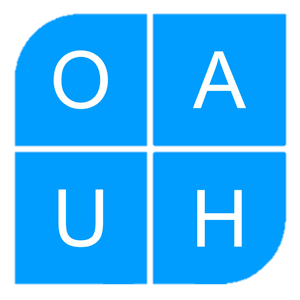
Obchodní akademie, Vyšší odborná škola a Jazyková škola s právem státní jazykové zkoušky Uherské Hradiště



Icon

Description automatically generated

Dokumentace – GambleGO

Tomáš Gabriel & Adam Lukáš

Programátorská část

# Obsah

[Obsah 2](#_Toc105355479)

[Shrnutí 4](#_Toc105355480)

[Použitý jazyk a knihovny 4](#_Toc105355481)

[Uživatelská dokumentace 4](#_Toc105355482)

[Před prvním spuštěním 4](#_Toc105355483)

[Jak spustit server 5](#_Toc105355484)

[Ovládání stránky 5](#_Toc105355485)

[Vytváření účtu 5](#_Toc105355486)

[Mazání účtu 5](#_Toc105355487)

[Výběr účtu 5](#_Toc105355488)

[Přidávání tokenů 5](#_Toc105355489)

[Zobrazení a prodej skinů 5](#_Toc105355490)

[Otevírání beden (Cases) 6](#_Toc105355491)

[Vylepšování skinů (Upgrader) 6](#_Toc105355492)

[Kontrakty (Contracts) 6](#_Toc105355493)

[Technická dokumentace 7](#_Toc105355494)

[Wireframe 7](#_Toc105355495)

[Analýza projektu 9](#_Toc105355496)

[Harmonogram 9](#_Toc105355497)

[Funkce projektu 9](#_Toc105355498)

[Zpráva o testování 10](#_Toc105355499)

[Server 11](#_Toc105355500)

[Index.js 11](#_Toc105355501)

[Client 12](#_Toc105355502)

[App.jsx 12](#_Toc105355503)

[CaseDetail.jsx 12](#_Toc105355504)

[Vlastní hodnocení 14](#_Toc105355505)

[Závěr 15](#_Toc105355506)

[Splněno: 15](#_Toc105355507)

[Nesplněno: 15](#_Toc105355508)

[Věci navíc: 15](#_Toc105355509)

[Zdroje 16](#_Toc105355510)

# Shrnutí

**Název projektu:**  GambleGO

**Termín zpracování:** 21. 2. – 8. 5. 2022

**Typ projektu:** Web

**Tým:** Tomáš Gabriel - programátor

Adam Lukáš – databázový specialista

**Vývojové** **prostředí**: Visual Studio Code, Mozilla Firefox

**Odkaz na GitHub:** https://github.com/VirtualCZ/GambleGO

**Způsob zpracování:**  Stránka je napsaná v JavaScriptu s klihovnama: ReactJS, Axios, Express, …

Pro design webu jsem zvolil TailWind, kvůli jeho jednoduchosti. Design je originál.

Web obsahuje 5 stránek: Správa účtů, inventář, zobrazení truhel, podrobnosti a otevírání truhel a upgrader (vylepšovač skinů).

Webová aplikace využívá databázi, kterou vytvořil Adam Lukáš. Pro komunikaci s databází jsem vytvořil API se kterou mi pomohl Adam.

