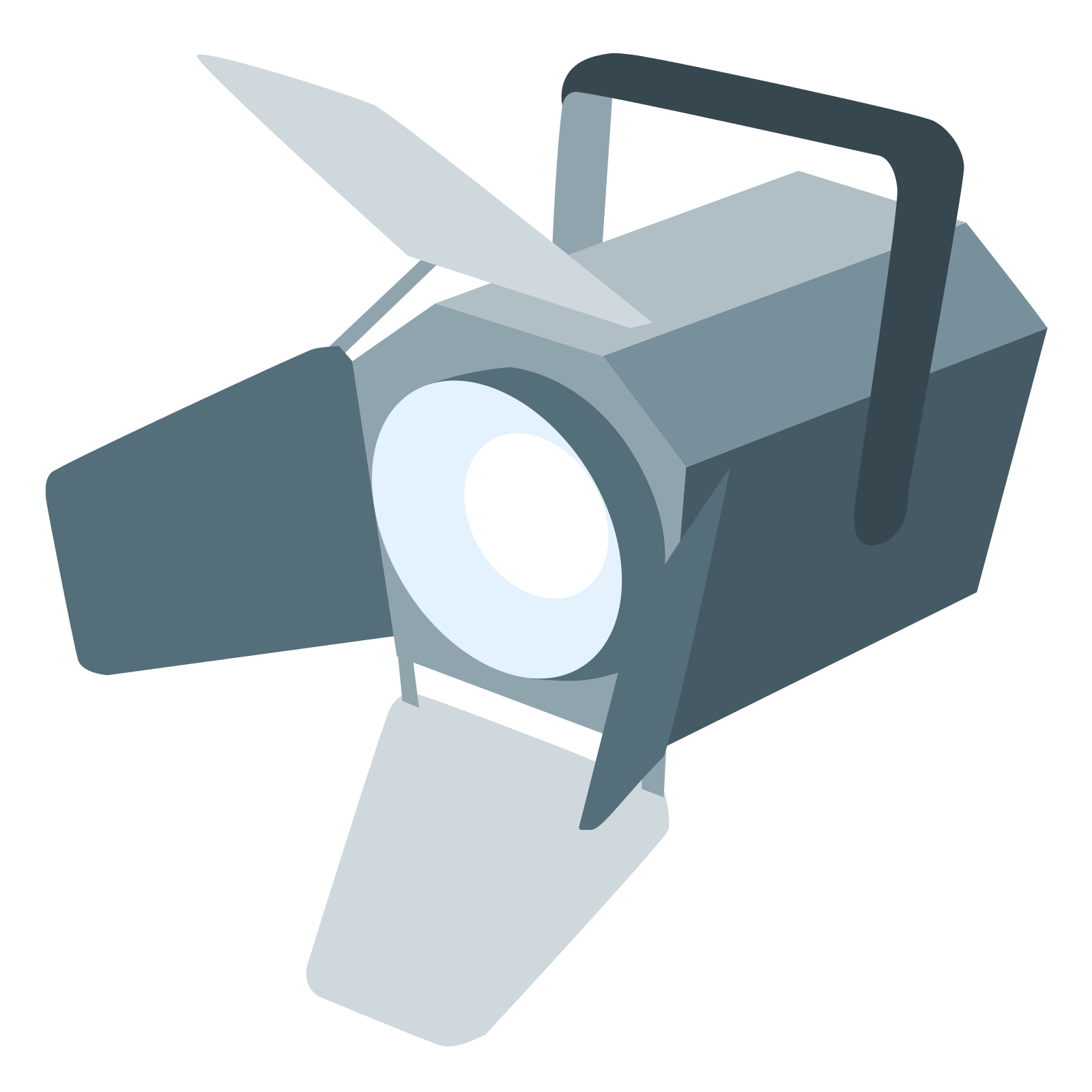
**Rapport d’activité**





**Mission 1 - FaceCast**

Développement d’une application web basé sur un serveur **Node.js**, le framework **Express.js** ainsi qu’une base de données MongoDB **via l’approche par Model de Mongoose** .

