

EXPLORER

- CLIFFJUMPS
 - .git
 - cloudinary
 - controllers
 - models
 - node_modules
 - public
 - routes
 - seeds
 - utils
 - views
 - cliffjumps
 - code.ejs M
 - disc-edit.ejs
 - discussions.ejs
 - edit.ejs
 - index.ejs
 - new.ejs
 - photos-edit.ejs
 - replies.ejs
 - reply-edit.ejs
 - show.ejs M
 - videos-edit.ejs
 - layouts
 - partials
 - users
 - error.ejs
 - home.ejs
 - .env
 - fixDb.js
 - index.js
 - joiSchemas.js
 - LICENSE.txt
 - middleware.js M
 - package-lock.json
 - package.json

OUTLINE

TIMELINE

```

discussion.js  JS  reply.js  cliffjump.js  photos-edit.ejs  photoEditpage.js  middleware.js M
JS  middleware.js > [O] axios
1  const { JoiCliffjumpSchema, JoiDiscussionSchema, JoiReplySchema } = require('./joiSchemas.js');
2  const ExpressError = require('./utils/ExpressError');
3  const Cliffjump = require('./models/cliffjump');
4  const Discussion = require('./models/discussion');
5  const Reply = require('./models/reply');
6  const User = require('./models/user');
7  const axios = require('axios');
8
9
10 module.exports.isLoggedIn = (req, res, next) => {
11   if (!req.isAuthenticated()) {
12     req.session.returnTo = req.originalUrl
13     req.flash('error', 'you need to be logged to do that');
14     return res.redirect('/login');
15   }
16   next();
17 }
18
19 module.exports.validateCliffjump = (req, res, next) => {
20   const { error } = JoiCliffjumpSchema.validate(req.body);
21   if (error) {
22     const msg = error.details.map(el => el.message).join(',');
23     throw new ExpressError(msg, 400)
24   } else {
25     next();
26   }
27 }
28
29 module.exports.validateDiscussion = (req, res, next) => {
30   const { error } = JoiDiscussionSchema.validate(req.body);
31   if (error) {
32     const msg = error.details.map(el => el.message).join(',');
33     throw new ExpressError(msg, 400)
34   } else {
35     next();
36   }
37 }
38
39
40 module.exports.isAuthor = async (req, res, next) => {
41   const { id } = req.params
42   var Id_error = 0
43   var cliffjump
44   try {
45     cliffjump = await Cliffjump.findById(id);
46   }
47   catch (err) {
48     if (err.kind === 'ObjectId') Id_error = 1

```