# Použitý jazyk a knihovny

**Jazyk**: JavaScript, CSS, HTML, SQL

**Knihovny**: JavaScript: ReactJS, Axios, react-router-dom, body-parser, cors, express

CSS: TailWind

**Server**: Xampp (PHPmyadmin, Apache), NodeJS

# Uživatelská dokumentace

## Před prvním spuštěním

* Stáhněte a nainstalujte [Xampp with PHP version 7.4.9](https://sourceforge.net/projects/xampp/files/XAMPP%20Windows/7.4.9/xampp-windows-x64-7.4.9-0-VC15-installer.exe/download).
* Spusťte Xampp a v něm klikněte na “Start” u Apache a MySQL
* Stiskněte “Admin” u MySQL a levém sloupečku klikněte na “New”
* Do pole “Database name” napište “gamblego1”, vyberte “utf8\_czech\_ci” a zmáčkněte tlačítko “Create”
* V horní navigaci stiskněte “Import”, najděte tlačítko “Browse…” a vyberte soubor s názvem “gamblego1.sql” (nachází se ve složce GambleGO”
* Stiskněte “Go”
* Stáhněte a nainstalujte [NodeJS LTM](https://nodejs.org/dist/v16.15.0/node-v16.15.0-x64.msi)
* Otevřete powershell ve složce “client” a spusťte příkaz “npm install vite”

## Jak spustit server

1. Přes skripty
   1. Před prvním spuštěním skriptů je třeba:
      1. Spusťte powershell jako admin a spusťte příkaz "Set-ExecutionPolicy Unrestricted"
      2. Zmáčkněte „A“
   2. Zmáčkněte pravé tlačítko u souboru „Run client.ps1“ a zmáčkněte „Run with Powershell“
   3. Zmáčkněte pravé tlačítko u souboru „Run server (api).ps1“ a zmáčkněte „Run with Powershell“
2. Manuálně
   1. Spusťte 2 powershelly – jeden ve složce „client“ a druhý ve složce „server“
   2. U obou spusťte příkaz „npm run dev“

## Ovládání stránky

Výchozí odkaz na stránku je [localhost:3000](http://localhost:3000/)

### Vytváření účtu

* Pod sekcí „Account“ napište do pole „username“ Vaše uživatelské jméno a do pole „tokens count“ počet tokenů, které má účet mít
* Zmáčkněte „Add an account“ a znova načtěte stránku
* Účet by se měl oběvit v seznamu

### Mazání účtu

* Pod sekcí „Account“ najděte účet, který chcete vymazat a klikněte na „Delete“
* Znova načtěte stránku a účet by měl být vymazán

### Výběr účtu

* Pod sekcí „Account“ najděte účet, který chcete používat a klikněte na „Select“
* V pravém horním rohu by se měly načíst Tokeny, uživatelské jméno a uživatelské ID

### Přidávání tokenů

* V pravém horním rohu stiskněte „Add Tokens“ a mělo by se přičíst 1000 Tokenů na právě zvolený účet

### Zobrazení a prodej skinů

* Klikněte na Vaše uživatelské jméno v pravém horním rohu
* Klikněte na tlačítko „Sell“ u jakéhokoliv skinu
* Znovu načtěte stránku
* Měly by se Vám přičíst tokeny
  + V případě, že máte více než 1 „amount – množství“, tak se jeden odečte
  + V případě, že máte 1 „amount - množství“, tak se skin vymaže

### Otevírání beden (Cases)

* V sekci „Cases“ si vyberte jakoukoliv bednu a klikněte na ní
* Stiskněte tlačítko „OPEN“
* V případě, že máte dostatek Tokenů, tak by se měl ukázat skin, co jste dostali a odečetly se Tokeny
* Po otevření skinu zmáčkněte tlačítko „Close“

