

Universitatea Alexandru Ioan Cuza Iași  
Facultatea de Informatică

# FaceMaze

Iacobeț Vlad Adrian, grupa A2  
Ilieș Norbert Mario, grupa A2  
Edu Dorin, grupa A2

Anul universitar 2015 – 2016

# Cuprins

1. Introducere
2. Scop/Public țintă
3. Mod de utilizare prevăzut
4. Arhitectura aplicației și etape de dezvoltare
5. API-uri utilizate
6. Concluzii

# 1. Introducere


FaceMaze este un joc web, bazat pe legendarul Pac-Man, care a devenit hit direct la lansare. Considerat imaginea anilor 80, Pac-Man reprezintă o legendă a jocurilor video. Inițial fiind un joc de tipul arcade ce se putea juca doar pe o cutie imensă, de-a lungul anilor acesta a fost portat pe diverse console de jocuri video sau device-uri, iar în zilele noastre a ajuns să poată fi jucat în browser. FaceMaze își propune o variantă mai interactivă, integrând noi funcționalități sociale conceptului de Pac-Man clasic.

## 2. Scop/Public țintă

**Scopul** acestei aplicații web este de a oferi posibilitatea de a descoperi jocul Pac-Man într-o forma mai interactivă, cu o experiență de joc personalizată fiecărui utilizator.

**Publicul țintă** este format din toate tipurile de gameri, ocazionali sau pasionați.

## 3. Mod de utilizare prevăzut


The image shows a user login and registration interface on a light yellow background. It features a white rounded rectangle containing the following elements: a 'Username:' label next to a text input field; a 'Password:' label next to another text input field; two social media login buttons on the right, one for Facebook (blue with 'f' icon) and one for Twitter (blue with bird icon); and a bottom bar with a green 'LOGIN' button, a grey 'or' separator, and an orange 'REGISTER' button.

Interfața de login a unui utilizator, prevăzută cu opțiunea de a se conecta cu informațiile contului acestuia de pe una din rețelele de socializare aferente.

1 - **Username:** Utilizatorul va scrie numele de utilizator cu care este înregistrat în baza noastră de date.

2 - **Password:** Acesta va introduce parola contului respectiv.

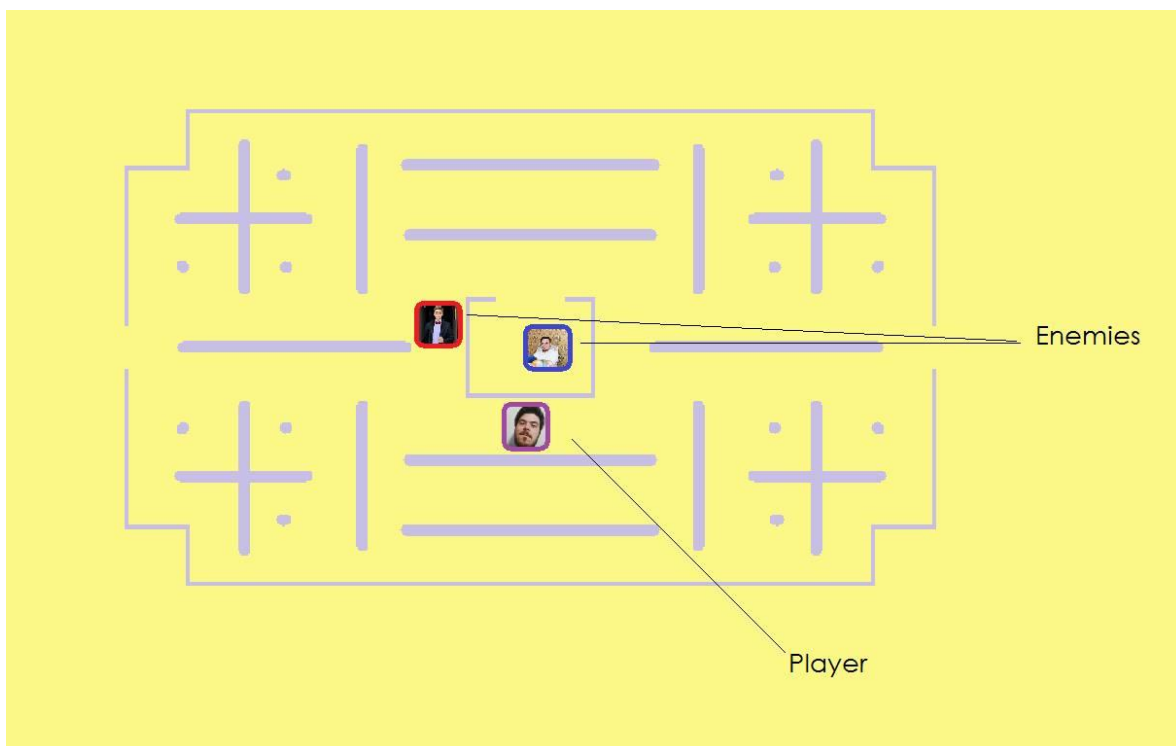
În cazul în care utilizatorul dorește să se înregistreze în baza noastră de date cu un cont nou (apăsând butonul "Register Now"), acesta va fi redirecționat către o nouă pagină.



