FILE NAMING WEB APP
WITH PYTHON / SQL

Erim Çelen & Lasha Gochiashvili

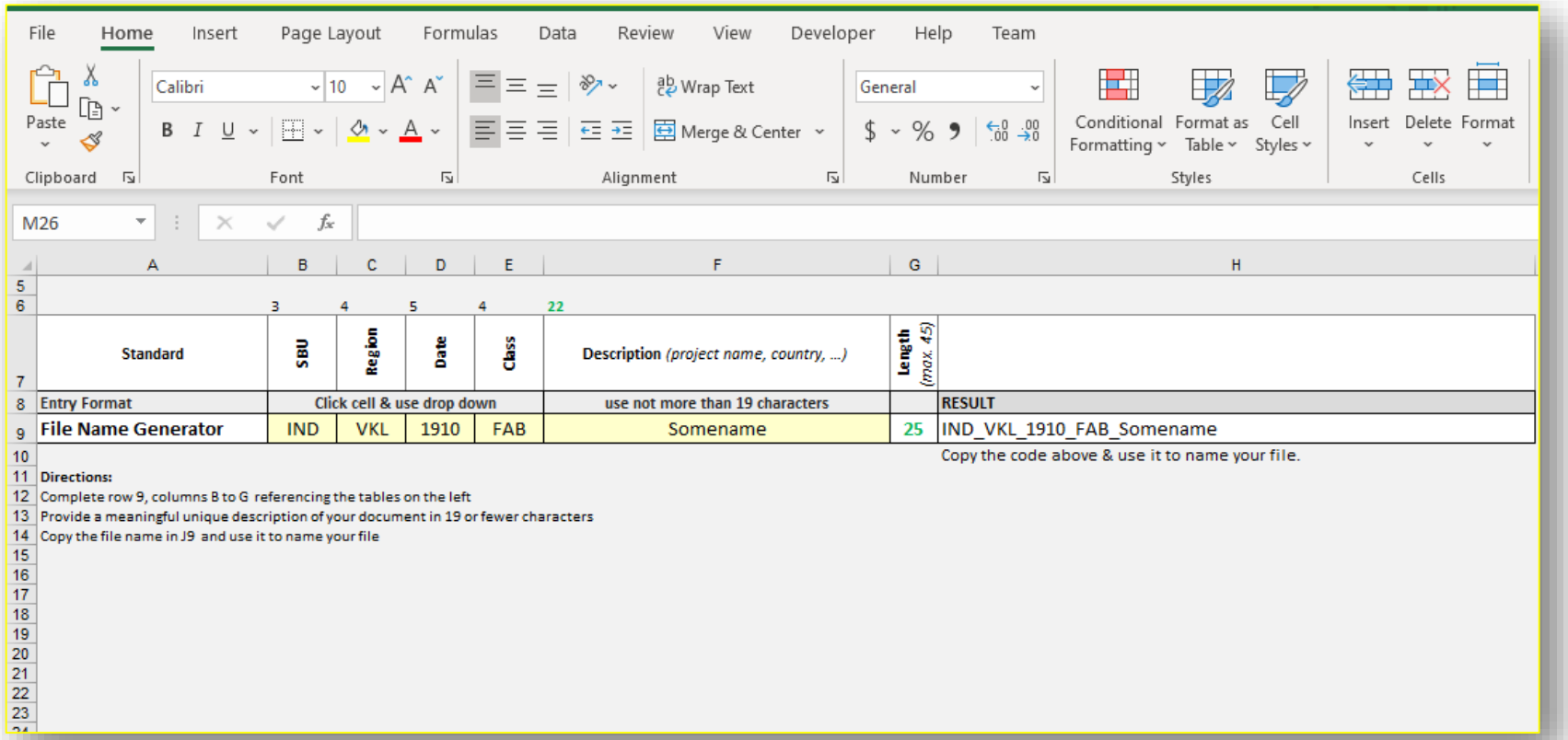
AGENDA:

1. Problem
2. Solution
3. Technology
4. Database Scheme
5. Web App
6. Work Distribution
7. Conclusions

WHAT IS A PROBLEM?

Source: https://me.me/i/naming-files-you-dont-care-about-u7gtli8iguuu-f71c719385f744bca2bc5dbfa9391b40

EXISTING SOLUTION… HMM…



	Standard	SBU	Region	Date	Class	Description *(project name, country, …)*	Length *(max. 45)*	
		3	4	5	4	22		
Entry Format		Click cell & use drop down				use not more than 19 characters		RESULT
File Name Generator		IND	VKL	1910	FAB	Somename	25	IND_VKL_1910_FAB_Somename

Copy the code above & use it to name your file.

Directions:

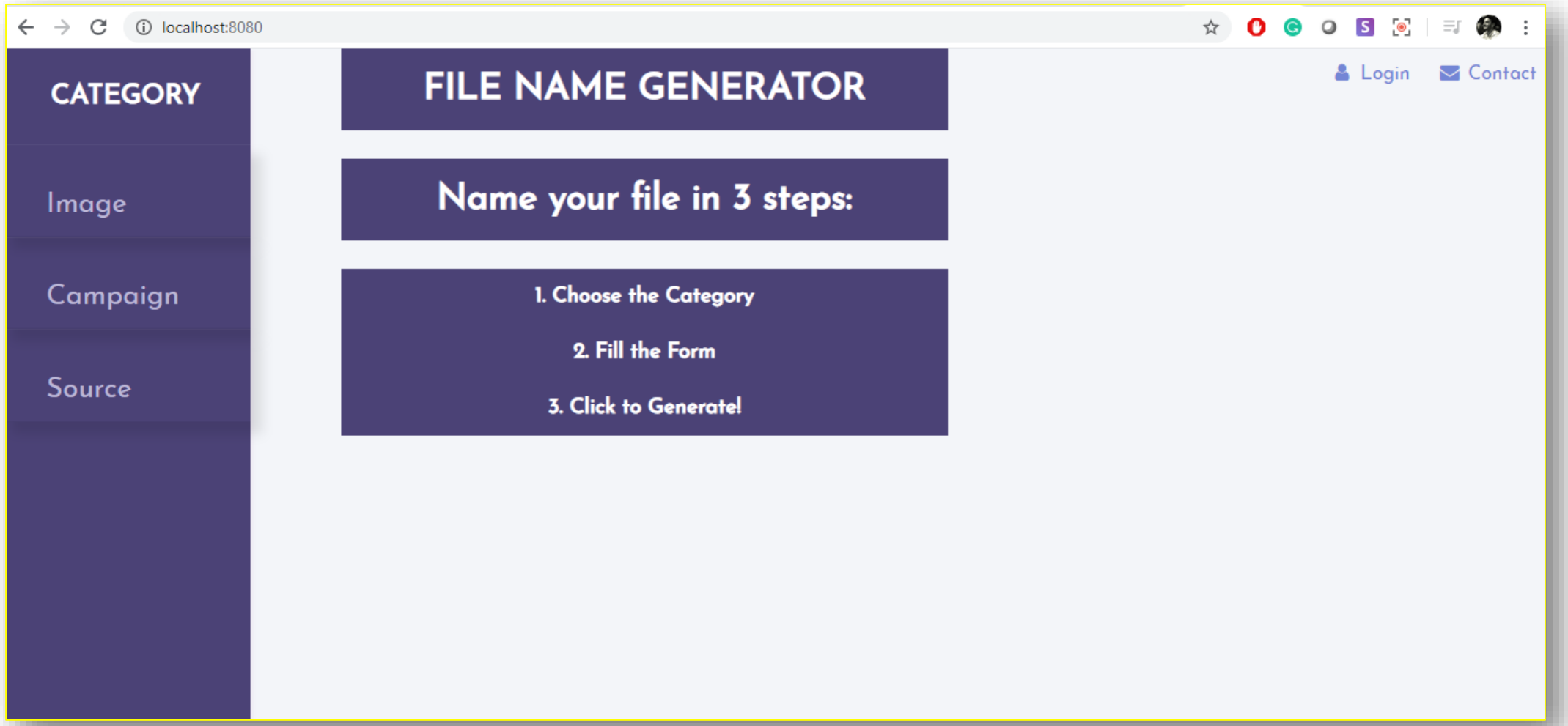
Complete row 9, columns B to G referencing the tables on the left

