

DM Jhipster

Oumaima Hajji

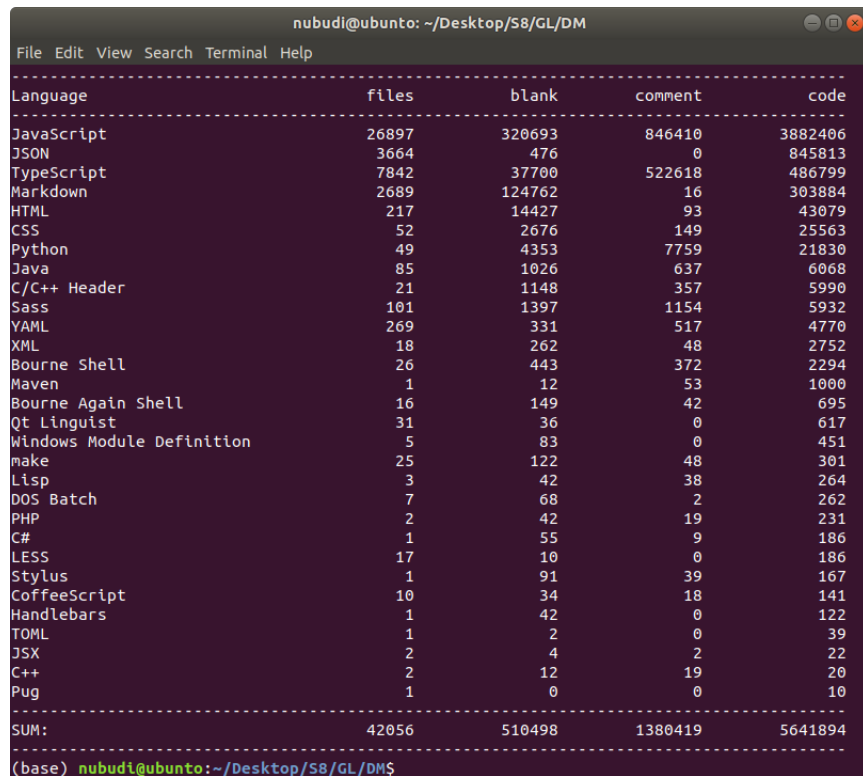
April 2021

1 Partie I

1.1 Page 1

1.1.1 Le nombre de lignes de code générées avec le modèle

Après la génération de l'application avec Jhipster, on utilise l'outil cloc qui calcule le nombre de lignes de code générés par l'application. On trouve comme résultat:



Language	files	blank	comment	code
JavaScript	26897	320693	846410	3882406
JSON	3664	476	0	845813
TypeScript	7842	37700	522618	486799
Markdown	2689	124762	16	303884
HTML	217	14427	93	43079
CSS	52	2676	149	25563
Python	49	4353	7759	21830
Java	85	1026	637	6068
C/C++ Header	21	1148	357	5990
Sass	101	1397	1154	5932
YAML	269	331	517	4770
XML	18	262	48	2752
Bourne Shell	26	443	372	2294
Maven	1	12	53	1000
Bourne Again Shell	16	149	42	695
Qt Linguist	31	36	0	617
Windows Module Definition	5	83	0	451
make	25	122	48	301
Lisp	3	42	38	264
DOS Batch	7	68	2	262
PHP	2	42	19	231
C#	1	55	9	186
LESS	17	10	0	186
Stylus	1	91	39	167
CoffeeScript	10	34	18	141
Handlebars	1	42	0	122
TOML	1	2	0	39
JSX	2	4	2	22
C++	2	12	19	20
Pug	1	0	0	10
SUM:	42056	510498	1380419	5641894

On utilise ainsi le fichier jdl que l'on implémente et on regénère l'application avec ce fichier. On trouve alors comme nombre de ligne de code:

nubudi@ubuntu: ~/Desktop/S8/GL/DM				
File Edit View Search Terminal Help				
Language	files	blank	comment	code
JavaScript	26897	320693	846410	3882406
JSON	3679	476	0	846398
TypeScript	7949	38905	522890	493328
Markdown	2689	124762	16	303883
HTML	246	14644	93	45278
CSS	52	2676	149	25563
Python	49	4353	7759	21830
Java	192	3434	2667	15168
C/C++ Header	21	1148	357	5990
Sass	101	1397	1154	5932
YAML	273	353	635	4975
XML	32	318	183	3261
Bourne Shell	26	443	372	2294
Maven	1	13	52	1048
Bourne Again Shell	16	149	42	695
Qt Linguist	31	36	0	617
Windows Module Definition	5	83	0	451
make	25	122	48	301
Lisp	3	42	38	264
DOS Batch	7	68	2	262
PHP	2	42	19	231
LESS	17	10	0	186
C#	1	55	9	186
Stylus	1	91	39	167
CoffeeScript	10	34	18	141
Handlebars	1	42	0	122
TOML	1	2	0	39
JSX	2	4	2	22
C++	2	12	19	20
Pug	1	0	0	10
SUM:	42332	514407	1382973	5661068

