči gdje uređuje svoj profil odabirom profilne i naslovne fotografije te unosi podatke pomoću kojih će sustav izraditi profil. To mogu biti podaci koje korisnik direktno unosi, kao na primjer koje hardverske platforme imaju, ili vjerodajnice za igrače platforme pomoću kojih sustav može nabaviti informacije o korisniku. Pomoću unesenih vjerodajnica PlayTracker se spaja na igrače platforme i povlači potrebne podatke, te ih sprema u vlastitu bazu podataka iz koje se podaci pozivaju i prikazuju na korisničkim profilima.

### 2.2. Kome je projekt namijenjen i zašto je potreban

Svaka igrača platforma nudi svoj komplet svojstava, no većina njih nudi funkcije društvene mreže (chat, profil, dodavanje prijatelja), te zapisuje vrlo sličan osnovni set podataka kao što su broj odigranih sati, igranih igara i otključanih postignuća. Postignuća (također na nekim platformama poznata kao „trofeji“) igrači dobivaju tako da obavljaju posebne zadatke unutar igara, na primjer „pređi razinu 5 u manje od 7 minuta bez da koristiš oružje“.

Kako su se tehnologije mreža razvijale tijekom godina, tako su i tvrtke koje proizvode videoigre razvijale svoju online prisutnost u obliku digitalne distribucije igara i raznih

popratnih usluga kao što su praćenje statistika o igranju i sinkronizacija podataka. S vremenom to je rezultiralo u fragmentiranom tržištu gdje svaki veći izdavač ima svoju vlastitu igraču platformu na kojoj su ekskluzivno dostupne njihove igre.

U želji da igraju igre koje ih zanimaju, igrači su po potrebi koristili razne igrače platforme što je rezultiralo u fragmentaciji njihove biblioteke igara i popratnih podataka te nesuglasnosti njihovih korisničkih imena među raznim mrežama.

Stvorila su se dva problema:

- kada igrač upozna novu osobu s kojom želi igrati igre, mora s tom osobom podijeliti niz korisničkih imena kako bi se mogli dodati kao prijatelji na raznim igračim platformama
- igrači kojima je stalo do statistike o njihovom igranju i otključanim postignućima te informacije imaju razbacane po raznim igračim platformama

PlayTracker svojim skupljanjem podataka rješava te probleme. Korisnik koji je izradio PlayTracker profil može pratiti zbrojenu i ažurnu statistiku te jednim klikom pristupiti detaljnijim informacijama za svaku pojedinačnu igraču platformu koju je dodao. Pri upoznavanju novih prijatelja, korisnik može podijeliti samo svoje PlayTracker korisničko ime pomoću kojega se lagano dolazi do ostalih.

## 2.3. Pregled izvora podataka

Osnovne podatke kao korisničko ime i profilnu sliku PlayTracker korisnik unosi u svojoj korisničkoj ploči, no većina podataka koji se prikazuju na njegovom profilu PlayTracker sustav dolaze iz trećeg izvora. U trenutnoj verziji sustava postoji pet izvora podataka. Oni su:

- Steam - digitalna distribucijska platforma za osobna računala tvrtke Valve
- PlayStation Network – Sonyeva igrača platforma za njihove konzole
- Origin – digitalna distribucijska platforma za osobna računala tvrtke EA
- Overwatch – jedna od najigranijih igara na svijetu
- Guild Wars 2 – jedna od najpopularnijih masovnih online<sup>1</sup> igara na svijetu

---

<sup>1</sup> online igra za veliki broj igrača u kojoj igrač preuzima kontrolu nad svojim virtualnim likom koji postoji u dijeljenom svijetu između svih igrača te igre.

Podaci sa Steam platforme uključuju informacije o samom korisniku kao što su njegovo korisničko ime i broj kupljenih igara, no također i detaljne statistike o svim igrama koje je korisnik igrao (na primjer vrijeme igranja, postignuća). Iste ove podatke sustav dohvaća i s PlayStation Networka te Origina, s obzirom da se radi o sličnim platformama.

S Overwatch i Guild Wars 2 sustav dohvaća opće podatke (vrijeme igranja i postignuća), ali i neke bitne detalje specifične za te dvije igre kao što su najigraniji likovi.

## **2.4. Pregled programskih jezika i alata koji se koriste u izradi**

Za podešavanje poslužitelja korišten je Linux terminal command-line sučelje. Ovo sučelje u širokoj je upotrebi jer olakšava instalaciju i podešavanje potrebnih komponenti na serveru, što je ponekad moguće kroz samo jednu naredbu.

Baza podataka podešena je koristeći programski jezik SQL i PhpMyAdmin alat. Web sustav programiran je pomoću PHP-a, HTML-a, CSS-a i Javascripta s jQuery frameworkom.

Za grafičku pripremu sustava korišteni su Adobe alati Photoshop i Illustrator.

## **2.5. Planiranje projekta – obuhvat i organizacija**

Projekt obuhvaća cjelokupnu izradu web sustava od nule do javno dostupnog funkcionalnog servisa, zbog čega je iznimno bitno bilo odrediti precizan obuhvat. Obuhvat je definiran dvjema ključnim granicama:

- sustav ne obuhvaća mogućnost komunikacije među korisnicima niti ikakve druge tradicionalne mogućnosti društvenih mreža već je njegova funkcionalnost isključivo u izradi i podešavanju profila
- sustav će u prvoj verziji podržavati točno pet pažljivo odabranih izvora podataka, ali sustav mora biti fleksibilan kako bi bilo moguće dodavanje novih izvora podataka kada ih korisnici zahtijevaju

Izvori podataka odabrani su zato što su među najpopularnijim platformama i jer skupa predstavljaju dobro zaobljenu kombinaciju izvora podataka s predstavnikom iz svakog područja interesa.

Projekt je organiziran u pet kronoloških faza:

- definiranje interne logike i organizacije sustava
- priprema infrastrukture
- izrada baze podataka i back-end rješenja za dohvaćanje podataka
- dizajn aplikacije i izrada front-end rješenja za vizualni prikaz
- testiranje i prilagođavanje kritikama

Taj redoslijed osigurao je kvalitetu finalnog sustava i omogućio ostvarenje tako ambicioznog projekta.

## 2.6. Infrastruktura projekta

Da bi web sustav bio javno dostupan, potreban je poslužitelj. PlayTracker se pokreće na poslužitelju *cloud* platforme Digital Ocean. Na poslužitelj je instaliran takozvani „LEMP“ *stack* što znači da je operacijski sustav poslužitelja Linux, odnosno u slučaju PlayTrackera Ubuntu 16.04, s NGINX web poslužiteljem i tehnologijama PHP i MariaDB. Podešavanje poslužitelja obavljeno je putem Linux terminal *command-line* sučelja na koje se pristupa SSH tehnologijom kako bi vjerojatnost ugrožene sigurnosti bila minimizirana.

