```
import { router, userProcedure } from '@/app/api/trpc/trpc-router';

import { TRPCError } from '@trpc/server';


const dashboardRouter = router({
  getUserRequestCount: userProcedure.query(async ({ ctx }) => {
    const user = ctx.session.data.session?.user;
    if (!user) {
      throw new TRPCError({
        code: 'UNAUTHORIZED',
        message: 'Unauthorized',
      });
    }


    const userRequestCount = await ctx.supabase
      .from('swarms_cloud_api_activities')
      .select('request_count')
      .eq('user_id', user.id)
      .is('organization_id', null);


    if (userRequestCount.error) {
      console.error(
        'Error fetching user request count:',
        userRequestCount.error.details,
      );
      return 0;
    }
```

```
  const orgRequestCount = await ctx.supabase

    .from('swarms_cloud_api_activities')

    .select(

      `request_count, organization_id, swarms_cloud_organizations!inner(owner_user_id)`,

    )

    .not('organization_id', 'is', null)

    .eq('swarms_cloud_organizations.owner_user_id', user.id);


  if (orgRequestCount.error) {

    console.error(

      'Error fetching org request count:',

      orgRequestCount.error.details,

    );

    return 0;

  }


  // TODO: GET NUMBER OF MODELS USED BY USER


  const allCounts = [...userRequestCount.data, ...orgRequestCount.data];

  const totalCount = allCounts.reduce((acc, curr) => {

    return acc + (curr.request_count || 0);

  }, 0);


  return totalCount;

}),
```

```
});
```

```
export default dashboardRouter;
```