Alors, le nombre de lignes de code générées avec le modèle est: $5661068 - 5641894 = 19174$

1.1.2 Le lien github du projet

Le lien vers le dépôt github/gitlab du projet **coopcycle-jhipster**: <https://github.com/oumaimahajji/Coopcycle>

1.1.3 Le lien github du fichier jdl

Le lien dans le dépôt github/gitlab vers le fichier **coopcycle.jh**: <https://github.com/oumaimahajji/Coopcycle/blob/master/coopcycle.jh>

1.1.4 Le lien github vers la présentation graphique du modèle

Le lien dans le dépôt github/gitlab vers la représentation graphique du modèle **coopcycle.png**: <https://github.com/oumaimahajji/Coopcycle/blob/master/coopcycle.png>

1.2 Page 2

1.2.1 Les commandes à exécuter pour générer l'application

La liste des commandes à exécuter pour générer l'application "coopcycle" est:

jhipster : Pour générer l'application initiale avec Jhipster.

jhipster import-jdl coopcycle.jh : pour utiliser le modèle que l'on crée.

./mvnw : pour lancer l'application.

1.2.2 Copie d'écran de l'entité 'Compte'

Voici une capture d'écran de l'entité **Compte**:

Comptes										Refresh list + Create a new Compte	
ID	Name	Surname	Email	Categorie	Phone Number	Address	Postal Code	City	Member Of		
1	IB connecting	protocol azure fuchsia	TOa@hotmail.fr	CLIENT	0111111111	Virginia overriding drive	84854	West Kirahaven		View	Edit
2	background Enhanced archive	Division calculating	ggrj@hotmail.com	CLIENT	0123456789	Dynamic District Rapid	55555	Candaceville		View	Edit
3	Franc	turn-key Rustic teal	Akwyw4@imag.fr	LIVREUR	0782121548	Arkansas Brand navigate	84568	Port Damontown		View	Edit
4	Customer architect transmitter	copy internet Loan	j7@gmail.fr	CLIENT	0721202154	Steel	84521	West Dario		View	Edit
5	blue Peso	Pound eveballs	SvPUt4@imag.com	COMMERCANT	0845125872	Ball	87111	New Gaston		View	Edit

1.3 Page 3

Voici des copies d'écran de quelques ressources de l'API Swagger de l'application **CoopCycle**

Coopcycle

v0.0.1-SNAPSHOT

Home

Entities

Administration

Account

Development

account-resource

Account Resource

GET

/api/account

getAccount

POST

/api/account

saveAccount

POST

/api/account/change-password

changePassword

POST

/api/account/reset-password/finish

finishPasswordReset

POST

/api/account/reset-password/init

requestPasswordReset

GET

/api/activate

activateAccount

GET

/api/authenticate

isAuthenticated

POST

/api/register

registerAccount

commande-resource

Commande Resource

GET

/api/commandes

getAllCommandes

POST

/api/commandes

createCommande

GET

/api/commandes/count

countCommandes

GET

/api/commandes/{id}

getCommande

PUT

/api/commandes/{id}

updateCommande

<div> <div>Coopcycle</div> <div>v0.0.1-SNAPSHOT</div> </div> <div> Home Entities Administration Account </div>	
<div> <div>compte-resource</div> <div>Compte Resource</div> </div>	
GET	/api/comptes getAllComptes
POST	/api/comptes createCompte
GET	/api/comptes/count countComptes
GET	/api/comptes/{id} getCompte
PUT	/api/comptes/{id} updateCompte
DELETE	/api/comptes/{id} deleteCompte
PATCH	/api/comptes/{id} partialUpdateCompte
<div> <div>cooperative-resource</div> <div>Cooperative Resource</div> </div>	
GET	/api/cooperatives getAllCooperatives
POST	/api/cooperatives createCooperative
GET	/api/cooperatives/count countCooperatives
GET	/api/cooperatives/{id} getCooperative
PUT	/api/cooperatives/{id} updateCooperative
DELETE	/api/cooperatives/{id} deleteCooperative