A registration form on a light yellow background. The form is enclosed in an orange rounded rectangle. It contains four text input fields stacked vertically, labeled 'Username:', 'E-mail:', 'Password:', and 'Confirm password:'. Below these fields is an orange button with the text 'Register Now' in white.

- 1 - **Username**: Utilizatorul scrie numele pe care acesta îl dorește pentru contul sau.
- 2 - **E-mail**: Câmpul care cere adresa de e-mail asociată contului respectiv.
- 3 - **Password**: Parola pe care utilizatorul o dorește.
- 4 - **Confirm password**: Un camp care trebuie să fie identic cu cel anterior pentru a evita cât mai multe erori de scriere a acesteia (aceste câmpuri fiind criptate la scris).

După ce utilizatorul este logat, acesta va putea să pornească un joc de FaceMaze, cu labirint generat aleatoriu, unde fantomele sunt prieteni de pe rețeaua de socializare utilizată pentru a se loga sau cu fantome clasice în cazul în care este logat cu ajutorul bazei noastre de date.



## 4. Arhitectura aplicației și etape de dezvoltare



Aplicația Web FaceMaze va implementa o arhitectură bazată pe șablonul MVC (Model – View – Controller). Pentru a realiza această implementare se va utiliza CodeIgniter, un framework de PHP lansat în 2006 care în prezent are o reputație foarte bună când vine vorba de performanță și de modul ușor în care poate fi utilizat. Astfel, aplicația va fi împărțită în trei mari module ce urmează să fie implementate:

1. *Controller* – reprezintă nucleul aplicației, având rolul de a stabili legătura dintre Model și View, fiind necesar pentru a procesa cererile de tipul POST/GET.
2. *Model* – reprezintă partea logică a aplicației, având rolul de a gestiona resursele aplicației precum baza de date utilizată pentru a stoca informații despre utilizatori. De asemenea, tot aici vor fi implementate diverse clase PHP ce descriu mecanica jocului (generarea labirintului, condițiile pentru a câștiga, comportamentul personajelor).
3. *View* – reprezintă partea de front-end prin care utilizatorul va putea să interacționeze cu aplicația. Conține fișiere HTML, CSS și JavaScript. Pe partea de front-end se va oferi un design responsive.

În general, organizarea aplicației va fi influențată puternic de framework-ul CodeIgniter. După cum se observă în imaginea din dreapta, framework-ul CodeIgniter se împarte în trei foldere principale:

1. *application* – conține codul sursă al aplicației ce va fi împărțit pe module cum ar fi *controllers*, *models*, *views*, *language*, etc.
2. *system* – conține codul sursă al framework-ului CodeIgniter.
3. *user\_guide* – conține user guide-ul framework-ului

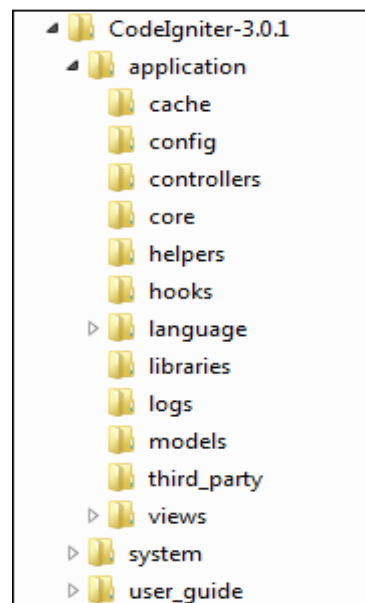
Principalele etape de dezvoltare a aplicației sunt următoarele (nu neapărat în această ordine) :

1. Proiectarea bazei de date Oracle 12c ce va stoca diverse informații despre useri (user, parolă, email, cel mai bun scor, ultimul scor). Această etapă presupune atât construirea bazei de date, cât și implementarea unei clase PHP ce are ca rol comunicarea cu baza de date.

