Information architecture

News Article

```
1  ∨ export type Votes = {
2        upVotes: number;
3        downVotes: number;
4     };
5
6  ∨ export type Comment = {
7        commentId: string;
8        userId: string;
9        text: string;
10       votes: Votes;
11       comments: Array<Comment>;
12    };
13
14 ∨ export type NewsArticle = {
15       id: number;
16       articleImageUrl: string;
17       articleUrl: string;
18       author: string;
19       content: string;
20       description: string;
21       newsOutletId: null | string;
22       newsOutletName: string;
23       publishedAtDate: Date;
24       title: string;
25       votes: Votes;
26       comments: Array<Comment>;
27    };
```

News Articles are defined by this type, and will be fetched in the Frontend like this.
Every Article also contains a Comment array, which can also contain another Comments
array. Thus the child comments can be easily determined by the structure of the data itself.
Up-Votes and Down-Votes are simply stored in every Article and Comment.

User

```
1 ∨ export type User={
2        id:string;
3        authenticationToken:string;
4        name:string;
5        mail:string;
6    }
7
```

Our initial User-Data will only contain the necessary:

- **Id** for comments
- **AuthenticationToken** to authorize the user activity
- **Name** as the username
- **Mail** for password recovery and login purposes

A **password** Attribute will be stored in the database for login purposes as well, which will not be retrieved in the frontend.