1.4 Page 4

1.4.1 Le rapport COCOMOII

On utilise le calculateur COCOMOII pour calculer la durée et la charge du projet de l'application **Coop-cycle**. Pour ce faire, on insère le nombre de lignes de code trouvé dans l'étape 1 et on trouve comme résultat:

Results

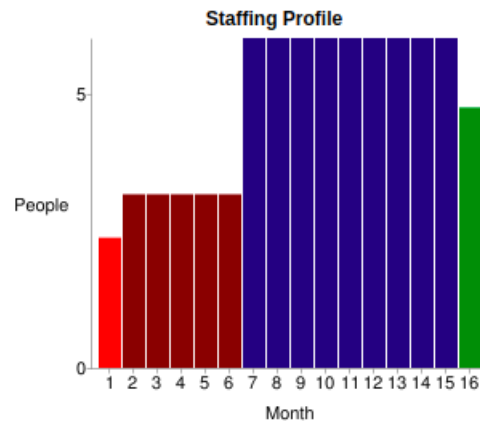
Software Development (Elaboration and Construction)

Effort = 75.7 Person-months
Schedule = 15.3 Months
Cost = \$0

Total Equivalent Size = 19174 SLOC
Effort Adjustment Factor (EAF) = 1.00

Acquisition Phase Distribution

Phase	Effort (Person-months)	Schedule (Months)	Average Staff	Cost (Dollars)
Inception	4.5	1.9	2.4	\$0
Elaboration	18.2	5.7	3.2	\$0
Construction	57.5	9.6	6.0	\$0
Transition	9.1	1.9	4.7	\$0



Software Effort Distribution for RUP/MBASE (Person-Months)

Phase/Activity	Inception	Elaboration	Construction	Transition
Management	0.6	2.2	5.8	1.3
Environment/CM	0.5	1.5	2.9	0.5
Requirements	1.7	3.3	4.6	0.4
Design	0.9	6.5	9.2	0.4
Implementation	0.4	2.4	19.6	1.7
Assessment	0.4	1.8	13.8	2.2
Deployment	0.1	0.5	1.7	2.7

Your output file is http://softwarecost.org/tools/COCOMO/data/COCOMO_April_11_2021_15_13_50_383131.txt

Vous pouvez retrouver ce rapport en format txt en cliquant sur ce lien github: <https://github.com/oumaimahajji/Coopcycle/blob/master/cocomoii.txt>

1.4.2 Les coûts de développement

Pour une équipe de 5 personnes située à chacune de ces villes, on trouve le salaire de l'équipe en calculant la somme des salaires de chaque membre de l'équipe.

Les coûts de développement en Grenoble:

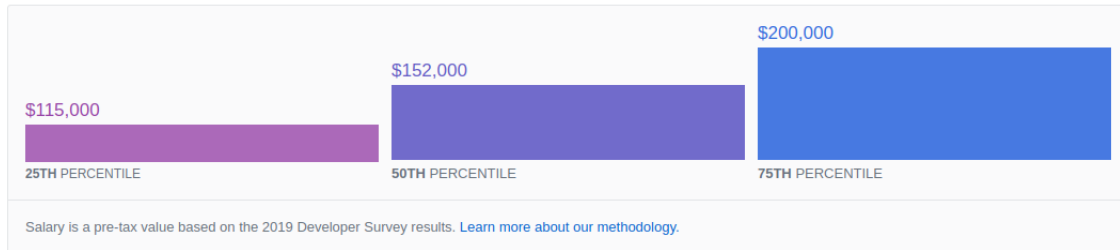


En considérant que le salaire annuel est: 53000 euros = 63074 dollars, on trouve un salaire mensuel de 5256 dollars. Et donc pour toute l'équipe, on trouve: 26280 dollars.