### Vylepšování skinů (Upgrader)

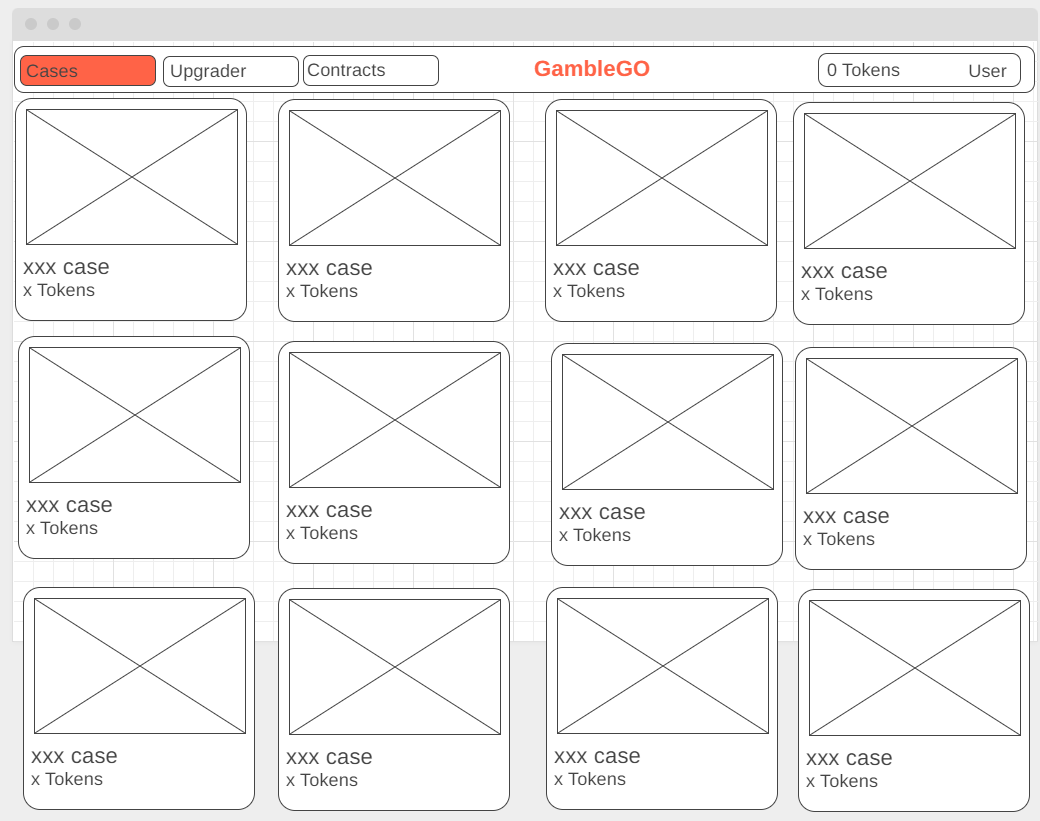
* V sekci „Upgrader“ se v levém sloupci ukážou Vaše skiny
* Vyberte jeden z nich
* V prostředním sloupci si vyberte o kolik % se skin zlepší
* V pravém sloupci si vyberte skin, který chcete dostat
  + Pod každým skinem je v % napsána šance že ten skin získáte
  + V případě, že skin získáte se skin z Vašeho inventáře vymaže a bude nahrazen novým
  + V případě, že skin nezískáte se vymaže skin z Vašeho inventáře
* Stiskněte tlačítko „Upgrade“