Provide a meaningful unique description of your document in 19 or fewer characters

Copy the file name in J9 and use it to name your file

Source: my file with changed names

HOW TO SOLVE PROBLEM?



WHAT TECHNOLOGY DID WE USE?

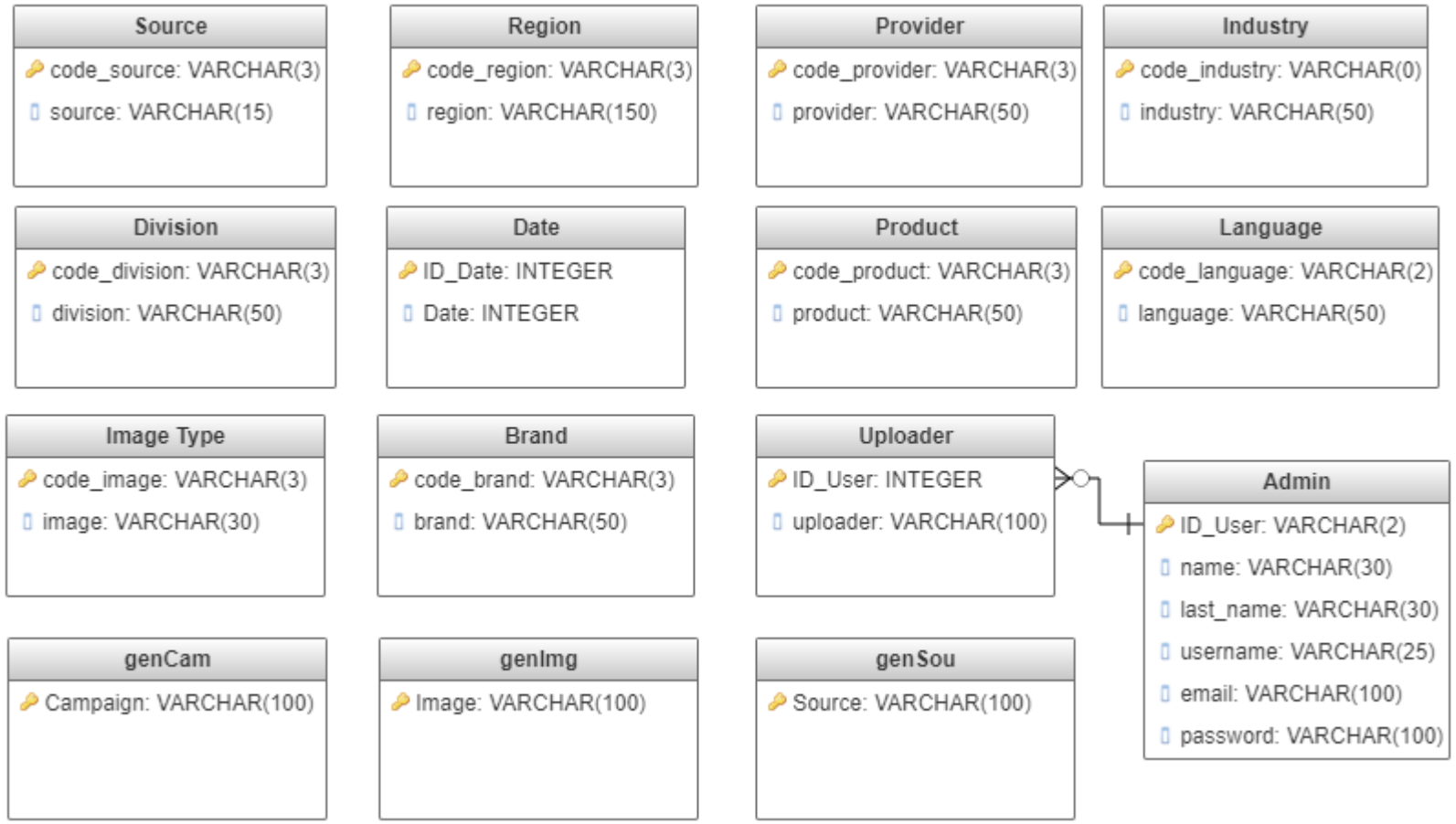
Database

Web framework

Back-end

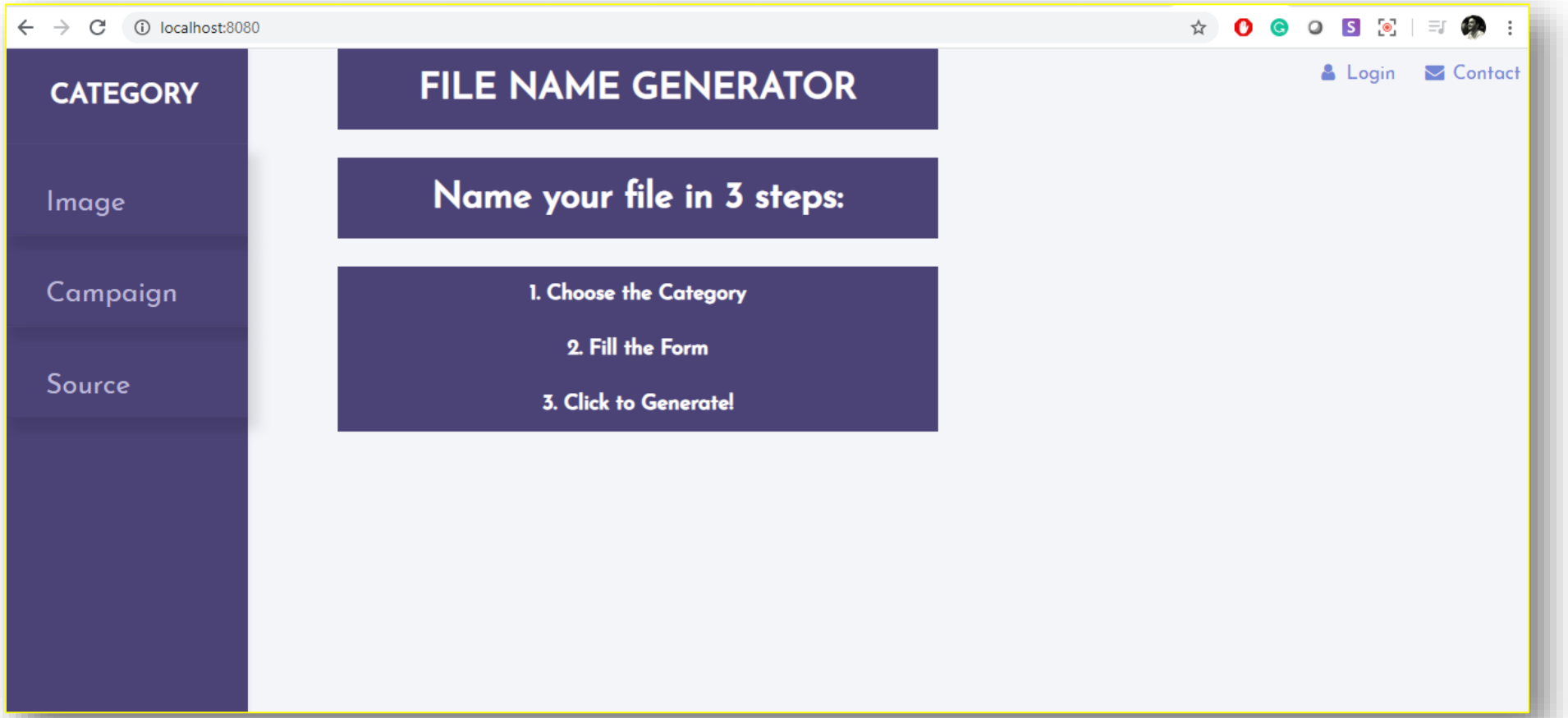
Front-end

DATABASE SCHEME



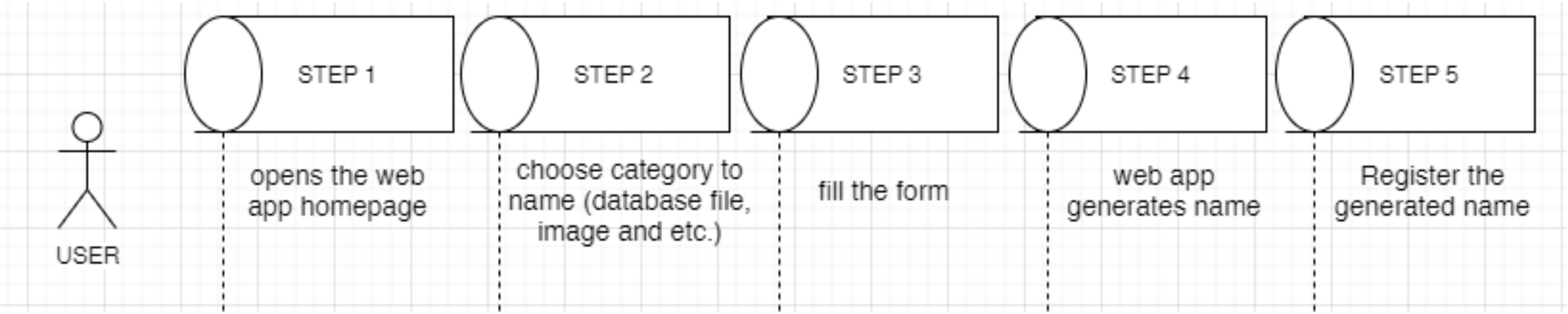
WEB APP

Main page



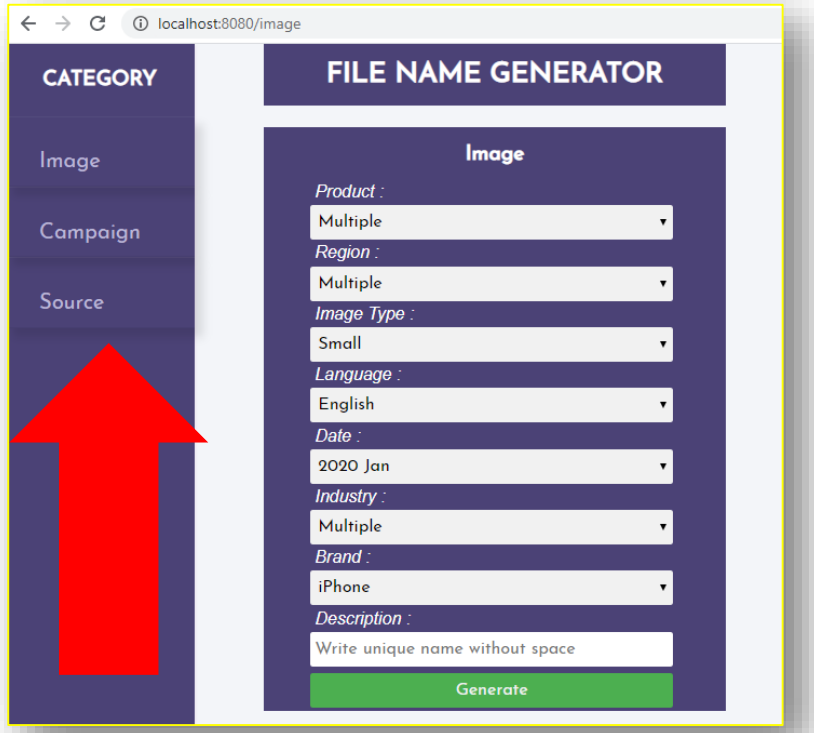
