

# MP3 přehrávač s databází skladeb

*Projekt ITU, 2019/2020, Z*

**Autor:** Martin Macháček (xmacha73)

**Další členové týmu:** David Gajdoš (xgajdo22), Václav Trampeška (xtramp00)

**Datum odevzdání:** 08-12-2019

---

## Úkol & popis aplikace

Cílem našeho projektu je vytvořit aplikaci, která bude sloužit jako zdroj informací pro návštěvníky zoo, ale velká část aplikace bude zaměřená na audioguide, poskytující zajímavosti o jednotlivých zvířatech. Tato aplikace bude dostupná na mobilních platformách Android a iOS.

Aplikace má za úkol poskytnout jak informace pro návštěvníky, tak i administrátorský panel pro vlastníky, aby mohly editovat informace o své zoo.

## Uživatelé a potřeby

Uvědomili jsme si, že aplikaci jako mp3 přehrávač potřebujeme více zaměřit na cílovou skupinu. Po mnoha průzkumech jsme narazili na fakt, že téměř všechny, již existující zoo, mají velice zastaralý způsob předávání informací a příběhů o svých zvířatech, a to pomocí informačních tabulí a dlouhých odstavců textu u každého zvířete/druhu.

Tabule splňují sice úkol “předávat informace návštěvníkům”, ale úplně se zapomíná na fakt, že když člověk čte, tak se dívá na text, ale ne na zvíře, o kterém se chce něco dozvědět. Kvůli tomuto zákazníci mohou prošvihnout zajímavý zážitek, kdy například tygr jde lovit, zatímco oni jsou zahleděni do tabule.

Proto naše aplikace bude obsahovat zmíněnou audioguide, díky které návštěvníci budou nejen moci pozorovat zvířata, ale zároveň se o nich dozvědět jejich příběh, zajímavé fakta a i například jejich zvuky.

Aplikace bude tudíž zaměřena na tyto pojmy:

- Optimalizace pro user flow
- Audioguide pro jednotlivé zvířata/druhy.
- Minimalistický , novodobý design
- Možnost lokace zvířete na mapě
- Opravdu jednoduchá a intuitivní navigace
- Události v zoo a historie

## Cílová skupina

Cílovou skupinou pro naši aplikaci jsou návštěvníci zoo, ale také i majitelé zoo, kterým naše aplikace poskytne rozhraní pro správu informací zobrazených v aplikaci.

- Persona 1  
Mladý jedinec, který se zajímá o zvířata a nebojí se používat mobilní aplikaci a sluchátka
- Persona 2  
Děvče, které má problém se čtením a chce informace zjistit posloucháním.
- Persona 3  
Pravidelný návštěvník zoo, který chce prožít procházku v zoo na maximum

## Rozhraní & Klíčové prvky GUI

Zde jsou uvedené hlavní komponenty a funkce, které bude rozhraní aplikace obsahovat:

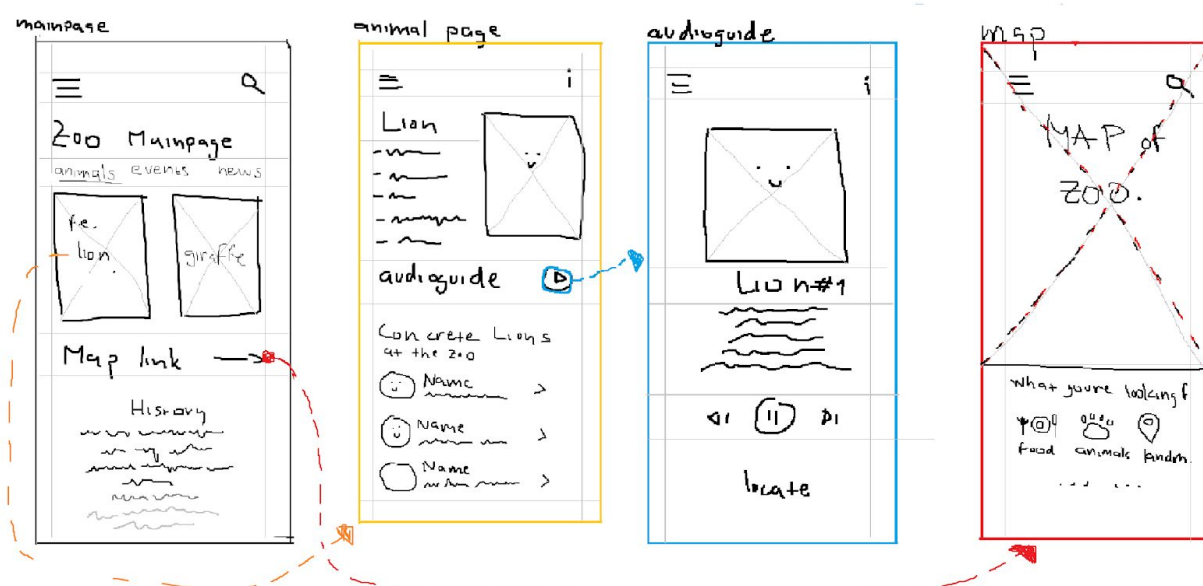
- Úvodní okno, na které bude přehled zoo a např. novinky, události a historie
- Okno zvířete/druhu, na kterém jsou základní informace, vypsání konkrétní zvířata i se jmény a informacemi + tlačítko pro přehrání zvuku, který tento zvíře vydává. Toto okno také bude obsahovat tlačítko "play" pro audioguide, které po zmáčknutí začne přehrávat audioguide a také zobrazí vespod obrazovku panel s ovládacími tlačítky.
- Mapa zoo, ve které lze hledat restaurace, zvířata, spouštět si audio informace o zvířatech, události a například nějaké zajímavé místa.
- Menu, ze kterého z hlavní obrazovky, které bude sloužit jako "hub" pro celou aplikaci.
- Možnost načtení audioguide pomocí QR kódu, aby uživatel nemusel hledat zvíře v aplikaci, ale aby se mohl plně zaměřit na jeho zážitek v zoo.