**21/11/2017**

**Crété Jonathan – Adelaïde Donovan - Lycée Léonard de Vinci**

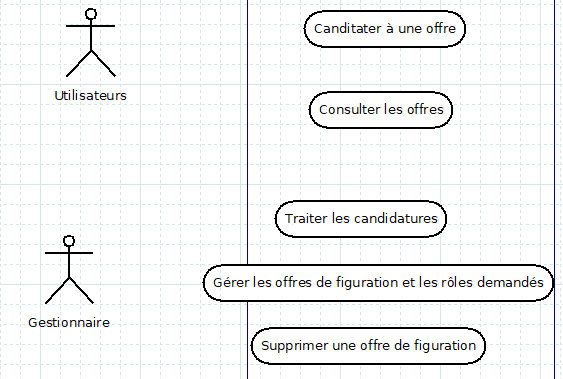
**PLAN :**

1. Cahier des charges
2. Expression des besoins
3. Analyse
4. Mise en œuvre de la mission
5. Structure de la mission
6. API REST
7. Test(s) Unitaire
8. Conclusion
9. Cahier des charges
10. Expression des besoins

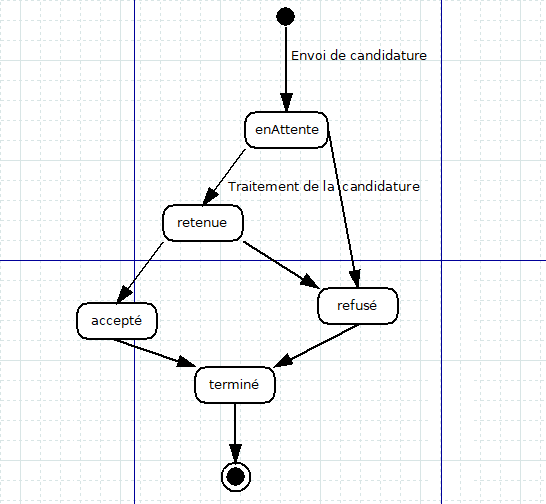
L’agence cliente FaceCast, demandeuse de l’application, a besoin d’une application web afin de traiter au mieux ses candidatures. Le projet se découpant en 2 applications (web & Android) nous intervenons alors dans la conception de l’application destinée au « back Office » à savoir l’application web.

Cette application étant destinée au gestionnaire de l’agence, celui-ci doit pouvoir ajouter une offre ainsi que gérer en temps réel le traitement des candidatures (différents états).

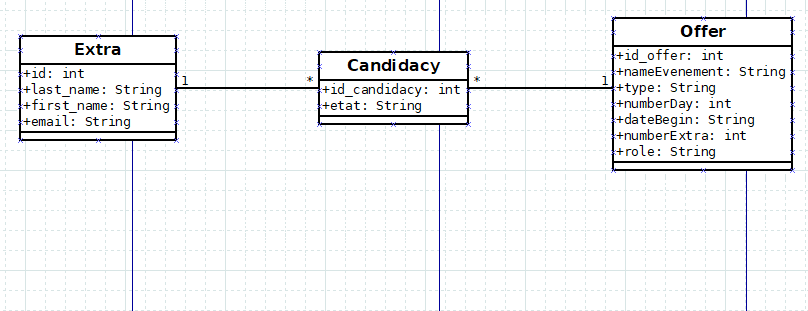
1. Analyse
2. Diagramme des cas d’utilisation de FaceCast