2. Implementarea unor clase PHP ce descriu diverse componente ale jocului FaceMaze, cum ar fi labirintul generat aleatoriu, personajul controlat de jucător, personajele „rele”, condițiile pentru a câștiga, etc.

3. Realizarea componentei de front-end prin care user-ul va interacționa cu aplicația. Partea de front-end va avea un design responsive.

4. Implementarea unor API-uri precum Facebook sau Twitter cu rolul de a permite user-ului să se logheze folosindu-și contul de pe rețeaua socială respectivă și de a beneficia de o experiență de joc superioară datorită anumitor informații legate de contul său de socializare, informații ce vor fi folosite în joc (precum înlocuirea personajului principal din



joc cu poza de profil de pe Facebook a jucătorului). De asemenea, aplicația va putea posta pe Twitter anumite mesaje generate automat.

5. Implementarea formularelor de login și de register.

6. Implementarea unui Controller care să stabilească legătura dintre partea logică și partea vizuală.

## 5. API-uri utilizate

### 5.1 Facebook Graph API



API-ul Facebook va fi folosit pentru logarea cu contul de Facebook și pentru a prelua poza de profil a utilizatorului, plus pozele de profil a unor prieteni, poze ce vor înlocui personajele din Pac-Man: poza de profil a jucătorului va înlocui personajul Pac-Man iar pozele de profil a prietenilor vor înlocui fantomele din joc.

Următorul request este utilizat pentru a obține informații de bază despre jucător (id, nume și poză de profil):

```
GET graph.facebook.com/me?fields=id,name,picture
```

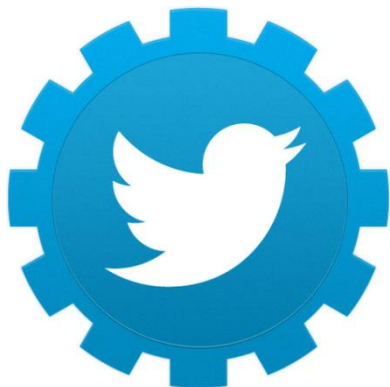
Următorul request este utilizat pentru a obține lista de prieteni a jucătorului:

```
GET graph.facebook.com/me/friends
```

Următorul request este utilizat pentru a obține poza de profil a unui prieten:

```
GET graph.facebook.com/{user-id}/picture
```

## 5.2 Twitter API



API-ul de Twitter va fi folosit în aceeași manieră ca API-ul de Facebook dar va permite și postarea unor mesaje generate automat atunci când o fantomă a atins jucătorul, precum "Jucand #FaceMaze, l-am înghitit pe @utilizator".

Pentru login se utilizează următorul request:

GET

[https://api.twitter.com/1.1/account/verify\\_credentials.json](https://api.twitter.com/1.1/account/verify_credentials.json)

Pentru a posta un tweet se utilizează următorul request:

POST <https://api.twitter.com/1.1/statuses/update.json?status={mesaj}>

Pentru a obține informații de bază legate de jucător se utilizează următorul request:

GET [https://api.twitter.com/1.1/users/show.json?screen\\_name={username}](https://api.twitter.com/1.1/users/show.json?screen_name={username})

Pentru a obține lista cu prieteni (utilizatori de Twitter pe care jucătorul îi urmărește) se utilizează următorul request:

GET [https://api.twitter.com/1.1/friends/list.json?screen\\_name={username}](https://api.twitter.com/1.1/friends/list.json?screen_name={username})

## 6. Concluzii

Prin FaceMaze am dorit să realizăm un joc pentru toată lumea, dedicat relaxării și amuzamentului, care să se diferențeze de alte implementări ale jocului Pac-Man prin caracterul social al jocului. Astfel, jucătorii vor putea să înlocuiască clasicele personaje ale jocului Pac-Man cu prietenii lor de pe rețelele de socializare iar faptul că vor putea posta mesaje legate de activitatea lor din joc va atrage noi jucători. Datorită design-ului responsive, jocul va putea fi jucat pe o varietate largă de device-uri.