Pomoću registra domene Namecheap registrirana je playtracker.net domena koja je glavni identifikator sustava na webu. Na Namecheapu definirani su *nameserveri* domene čime je svaki zahtjev na nju preusmjeren na pripremljeni poslužitelj na Digital Ocean *cloud* platformi.

Kako bi web sustav ispunio moderne standarde sigurnosti, potrebno je da podržava HTTPS protokol odnosno da ima validan SSL certifikat. Taj proces izveden je pomoću usluge Let's Encrypt koja nudi besplatne SSL certifikat te je iznimno lagana za instalirati i podesiti na Linux poslužitelju jer zahtjeva samo par naredbi u terminalu.

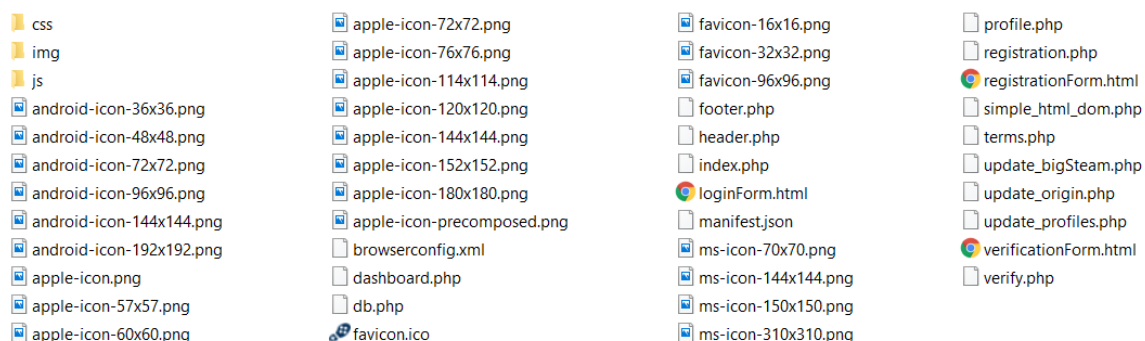
S obzirom da sustav obuhvaća i verifikaciju email adrese korisnika, obavezna je i email komponenta. Tu ulogu ispunio je Postfix mail poslužitelj koji adekvatno ispunjava potrebe, ali minimalno opterećuje poslužitelj.

Iako je moguće bazu podataka podesiti i putem terminala, kako bi taj zadatak bio lakši instalirana je komponenta PhpMyAdmin koja je pružila potrebno grafičko sučelje za bazu podataka.

## 3. Front-end i dizajn aplikacije

### 3.1. Programska struktura

Datoteke koje čine front-end projekta nalaze se na poslužitelju i određuju što korisnik vidi i kada.



Slika 3.1 Datoteke projekta

U mapi „css“ nalaze se datoteke koje određuju stil projekta pisane CSS programskim jezikom te mapa koja sadrži datoteke za font koji projekt koristi. Tri su CSS datoteke, „rawline.css“ sadrži poziv datoteka za font, „style\_main.css“ sadrži generičke klase za raspored stranice putem koje su primjenjive na bilo koji projekt, a „style\_playtracker.css“ sadrži ostali stilski kod specifičan za PlayTracker.

U mapi „img“ nalaze se svi vizualni materijali kao što su logotipi i naslovne fotografije. Mapa „js“ sadrži Javascript skripte – „dashboard\_functions.js“ i „search\_functions.js“ sadrže kod za elemente korisničkog sučelja, dok „md5.min.js“ sadrži funkciju za MD5 enkripciju teksta koja se koristi isključivo pri generaciji URL-a za trenutni pregled profilne fotografije kada korisnik unosi svoju Gravatar email adresu.

Bitne datoteke koje sadrže front-end projekta su sljedeće .php i .html datoteke:

- „dashboard.php“ – kontrolna ploča korisnika, sadrži i back-end kod
- „footer.php“ – kod za ispis podnožja stranice, poziva se u drugim skriptama
- „header.php“ – kod za ispis zaglavlja stranice, poziva se u drugim skriptama
- „index.php“ – naslovna stranica
- „loginForm.html“ – forma za prijavu korisnika, poziva se u drugim skriptama

- „profile.php“ – profil korisnika
- „registrationForm.html“ – forma za registraciju korisnika
- „terms.php“ – uvjeti korištenja koje korisnik mora prihvatiti pri registraciji
- „update\_origin.php“ – stranica za povezivanje korisnikovog Origin računa
- „verificationForm.html“ – forma za verifikaciju emaila korisnika

Sve .png datoteke, „favicon.ico“ i datoteke „browserconfig.xml“ i „manifest.json“ služe za prikaz ikonice PlayTrackera, tzv. *favicon* na svim uređajima. Ostale datoteke služe isključivo za back-end projekta.

## 3.2. Vizualni identitet i brend

Publika ljubitelja videoigara tehnološki je suvremena i vizualno zahtjevna, stoga je bitno da je vizualni identitet i brend PlayTracker web sustava jednako suvremen i atraktivan. Cilj je koristiti dokazane paradigme kako bi vizualni identitet novim korisnicima dao dojam kao da ga već poznaju i vjeruju mu te ga automatski povezuju sa vjerodostojnosti i kvalitetom. Laički rečeno, traži se „*mainstream*“ dojam, a ne upečatljivo „hipsterski“.

### 3.2.1. Ime i logo

Ime „PlayTracker“ složenica je dviju riječi koje predstavljaju glavne značajke sustava. Riječ *play* koja na engleskom znači igrati, predstavlja temu sustava koja se vrti oko videoigara. Riječ *tracker* koja na engleskom znači tragač, označava prikupljanje ili praćenje podataka iz raznih izvora koje je u srži PlayTrackera kao usluge.

Logo za PlayTracker inspiriran je riječima u njegovom imenu. Sadrži se od otiska stopala koji je česti motiv za traganje ili praćenje, te klasičnih tipki jednog kontrolera. Koristi se isključivo monokromatska verzija, primarno bijela na tamnoj podlozi ili crna za svijetle podloge.



Slika 3.2 Logo PlayTracker brenda

### 3.2.2. Vizualni identitet

Glavna boja brenda je #012f59 odnosno tamno plava. Brendovi u industriji videoigara najčešće se drže tamnih boja te ova boja iako nije jedinstveno upečatljiva, slijedi taj trend i doprinosi cilju dojma kvalitete.

Značajni dio vizualnog identiteta je upotreba prozirnih crnih ili bijelih elemenata nad fotografijama. Ova praksa, vrlo česta na webu, stavlja pritisak na kvalitetu fotografija. Kvaliteta fotografija na PlayTrackeru osigurana je koristeći, uz dopuštenje, odabrane fotografije iz igara koje je uhvatio poznati „video game photographer“ Duncan Harris.

