

Gamification API

Joseph EL MALLAH

Diana AFFI

Ahmed Hachmi

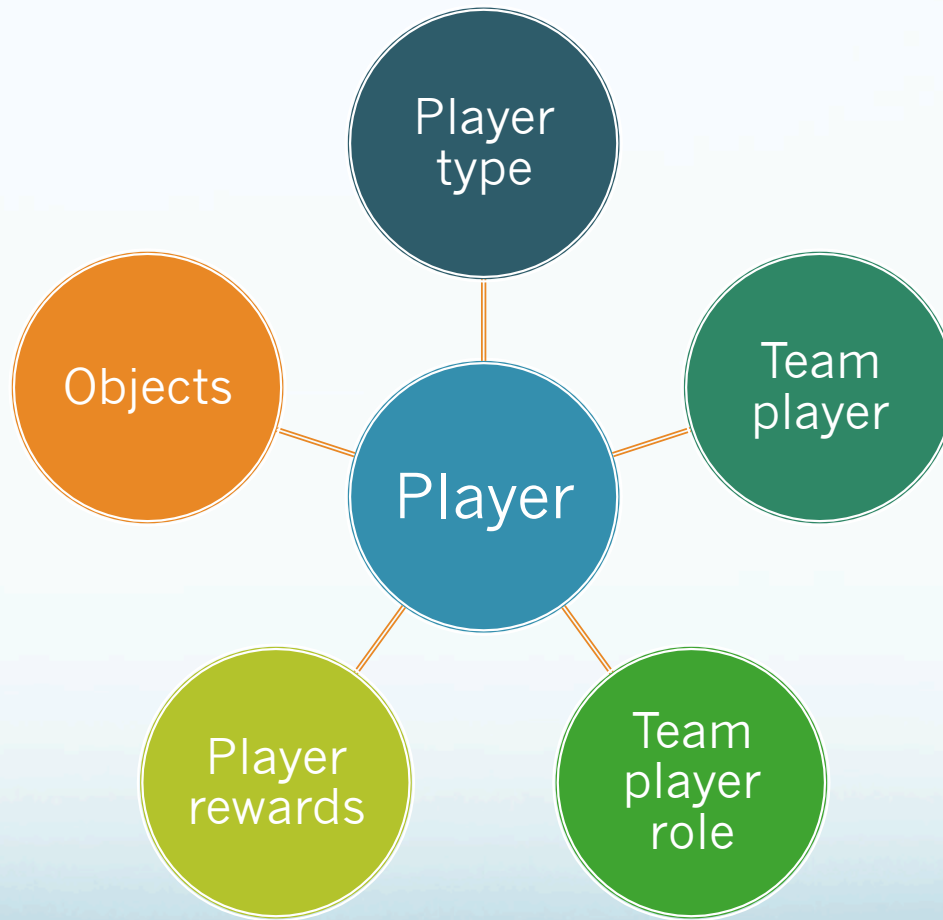
Sommaire

- Object Model
- Architecture de l'API
- Web services
- Conclusion

Object Model 1 of 5



Object Model 2 of 5



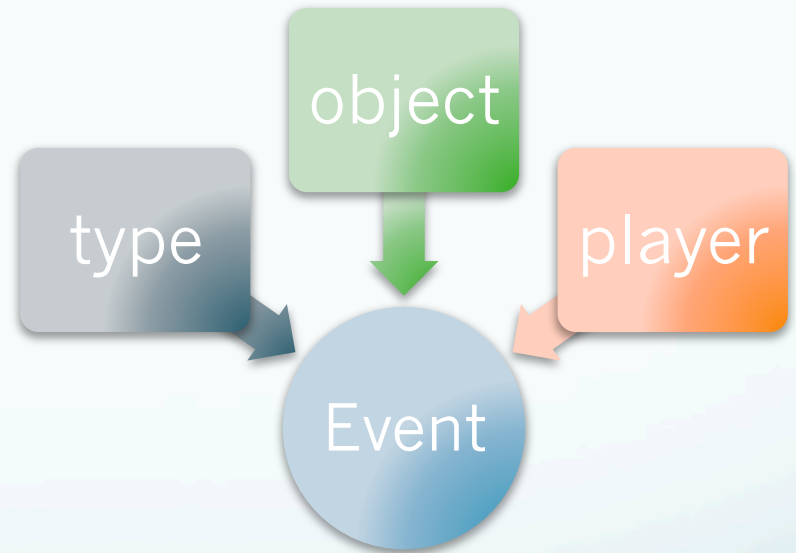
Object Model 3 of 5



Object Model 4 of 5

Event:

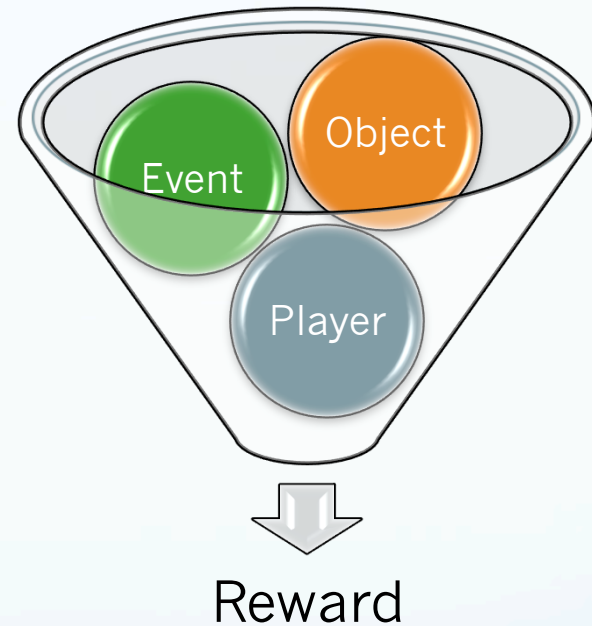
- Has a type
- Invoked by a player
- Invoked on an object
- That object is possessed by the same or another player