Les coûts de développement en San Francisco:

Annual Salary

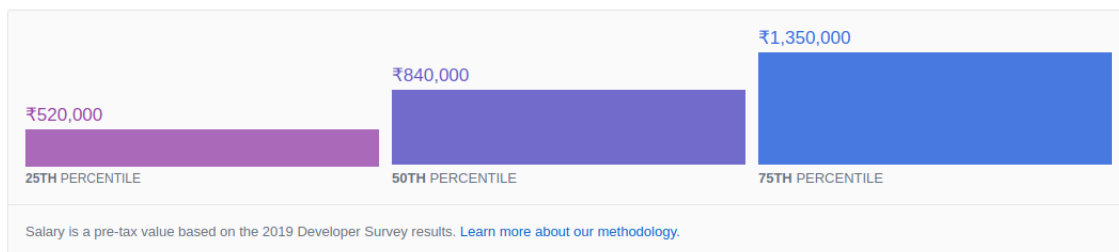
full stack developer san francisco, ca 5 years experience graduate degree (e.g. ma, ms) html java javascript python typescript



En considérant que le salaire annuel est: 200000 dollars, on trouve un salaire mensuel de 16666 dollars. Et donc pour toute l'équipe, on trouve: 83333 dollars.
Les coûts de développement en Bengaluru:

Annual Salary

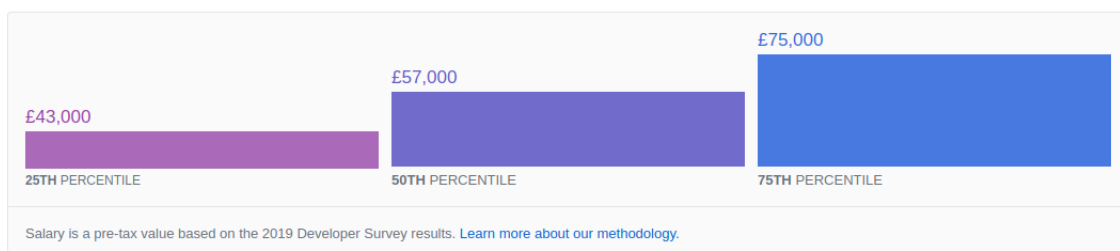
full stack developer bengaluru, india 5 years experience graduate degree (e.g. ma, ms) html java javascript python typescript



En considérant que le salaire annuel est: 1350000 rupee = 18066 dollars, on trouve un salaire mensuel de 1505 dollars. Et donc pour toute l'équipe, on trouve: 7527 dollars.
Les coûts de développement en London (stackoverflow ne donne pas la possibilité de calculer en Switzerland):

Annual Salary

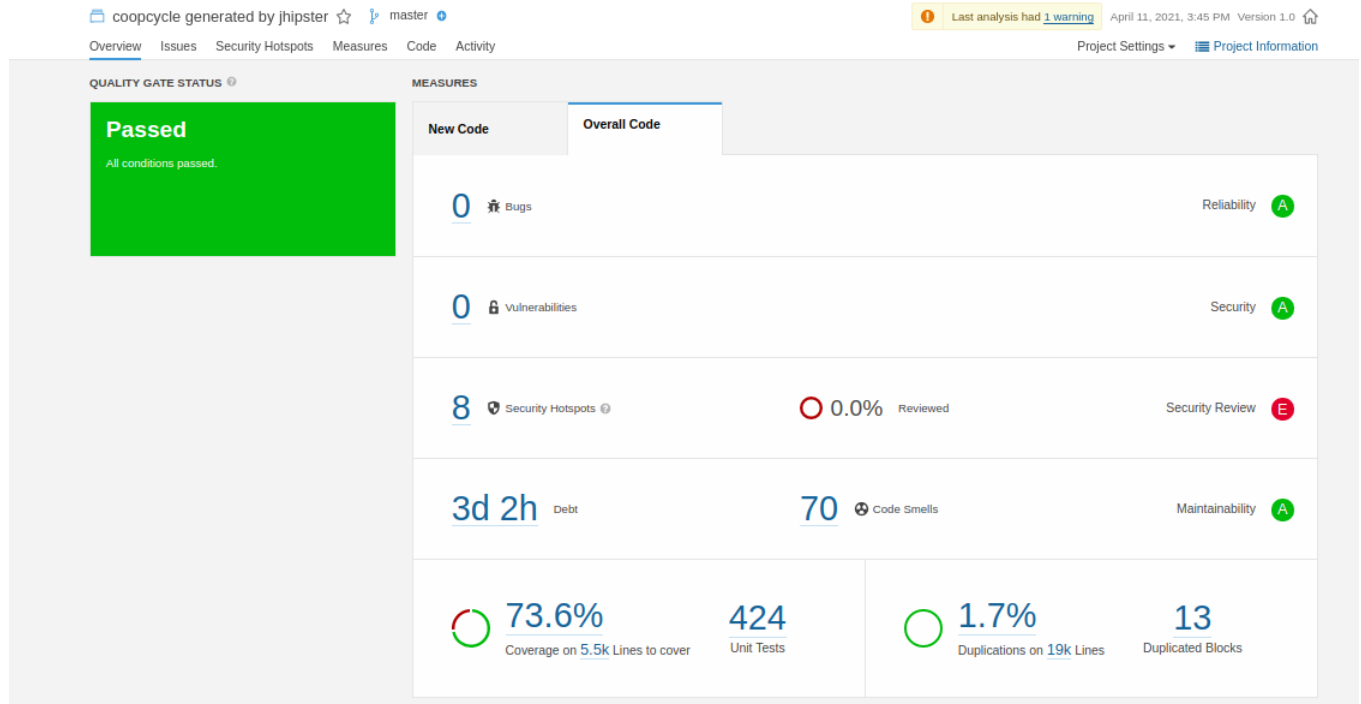
full stack developer london, uk 5 years experience graduate degree (e.g. ma, ms) html java javascript python typescript



