

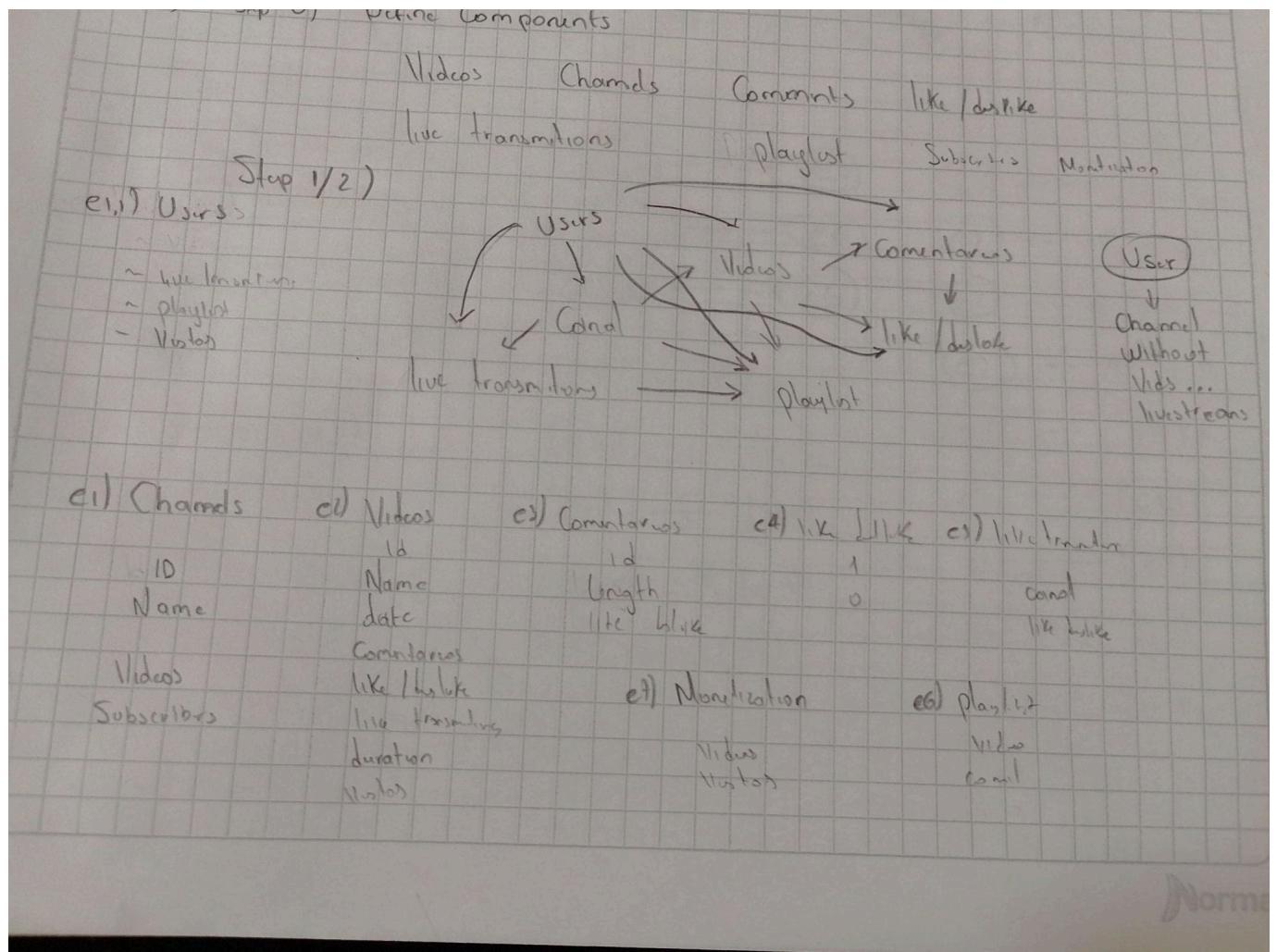
User story:

as an system engineer, I seek to make in a simple way how works the relation between the entities that are part of youtube database

Steps

Step 3)						
e1) Channels	e2) Videos	e3) Comments	e4) Like/Dislike	e5) LiveTransitions	e6) PlayList	e7) Merchandise
e1.) User						
e1)	✓	✓	✗	✗	✓	✗
e1.)	✓	✗	✓	✓	✓	✗
e2)	✓	✗	✓	✓	?	✓
e3)	✗	✓	✓	✓	✓	✗
e4)	✓	✓	✓	✓	✓	✗
e5)	✓	✗	✓	✓	✓	✓
e6)	✓	✓	✓	✗	✓	✗
e7)	✗	✗	✓	✗	✓	✗

Step 4)						
(e1) --- - (e1,1)	(e1,1) - - -< (e3)	(e2) —— (e)				
(e1) —— < (e2)	(e1,1) —— (e4)	(e2) -- -< (e3)				
(e1) — - - (e5)	(e1,1) — - -< (e6)	(e2) --- < (e4)				
(e1) — - -< (e6)	(e3) —> (e1,1)	(e2) --- < (e6)				
(e4) —— (e1,1)	(e3) --- - (e1)	(e2) --- - (e7)				
(e4) — - (e1)	(e2) —— (e2)	(e5) ---- (e1)				
(e4) — - - (e2)	(e3) — - < (e4)	(e5) --- - (e7)				
e4 — - - (e3)	(e3) —— (e6)	(e6) —~ (e1)				
(e4) — - - (e7)		(e6) — - - (e1,1)				



the others steps are in the img from draw io

Technical decision / consideration

I decided to mark a difference between user and channel;

User -> no vids, no live streams, just a viewer and a consumer of the content.

channel -> a user who makes vids, livestreams and also does the same things as a user.

challenges

I have had a problem trying to make the database draw look good, and easy to understand, mostly because of the connectors between entities.