Dio vizualnog identiteta PlayTrackera je i implementacija tuđih brendova. Zato što PlayTracker služi kao agregator informacija s više platformi, bitno je brendove tih platforma vjerno prikazati u svakom kontekstu. Dizajn ključne stranice – korisničkog profila – delikatan je balans između PlayTracker brenda i brendova platformi s kojih se informacije dohvaćaju. Za informacije koje generira PlayTracker odnosno koje su zbrojene s više platformi, koristi se PlayTracker boja, dok se za informacije koje se prikazuju bez promjena koristi brend platforme s koje su dohvaćeni. Informacije su po toj podjeli u grupirane u kartice. Ovakav pristup uspostavlja vizualnu hijerarhiju i podjelu informacija koje olakšava čitljivost podataka. Implementacija tuđih brendova također podiže vjerodostojnost našeg brenda asocijacijom.

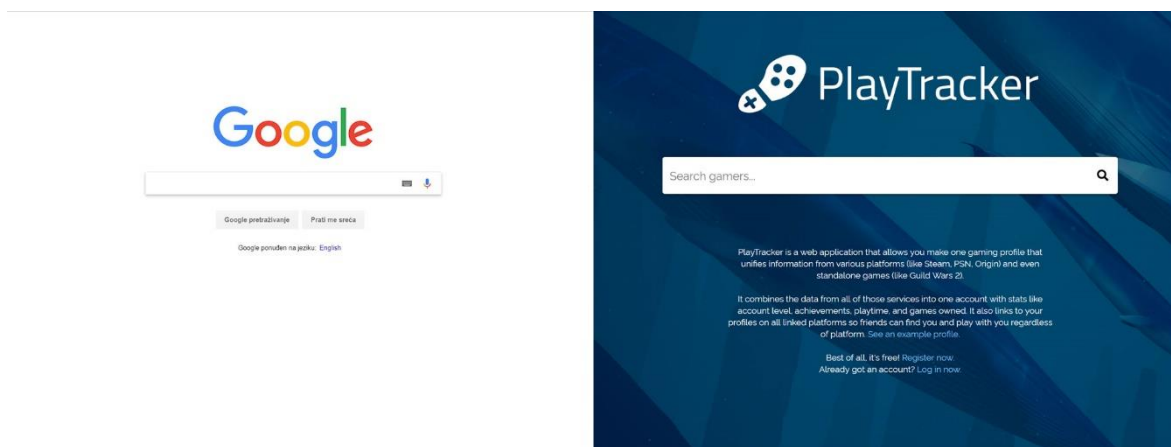


### 3.3. Struktura i osnovni raspored i stil stranice

Tri su ključne stranice PlayTracker sustava koje su zajedno odgovorne za gotovo svo vrijeme koje korisnici provedu na PlayTrackeru. Radi se o naslovnici, korisničkom profilu i kontrolnoj ploči.

#### 3.3.1. Naslovnica

Naslovnica web sustava ima jednu glavnu funkciju – omogućiti korisnicima pretragu baze igrača. Za strukturu te stranice inspiracija je bila jedna od najpoznatiji stranica na webu – Googleva naslovnica.



Slika 3.3 Google naslovnica (lijevo) i finalna verzija PlayTracker naslovnice (desno)

Centar stranice i fokusna točka svakog posjetitelja je okvir za pretraživanje iznad kojega se nalazi logo, a ispod kojega se nalazi kratki opis web sustava. No, jedan od ključnih faktora u uspješnosti Googleove naslovnice je njihov veseli šareni logo koji se ističe na bijeloj podlozi. Ovo nije slučaj za PlayTracker brend zbog monokromatskog crnog logoa. Nakon testiranja s fokus grupom, u stil stranice uključena je i zatamnjena pozadinska fotografija i bijela verzija logoa na kojoj se bijeli okvir za pretraživanje još više istaknuo.

#### 3.3.2. Kontrolna ploča korisnika

Kontrolna ploča korisnika stranica je na kojoj korisnik podešava sve opcije vezane uz svoj profil. Njen raspored je dva stupca jednake širine koji se spoje u jedan stupac na mobilnom sučelju. Svaki stupac sadrži se od kartica za unos podataka. Prvi (lijevi) stupac sadrži kartice za unos podataka koji će se direktno prikazivati na PlayTrackeru kao što su korisničko ime i profilna fotografija, dok drugi (desni) stupac sadrži polja za unos

vjerodajnica pomoću kojih će se PlayTracker spajati na druge platforme s kojih se dohvaćaju podaci.

Na dnu stranice, ispod svih kartica, nalazi se velika tipka za spremanje unesenih podataka.

The screenshot shows the 'LittleDinamit's Dashboard' in PlayTracker. The dashboard is divided into several sections:

- Your info:** Username 'LittleDinamit', Gravatar Email 'marijan.mikolic@gmail.com', and a profile picture of a character in a blue hood.
- Your cover image:** A grid of 12 image thumbnails for selection.
- Your platforms:** A list of gaming platforms with checkboxes: Windows PC, Mac, Linux, PlayStation 4, PlayStation 3, PlayStation Vita, XBOX One, XBOX 360, Nintendo Switch, Nintendo Wii U, Nintendo 3DS, Android, and iOS.
- Your profiles:** A list of external profiles with their IDs and checkboxes to show cards on the profile:
  - Steam ID: 76561198014612192 (checked)
  - PlayStation Network Username: Little\_Dinamit (checked)
  - Battle.Net Username: Dinamit#1723 (checked)
  - Guild Wars 2 API Key: 5706C69B-3417-D741-B656-F38664F7 (checked)
  - GOG Username: Imaooo
  - XBOX Gamertag: ayyy
  - Origin account: (checked)

At the bottom, there is a 'Save changes' button and a note: 'If you've just entered a lot of new profiles, please be patient - it might take a few seconds.'

Slika 3.4 Kontrolna ploča korisnika

### 3.3.3. Profil korisnika

Najbitnija stranica cijelog sustava je profil korisnika. Profil korisnika je usluga koju web sustav nudi – prikazivanje svih podataka na jednom mjestu.

Izazov profila korisnika je u tome što sastoji veliku količinu bitnih informacija koje je potrebno organizirati tako da imaju smisla i ne daju osjećaj zasipanja korisnika.

Problem zasipanja korisnika riješen je guranjem svih informacija niže, prema crti prijeloma tako da je većina sadržaja ispod nje. Glavu stranice čine:

- traka na vrhu sa logotipom, okvirom za pretraživanje i tipkama koje vode na korisnikov profil i kontrolnu ploču
- korisnikova naslovna fotografija preko cijelog ekrana
- korisnikova profilna fotografija i korisničko ime na samoj sredini naslovne fotografije

Atraktivna glava stranice s malom količinom informacija korisnika opušta i zasipanje podacima odgađa dok korisnik ne pomakne sadržaj dolje.

