

DO GRUPO	NOME DO GRUPO	INTEGRANTES	ATIVIDADE	OBSERVAÇÕES
1	André	André	Calendário	Layout e funcionamento ok.
2	Cristian e Alexander	Cristian e Alexander	look.controller.js	Gerar: não há necessidade de checar se o req.user está no banco. Se o usuário está fazendo uma request com um token válido, significa que ele já está cadastrado no sistema. Não há necessidade de criar um novo User com o req.user (e.g. user = User(req.user)), os campos podem ser acessados diretamente (e.g. req.user._id) Na função getall: não há necessidade de checar se o usuário existe, basta fazer a consulta no banco e filtrar pelo owner/parâmetro do sistema. Na função create: não há necessidade de utilizar o join_example. Exemplo de uso correto deve vir na documentação apropriada e não como resposta a uma requisição HTTP. Além disso, a resposta mal estruturada gera um erro 400 ao invés de 422. Gerar: O código não executa. Faz acesso a variáveis (findOne, createCalendar) que não foram declaradas ou importadas. Analisando o código é possível perceber que está levando em conta somente o dia do mês, mas é preciso levar em conta também o mês e o ano.
3	COVID-19	Gabriel Righi e Bruno Alves	wardrobe.controller.js	Revisar o modo como está recebendo a data. Ela virá como a quantidade de milissegundos decorridos desde 1 de janeiro de 1970 00:00:00 UTC (ver Data now()). Quando o look não é encontrado, deve retornar uma response (e.g. res.json) onde o look é settado como null, e não retornar null de cara (isso faz com que a requisição fique esperando uma resposta por tempo indeterminado). Layout ok.
4	Bruno	Bruno Gottlieb	calendar.controller.js	Compor looks: erro 401, falta o token de verificação na request da api. Em ComposeAndMount, não cancelar variáveis? deveria ser "acc.cancelar()" para passar o objeto atual inteiro no estado, não somente as variáveis. Deve ficar assim: https://github.com/COZUEH/next.js/blob/master/src/app/page.tsx#L10 Gerar: Não devem merge do response original no look, eles, por isso, o ahh, o ahh, e a look.metas já estão incorretos. Não há falta nenhuma levando os dados verem do look na request. Não tem problema nenhum criando o arquivo look.repo.js, mas as funções que foram criadas lá poderiam ser escritas direto dentro do look.controller.js.
5	Grupo da Goleadora	Frederico e William	Compor Look	Layout ok.
6	Backteam	Fernando Garcia e Felipe Mello	look.controller.js	Meus looks
7	TeamName is Undefined	Fernando Paim e Matheus Enloff	Meus looks	Layout ok.
8	DDVARGAS	Daniel de Vargas	Sobre	Gerar: 202 não é o código de resposta apropriado para uma request bem sucedida. Ver API Javascript e Códigos de resposta HTTP .
9	Orion Team	Matheus Orsini e Pedro Santos	user.controller.js	Visualizar Peça
10	Deborah Moreira da Silva e Lucas de Oliveira Dias	Deborah e Lucas de Oliveira Dias	Visualizar Peça	Aprentou erro na chamada da API do tempo/temperatura. (Request failed with status code 400) Layout: Tamanho longo das figuras, posição dos cones temerária deslocada.
11	Grupo de Risco	Alisson Machado e Eduardo Vedotto	Visualizar Peça	Meus looks
12	Matheus Dalmolin	Matheus Dalmolin	Menu Principal	Tudo certo
13	Grupo do João	João Gonçalves Machado	Meus looks	Gerar: não foi dado import na classe Look.
14	Daniel e Deivis	Daniel B. Selterius e Deivis	wardrobe.controller.js	Na função getLook: Revisar o modo como está recebendo a data. Ela virá como a quantidade de milissegundos decorridos desde 1 de janeiro de 1970 00:00:00 UTC (ver Data now()). Quando o look não é encontrado, deve retornar uma response (e.g. res.json) onde o look é settado como null, e não retornar null de cara (isso faz com que a requisição fique esperando uma resposta por tempo indeterminado). Na função setLook: Mesmo problema do recebimento da data. O acesso aos parâmetros da request estão incorretos (req.user.params.day?). O ID do look que deve ser salvo no banco vem do corpo (req.body) da requisição, e não como parâmetro. Na função create: Da forma que está, a criação de usuários não suporta desenhos sobrenomes (e.g. "Fulano da Silva" salvaria como "Fulano da Silva"). Melhor receber um objeto name que contém os campos firstName e lastName (e.g. name: { firstName: "Fulano", lastName: "da Silva"}). Além disso, é recomendável que validações sejam feitas com certa precisão o que de fato está tentando (evitar utilizar express "Missing information"). Na função login: Poderia ser usado o método user.generateAuthToken ao invés de gerar token do jwt sign. O código está exigindo que tanto o e-mail quanto o username estejam corretos (validUser && validEmail), mas basta que um deles esteja correto. O retorno da API deve seguir o padrão "name": { "_id": "5e646b2e89e6035ea89277", "first": "John", "last": "Doe", "username": "jdoe" } (Precisa corrigir o nome) É necessário corrigir o token de autenticação.
15	Zezed Camaro	Pedro Leonel e Tiago Chagas	calendar.controller.js	Layout ok, informação da temperatura/tempo estática (deverá ser implementado futuramente)
16	Quarentena de Legends	Luigi Perotti e João Pedro da Silva	user.controller.js	
17	Multitask	Emílio	Perfil	
18	Reger natives	Julia Accosta e Gabriel Gomes	Menu Principal	
19	Joggy e Jaga	Felipe Imhoff	Perfil	
20	Laurence	Laurence	Sobre	