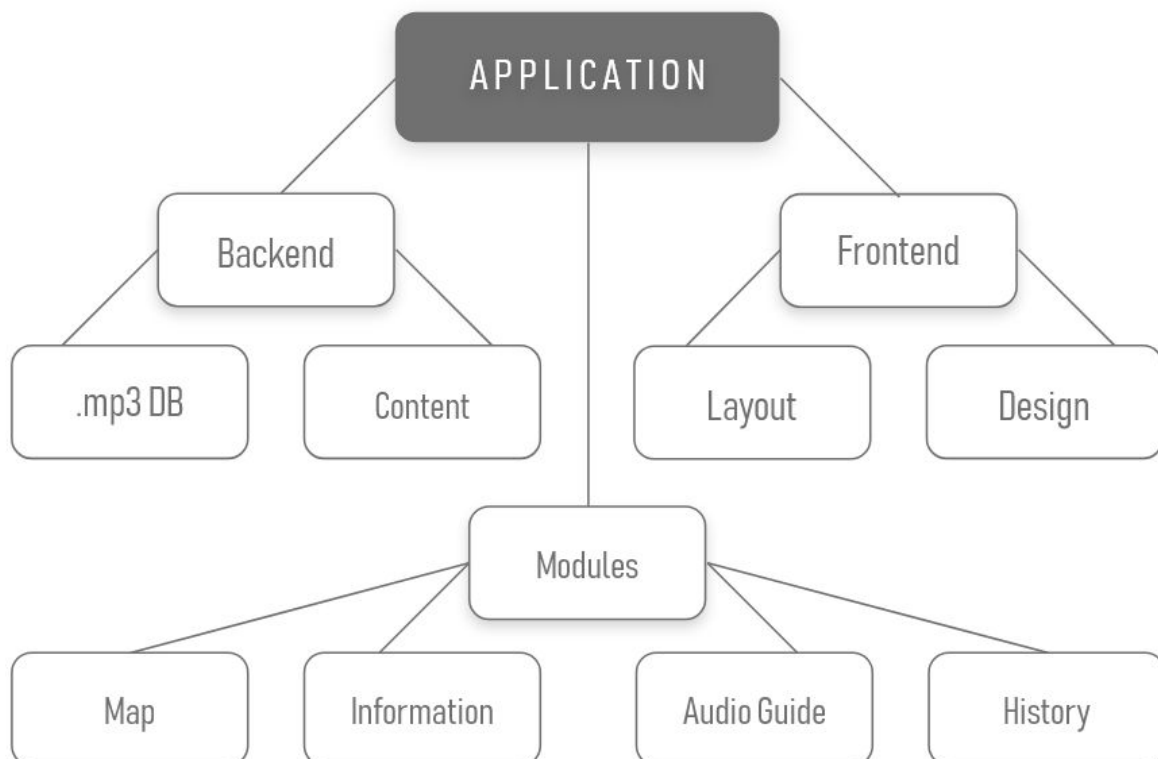
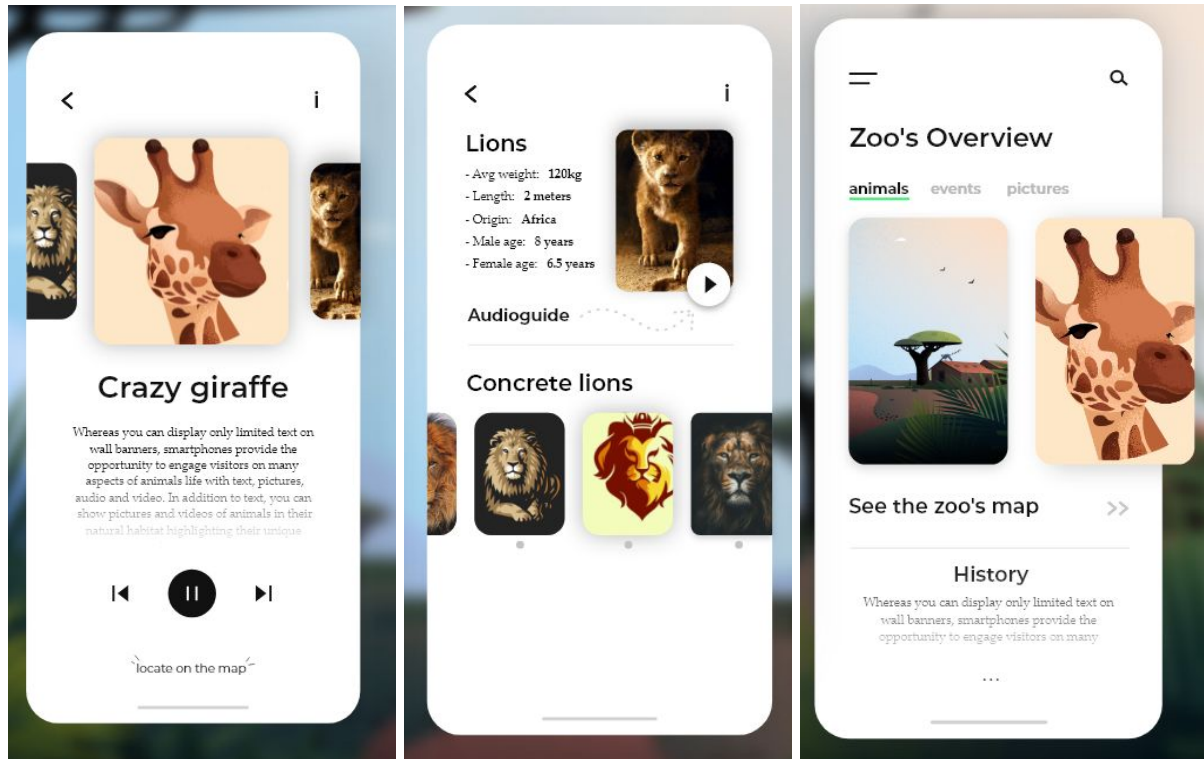
Při návrhu je potřeba zajistit co nejjednodušší rozhraní pro uživatele, nesmí se stát, že uživatel bude čekat na aplikaci než se načte, musí mu pomáhat, ne ho otravovat. Aplikace bude mít velice novodobý a minimalistický design, s velkým důrazem na jednoduchost a plynulost.

Důležitou částí aplikace je databáze mp3 souborů pro audioguide NEBO schopnost aplikace číst uvedený text.

## Počátky návrhu & GUI

Začal jsem rychlým náčrtem, jak by aplikace mohla být uspořádaná. Mám již nachystaný návrh designu a UI, ovšem není ještě plně hotov, tudíž zde přidávám první rychlý náčrt a dosavadní stav propracovanějšího návrhu.





## **Diskuze s uživateli**

Osoby se kterými jsem komunikoval řekli, že jsou s návrhem velice spokojeni, že nepotřebují žádnou přepřácanou aplikaci. Dozvěděl jsem se, že chtějí také dark mode, a že preferují jednoduchost použití nad vzhledem. V další fázi budeme sledovat reakci uživatelů na již implementovanou aplikaci.