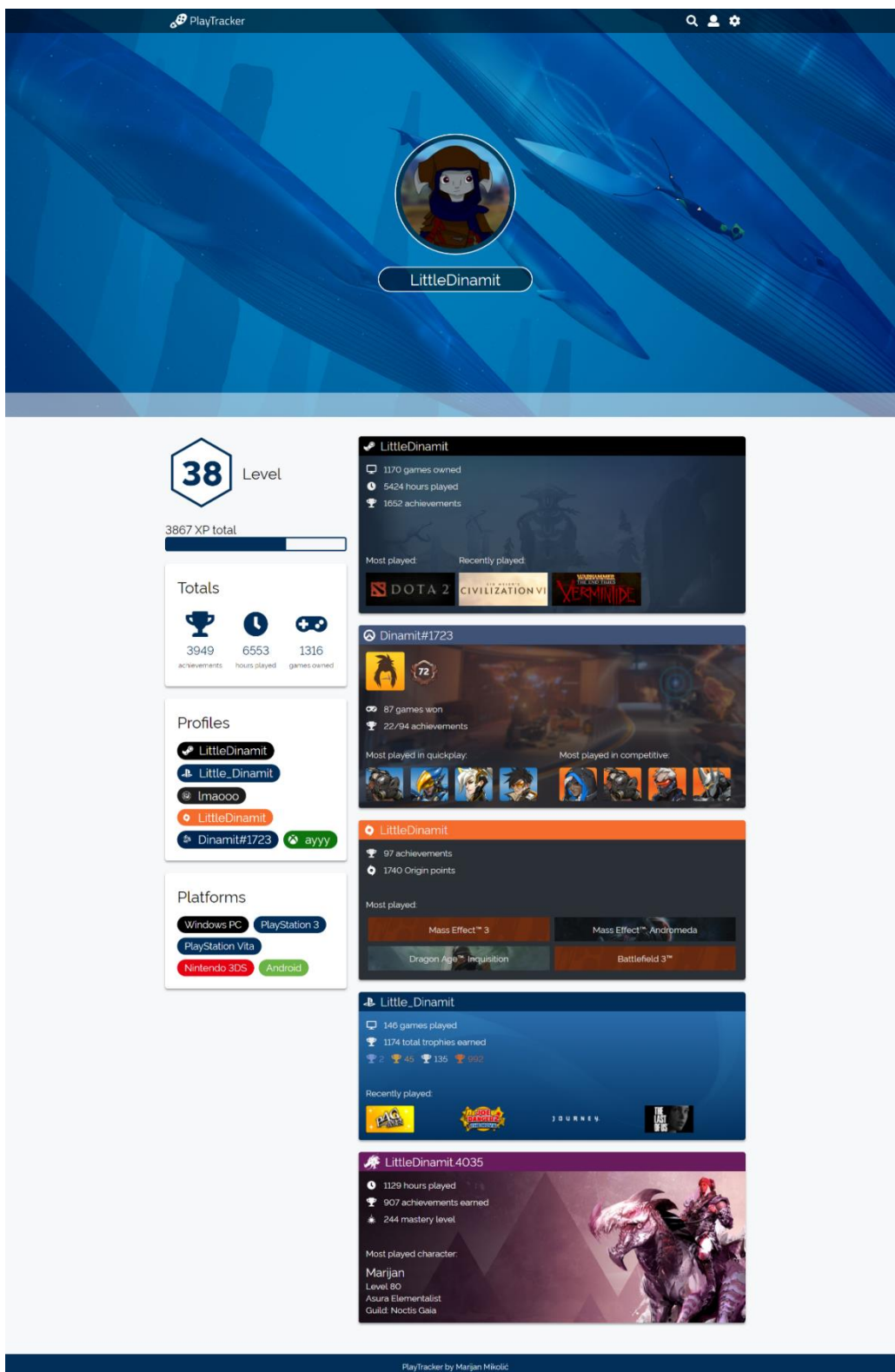
Problem organizacije riješen je podjelom sadržaja stranice u dva stupca omjera zlatnog reza koji se za mobilno sučelje spoje u jedan stupac. Prvi stupac sadrži podatke čiji je izvor podataka sam PlayTracker odnosno podatke koji su rezultat zbrajanja svih platformi ili su uneseni direktno u PlayTracker, npr. zbrojevi statistika i platforme. Na samom vrhu ovog stupca je korisnikov level i količina iskustva<sup>2</sup> koje korisnik ima što je zapravo najkraći sažetak korisnikove aktivnosti preko svih platformi. Ispod toga nalaze se kartice s zbrojevima korisničkih statistika (postignuća, vrijeme igranja, broj igara), korisnikovim profilima (korisničkim imenima preko kojih se može pronaći na drugim platformama) i hardverskim platformama (npr. PlayStation 4).

Drugi stupac sadrži podatke sa svih povezanih platformi za koje je korisnik odobrio da se prikazuju. Svaki izvor podataka ima svoju karticu koja slijedi vizualni identitet platforme s koje su podaci dohvaćeni. Ovakav prikaz podataka nije nov – takozvani „gamercards“ koriste se od ranih dana weba kada su ih korisnici imali u svojim potpisima na forumima stoga će većini korisnika biti prepoznatljiv i logičan.

---

<sup>2</sup> Iskustvo ili XP je valuta česta u igrama s kojom igrač podiže svoj level i otključava novi sadržaj.

Raspored lijevo (ili gore na mobilnom sučelju) za PlayTracker podatke te desno (ili dolje na mobilnom sučelju) za podatke s drugih platformi konzistentan je sa kontrolnom pločom i pomaže orijentaciji korisnika.



Slika 3.5 Korisnički profil sa svim mogućim karticama

### 3.4. Korisničko sučelje

Cilj korisničkog sučelja PlayTrackera je da profil korisnika bude čitljiv i atraktivan, a da unos podataka putem kontrolne ploče bude što je više moguće olakšan korisniku. Element korisničkog sučelja koji te dvije stranice dijele je traka na vrhu koja služi za navigaciju sustavom. Ona sadrži logo koji vodi na naslovnu stranicu, okvir za pretraživanje, tipku koja povezuje na kontrolnu ploču i, ukoliko je korisnik prijavljen, tipku koja vodi na njegov profil.

Dizajn korisničkog sučelja bazira se na pravilima koja je uspostavio Google za svoju *material design* filozofiju dizajna. Generalni principi organizacije podataka u kartice, isticanja kontrastom boja i izgledom konzistentnim preko svih platformi prisutni su u korisničkom sučelju PlayTrackera, s promjenama gdje je bilo potrebno.

Iako se većina podataka na cijelom sustavu organizira u kartice, postoje iznimke. Na naslovnoj stranici sažetak funkcionalnosti nema pozadinu kartice kako bi fokus ostao na okviru za pretraživanje. Na korisničkom profilu prva kartica u prvom stupcu koja sadrži level i iskustvo korisnika nema pozadinu kartice kako bi se razbila stroga mreža i dobio dojam dinamičnosti.

Kontrolna ploča dom je mnogim poljima za upis teksta koji uglavnom traže vrlo različite i ponekad neintuitivne informacije za unos, s obzirom da su vjerodajnice za pristup platformama različite. Nema nikakvog rješenja koje će to standardizirati, no nejasnoća se može olakšati tako da se uz svako polje potreban podatak ukratko pojasni.

