

# ChimeShare

Huțanu Andrei Leontin

Colegiul Național de Informatică „Tudor Vianu” - Clasa a X-a

Proiect realizat sub îndrumarea:

- Prof. Carmen Nicoleta Mincă
- Prof. Simona Ionescu
- Prof. Rădulescu Magdalena

# Cuprins:

## 1. O scurtă descriere

- 1.1. Aspectul social al proiectul
- 1.2. Istoria lui

## 2. Prezentarea proiectului

- 2.1. Designul și structura
- 2.2. Cum funcționează în fundal
- 2.3. Planuri de viitor

## 3. Tehnologii utilizate

- 3.1. Frontend
- 3.2. Backend
- 3.3. Externe
- 3.4. Cerințe de sistem

**1.**

**0 scurtă descriere**

# 1. 0 scurtă descriere

**ChimeShare este un proiect web online care încearcă să facă muzica accesibilă pentru toată lumea, cu un twist social.**

Cu toții cunoaștem acele idei care apar aleatoriu în timpul zilei și care par incredibile, dar pe care nu reușim să le salvăm la timp. Cu acest site, împărtășirea și scrierea propriilor tale melodii scurte devine mai ușoară ca niciodată. Toate necesitățile de bază pentru crearea de melodii simple sunt aduse aici, pe aceeași platformă, pentru a elimina stresul de a nu putea găsi un workflow confortabil pentru crearea muzicii tale.

Această platformă permite crearea de melodii scurte, de 32 de note lungime, acoperind 2 octave.

## 1.1. Aspectul Social

Acest proiect este realizat și cu aspectul social în minte. Pagina principală este concepută special pentru a permite descoperirea celor mai noi și populare melodii create de alți utilizatori. Având acest lucru în vedere, am creat o modalitate prin care oamenii să se conecteze între ei, permițând crearea de conturi pe site. Acest lucru ne oferă, de asemenea, posibilitatea de a urmări creatorii preferați pe platformă.

Fiecare melodie creată are opțiunea de a fi afișată publicului, unde poate aduna aprecieri de la alți utilizatori, aducând posibilitatea de a deveni următorul mare „Chime-Maker”!

## 1.2. Istoria lui

În prezent, proiectul este bazat pe **framework-ul PHP Yii** și **biblioteca Tone.js**, pentru sunete.

Acest site a fost scris inițial doar în **HTML, CSS, JavaScript** și puțin **PHP**, asta fiind ceea ce știam, dar pe măsură ce timpul a trecut și au fost adăugate funcții noi, am realizat că acestea nu vor fi suficiente.

Pasul logic următor a fost portarea proiectului pe un **micro-framework PHP, Limonade**, care se ocupa de rutare și view-urile site-ului. Odată cu această schimbare, am profitat de ocazie pentru a îmbunătăți partea de sunet a proiectului. Înainte, fiecare sunet era un fișier mp3 care era redat, ceea ce însemna că trebuia să creez un fișier mp3 pentru fiecare dintre cele 24 de note, pentru fiecare instrument pe care intenționez să îl adaug. Vă puteți imagina că acest lucru era foarte plictisitor și nici nu performa grozav.

Noua implementare se baza pe **librăria audiosynth**, de **Keithwhor**, dar am descoperit curând că nici aceasta nu putea reda multiple note simultan. Faptul că **JavaScript** este un limbaj single-threaded a făcut ca implementarea polifoniei să fie foarte dificil de realizat de unul singur, așa că am știut că găsirea unei noi biblioteci era necesară.

## 1.2. Istoria lui

Aici intervine **Tone.js**. Această bibliotecă rezolvă problema polifoniei notelor și poate, de asemenea, să genereze sunete de instrumente pe loc. Acest lucru înseamnă că sunetele sunt create pe baza diferitelor variabile pe care le pot schimba ușor, fără a fi nevoie să refac vreun fișier eu însumi. Acest lucru înseamnă că în viitor pot adăuga o funcție care permite utilizatorilor să-și creeze propriile instrumente sau să facă sample-uri la diferite sunete de ale lor.

Pentru ca această schimbare să fie posibilă, am rescris, pentru a treia oară, întregul backend pentru pagina **Chime Maker**, schimbând modul în care notele sunt stocate în fundal pe mașina utilizatorului și ulterior în baza de date. Cu acest nou design de structură, sunt, de asemenea, capabil să stochez diferite lungimi ale notelor, permițând posibilități nelimitate! (teoretic 2 la puterea 3072, dar cine stă să numere :) E încă mult )

Odată cu trecerea la **Tone.js**, am decis să portez proiectul și pe un **framework** mai puternic, **Yii**. **Yii** aduce foarte multe îmbunătățiri la partea de securitate, dar implementarea lui a însemnat o rescriere a întregului site și a modului în care funcționează. Pe parcursul acestui proiect, am învățat foarte multe lucruri noi.

**2.**

# **Prezentarea proiectului**

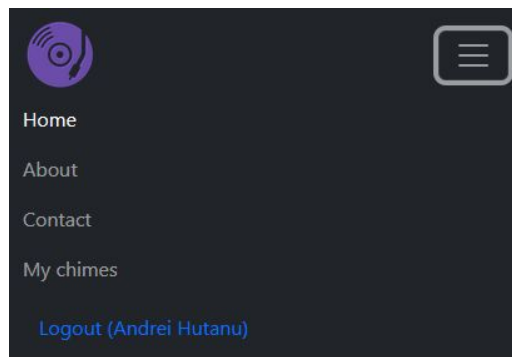


## 2.1. Designul

Designul este unul **simplic, modern și minimalist**, creat atât din clase utilitare, cât și cu ajutorul claselor oferite de **Bootstrap**.

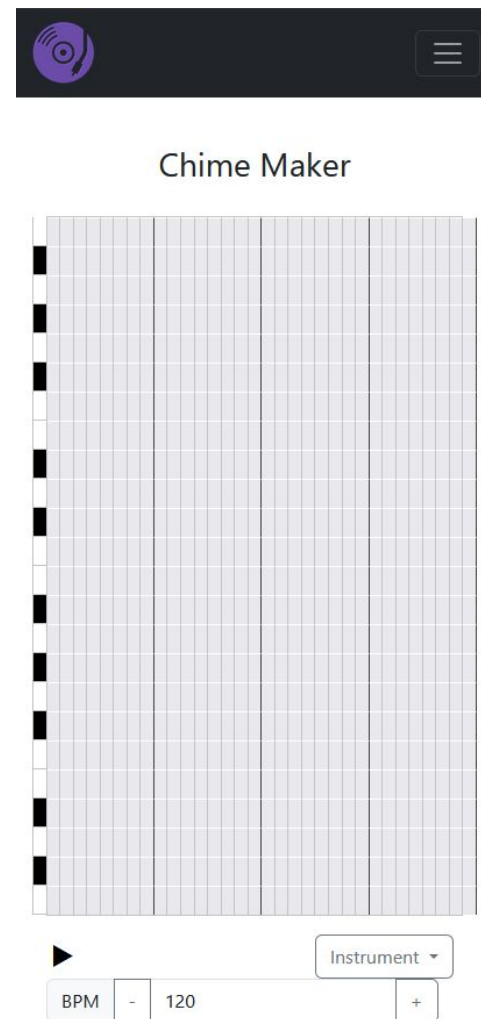
Am avut grijă ca aplicația să se **adapteze** corect atât unui ecran cu lățime mare (desktop/laptop), cât și pe un ecran de mărime medie (ca o tabletă). Pentru ecranele mai mici, asemenea celor găsite pe telefoane, interfața se adaptează și reușește să își pastreze în mare parte aceleași capabilități.

Probleme pot apărea însă atunci când se dorește crearea unei melodii de pe telefon, interfața având de suferit datorită **limitărilor impuse** de designul ales și de modalitatea de control pusă.



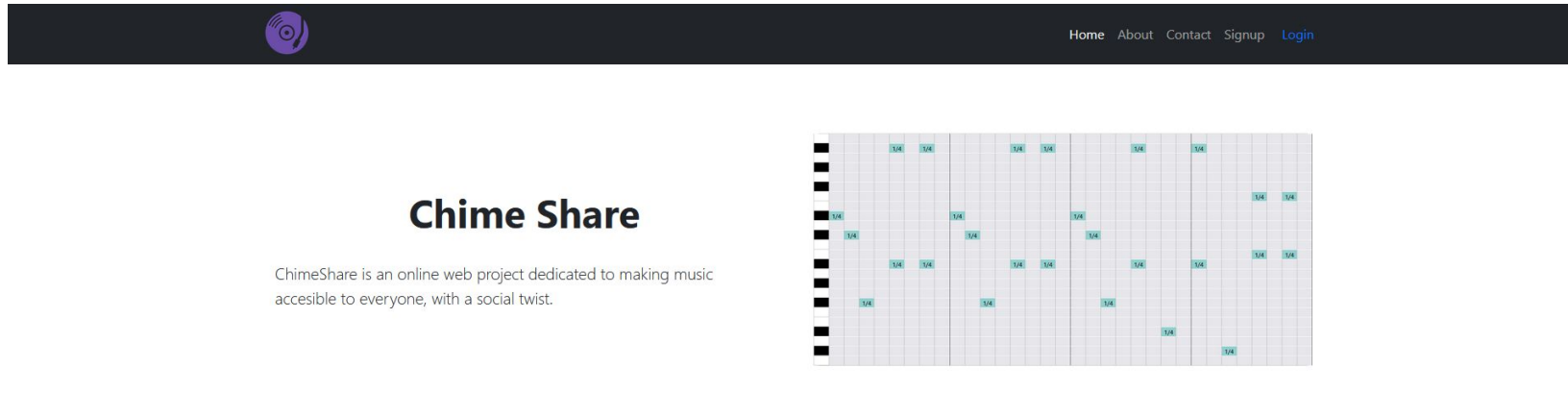
### Chime Share

ChimeShare is an online web project dedicated to making music accesible to everyone, with a social twist.



## 2.1. Structura

Structura pentru un utilizator fără cont (guest) este un simplă, asemenea majorității site-urilor de pe internet.



Utilizatorul are acces la prima pagina, de unde poate asculta ultimele melodii urcate, sau melodiile cele mai apreciate de comunitate, dar și la paginile 'About' și 'Contact'.

## 2.1. Structura

Exemplu pentru pagină de contact.


### Contact

For any questions regarding this project, please contact us using by filling the form below. Thank you.

ⓘ

Name cannot be blank.

Body

 **Spring**  
boot

Verification Code

Submit

## 2.1. Structura

După ce utilizatorul își face cont, el primește dreptul de a:

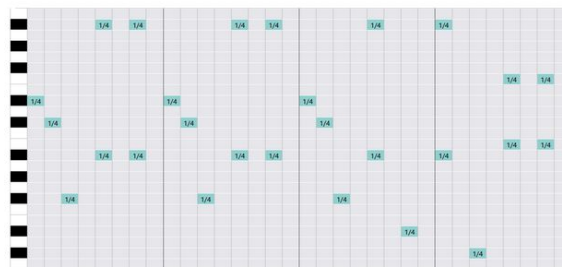
- Își arăta aprecierea pentru melodia altui utilizator
- Își face propriile melodii

Așa arată noua lui bară de navigație:



### Chime Share

ChimeShare is an online web project dedicated to making music accesible to everyone, with a social twist.



## 2.1. Structura

Pentru a își crea propria melodie utilizatorul trebuie sa ajungă la pagina 'My Chimes' de unde va apăsa butonul 'Create Chime', care îl va duce pe aceasta pagina:

Aici utilizatorul poate crea propriile melodii, el putând să le asculte la diferite instrumente și tempo-uri, în funcție de preferințe. El poate alege de asemenea să salveze melodia după terminarea acesteia, moment în care are de completat un formular.



Utilizatorul este liber să aleagă dacă își dorește ca melodia lui să fie publică, practic dacă apare și pe prima pagina, la secțiunea 'Latest Chimes', sau sa o pastreze privată.

## 2.2. Cum funcționează în fundal?

Gridul în care utilizatorul își face muzica este de fapt **generat pe loc** cu ajutorul **php-ului**, în funcție de anumite variabile decise de mine (lucru care îmi permite pe viitor să extind tot proiectul, sau poate chiar să îl fac compatibil cu mai multe instrumente în același timp). Fiecăruia din aceste butoane îi este dat un id unic, care urmează regula: "**c\_r\_**", unde 'c' vine de la 'column/coloana', iar 'r' de la 'row/rând'. Fiecare buton are de asemenea o **legătură 'onclick'** la o funcție **JavaScript** care sparge id-ul în bucăți pentru a recunoaște exact ce nota are nevoie să cânte (în funcție de randul în care se afla), iar care mai apoi generează un obiect care ține toate informațiile melodiei, a notelor cantate. Astfel, acest obiect poate fi foarte ușor salvat **sub forma JSON**, pentru a fi salvat eficient într-o bază de date.

Fiecărei melodii îi este atribuită, pe langa lucrurile de bază, precum un **nume și un owner**, și un **id public, diferit fata de cel din baza de date**. Acest lucru va permite deschiderea pentru ascultarea melodiilor altor persoane pe baza unui simplu link, care poate fi distribuit ușor și care va conține acel **id public** în structura lui.

În plus, fiecare melodie publică are de asemenea și un contor de **like-uri**, lucru care permite formarea acelei liste cu cele mai bune melodii pe care o găsim pe prima pagină și care aduce elemente asemănătoare unui **'social media'** pe platformă.

## 2.3. Planuri de viitor

Îmi doresc să aduc pe platformă cât mai multe elemente sociale în viitor. Vreau, de exemplu, să poți deschide pagina unui alt utilizator ca să explorezi toate melodiile lui publice, și ca să îl urmărești pe platforma, astfel arătându-ți aprecierea. Structura proiectului în prezent permite dezvoltarea acestor funcții cu ușurință, astfel că lista de melodii ale unui utilizator este strâns legată de acesta și poate fi extrasă din baza de date cu foarte puține interogări, fiind totul foarte rapid.

**3.**

# **Tehnologia utilizată**



### 3. Tehnologia utilizată

Acest proiect este bazat pe **frameworkul** de **PHP Yii** si pe **libraria Tone.js**, utilizată pentru a genera sunetul instrumentelor.

#### Lista tehnologiilor:

- JavaScript
  - GitHub
  - HTML
  - CSS
  - Bootstrap
  - PHP 8.2
  - MariaDB 11.3
- } Frontend
- } Backend

### 3. Tehnologia utilizată

#### Framework-uri și librării externe:

- **Yii** - framework care se ocupă de rutare, viewuri și alte elemente de securitate care previn atacurile nedorite.
- **Tone.js** - librărie de **JavaScript** care generează sunetele diferitelor instrumente pe loc, în funcție de anumiți parametri setați în cod. Va permite și user-ului să își creeze propriile instrumente pe viitor.

### 3. Tehnologia utilizată

Documentație pentru resursele folosite:

- <https://getbootstrap.com/docs/5.3/getting-started/introduction/>
- <https://www.yiiframework.com/doc/guide/2.0/en>
- <https://tonejs.github.io/docs/15.0.4/index.html>
- <https://github.com/sofadesign/limonade/blob/master/README.mkd>
- <https://keithwhor.github.io/audiosynth/>

## 3.4. Cerințe de sistem

- Un browser relativ modern (Chromium 113+)
- Conexiune la internet

Pentru a rula **local** proiectul este nevoie de:

- Server Apache
- PHP 8.2
- MariaDB 11.3