WEB APP

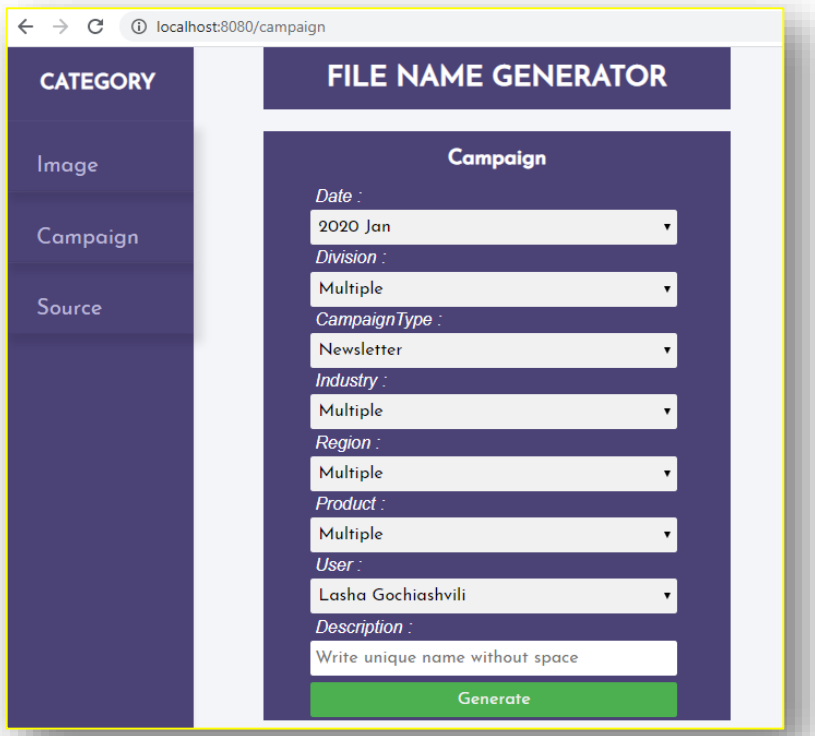
User’s path on our web app

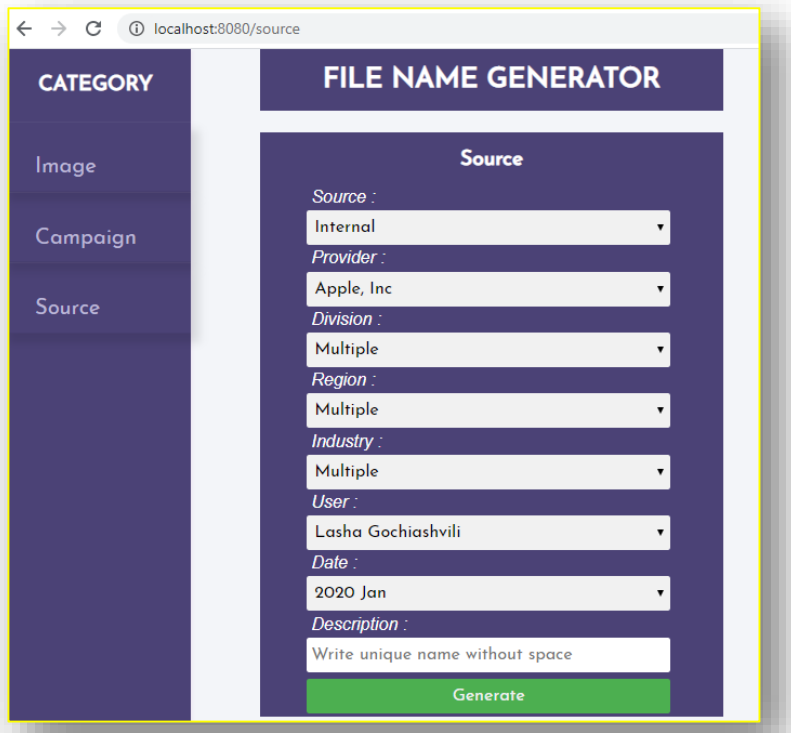


WEB APP

Categories



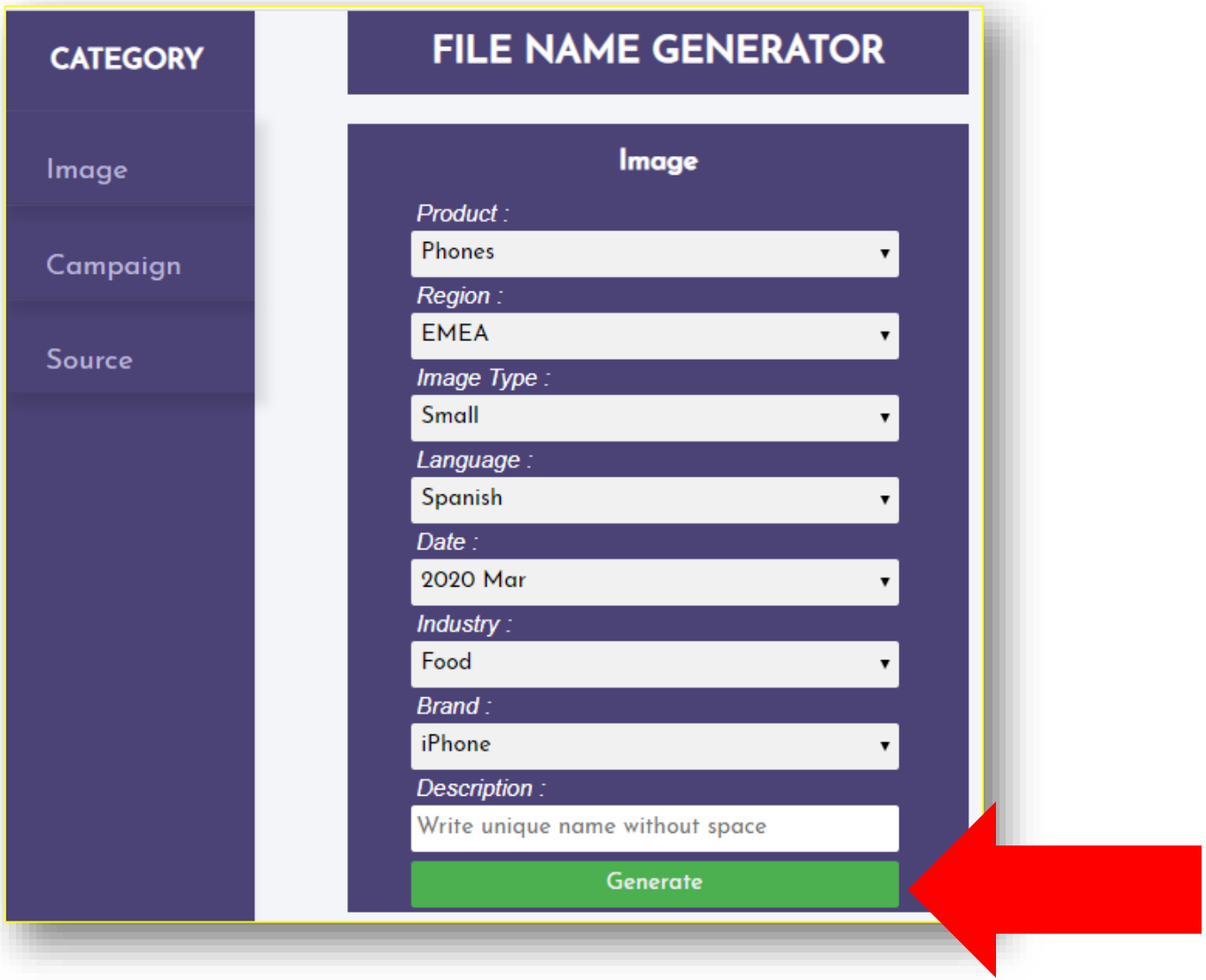




WEB APP

Fill the form
to name the “image”

