

“Project Stream”

- Card game like “Hearthstone”
 - Each card is represented by a **twitch streamer / league of legends champion**
 - Cards have talents, roles, characteristics..
-
- Worked with a web developer that manage learn2play.fr
 - Designed the database of the card game
 - Developed Unity application about the collection of cards

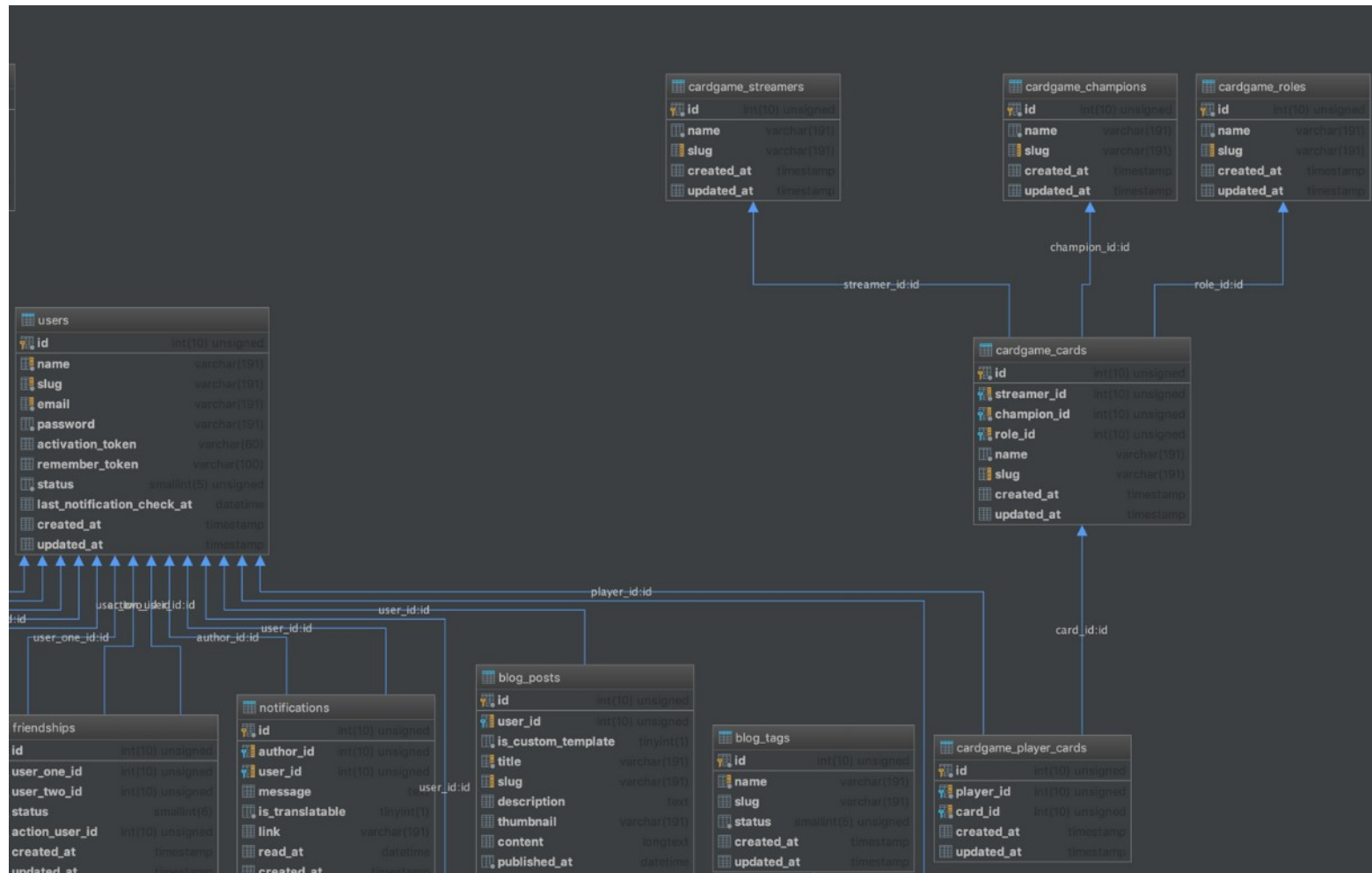
Why this project ?

