No.	Description	Mark Allocation
Part 1: MongoDB / Data / API		
This is the only technical aspect of the marking, so just ensure that you tick all these boxes.		
a	Pulling data from MongoDB	1
b	Using the "mongodb" package	1
c	Data already entered (2 user profiles, 2 playlists, and 3 songs)	1
d	No dummy data	1
e	Express is used	1
f	Folder structure is correct (/backend and /frontend)	1
Part 2: Linking the Frontend and Backend		
*For this section, we want to see that you have implemented your API routes into your frontend. You can do the backend how you want, as long as it is using ExpressJS. How you actually do the requests is up to you, so we are going to look exclusively at the frontend functionality (**via the browser**) to see changes rather than looking at the code behind it.*		
g	No errors in browser console	1
h	Native Fetch API being used	1
i	Authentication API Request (Login, Signup, Logout)	3
j	Profile API Request (View, Edit, View someone elses, Delete your profile)	4
k	Friend / Unfriend API Request	2
l	Your Playlist API Requests (Create playlist, Add songs to playlist, View Playlist, Edit Playlist Details, Delete Playlist)	5
m	Others Playlists API Requests (View Playlist)	1
n	Songs API Requests (Create song, Delete song)	2
o	Searching API Requests (Playlist, Songs, Users)	3
p	Using existing components	1
Part 3: Basic TailwindCSS Styling		
q	No default Tailwind fonts	1
r	No default HTML fonts	1
s	No default bootstrap components	1
t	Implemented colours across entire website	2
u	Implemented theme (fonts / general layout) across entire website	2
Part 4: Questions		
Two random questions that we will ask on the day, just like how we have been doing it in the practicals.		
v	On the day question 1	2
w	On the day question 2	2
	Minus	0
	Bonus	0
	Total	**40**

If your application does not run using Docker, you will recieve 0 for this deliverable.

Always refer back to the D2 spec and the main project spec to ensure that you cover everything and ensure that everything works as intended