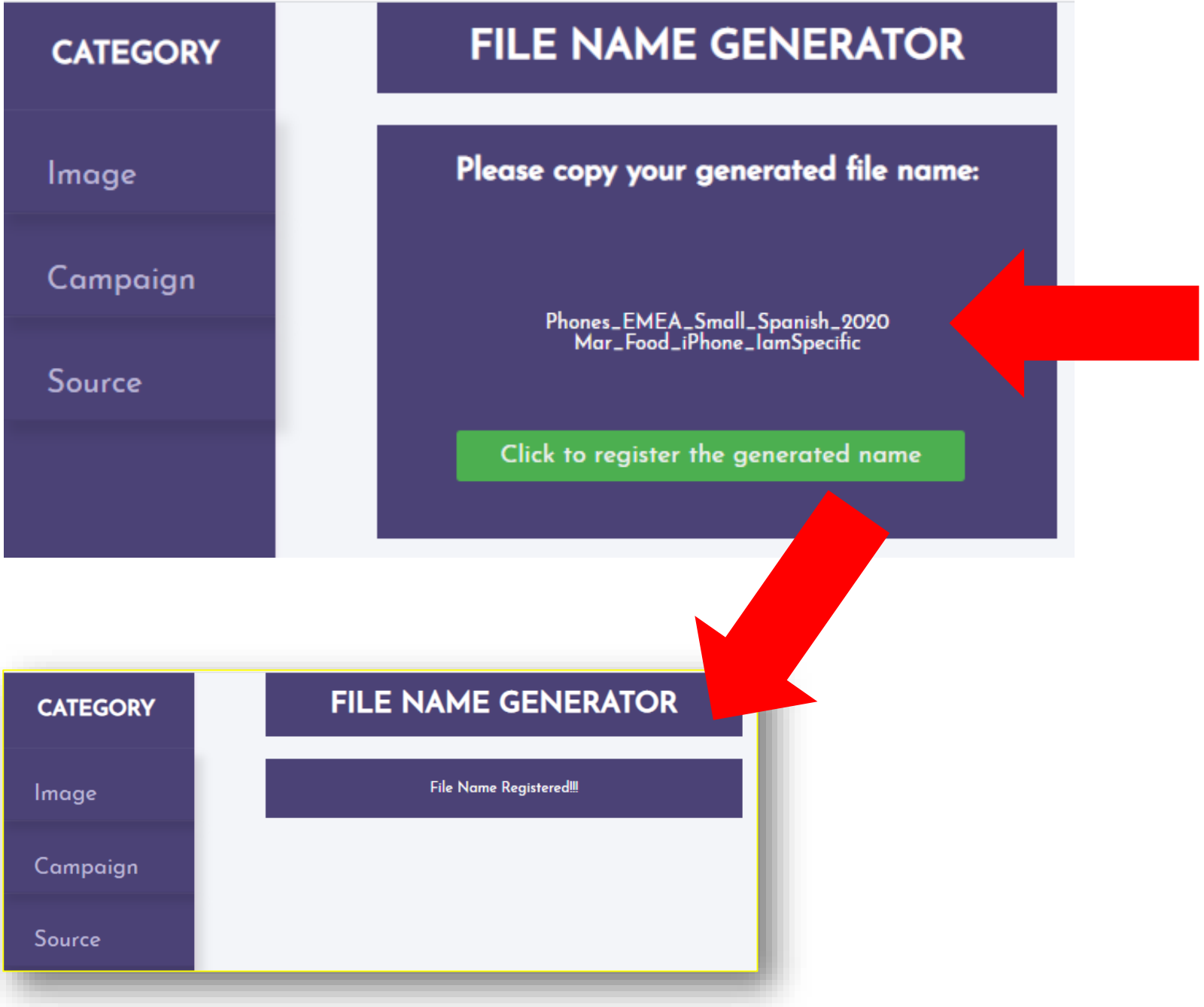
Click to “Generate”



WEB APP

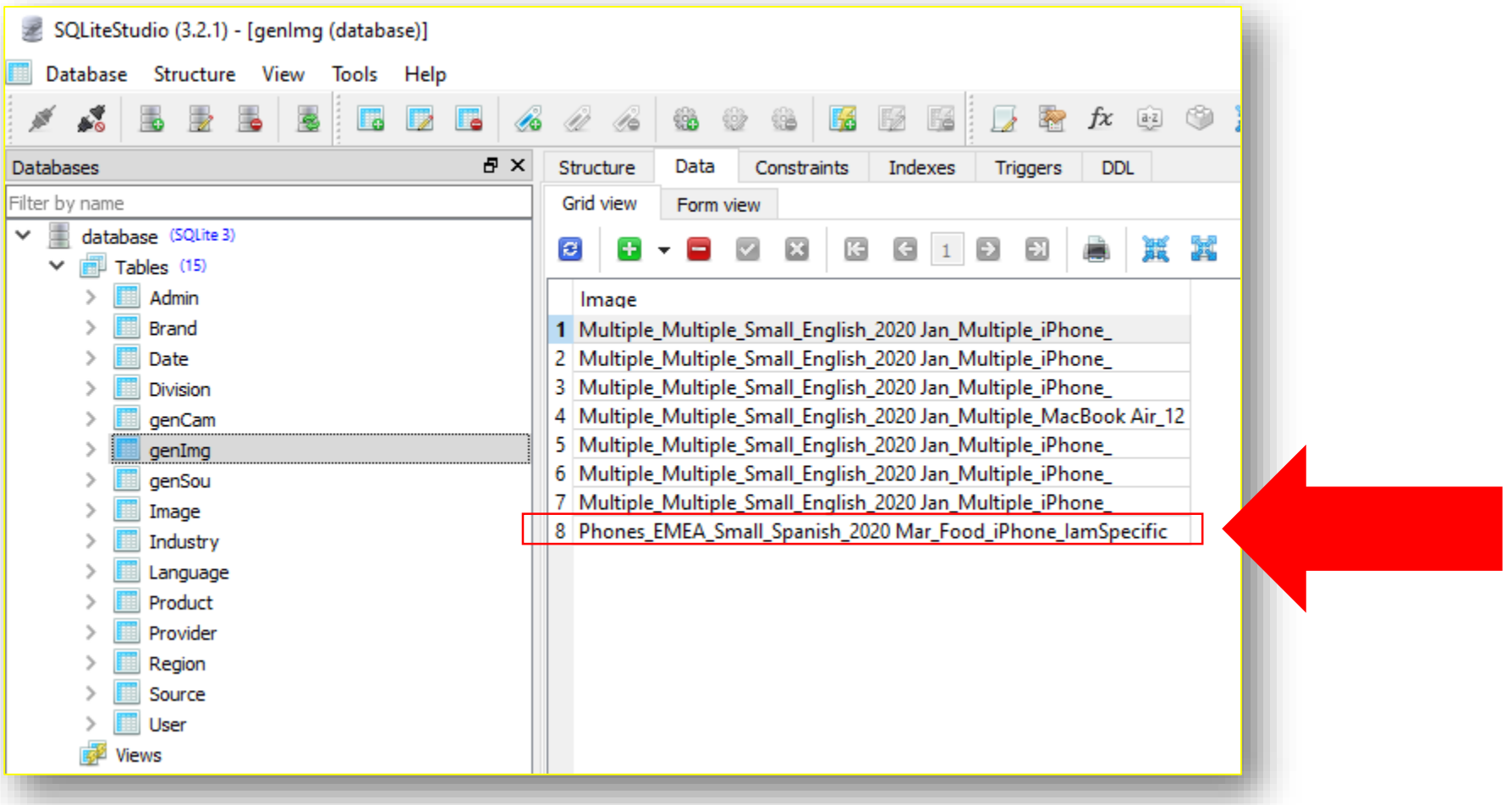
Name is generated!