- Serious games / Gamification
- Database
- Manipulate data
- Discover Web development and API's
- + Potential of the project (Narkuss)

How did I developed this project ?

- Worked with a team to define the card game (rules, cards effects...)
- Designed the database & implement it with the web developer
- Developed the Unity application, matching with the already existing database from L2P website

Database of the card game



Routes of the API

// Card game routes

```
$router->group([
    'prefix' => 'cardgame',
    'namespace' => 'CardGame',
], function () use ($router) {
    $router->get('card/{page?}/{limit?}', 'CardController@index');
    $router->post('register-card/{card}/{player}', 'CardController@register');
    $router->get('player/{player}/{page?}/{limit?}', 'PlayerController@index');

    $router->get('streamer', 'DataController@streamer');
    $router->get('champion', 'DataController@champion');
    $router->get('role', 'DataController@role');
    $router->get('talent', 'DataController@talent');
    $router->get('ability', 'DataController@ability');
    $router->get('characteristic', 'DataController@characteristic');
});

(new GameRoute())->gameApiRoutes();
});
```

Example : register a card

```
public function register(Card $card, Player $player): Response
{
    $card->players()->attach($player->id);

    return $this->player($player);
}
```

Generate data : The dash board

m **misteryeti**
Mon compte Déconnexion

Tableau de bord

Plateformes

Jeux

League of Legends

Card game

- Streamers
- Champions
- Roles
- Talents
- Capacités
- Caractéristiques
- Cartes

Utilisateurs, Roles, Permissions

Cartes

Tous les cartes dans la base de données.

+ Ajouter carte

25 enregistrements par page

Nom	Actions
Tiorianna	Caractéristiques Modifier Supprimer
Narkussazor	Caractéristiques Modifier Supprimer
Lamastick	Caractéristiques Modifier Supprimer
Trayl	Caractéristiques Modifier Supprimer
Katafiz	Caractéristiques Modifier Supprimer
Walle	Caractéristiques Modifier Supprimer
Shaatri	Caractéristiques Modifier Supprimer
Melou	Caractéristiques Modifier Supprimer
Chati	Caractéristiques Modifier Supprimer

GET & POST on Unity

Problems :

- Find a way to execute GET & POST requests
- Make them *generics* ?
- *Return data* from the requests. Still need *generic*
- Allow the developer to execute methods :
 - *before/after* executing the request
 - In case of *failure/successful* request

=> Needed lot of re-factorization

Generic GET method

How ?

- Use Delegates
- Use Coroutines
- Generic objects
- Convert Json data to the generic object
- Execute method before/after the request
- Execute method in case of failure/successful request

