ChillBox

from Pausetta.org

for KONVERTO LAB

Our Team - Pausetta.org

DB creation & management

SongRank Algorithm
Home Page

Backend (API Endpoints)
Gathering song data

Frontend (user view & host view)

ChillBox

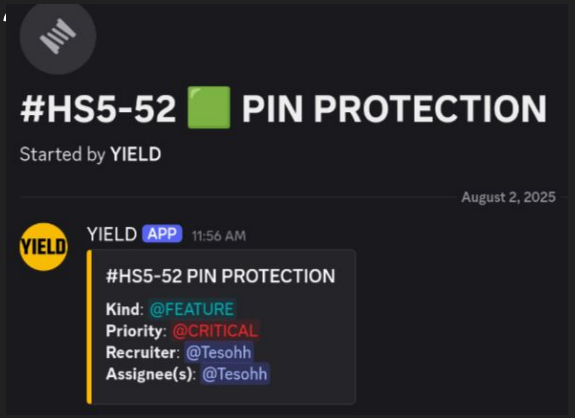
A **fast and easy-to-use** web app that lets users listen to music together by creating **shared listening "rooms"** based on their physical location.

Our goals are

- create a **seamless social music experience**
- **automatically** choose the **best tracks** for the moment
- encourage **positive participation** of users
- hands off approach for the operator

How did you solve the challenges

- First plan & then act
- Fast & Efficient communication
- Time & Task Management with a **Custom Discord bot**




```
    // Setup websocket connection
    socket = io("/", { path: "/ws", transports: ["websocket"] })
    await socket.emitWithAck("join_room", { id: data.roomId })

    socket.on("queue_update", async (d) => {
        const songs = await Promise.all(d.queue.map(parseSong))
        queueSongs = songs
        playingIndex = d.index
    })

    socket.on("new_vote", async (d) => {
        const updated = await parseSuggestion(d.song)
        suggestions = suggestions.map((s) => (s.uuid === updated.uuid ? updated : s))
    })

    socket.on("new_song", async (d) => {
        const song = await parseSuggestion(d.song)
        suggestions = [...suggestions, song]
    })
})
onDestroy(() => {
    if (socket) socket.disconnect()
})
```

- Very important to take breaks to relax & think
- Testing features before merging
- Having fun in the process

Demo prototype/solution