Click to “Register”



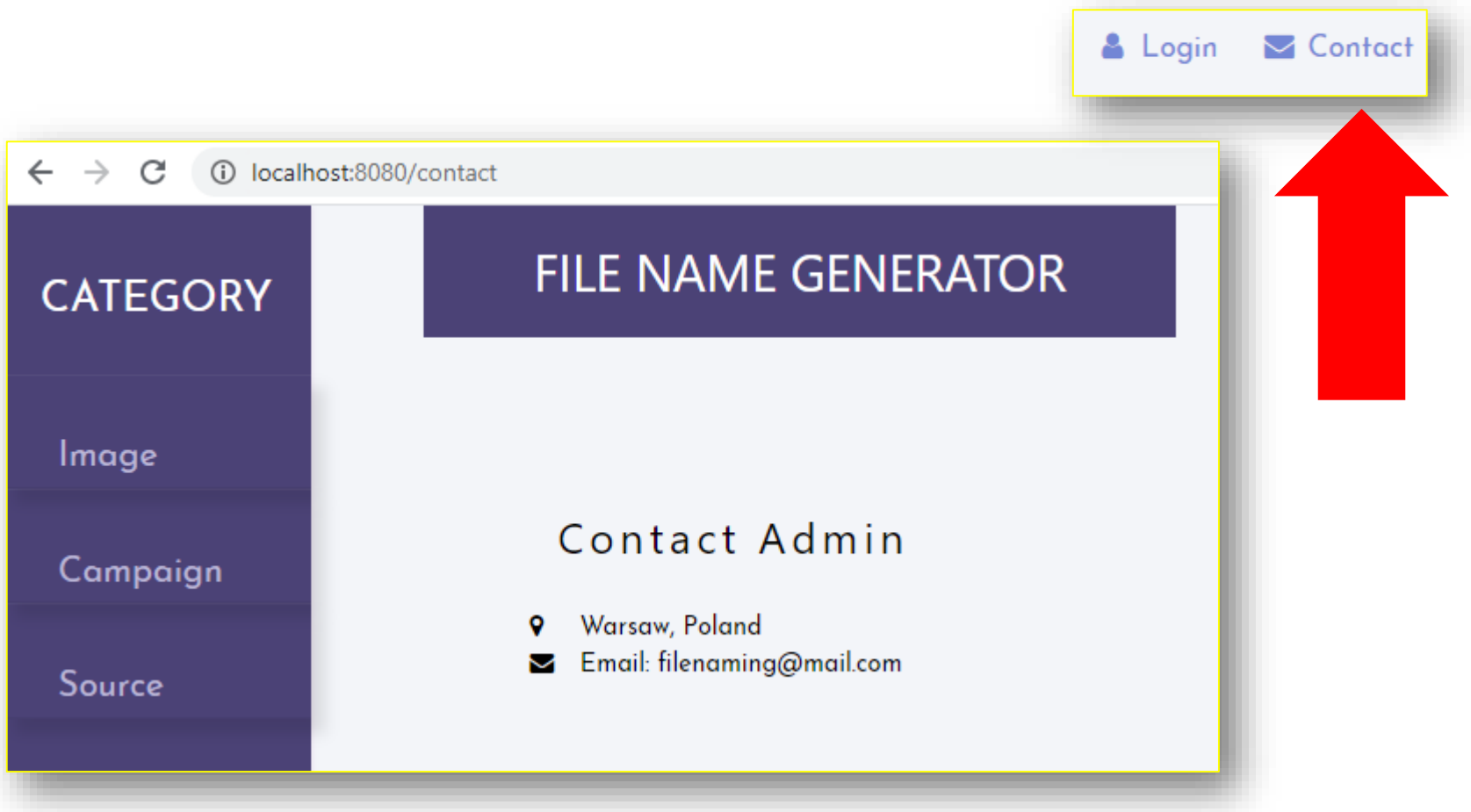
WEB APP

Name registered in database



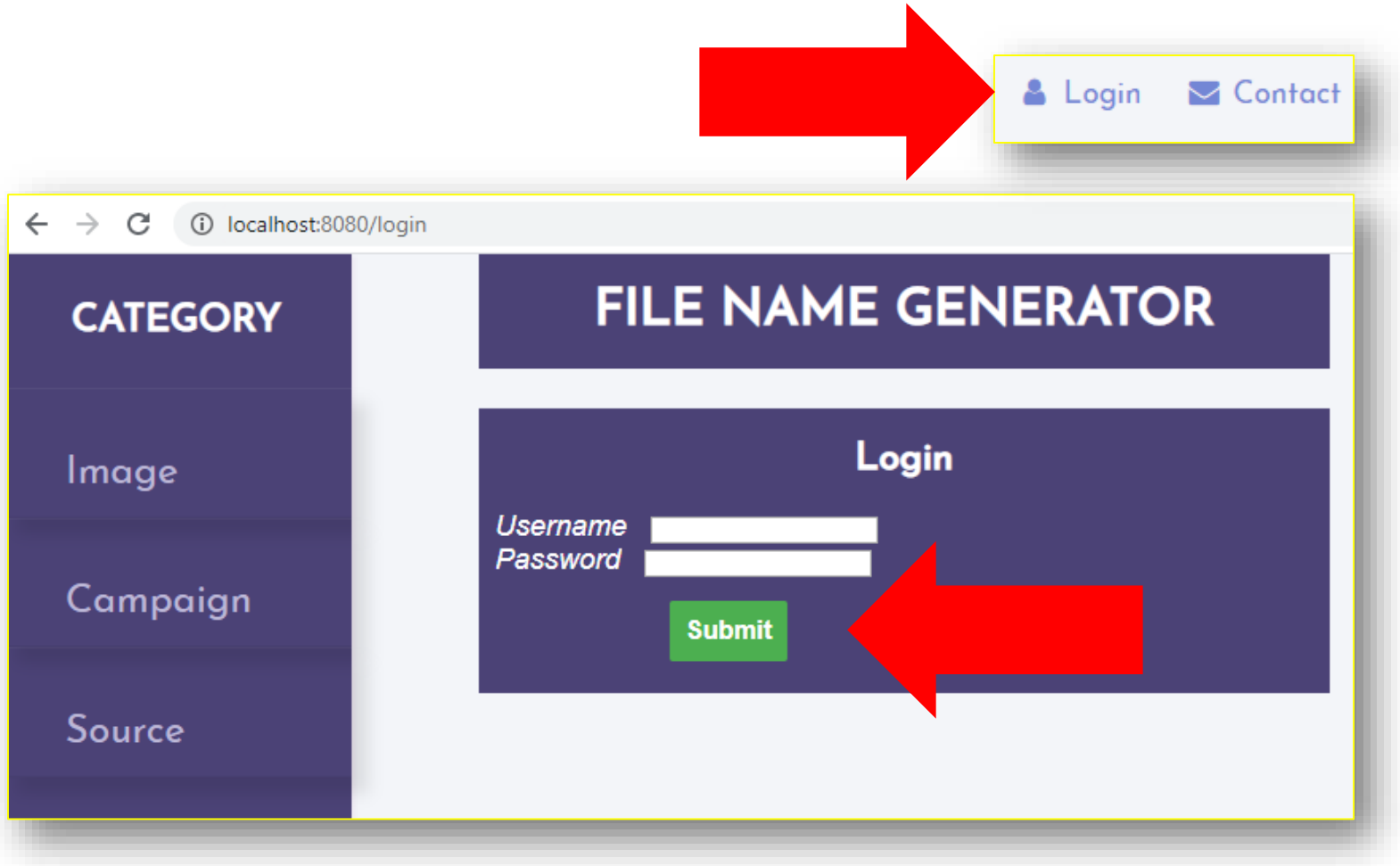
WEB APP

Name registered
in database



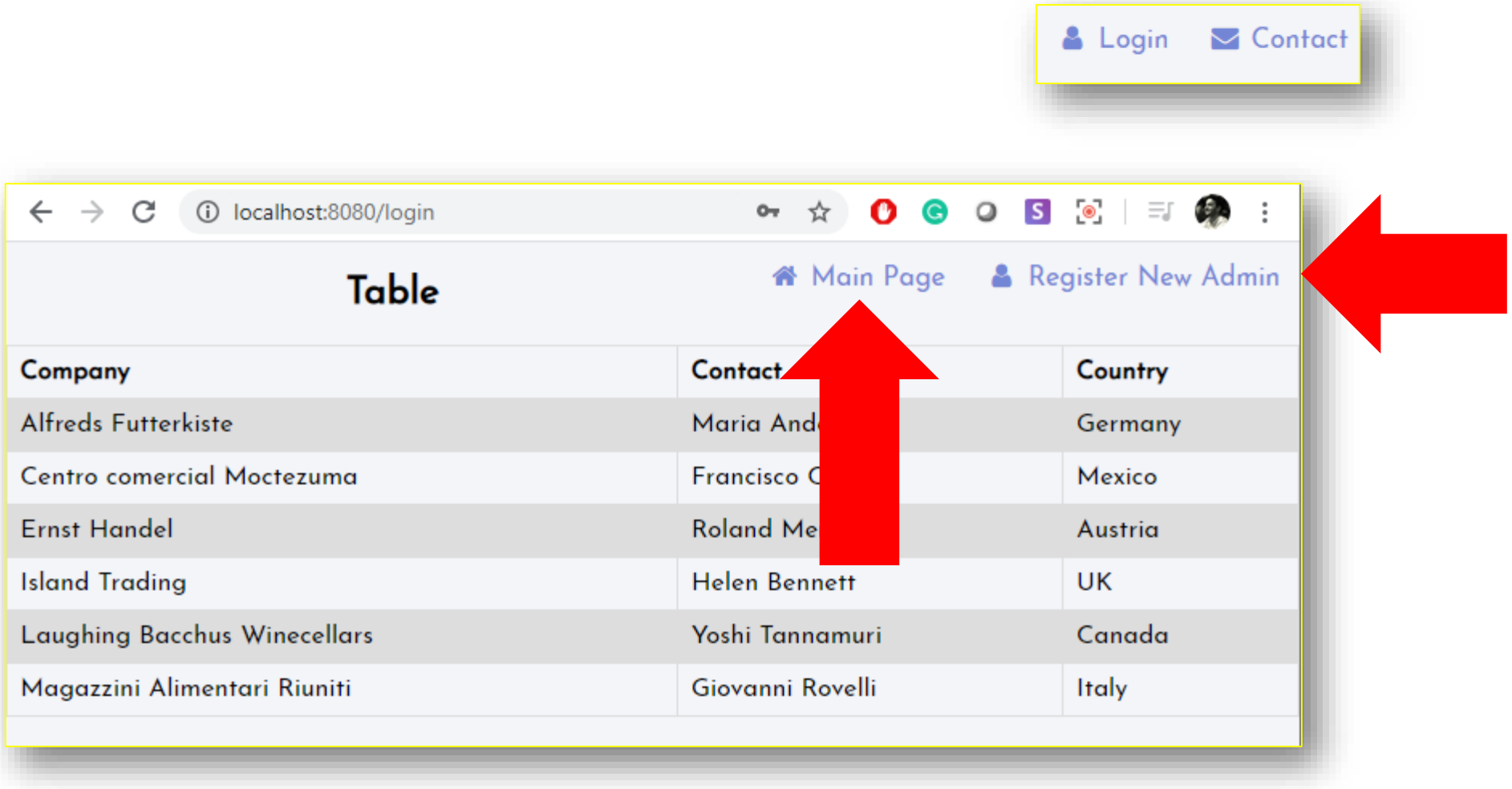
WEB APP

Login



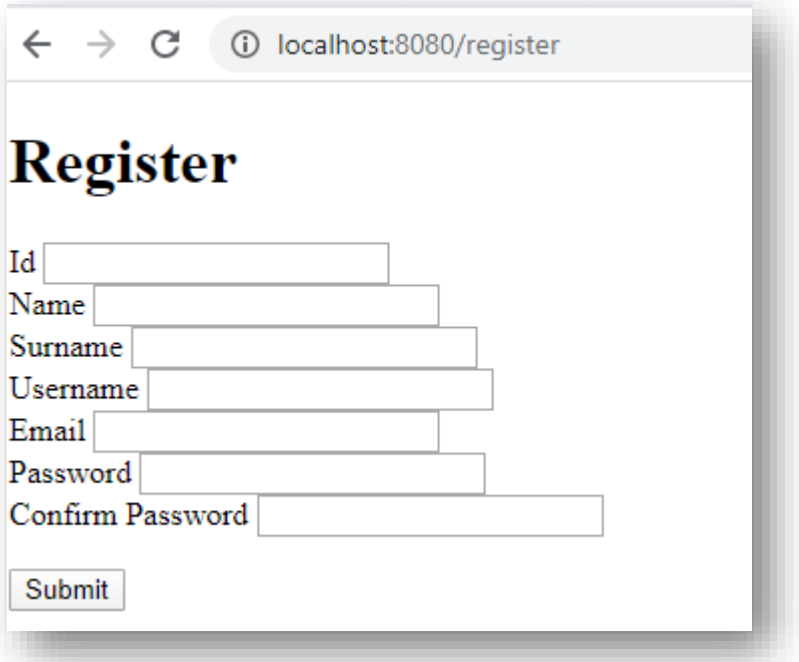
WEB APP

Login



WEB APP

Register new admin



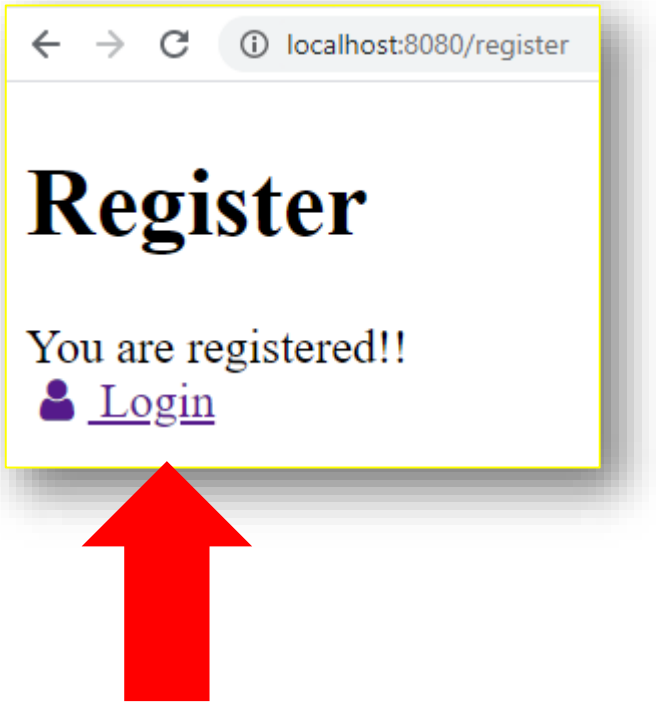
Source: http://www.quickmeme.com/meme/3r616q

WEB APP

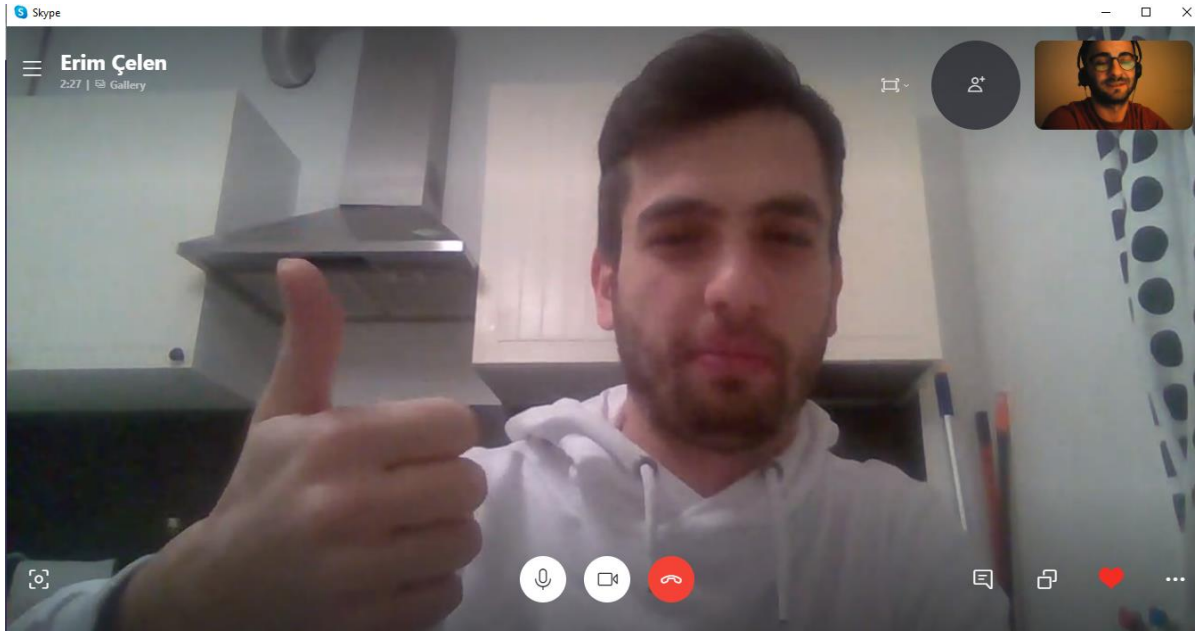
Register new admin

Mismatched passwords…

After registering correctly… ☺
you are redirected here:



HOW DID WE WORK ON IT?



HOW DID WE WORK ON IT?

#	ACTION	RESPONSIBLE
1	**Proposal**	
1.1	Brainstorming about ideas I	MEÇ
1.2	Brainstorming about ideas II	LG
1.3	Mockup of final idea I	MEÇ
1.4	Mockup of final idea II	LG
1.5	Work plan breakdown I	MEÇ
1.6	Work plan breakdown II	LG
1.7	Writing proposal I	MEÇ
1.8	Writing proposal II	LG
2	**Web app**	
2.1	Login	LG
2.2	Registration	MEÇ
2.3	Web interface - categories	MEÇ