1. Diagramme UML E/T du cycle de vie d’une demande de figuration



1. Diagramme UML des entités



1. Exemple Json

/\* 1 \*/

{

"\_id" : ObjectId("59fed81b069a78003b0bc631"),

"nameEvenement" : "Gary Movie",

"type" : "Spectacle",

"dateBegin" : "2018-01-25",

"numberDay" : 15,

"numberExtra" : 5,

"role" : "Plombier",

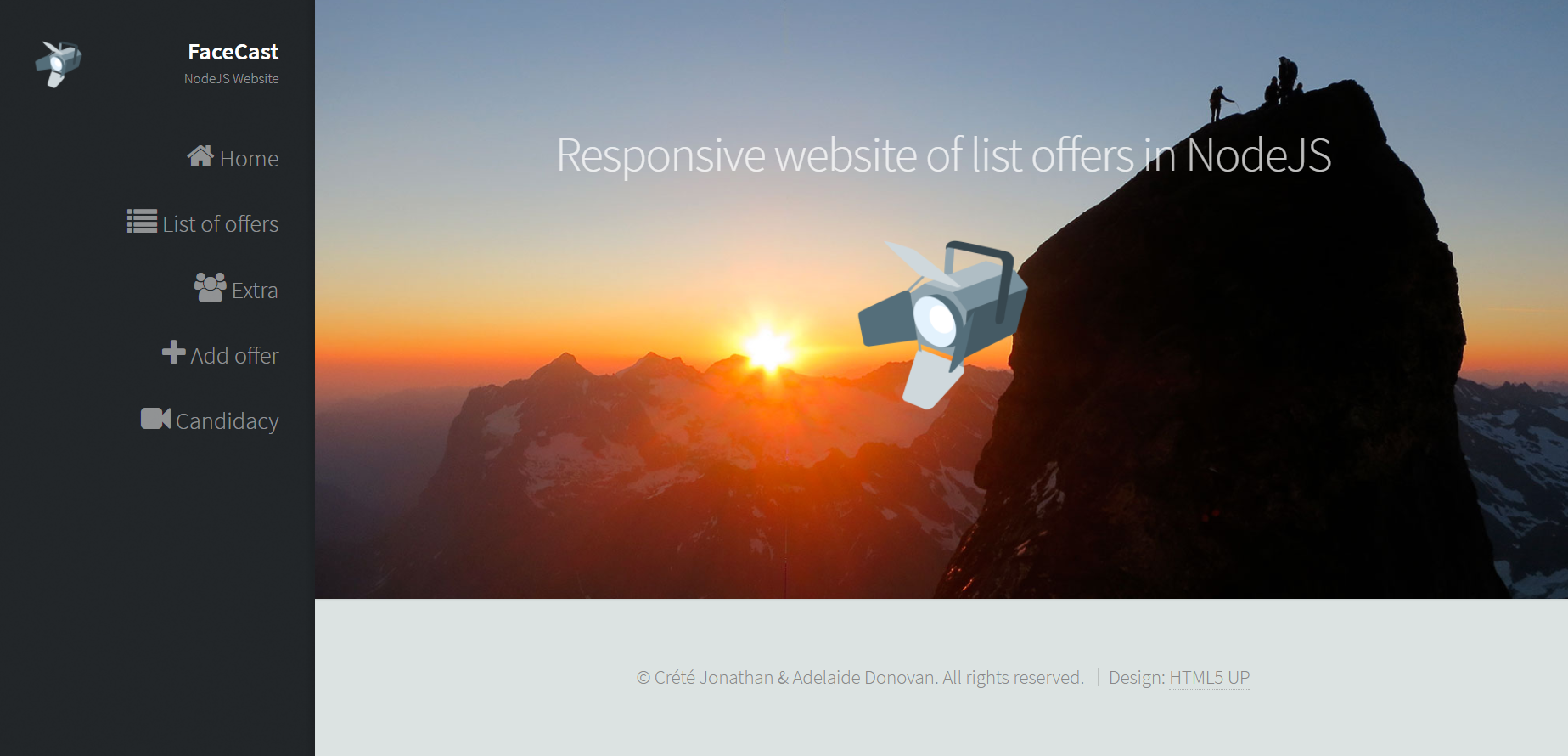
"\_\_v" : 0

}

1. Tableau URI

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Méthode HTTP | URI | Action résultante | Format échangé | Cas d’utilisation |
| Liste des Offres | FaceCast/Offers | GET | Json | Affichage des offre |
| Candidature a une Offre | FaceCast/Offers/Add | GET | Json | Candidature a une offre |
| Offre déjà postuler | FaceCast/Offers/id figurant, id offer | Post | Json | Affichage des offre déjà postuler par le figurant |

1. Mise en œuvre de la mission
2. Structure de la mission



1. API REST

*// JavaScript Document*

**var** express = require('express');

**var** router = express.Router();

**var** path = require('path');

**var** fs = require('fs');

**var** modelCandidacy = require('../models/candidacy');

**var** modelExtra = require('../models/extra');

**var** modelOffer = require('../models/offer');

**var** objectId = require('mongoose').Types.ObjectId;

router.get('/', **function** (req, res, next) {

res.send('Hi ApiRest , Choose view : /offer or /extra or /candidacy');

});

