

```

import { Router } from 'express';
import { StatusCodes } from 'http-status-codes';
import authMiddlewares from "../../middlewares/auth/index.js";
import * as Ably from "ably/promises.js";
import ablyService from '../../ably/ably-service/index.js';

const router = Router();

router.get("/token-request", async (_req, res) => {
  try {
    const client = new Ably.Realtime({
      key: process.env.VERCEL_ENV === "production" ?
process.env.ABLY_API_KEY_PRODUCTION :
process.env.ABLY_API_KEY_DEVELOPMENT,
      log: { level: 4 },
    });

    const tokenRequestData = await
client.auth.createTokenRequest({ clientId:
process.env.VERCEL_ENV === "production" ?
process.env.ABLY_CLIENT_ID_PRODUCTION :
process.env.ABLY_CLIENT_ID_DEVELOPMENT });

    return
res.status(StatusCodes.OK).json(tokenRequestData);

  } catch (err) {
    return
res.status(StatusCodes.INTERNAL_SERVER_ERROR).send({ err:
(err.message || err) });
  }
});

router.get("/initialize", async (_req, res) => {
  try {
    await ablyService.initialize();

    return res.status(StatusCodes.OK).send("Initialized
ably");

  } catch (err) {
    return
res.status(StatusCodes.INTERNAL_SERVER_ERROR).send({ err:
(err.message || err) });
  }
});

```

```
export default router;
```