Generic GET method

//Get generic method

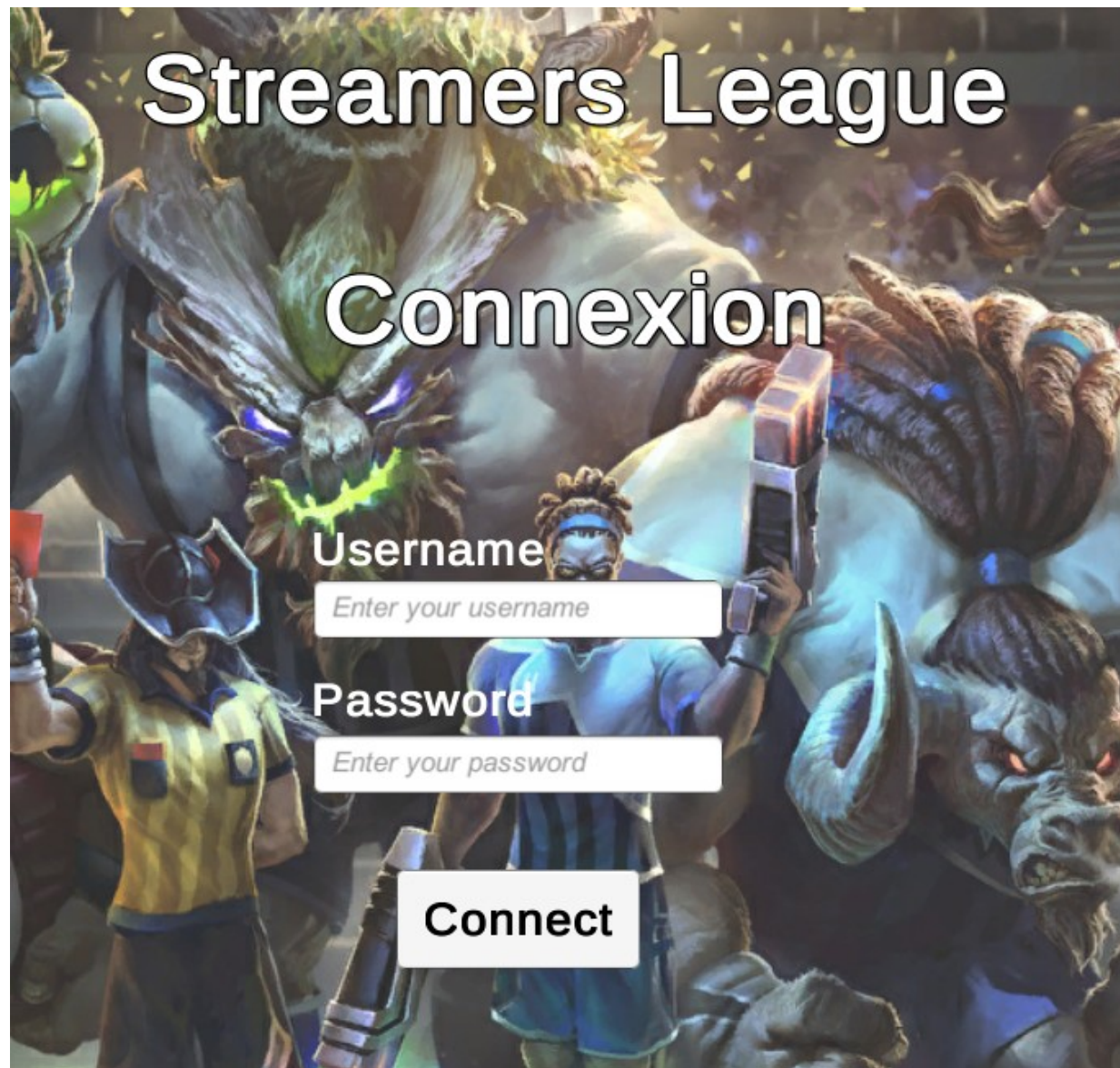
```
public static IEnumerator RequestCoroutineGet<T>(string url, Action<T> getObject = null, Action actionFail = null, Action actionBefore = null, Action actionAfter = null)
{
    if (actionBefore != null)
        actionBefore();

    UnityWebRequest www = UnityWebRequest.Get(url);
    yield return www.SendWebRequest();

    if (www.isNetworkError || www.isHttpError)
    {
        actionFail();
    }
    else
    {
        string jsonResult =
            System.Text.Encoding.UTF8.
                GetString(www.downloadHandler.data);
        T obj = JsonHelper.GetJsonObject<T>(jsonResult);
        getObject(obj);

        if (actionAfter != null)
            actionAfter();
    }
}
```

Example : The authentication



Example : The authentication

What do we want after clicking on “connect” ?