*// Offers*

router.get('/offer', **function** (req, res, next) {

modelOffer.Offer.find({}, **function** (err, offres) {

**if** (err) **throw** err;

res.json(offers);

});

});

router.get('/offer/:nb', **function** (req, res, next) {

model.Offer.find({}, **function** (err, offers) {

**if** (err) **throw** err;

**if** (req.params.nb == 0) {

res.json(offres[0]);

}

**else** {

res.json(offers.splice(0, req.params.nb));

}

});

});

*// Extras*

router.get('/extra', **function** (req, res, next) {

modelExtra.Extra.find({}, **function** (err, extras) {

**if** (err) **throw** err;

res.json(extras);

})

});

*// candidacys*

router.get('/candidacy', **function** (req, res, next) {

modelCandidacy.Candidacy.find({}, **function** (err, candidacys) {

**if** (err) **throw** err;

res.json(candidacys);

});

});

*// candidacy*

router.post('/candidacy', **function** (req, res, next) {

**var** candidacy = **new** model.Candidacy({

etat: req.body.etat,

idOffer: req.body.idOffer,

idExtra: req.body.idExtra

});

candidacy.save(**function** (err) {

**if** (err) **throw** err;

res.json(candidacy);

});

});

*// My applications*

router.get('/candidacy/:idExtra', **function** (req, res, next) {

modelCandidacy.Candidacy.aggregate([

{

$match: { idExtra: objectId(req.params.idExtra) }

},

{

$lookup: {

from: "offers",

localField: "idOffer",

foreignField: "\_id",

as: "offer"

}

}]).exec(**function** (err, candidacy) {

res.json(candidacy);

});

});

module.exports = router;

1. Test(s) Unitaire(s)

**package** testFaceCast;

**import** **static** org.junit.Assert.\*;

**import** java.io.UnsupportedEncodingException;

**import** org.apache.http.entity.StringEntity;

**import** org.json.JSONArray;

**import** org.json.JSONObject;

**import** org.junit.Before;

**import** com.mashape.unirest.http.HttpResponse;

**import** com.mashape.unirest.http.JsonNode;

**import** com.mashape.unirest.http.Unirest;

**import** com.mashape.unirest.http.exceptions.UnirestException;

**import** org.junit.Test;

**public** **class** TestApp {

JsonNode body;

HttpResponse<JsonNode> response;

HttpResponse<JsonNode> responseExtra;

@Before

**public** **void** setUp() **throws** UnirestException {

response = Unirest.get("http://localhost:3000/rest/candidacy").asJson();

body = response.getBody();

responseExtra = Unirest.get("http://localhost:3000/rest/extra").asJson();

}

@Test

**public** **void** testEtat() {

assertNotNull(response);

assertEquals("Waiting", body.getArray().getJSONObject(0).getString("etat"));

}

@Test

**public** **void** testIdOffreAndIdFigurant() {

*// Test IdOffer*

assertEquals("59ede93db8b17d0068cc3e23", body.getArray().getJSONObject(0).getString("idOffer"));

*// Test IdExtra*

assertEquals("59ec70a849a17b4da7e07f27", body.getArray().getJSONObject(0).getString("idExtra"));

}

*////////////////////////////////////////////////////////////*

@Test

**public** **void** addCandidacy() {

**try** {

HttpResponse<JsonNode> requestCandidacy = Unirest.post("http://localhost:3000/rest/candidacy")

.header("accepted", "application/json").field("etat", "Waiting")

.field("idFigurant", "59ec6fdf49a17b4da7e07ec5").field("idOffer", "59ede93db8b17d0068cc3e23")

.asJson();

assertEquals(200, requestCandidacy.getStatus());

assertEquals(8, response.getBody().getArray().length());

} **catch** (UnirestException e) {

*// TODO Auto-generated catch block*

e.printStackTrace();

fail("error to add candidacy");

}

}

@Test

**public** **void** myApplications() {

**try** {

HttpResponse<JsonNode> responseCandidacy = Unirest

.get("http://localhost:3000/rest/candidacy/59ec70a849a17b4da7e07f27").asJson();

assertEquals(2, responseCandidacy.getBody().getArray().length());

} **catch** (UnirestException e) {

*// TODO Auto-generated catch block*

fail("Error : number of objects in th json");

e.printStackTrace();

}

}

}

1. Conclusion