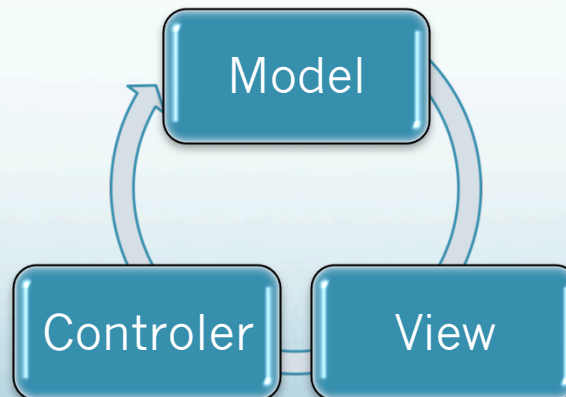
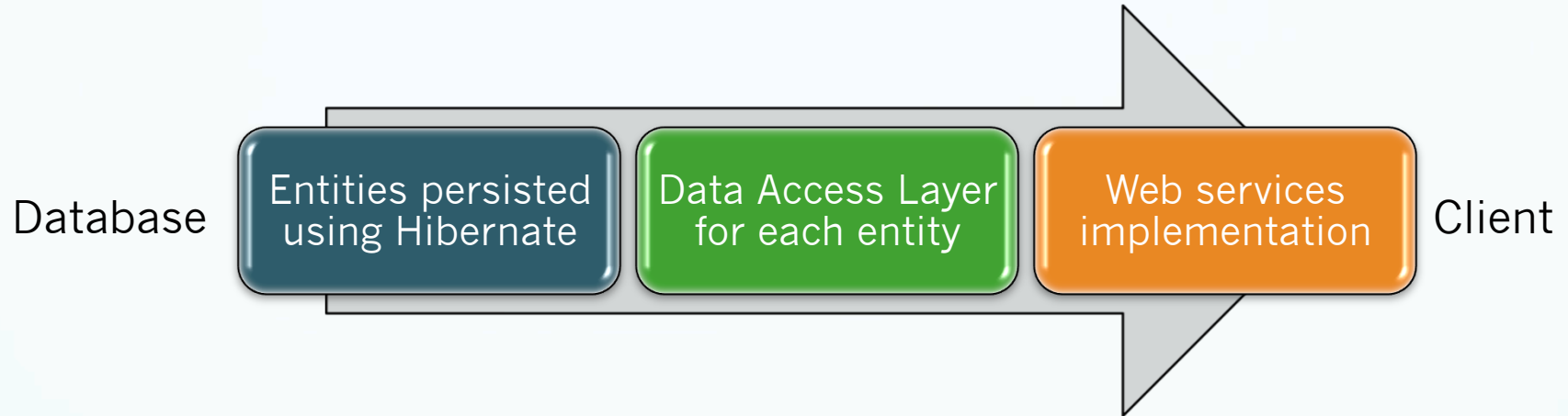
Object Model 5 of 5

Action:

- Takes an object, an event and the player who invoked the event.
- Checks if there is an action conform to the object and event type.
- If yes, the output reward is assigned to the player



Architecture de l'API



Web service - Players

Players

Players related resources of the **Gamification API**

Players Collection

List all Players

GET

/players{?appId, q}

Add Comment  

Parameters

Name	Description	Details
q	filter.	string, required
appId	ID of the application.	int, required

Request

raw

Response

```
1 200 (OK)
2 Content-Type: application/json
1 [{
2   "id": 1, "name": "Diana AFFI", "userType": "person"
3 }, {
4   "id": 2, "name": "Team 1", "userType": "team"
5 }]
```

Web service - Player

Player

A single Player object with all its details

Retrieve a Player

GET

/players/{id}

Add Comment  

Parameters

Name	Description	Details
id	Numeric <code>id</code> of the Player to perform action with. Has example value.	number, required example: 1

Web service – Player rewards

Player Rewards

A single Player's rewards object with all its details

Retrieve Rewards

GET

/players/{id}/rewards

Add Comment  

Parameters

Name	Description	Details
id	Numeric <code>id</code> of the Player to perform action with. Has example value.	number, required example: 1

Web service – POST

Create a Badge

POST

/badges{?appId}

Add Comment  


Example

Debugger

Comments

Try It

Show code sample

none (raw data) 

Request

raw

```
1 | Content-Type: application/json
1 | { "name": "Badge1", "description": "the badge 1", "icon":"icon.png" }
```

Response

```
1 | 201 (Created)
2 | Content-Type: application/json
1 | { "id": 1, "name": "Badge1", "description": "the badge 1", "icon":"icon.png" }
```

Web service – PUT

Update a Badge

PUT

/badges/{id}

Add Comment 

Parameters

Name	Description	Details
id	Numeric <code>id</code> of the Badge to perform action with. Has example value.	number, required example: 1

Example

Debugger

Comments

Try It

Show code sample none (raw data) ▾

Request

raw

```
1 | Content-Type: application/json
1 | { "name": "Badge1", "description": "the badge 1", "icon":"icon.png" }
```

Response

```
1 | 204 (No Content)
```

Web service – DELETE

Remove a Badge

DELETE

/badges/{id}

Add Comment 

Parameters

Name	Description	Details
id	Numeric <code>id</code> of the Badge to perform action with. Has example value.	number, required example: 1


Example

Debugger

Comments

Try It

Show code sample

none (raw data) 

Request

raw

Response

1 | 204 (No Content)

Conclusion

- Token based authentication
- Query param
- More documentation
- More tests and error management