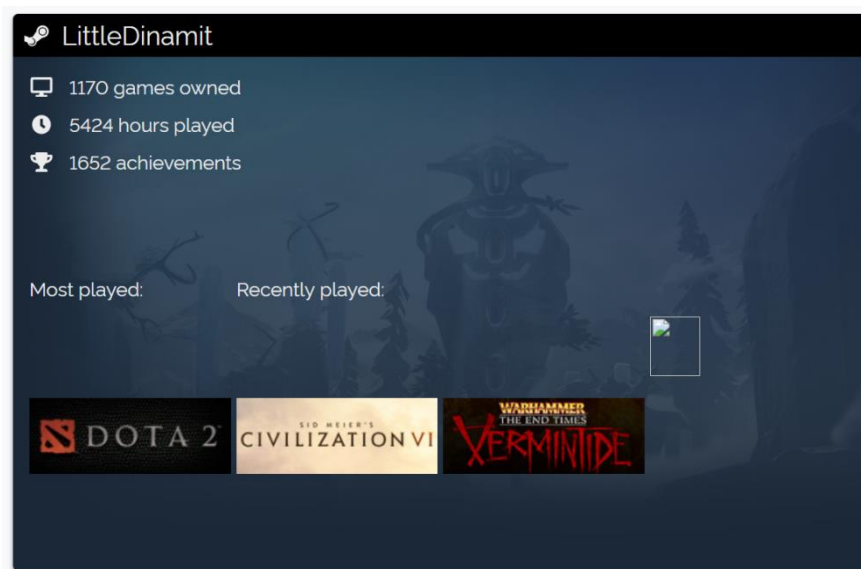
Jedan od izvora podataka zahtjeva da korisnik učita datoteku, te je taj proces odvojen na svoju vlastitu stranicu koja je povezana na kontrolnu ploču.

Kako bi korisničko sučelje radilo brzo i izgledalo dobro, bitno je držati se dobrih praksi u izradi. Za raspored stranice korištena je flex metoda putem CSS-a koja osigurava fleksibilnost svakog elementa i podržana je u 97.8% preglednika globalno (CanIUse, 2018.). Za ikonice koristi se FontAwesome set vektorskih ikonica, a tip slova je Rawline.

Za određene naprednije funkcije koristi se jQuery programski jezik, npr. da bi klik na tipku povećala fokusirao okvir za pretraživanje.

### 3.5. Testiranje korisničkog sučelja

Korisničko sučelje testirano je fokus grupom 5 osoba koje su se registrirale i podesile svoj profil. Tijekom tog procesa javljali su nejasnoće i probleme na koje su naišli. Rezultat ovog testiranja bila je reorganizacija podataka na kontrolnoj ploči i dodatna pojašnjenja za neka polja unosa te popravak velikog broja otkrivenih problema s prikazom podataka koji su uzrokovani raznovrsnošću podataka koje sustav može dohvatiti. Npr. u kartici platforme Steam prikazuju se ikonice tri igre koje je korisnik igrao u zadnja dva tjedna, što je značilo da ukoliko je korisnik u zadnja dva tjedna igrao manje od tri igre, rezultat bi bila greška 404 na pokušaju dohvaćanja ikonice tih igara. Ovo je popravljeno tako da je dodana provjera broja igara koja utječe na broj pokušaja dohvaćanja ikonica.



Slika 3.6 Primjer greške pronađene pri testiranju korisničkog sučelja

Zbog povratne informacije fokus grupe došlo je i do nekih subjektivnih vizualnih promjena kako bi sustav bio atraktivniji. Npr. u kartici Overwatch na dnu se ispisuju ikonice likova koje je korisnik najviše igrao, no nije bilo vizualne podjele između dva različita moda igranja koji postoje. Ta vizualna podjela postignuta je razmakom i različitom bojom pozadine ikonice likova.



Slika 3.7 Primjer promjene dizajna kao rezultat testiranja korisničkog sučelja (staro lijevo, novo desno)

### 3.6. Praćenje i analitika

Praćenje i analitika PlayTrackera obavlja se koristeći Googleov Analytics softver koji je uključen u svaku stranicu web sustava. Analytics je softver koji prati statistike o posjećenosti stranica, ponašanju korisnika, duljine njihovog posjeta i mnogo više. Uz opcije praćenja podataka na web sustavu u koji je implementiran, Analytics pruža i profiliranje korisnika pomoću informacija koje Google ima o korisniku. Ove informacije omogućuju naprednu demografsku analizu korisnika web sustava koja ne bi bila moguća uz samostalni sistem analitike i praćenja korisnika te je razlog zašto je Analytics izabran za ovaj projekt.

Unatoč svojim opsežnim mogućnostima, Analytics je jednostavan za implementaciju. Implementacija Google Analytics u PlayTrackeru izvodi se sljedećim kodom:

```
<script async
src="https://www.googletagmanager.com/gtag/js?id=UA-97259968-
1"></script>
<script>
window.dataLayer = window.dataLayer || [];
function gtag(){dataLayer.push(arguments);}
gtag('js', new Date());

gtag('config', 'UA-97259968-1');
</script>
```

Taj kod ubačen je u „header.php“ datoteku koja se poziva u svakoj stranici okrenutoj prema korisniku. Google Analytics

Iz Google Analytics podataka cilj je izvući podatke o korisničkim navikama pomoću kojih će se sustav unaprjeđivati s vremenom. Posebno zanimljiv podatak biti će omjer vremena

koji korisnici provode podešavajući i pregledavajući vlastiti profil naspram vremena koji provode gledajući tuđe – ovaj podatak odrediti će koristiti li sustav korisnici više da pronađu druge ili da prate sebe. Ovisno o tome, prioritet daljnjeg razvoja će biti na mogućnostima vezanih uz taj korisnički cilj.



## 4. Back-end aplikacije

### 4.1. Programska struktura

PHP back-end projekta u pojedinim datotekama isprepleten je s front-endom. Te datoteke osim što ispisuju stranicu korisniku obavljaju i potrebnu komunikaciju s bazom. One su:

- „dashboard.php“ – osim što korisniku ispisuje kontrolnu ploču također unesene podatke dohvaća POST metodom, provjerava i upisuje u bazu podataka
- „update\_origin.php“ – sadrži sve faze povezivanja korisnikovog Origin računa te ispisuje potrebne informacije za trenutni korak, rezultat procesa provjerava i upisuje u bazu podataka.

Datoteke koje sadrže back-end funkcije projekta su sljedeće .php datoteke:

- „db.php“ – kod za povezivanje na bazu podataka, poziva se u drugim skriptama
- „registration.php“ – obavlja funkciju registracije korisnika, ispisuje potrebnu formu u odgovarajućem trenutku te provjerava i upisuje podatke u bazu podataka
- „simple\_html\_dom.php“ – funkcija za struganje stranica, poziva se u drugim skriptama
- „update\_bigSteam.php“ i „update\_profiles.php“ skripte za ažuriranje podataka korisnika koje server redovito automatski izvodi
- „verify.php“ – obavlja funkciju verifikacije email adrese korisnika provjeravanjem unesenih podataka i upisom u bazu podataka

### 4.2. Asinkroni prikaz i dohvaćanje podataka

Točnost i ažurnost podataka ključni su elementi funkcioniranja ovog sustava, no ne smiju rezultirati u usporavanju prikaza za korisnika. Kada bi profil korisnika dohvaćao sve podatke u pravom vremenu, trebalo bi mu oko 5 minuta da se učita. Rješenje za ovo je stroga podjela između dohvaćanja i prikazivanja podataka.