2.4	Web interface - forms	MEÇ
2.5	Web app - connecting categories with forms	MEÇ
2.6	Web app - name generator	MEÇ
2.7	Web interface - save/register button for generated name	MEÇ
2.8	Create data for each table - in scheme and SQL	LG
2.9	Create data for each table - in python	MEÇ

#	ACTION	RESPONSIBLE
2	**Web app II**	
2.10	Routing in python I	MEÇ
2.11	Routing in python II	LG
2.12	HTML templates I	MEÇ
2.13	HTML templates II	LG
2.14	CSS I	MEÇ
2.15	CSS II	LG
2.16	Test web app	LG
2.17	Create environment to run web app	LG
3	**Report**	
3.1	Introduction	LG
3.2	Databases scheme with description	LG
3.3	User manual describing the use of application with screenshots	MEÇ
3.4	Final SQL queries	MEÇ
3.5	How work was distributed among team members	LG
4	**Presentation**	
4.1	Slides about introduction	LG
4.2	Slides about database scheme	LG
4.3	Demo of final app	MEÇ
4.4	Slide about how work was done	LG

CONCLUSIONS

1. Real problem with its solution identified
2. We learned lots of technologies while working on building the solution
3. Without good community and documentation it is not easy to use even “cool” technology. Due to this problem we switched “bottle” and moved entire web app to “flask”
4. Team work is a key
5. Lots of areas to develop for this web app. E.g. admin panel, better front end and etc.☺

REFERENCES

- Most of the knowledge we acquired from Stackoverflow.com & youtube.com
- http://flask.palletsprojects.com/en/1.1.x/
- https://www.youtube.com/watch?v=MwZwr5Tvyxo
- https://wiki.python.org/moin/WebFrameworks
- Python Flask From Scratch series: https://www.youtube.com/watch?v=zRwy8gtgJ1A
- https://jinja.palletsprojects.com/en/2.10.x/
- https://www.w3schools.com/w3css/w3css_examples.asp
- https://training.talkpython.fm/courses/explore_flask/building-data-driven-web-applications-in-python-with-flask-sqlalchemy-and-bootstrap
- https://flask-wtf.readthedocs.io/en/stable/install.html
- https://www.palletsprojects.com/p/flask/
- http://bottlepy.org/docs/dev/
- https://flask-wtf.readthedocs.io/en/stable/