En considérant que le salaire annuel est: 75000 pound = 102813 dollars, on trouve un salaire mensuel de 8567 dollars. Et donc pour toute l'équipe, on trouve: 42838 dollars.

1.5 Page 5

Après avoir utilisé le docker pour lancer **SonarQube**, on utilise ses services pour trouver des informations sur la qualité du code de l'application **Coopcycle**. Voici le rapport **SonarQube** de qualité:



1.6 Page 6

J'ai passé sur la partie API de l'application pour faire cette manipulation et pour avoir les url et la commande curl à executer mais j'ai pas pu récupérer les bons résultats à cause d'un problème d'authentification. Vous pouvez voir ça dans la figure suivante:

GET

/api/commandes/{id} getCommande

Parameters

Name	Description
id ★ required	
integer(\$int64)id	
(path)	
	0

Execute

Clear

Responses

Curl

```
curl -X 'GET' \
  'http://localhost:8080/api/commandes/0' \
  -H 'accept: */*'
```

Request URL

```
http://localhost:8080/api/commandes/0
```

Server response

Code	Details
401	<div>Error: Unauthorized</div> <div> <div>Response body</div> <pre>{ "type": "https://www.jhipster.tech/problem/problem-with-message", "title": "Unauthorized", "status": 401, "detail": "Full authentication is required to access this resource", "path": "/api/commandes/0", "message": "error.http.401" }</pre> </div>

2 Partie II

2.1 Page 1

2.1.1 Le nombre de ligne de code ajouté

Le nombre de ligne de code ajoutée et modifiée pour la modification apportée au générateur: 2 dans le fichier : entity-management.module.ts.ejs 7 dans le fichier : entity-management-detail.component.html.ejs

2.1.2 Le lien du dépôt github du fork générateur

Voici le lien du github:

<https://github.com/oumaimahajji/generator-jhipster>

2.1.3 Le lien du dépôt github du commit

Voici le lien du github:

<https://github.com/oumaimahajji/generator-jhipster/commit/aadca3bbebeb5793a3b97c1409ad31b9ef97a2ab>

2.1.4 Le lien du dépôt github des deux fichiers modifiés

Voici les liens github:

<https://github.com/oumaimahajji/generator-jhipster/blob/main/generators/entity-client/templates/angular/src/main/webapp/app/entities/entity-management.module.ts.ejs>

<https://github.com/oumaimahajji/generator-jhipster/blob/main/generators/entity-client/templates/angular/src/main/webapp/app/entities/detail/entity-management-detail.component.html.ejs>

2.2 Page 2

Vous pouvez accéder à la nouvelle application sur github avec ce lien: <https://github.com/oumaimahajji/Coopcycle2>

2.2.1 La liste des commandes:

Voici la liste des commandes à exécuter pour régénérer l'application **Coopcycle**:

npm link /home/nubudi/Desktop/S8/GL/DM/generator-jhipster

jhipster --skip-jhipster-dependencies


jhipster import-jdl coopcycle.jh --with-entities

npm install angular2-qrc

./mvnw

2.2.2 Copie d'écran de l'entité 'Compte' de la nouvelle application

Compte

ID	1
Name	mary
Surname	doe
Email	fy@hotmail.fr
Categorie	CLIENT
Phone Number	0666066606
Address	Virginia overriding drive hell
Postal Code	66666
City	West Kira
Member Of	Unbranded directional
QRCode	

[← Back](#) [Edit](#)