Sinkronizacija korisničkih podataka obavlja se dvjema PHP skriptama koje se pokreću *cronjob* funkcionalnošću servera.

Glavna skripta, koja se pokreće svake minute, pronalazi najstariji podatak za svaki izvor podataka i osvježava ga. Skripta sadrži mnogo provjera validnosti podataka te ukoliko se problem pojavi, skripta zapisuje detalje problema u bazu podataka i označava zapis kako osvježavanje podataka ne bi zauvijek zapelo na tom jednom problematičnom retku.

Sporedna skripta, koja se pokreće svakih deset minuta, odgovorna je za prvu sinkronizaciju Steam izvora podataka za svakog korisnika. Zato što je Steam daleko najveći izvor podataka te je i najčešći izvor koji korisnici povezuju, povlačenje svih podataka za jednog korisnika traje više minuta. Sporedna skripta obavlja taj zadatak kada korisnik po prvi puta unese vjerodajnice za Steam (ili ih promjeni). No, pokretati cijelu skriptu za svakog korisnika kada god je potrebno osvježiti njegove informacije bilo bi iznimno zahtjevno za resurse poslužitelja. Zato je drugi element sinkronizacije podataka Steam izvora podataka u glavnoj skripti. Taj dio za korisnika dohvaća samo podatke o nedavno igranim igrama (zadnja dva tjedna) te dobivene podatke spaja sa starim podacima zapisanim u bazi kako bi dobio ukupne podatke kao što su broj postignuća i odigranih sati.

## **4.3. Baza podataka**

### **4.3.1. Odabir i uloga baze podataka**

Baza podataka koja pogoni PlayTracker web sustav je MariaDB. Ova baza podataka vrlo je slična poznatoj MySQL bazi, no dokazala se efikasnijom od nje u okruženju korištenog poslužitelja (Ubuntu 16.04. i NGINX) stoga je odabrana za projekt.

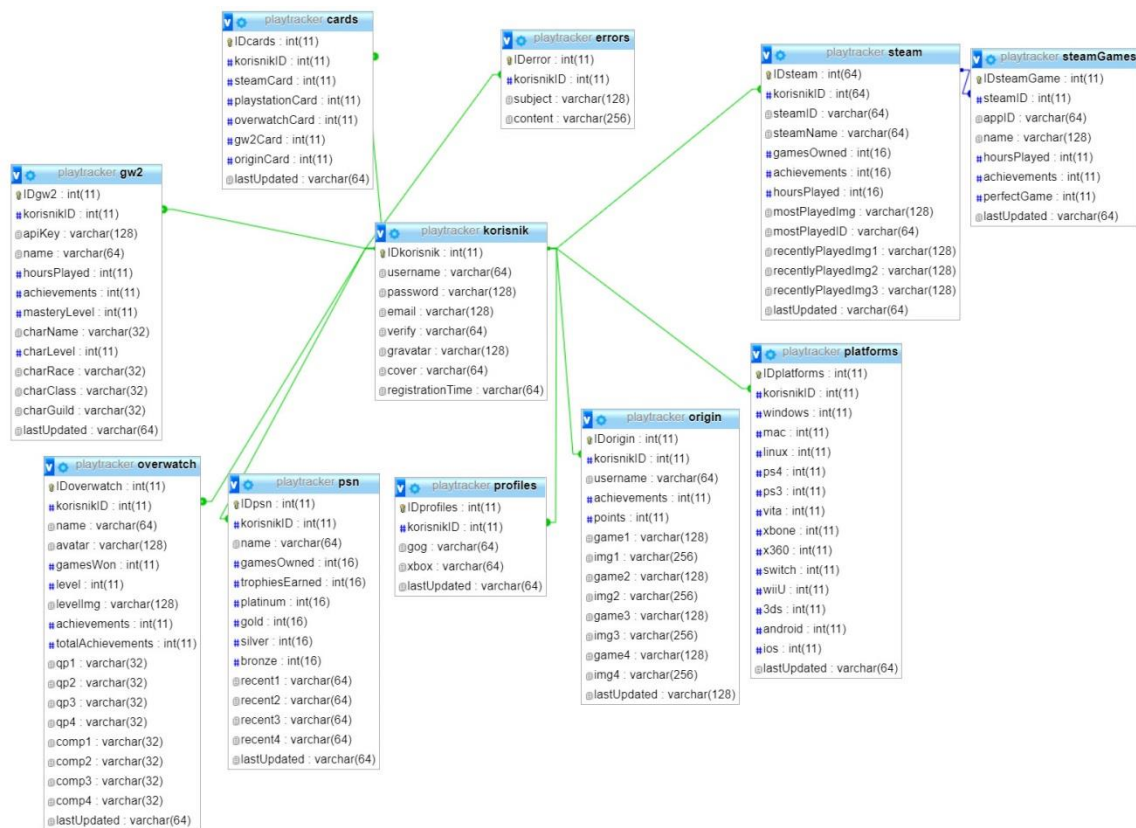
Baza podataka pogoni sve informacije koje korisnik vidi na profilu. Od njegovog korisničkog računa do svake informacije sa svakog izvora podataka koji je povezo, sve se za korisnika dohvaća direktno iz baze podataka.

### **4.3.2. Tablice i organizacija**

PlayTracker baza podataka sadrži se od 11 tablica čija je okosnica tablica korisnik koja sadrži podatke o korisniku kao što su njegovo korisničko ime i lozinka. Na ovu tablicu se stranim ključem povezuju sve tablice koje sadrže podatke povučene iz nekog izvora podataka kako bi dohvaćanje podataka za korisnika bilo jednostavno.

Svaki izvor podataka ima svoju tablicu, s iznimkom Steama koji ima dvije tablice – glavnu s podacima koji se ispisuju korisniku i sporednu koja sprema detalje o njegovim igrama

kako bi ranije spomenuti optimizirani sistem sinkronizacije podataka mogao sinkronizirati podatke bez da dohvaća ogroman broj podataka svaki puta.



Slika 4.1 Dijagram tablica baze podataka i njihovih veza

Na tablicu korisnik povezuju se i tablice koje sadrže dodatne direktno unesene podatke kao što su hardverske platforme koje korisnik ima i profili na platforma koji ne služe kao izvor podataka te tablica „cards“ koja sadrži korisnikov odabir koje će se kartice pokazivati na njegovom profilu, što također ubrzava i prikaz informacija jer skripta ne mora pozivati tablicu nekog izvora podataka samo da bi saznala da korisnik nije povezoao taj izvor podataka.

Zadnja tablica koja je povezana na tablicu korisnik je tablica „errors“ koja sadrži zapise o greškama pri sinkronizaciji podataka.