## **Použité technologie**

Jelikož již máme zkušenosti s prací s HTML a CSS, rozhodli jsme se pro využití některého z frameworků, které dokáží přeložit webové technologie do mobilní aplikace. Na výber bylo hned několik technologií, nejprve nás zaujal framework Phonegap, pracující na technologii Cordova. Poskytuje velice zajímavé prostředí pro testování aplikace přímo na telefonech pomocí zabudované mobilní aplikace, a není nijak náročný.

Po vyzkoušení se ovšem ukázalo několik nedostatků, jednou z nich byl fakt, že aplikace pro testování neustále “padala”, další nedostatek byl, že tento framework není dostatečně známý, tudíž pokud se zasekneme na nějakém hlubším problému, tak není tak snadné najít řešení skrz pomoc na internetu. Tento framework jsme tudíž vyřadili.

Shodli jsme se na frameworku Ionic (taktéž na Cordova). Je velice známý, také překládá webové technologie do mobilní aplikace, a navíc má zabudovanou podporu pro frameworky Angular a React, což hodně zlepšuje modularitu a správu struktury. Veškeré uvedené technologie jsou založené na JS, s tím, že Angular kód, který budeme používat, je nadstavba JS, aneb Typescript.

## **Komponenty**

Angular, který Ionic používá, má hlavní funkcionalitu - komponenty, které v tomto kontextu lze brát i jako komponenty architektury aplikace. Navíc, Ionic využívá tyto Angular komponenty, a dokáže z nich vytvořit “page”, která reprezentuje samotnou stránku, do které se uživatel může dostat.

## Struktura aplikace

Jak je již zmíněné v předchozí sekci, strukturu aplikace do určité části určuje Angular. Tuto strukturu lze pak ale seřadit do těchto pojmů, které určují obsah tak pojmenovaných složek:

### 1) Services

- V případě, že chceme sdílet například data, funkce napříč více komponenty v aplikaci, jsou použity tzv. Services, které obsahují privátní proměnné, funkce a samotné “data”, které pak lze jednoduše pomocí importování získat z ostatních komponent.

### 2) Pages

- Tento pojem definuje komponentu, která je v aplikaci zobrazena jako samotná stránka, se kterou uživatel pracuje. Je “univerzální” pro framework Ionic.

### 3) Other components

- Klasické komponenty, které nejsou použity jako stránky. Jsou použity v případě, že se na více stránkách by byl opakován stejný kód, například menu.

### 4) Modules

- Téměř každá komponenta má svůj vlastní modul, který nijak nejde poznat při interakci s aplikací, ale poskytuje velice důležitou logickou funkcionalitu pro Angularový styl struktury aplikace

### 5) Router

- Tento pojem je v Angularu velice známý, ovšem s využitím Ionic Pages je velice jednoduché s tímto Routerem pracovat, slouží v podstatě pro změnu “URL”, a odděluje velice chytře View od Logiky

Všechny tyto pojmy popisují princip, ve kterém se snaží co nejvíce oddělit vzhled od logiky aplikace, poskytují velké množství modularity v aplikaci.

## Relevantní části pro UI

### Pages

- Animal Kind
- Animals
- Events
- Event
- History
- Map
- Scan
- \*Home

### Services

- Animals
- Player
- Events

### Other Components

- Controls
- Sidemenu

\* Tato komponenta není strukturou mezi pages, ale chová se tak

## Návrhové vzory

Angular v kombinaci s Ionicem poskytují široký základ ve smyslu abstrakce, u čehož je konkrétním příkladem již zmíněná Service. Tato Service totiž zabaluje veškerá data a funkcionalitu do svého “modelu”, a lze k nim přistoupit pouze skrz využití “Angularového importu”, a zahrnutí této Service do modulu aplikace.

Dále je využit návrhový vzor Fasáda, která obaluje zmíněné services, aby se nestalo, že existuje jedna “vševědoucí service”, která obstarává veškerou logiku a data aplikace. Jedná se totiž o modulární aplikaci, kde se chceme co nejvíce vyhnout takovýmto případům.

## Datový model

Tato aplikace nemá nutnost obsahovat plnohodnotnou databázi, tudíž se spokojíme s datovým modelem v podobě Service, která bude obsahovat pole objektů, v nichž budou zakomponované všechny zvířata, informace o nich, odkazy na mp3 soubory, i cesty k obrázkům. Tento způsob velice připomíná JSON formát, ale jedná se pouze o pole objektů., ke kterým jednotlivé pages budou přistupovat.