1) Disable the connect button + Show loading bar

2) Execute the request :

- If fail → Do not connect + Enable button + Remove loading bar
- If success → Load all cards **THEN** connect (Load scene) + Remove loading bar

All parts need to be separated → One task at a time

Example : The authentication

```
public void Authentication()
{
    StartCoroutine(AuthenticateCoroutine());
}

IEnumerator AuthenticateCoroutine()
{
    WWWForm form = new WWWForm();
    form.AddField("identifiant", inputFieldUserName.text);
    form.AddField("password", inputFieldPassword.text);

    yield return Request.RequestCoroutinePost<RootObjectUser>(
        "http://learn2play.fr/api/authenticate",
        form,
        GetUser,
        RequestFailed,
        BlockButton,
        LoadCards
    );
}
```

Example : The authentication

```
public void Authentication()
{
    StartCoroutine(AuthenticateCoroutine());
}

IEnumerator AuthenticateCoroutine()
{
    WWWForm form = new WWWForm();
    form.AddField("identifiant", inputFieldUserName.text);
    form.AddField("password", inputFieldPassword.text);

    yield return Request.RequestCoroutinePost<RootObjectUser>(
        "http://learn2play.fr/api/authenticate",
        form,
        GetUser,
        RequestFailed,
        BlockButton,
        LoadCards
    );
}
```

Before

Example : The authentication

```
public void Authentication()
{
    StartCoroutine(AuthenticateCoroutine());
}

IEnumerator AuthenticateCoroutine()
{
    WWWForm form = new WWWForm();
    form.AddField("identifiant", inputFieldUserName.text);
    form.AddField("password", inputFieldPassword.text);

    yield return Request.RequestCoroutinePost<RootObjectUser>(
        "http://learn2play.fr/api/authenticate",
        form,
        GetUser,
        RequestFailed,
        BlockButton,
        LoadCards
    );
}
```

Fail
Before

Example : The authentication

```
public void Authentication()
{
    StartCoroutine(AuthenticateCoroutine());
}

IEnumerator AuthenticateCoroutine()
{
    WWWForm form = new WWWForm();
    form.AddField("identifiant", inputFieldUserName.text);
    form.AddField("password", inputFieldPassword.text);

    yield return Request.RequestCoroutinePost<RootObjectUser>(
        "http://learn2play.fr/api/authenticate",
        form,
        GetUser,
        RequestFailed,
        BlockButton,
        LoadCards
    );
}
```

Collect data

Fail
Before

Example : The authentication

```
public void Authentication()
{
    StartCoroutine(AuthenticateCoroutine());
}

IEnumerator AuthenticateCoroutine()
{
    WWWForm form = new WWWForm();
    form.AddField("identifiant", inputFieldUserName.text);
    form.AddField("password", inputFieldPassword.text);

    yield return Request.RequestCoroutinePost<RootObjectUser>(
        "http://learn2play.fr/api/authenticate",
        form,
        GetUser,
        RequestFailed,
        BlockButton,
        LoadCards
    );
}
```

Collect data

Fail
Before

After

Example : The authentication

Before executing the request

```
void BlockButton()  
{  
    button.interactable = false;  
    GameManager.Instance.SetLoadingLogo(true);  
}
```

BlockButton

Example : The authentication

In case of failure

```
void RequestFailed()  
{  
    Display(connexionInvalid);  
    button.interactable = true;  
    GameManager.Instance.SetLoadingLogo(false);  
}
```

RequestFailed

Example : The authentication

In case of success :

Collect data (Generic object → RootObjectUser)

```
void GetUser(RootObjectUser rootUser)
{
    dataManager.RootUser = rootUser;
}
```

GetUser

Example : The authentication

In case of success :

Load all cards → Execute another request to get the user's cards

```
void LoadCards()
{
    StartCoroutine(LoadCardsCoroutine());
}

IEnumerator LoadCardsCoroutine()
{
    yield return Request.RequestCoroutineGet<CardGame.RootObjectCard>(
        "http://learn2play.fr/api/cardgame/card",
        GetCards,
        null,
        null,
        ConnexionSuccess
    );
}
```

LoadCards

Repeat the process to **execute any other request !**

What I've learned

- Some aspect of web development
- API Request through Unity
- Use generics
- Architect a project
- Work with a remote team

=> Will help me finding a job in serious game industry

What's Next ?

- Online purchase of cards packs
- Customized selection of cards in the collection
- Find other way to optimize requests ?

The end