## 4.4. Application Interface – API

API je metoda za dohvaćanje podataka s neke usluge na Internetu službeno odobrena od strane kreatora te usluge. Upotreba API-a je vrlo česta u modernim aplikacijama i web sustavima te se mnoge usluge oslanjaju na njih da bi funkcionirale.

#### 4.4.1. Uloga API-a u projektu

U PlayTrackeru uloga API-a je način dohvaćanja podataka o korisniku. Unutar PlayTrackerovih skripti za sinkronizaciju podataka s API-a se dohvaćaju podaci u JSON formatu koje skripta dekodira, obrađuje i sprema u bazu.

Pristup API-u je putem tzv. krajnjih točki koje odgovaraju kojoj grupi podataka se pristupa. Pri implementaciji API-a, u dokumentaciji je potrebno pronaći krajnje točke koje sadrže željene podatke.

Sljedeći kod je primjer kako dohvatiti korisničko ime putem Steam API-a:

```
$userSteamID="76561198014812192";  
$apiKey = "5BCA3A60FAF613079A8AFE516AC1E5E0";  
$api =  
"http://api.steampowered.com/ISteamUser/GetPlayerSummaries/v0  
002/?key=5BCA3A60FAF613079A8AFE516AC1E5E0&steamids=".$userSte  
amID;  
$json = file_get_contents($api);  
$decoded = json_decode($json);  
Echo $decoded->response->players[0]->personaname;
```

#### 4.4.2. Pregled korištenih API-a

U projektu koriste se dva API-a: Valveov Steam API i Arena Netov Guild Wars 2 API.

Steam API nudi programerima pristup podacima sa Steam platforme koja je najveća na svijetu. Mogu se dohvaćati informacije o igrama i proizvodima u Steam trgovini ili o korisnicima. Za PlayTracker je bilo potrebno povlačiti podatke o korisnicima. Programer nabavlja svoj API ključ pa podatke o korisniku dohvaća putem ID-a korisnika, ukoliko korisnik svoj profil nije označio kao privatn.

Korištena su tri krajnje točke:

- *GetPlayerSummaries* – daje generalne informacije o korisniku kao što je korisničko ime
- *GetOwnedGames* – daje informacije o broju igara koje korisnik ima i koliko je koju igrao
- *GetPlayerAchievements* – daje informacije o postignućima korisnika za neku igru

Sve tri krajnje točke koriste se u obje skripte za sinkronizaciju podataka te obje skripte imaju petlju koja prolazi kroz dobiveno polje igara i koristi krajnju točku *GetPlayerAchievements* za dohvaćanje detalja korisnikovog napretka o toj igri, ukoliko igra podržava postignuća. Razlika je u tome što glavna skripta u petlju šalje polje igara igranih u zadnja dva tjedna, dobivenog putem krajnje točke *GetPlayerSummaries*, dok sporedna skripta prolazi sve igre u polju dobivenom putem krajnje točke *GetOwnedGames* koje je daleko veće.

Guild Wars 2 API nudi programerima pristup podacima s Guild Wars 2 platforme vezane uz igru istog imena. Za razliku od Steama, API ključ za Guild Wars 2 API nudi pristup podacima jedino korisnika čiji je to API ključ, što znači da svaki korisnik mora generirati svoj API ključ i upisati ga u PlayTracker.

PlayTracker pristupa mnogim krajnjim točkama vezanih uz korisnički račun općenito kao i njihove pojedinačne likove. Guild Wars 2 API strukturiran je tako da krajnje točke često ne daju gotov rezultat već daju ID koji je potrebno riješiti putem neke druge krajnje točke, što rezultira u mnogo poziva na API čak i za osnovne informacije.

## 4.5. Struganje podataka s web stranice

Struganje podataka s web stranice je proces u kojemu skripta dohvaća gotovi HTML dokument neke javno dostupne stranice te obrađuje taj HTML dokument kako bi izvukla potrebne podatke. Ovaj proces je sporiji i manje efikasan od dohvaćanja podatka API-em te često krši uvjete korištenja web stranice, no poneke web stranice dopuštaju i čak olakšavaju proceduru kako bi izbjegli stvaranje vlastitog API-a.

PlayTracker web sustav struga podatke s Overwatch i PSN profila dostupnih javno na webu, no jedan od izvora podataka, Origin, ne dopušta struganje. Unatoč tome, zato što je Origin jedna od najvećih platformi, mogućnost uvoza podataka s te platforme je morala biti moguća. Rješenje tog problema je bilo da korisnik sam skine HTML dokument svog profila te ga učita na PlayTracker koji onda struga tu datoteku.

Neovisno o tome radi li se o dokumentu koji je dohvatio sustav ili sam korisnik, struganje u PlayTrackeru obavlja se uz pomoć Simple HTML DOM PHP skripte koja omogućuje pretraživanje DOM-a HTML dokumenta kodom sličnim jQuery programskom jeziku. Potrebna informacija se pronalazi putem klase ili ID-a elementa u kojoj se nalazi, npr.

korisničko ime na PSN profilu nalazi se unutar elementa s klasom „username“ te ga sustav tako pronalazi u sljedećem kodu:

```
$html = file_get_html('https://psnprofiles.com/'. $userPSNID);  
$psnUsername = $html->find('.username');  
$psnUsername = $psnUsername[0]->plaintext;
```

U prvoj liniji koda koristeći funkciju „file\_get\_html“ definiranu u „simple\_html\_dom.php“ skripti sustav dohvaća cijeli HTML dokument stranice profila korisnika. U drugoj liniji koda pronalazi se HTML element s klasom „username“ te se u trećoj liniji koda u varijablu „psnUsername“ sprema čisti tekst sadržaj tog HTML elementa.

## 4.6. Obrada podataka

PlayTracker po potrebi dohvaćene podatke obrađuje i tijekom dohvaćanja odnosno prije zapisa u bazi, ali i tijekom ispisa iz baze.

Pri upisivanju bilo kakvih podataka koje je pružio korisnik skripta obavlja provjere kako bi se spriječila injekcija SQL koda:

```
$email=$mysqli->real_escape_string($_POST['email']);
```

Za tu svrhu koristi se funkcija „real\_escape\_string“ koja je ugrađena u *mysqli* PHP klasu te izbacuje neželjene znakove iz tekstualnog niza kako bi spriječila injekciju SQL koda.

Pri dohvaćanju podataka putem API-a ili struganjem, određeni podaci su nekompletni te ih je potrebno obraditi prije zapisa u bazu. Npr. iz Guild Wars 2 API-a mogu se povući informacije o korisnikovim *mastery* bodovima koji su oblik progresije u toj igri. No, API ne daje direktan podatak o njegovim bodovima, nego daje njegovu razinu u svakoj kategoriji. Stoga PlayTracker skripta izračuna faktorijelu razine koju je korisnik postigao u svakoj kategoriji, te ih sve zbroji da bi dobila točan broj *mastery* bodova:

```
function factorial($num) {  
    $factorial = 1;  
    if ($num<1){  
        return 0;  
    }  
    for ($x=$num; $x>=1; $x--)  
    {  
        $factorial = $factorial * $x;  
    }  
    return $factorial;  
}
```

```
}
```

Pri ispisu podataka iz baze na profilu, neki podaci se obrađuju za prikaz. Najčešće su to podaci koji imaju samostalno značenje te se nepromijenjeni ispisuju unutar kartica svojih izvora podataka, no također imaju značenje kada se zbroje sa sličnim podacima iz drugih platformi. Primjer takvih podataka su sve informacije pri vrhu lijevog stupca na stranici profila korisnika – postignuća, broj igranih sati i broj igara.

Najvažniji podatak koji se izračunava pri ispisu je iskustvo korisnika koje dohvaća podatke iz svih tablica izvora te ih ubacuje u algoritam koji pravedno dodjeljuje vrijednost iskustva za sve po sljedećem principu:

- Steam postignuća, brončani i srebrni Playstation trofeji vrijede jedan bod
- Overwatch postignuća i zlatni PlayStation trofeji vrijede dva boda
- Guild Wars 2 postignuća vrijede 0.75 bodova (broj se na kraju zaokružuje)
- Svakih 10 Origin bodova vrijede 1 bod iskustva na PlayTrackeru
- Svaka igra na Steam, PlayStation ili Origin platformi u kojoj je igrač otključao sva postignuća vrijedi 50 bodova

Konačan broj iskustva se dijeli s 100 kako bi se dobila korisnikova razina te zaokružuje na niže, s time da ja maksimalna razina 99:

```
$XP=$XPvals["SteamAch"]+($XPvals["OriginPoints"]/10)+(($XPvals["Platinum"]*50)+($XPvals["Gold"]*2)+$XPvals["Silver"]+$XPvals["Bronze"])+($XPvals["OwAch"]*2)+(round($XPvals["Gw2Ach"]*0.75));  
if($XP>=10000){  
    $level=99;  
    $XPprogress=99;  
}  
else {  
    $level=floor($XP/100);  
    $XPprogress=floor($XP%100);  
}
```

Rezultat koda je da je u varijabli „XP“ spremljen rezultat algoritma za pravedno dodjeljivanje vrijednosti iskustva, a u varijabli „level“ na niže zaokružena korisnička razina, te su ti podaci spremni za ispis.

## Zaključak

Svrha projekta bila je pružiti uslugu skupljanja podataka s raznih platformi na jedan profil korisnicima koji su ljubitelji računalnih igara te koji tu uslugu ne mogu dobiti nigdje drugdje.

Ostvarivanje te usluge zahtijevalo je stabilnu infrastrukturu u *cloud* servisu i održavanje sigurnosnih standarda da bi se spriječila mogućnost zlonamjernog napada.

Vizuali PlayTracker web sustava sastavljeni su od mnogih standardnih i provjerenih praksi u dizajnu i front-end programiranju te su postignuti najnovijim tehnologijama. Sustav je prilagođen mobilnim uređajima i brz, te je korisničko sučelje optimizirano prema povratnoj informaciji korisnika.

Back-end tehnologija sustava obuhvaća pažljivo složenu relacijsku bazu podataka koju pune automatizirane skripte za sinkronizaciju podataka koje koriste API-e i struganje podataka s web stranica kako bi dohvatili ažurne podatke o korisnicima.

Unatoč predznanju u web developmentu, izrada web sustava bila je zahtjevan zadatak zbog količine detalja koje je potrebno usavršiti, pogotovo sa sigurnosne strane, kako bi usluga bila spremna za javnost. Izazovno je i uskladiti tehnike dohvaćanja za svaku metodu dohvaćanja podataka jer svaki API ima svoja pravila te svaka stranica koja se struga ima drugačiju DOM strukturu.

Odabrane tehnologije dobro su poslužile izradi sustava, pogotovo PHP programski jezik koji je često na lošem glasu zbog svoje starosti, no u ovom projektu se pokazao kao fleksibilno rješenje prikladno za sve probleme na koje sam naišao.

Web sustav PlayTracker kompletan je i funkcionalan proizvod koji pruža uslugu korisnicima na webu, no može biti i temelj za daljnji razvoj u mnogo smjerova, ovisno o korisničkom cilju koji se pokaže dominantnim u analitici. Podrška za još platformi i dodatne opcije za prilagođavanje profila kao što je određivanje poretka kartica očiti su kandidati za daljnji razvoj. No, i veće dodatne funkcionalnosti kao što je izrada prilagođene kartice za neku igru koristeći korisnikovu statistiku mogle bi biti dio budućnosti PlayTracker projekta.



## Popis kratica

API	<i>Applicaton Programming Interface</i>	sučelje programiranja aplikacija
DOM	<i>Document Object Model</i>	model objekta dokumenta
ID	<i>Identification</i>	identifikacija
PSN	<i>PlayStation Network</i>	PlayStation mreža
URL	<i>Uniform Resource Locator</i>	usklađeni lokator resursa
HTML	<i>HyperText Markup Language</i>	hipertekst jezik za izradu web stranica
SQL	<i>Structured Query Language</i>	jezik za izradu strukturnih upita

## Popis slika

Slika 3.1 Datoteke projekta .....	6
Slika 3.2 Logo PlayTracker brenda .....	8
Slika 3.3 Google naslovnica (lijevo) i finalna verzija PlayTracker naslovnice (desno).....	9
Slika 3.4 Kontrolna ploča korisnika .....	10
Slika 3.5 Korisnički profil sa svim mogućim karticama .....	12
Slika 3.6 Primjer greške pronađene pri testiranju korisničkog sučelja .....	14
Slika 3.7 Primjer promjene dizajna kao rezultat testiranja korisničkog sučelja (staro lijevo, novo desno) .....	15
Slika 4.1 Dijagram tablica baze podataka i njihovih veza.....	19

## Literatura

- [1] CANIUSE, <https://caniuse.com/#feat=flexbox>, 10.02.2018.
- [2] E. Lupton; Thinking with type; Princeton Architectural Press; (2004, 2010); ISBN: 978-1-56898-969-3.
- [3] J.J. Garrett; The Elements of User Experience; New Riders; (2010.); ISBN: 978-0321683687.
- [4] A. Clarke; Hardboiled Web Design; Five Simple Steps; (2010.); ISBN: 978-1-907828-00-3
- [5] B. Frain; Responsive Web Design with HTML5 and CSS3; Packt Publishing; (2012.); ISBN: 978-1-84969-318-9
- [6] A. Kennedy, I. De León; Pro CSS for High Traffic Websites; Apress; (2011.); ISBN: 978-1-4302-3288-9
- [7] J. Ellingwood; How To Install Linux, Nginx, MySQL, PHP (LEMP stack) in Ubuntu 16.04; DigitalOcean Community;  
<https://www.digitalocean.com/community/tutorials/how-to-install-linux-nginx-mysql-php-lemp-stack-in-ubuntu-16-04>