Zde je například základ datového modelu pro zvířata:



**AnimalKind** (druh zvířete)

- Type
- Image
- Origin
- Feeding
- Temper
- Story
- Animals (pole objektů Animal)
- Audio

**Animal** (konkrétní zvíře)

- Name
- Age
- Gender
- Image
- Audio

**Závěrečný výběr návrhu** (jen návrhu)

Design aplikace jsme ve finální fázi volili podle výsledků uživatelského testování. V tomto testování jsme každý předložili svůj návrh designu, a nechali uživatele zvolit, který z poskytnutých se jim líbil nejvíce. Mezi uživateli vyhrál návrh, který mám zmíněný výše.

Skrz tohle testování jsme ale pouze zjistili, který návrh je nejvíce oblíbený, ale nezjistili jsme, jestli je skutečně použitelný a jestli se jedná o opravdu přehledný a funkční návrh, tudíž jsme pokračovali na samotné testování aplikace pomocí maket

**Testování na uživateliích**

Makety byly vytvořené v aplikaci Adobe Xd, každá stránka aplikace byla vyobrazena jako jedna maketa. Sledoval jsem, jak uživatelé reagují na každou stránku, s tím, že jsem jim na začátku prezentace zadal úkol :

- 1) Najděte jméno nejstaršího lva v naší Zoo.
  - Uživatel zprvu nevěděl, kde najít seznam všech zvířat, tudíž se odhalil první problém
- 2) Zapněte audioguide pro opice, a po chvílce vypněte audio.
  - U tohoto úkolu byl velký kámen úrazu možnost vypnout audio mimo stránku s audioguide, tudíž je potřeba implementovat panel, který umožní vypínat audio i mimo stránku s audioguide

3) Zapněte audioguide pro lvi, a přeskočte na další zvíře

- Tenhle úkol šel klidně, převážně bez problémů, až na jeden, a to ten, že uživatelé vůbec netrávili žádný čas na stránce s audioguide, jakoby byla úplně zbytečná, tento poznatek vedl k zrušení této stránky a byla nahrazena “vznášejícím se” panelem s ovládacími prvky a informacemi o hrajícím zvířeti.

4) Pustěte zvuk, které konkrétní zvíře vydává

- Tento úkol měl velice záporný ohlas, protože nebylo vůbec jasné, jestli konkrétní zvířata na stránce druhu fungují jako tlačítka, či ne.

### **Dodatečné poznatky**

- Nepotřebné ikonky “i” v pravém horním rohu

### **Závěrečný design**

Ve finální fázi tvorby aplikace byl kladen velký důraz na jednoduchost a přehlednost, a díky testování na uživateli bylo mnohem jasnější, kterým směrem se má aplikace vydat.

- Hlavní stránka obsahuje 3 nejbližší zvířata a události a 3 nadcházející události. Tyto panely také mají tlačítko, kterým se uživatel dostane na seznam všech zvířat, či seznam všech událostí. Dále je na úvodní obrazovce odkaz na mapu zoo, tlačítko pro načtení QR kódu (pro jednotlivé zóny, zvířata, atd..), odkaz na historii zoo a tlačítko pro menu.
- Menu aplikace je dostupné z hlavní stránky, jsou v něm odkazy na jednotlivé stránky v aplikaci (contact a help je funguje pouze jako placeholder)
- Stránka na QR kód poskytuje 2 možnosti zadání kódu, a to manuální napsání klávesnicí, či naskenování pomocí čtečky v telefonu
- Mapa v sobě bude mít aktuální pozici uživatele, a bude zobrazovat, podle nastaveného filtru pomocí tlačítek dole, buď zvířata, události, zajímavé místa, či restaurace. Na zvířata na mapě lze klepnout pro zapnutí audioguide.

- Stránka se seznamem zvířat má v sobě vyhledávací pole, aby uživatel mohl jednoduše vyhledat zvíře, které potřebuje
- Stránka s druhem zvířete, ve které lze také zapnout audioguide, zjistit informace o zvířeti, jsou zde také tlačítka na přehrání zvuků, které konkrétní zvířata vydávají.
- Stránka nesoucí seznam všech událostí, taktéž s vyhledávacím polem
- Stránka s konkrétní událostí, kde jsou vypsány informace jako datum, čas, podrobné informace

Konkrétní ukázky designu naleznete níže:

