

Project 1 Individual Assignment, IVIS_16

Emil Westin, Cmete_4

In order to create the best possible groups I entered the data into Excel to be able to easily handle it. In excel I then grouped some categories together and ignored others. The information I found relevant was the questions where the users evaluated their skills in different areas. I ignored the free text questions and the question about computer usage and user evaluation, since I find these topics not to be of main relevance in the course.

The questions about programming was calculated into a total programming average skill per person. IV, drawing and graphical programming was grouped and a total average graphic skill was calculated per person. Lastly, math and statistics were grouped and an average was calculated per person.

I then manually created the groups by first sorting the programmers and selecting them one by one into the 11 groups, starting and ending with group 1, ground round in a "loop". I this way, group 1 had the "best" and the "worst" programmer. Secondly and thirdly, I did the same with the graphical and math categories. I did it this way to try and get as homogenous groups as possible. There are 9 groups with seven members, and 2 groups with 6 members. Each group has at least 2 programmers, 2 graphics, and 2 math members. Below is a picture of the excel group distribution to better explain my reasoning:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
		Total average	Programming	Math average	Graphics				GROUP 1	GROUP 2	GROUP 3	GROUP 4	GROUP 5	GROUP 6	GROUP 7	GROUP 8	GROUP 9	GROUP 10	GROUP 11
2	1 Person 48	5.6	6.0	5.0	4.0			Programmers											
3	2 Person 30	7.2	8.7	4.5	6.7				48	30	46	49	59	61	44	25	6	36	65
4	3 Person 46	7.0	8.3	5.5	6.7			Graphics	7	24	70	1	4.5	40	19	82	7	39	42
5	4 Person 49	7.1	8.0	6.5	6.7														
6	5 Person 59	5.9	7.7	3.5	6.0			Math	27	60	47	63	9	34	23	16	32	54	53
7	6 Person 51	6.2	7.7	5.0	5.7				2	20	22	73	50	35	45	37	52	57	66
8	7 Person 44	6.9	7.0	5.0	7.3														
9	8 Person 25	6.0	7.0	7.0	4.7				41	39	10	51	3	26	15	74	75	21	14
10	9 Person 8	5.7	7.0	5.0	4.3			Average	28	67	11	17	6	68	64	31	5	58	13
11	10 Person 33	6.0	6.7	6.0	4.7				72	56	29	69	12	67	43	55	38		
12	10 Person 36	6.7	6.7	6.0	7.7														
13	11 Person 42	6.3	6.7	6.5	5.0				6.4	5.4	5.3	4.7	4.6	4.4	3.9	3.7	3.1	6.1	5.9
14	11 Person 65	7.0	6.7	5.0	7.8														
15	9 Person 4	6.1	6.7	5.0	4.7														
16	6 Person 40	5.9	6.3	5.0	5.7														
17	7 Person 19	6.3	6.3	5.0	6.0														
18	8 Person 62	6.8	6.3	6.0	6.3														
19	1 Person 41	6.1	6.0	8.0	3.3														
20	1 Person 7	6.7	6.0	8.5	4.7														
21	2 Person 24	5.9	6.0	5.5	5.0														
22	2 Person 39	4.9	6.0	3.5	4.0														
23	3 Person 70	6.0	6.0	3.5	5.8														
24	4 Person 1	6.1	6.0	6.0	5.3														
25	5 Person 71	6.0	6.0	4.5	6.3														
26	1 Person 2	4.9	5.7	4.5	4.3														
27	2 Person 60	7.0	5.7	6.0	6.3														
28	3 Person 10	5.7	5.7	6.0	4.0														
29	5 Person 9	6.2	5.7	6.0	5.8														
30	7 Person 23	6.1	5.7	4.5	5.7														
31	9 Person 32	6.2	5.7	6.0	5.3														
32	1 Person 27	6.3	5.3	6.0	6.7														
33	3 Person 47	6.1	5.3	5.5	6.0														
34	4 Person 51	4.0	5.3	3.0	2.7														
35	5 Person 3	5.3	5.3	4.5	4.3														
36	6 Person 26	4.8	5.3	4.5	4.0														
37	7 Person 15	4.4	5.3	4.5	2.7														
38	11 Person 53	5.4	5.0	5.5	5.8														
39	4 Person 63	6.1	5.0	5.5	6.0														
40	8 Person 74	4.9	5.0	5.5	3.3														
41	10 Person 21	4.2	4.7	3.5	4.3														
42	10 Person 57	5.4	4.7	7.0	5.0														
43	11 Person 13	5.1	4.7	4.5	3.7														
44	11 Person 14	5.4	4.7	3.5	4.3														
45	3 Person 22	5.1	4.7	3.0	4.7														
46	6 Person 34	5.6	4.7	5.0	5.7														
47	9 Person 75	4.1	4.7	3.5	4.0														
48	10 Person 54	5.7	4.3	6.0	5.3														
49	10 Person 58	5.6	4.3	6.5	3.7														
50	8 Person 31	4.9	4.3	6.0	2.7														
51	8 Person 34	2.6	7.4	5.4	2.4														

The data was then exported to JSON and I created an interactive visualization by using HTML and d3.js, although I don't think that I used d3.js the intended way. By hovering on the group "cards", a more detailed view of the group is presented on the side (See added files). I tried to follow the mantra of "Overview first, zoom and filter, then details-on-demand", but I couldn't implement the "zoom" feature, so I had to settle with the solution presented in the attached files.

The image to the right is a screenshot of the visualization. The mouse is hovering over Group 4 (although not visible in the screenshot - I don't know why).

