Roteiro da Aula 3

BACKEND

1. Adicionar a rota no arquivo routes.ts

app.patch('/habits/:id/toggle', async (request) => {

const toggleHabitParams = z.object({

id: z.string().uuid()

})

const { id } = toggleHabitParams.parse(request.params)

// recupera o dia de hoje, sem hora, minuto e segundo

const today = dayjs().startOf('day').toDate()

// verifica se o dia já existe

let day = await prisma.day.findUnique({

where: {

date: today

}

})

// caso o dia não tenha sido criado ainda, pois apenas dias com hábitos

// completados, existem

if (!day) {

day = await prisma.day.create({

data: {

date: today

}

})

}

// procura para saber se o hábito já foi completado

const dayHabit = await prisma.dayHabit.findUnique({

where: {

day\_id\_habit\_id: {

day\_id: day.id,

habit\_id: id

}

}

})

// se já foi completado

if (dayHabit) { // remove marcação

await prisma.dayHabit.delete({

where: {

id: dayHabit.id

}

})

} else { //não foi completada, então cria

await prisma.dayHabit.create({

data: {

day\_id: day.id,

habit\_id: id

}

})

}

})

1. Adicionar a rota no arquivo routes.ts

// [{day\_id, date, completed, ammount},

// {day\_id, date, completed, ammount},

// {day\_id, date, completed, ammount} ]

app.get('/summary', async () => {

const summary = await prisma.$queryRaw`

SELECT

D.id,

D.date,

(

SELECT

cast(count(\*) as float)

FROM day\_habits DH

WHERE DH.day\_id = D.id

) as completed,

(

SELECT

cast(count(\*) as float)

FROM habit\_week\_days HDW

JOIN habits H

ON H.id = HDW.habit\_id

WHERE

HDW.week\_day = cast(strftime('%w', D.date/1000.0, 'unixepoch') as int)

AND H.created\_at <= D.date

) as amount

FROM days D

`

return summary

})