# CSAP Project

AA 2024-2025

# Project

- BYOS (Build Your Own Spotify[tm])
- A music server, keeping audio (i.e. mp3) files on its local storage, allowing clients to play and rate songs.
- Administrator(s):
  - Create new users and give them credentials to access the server
  - Manage songs:
    - Add, delete, rename music files, change tags ...
- User(s):
  - Login, using the credentials provided by Administrator
  - List, download (play) and rate songs

# Project

- Songs:
  - Are stored on server filesytem
  - Have their own tags (i.e. id3) as Name, Author, Album, Year, ...
- Server(s):
  - One or more concurrent processes
  - Index songs, keeping their metadata in memory for speed
  - Allow multiple concurrent clients to:
    - Connect, on port defined in a configuration file
    - List, search, download (play) and rate available songs
    - Maintains statistics (i.e download counter) and user ratings **for each song**

# Project

- Clients:
  - Connect to server (on a known port) and authenticate
  - Interact with server, list songs, search for tags, download and play individual songs
  - Rate songs

# Notes

- Don’t reinvent the wheel
  - Use available tools/code when possible, e.g to manage tags on mp3 files or play audio
- Server must:
  - Allow for multiple **concurrent** users (including Admin)
  - Be able to recover after a failure, e.g. scanning and rebuilding data structures
  - Check for storage space before uploading new songs
  - Not leave stale IPC resources upon exit
- Client should:
  - Handle (unexpected) network failures
  - Use temporary storage for buffering incoming data
  - Cleanup stuff before exiting

# Notes

- Documentation should include:
  - Functional and Non-Functional design choices
  - High level design
    - Data structures
    - Main functions and their parameters
  - Known problems
  - Test cases
  - Instructions for building all modules
  - Etc.