### Kontrakty (Contracts)

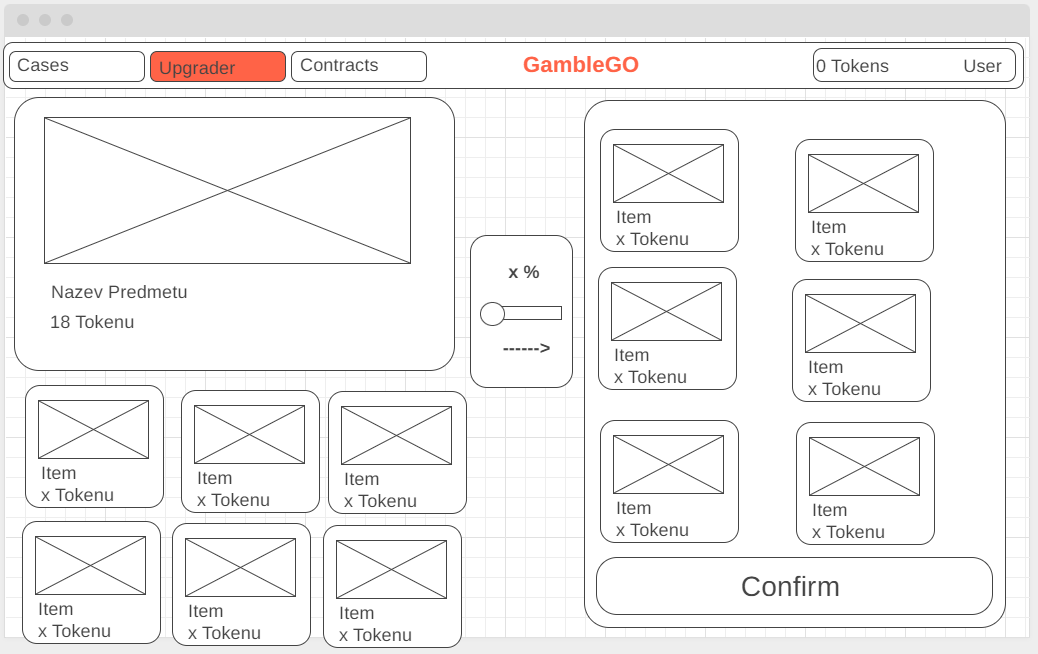
Nezprovozněno

# Technická dokumentace

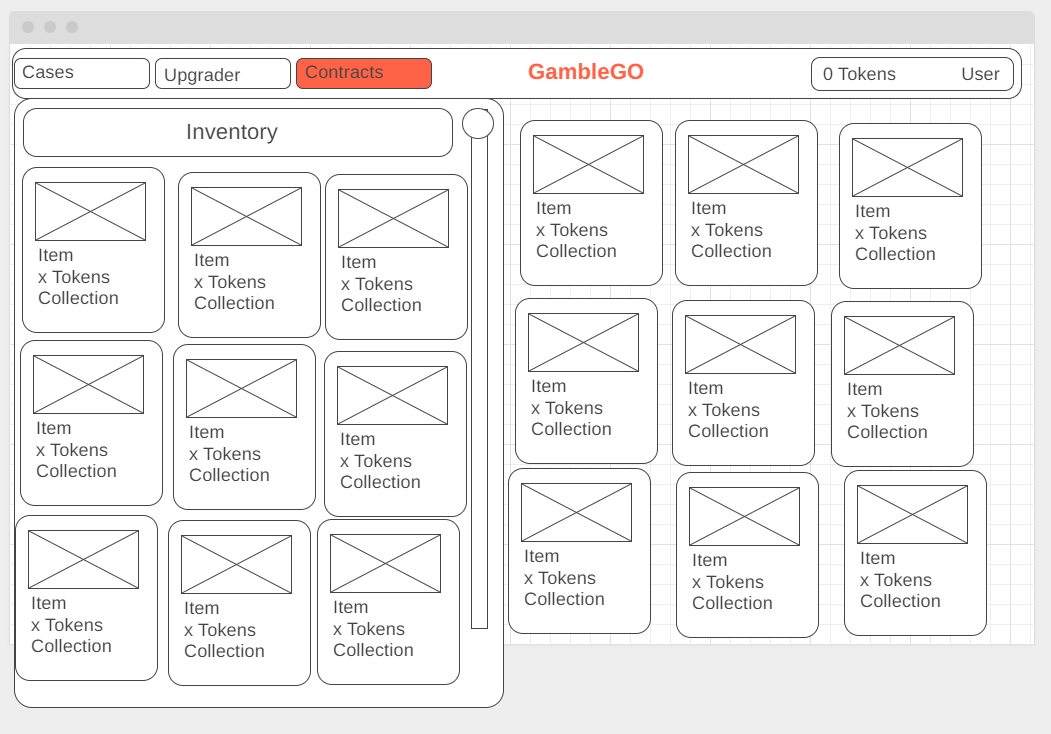
## Wireframe



Obrázek Wireframe - Cases



Obrázek Wireframe - Upgrader



Obrázek Wireframe - kontrakty

## Analýza projektu

### Harmonogram

|  |  |
| --- | --- |
| 21. 2. - 28. 2. | zadání projektu |
| 28. 2. - 13. 3. | analýza a návrh |
| 11. 4. | funkce 1, 2, 3, 4, 7, 8 |
| 25. 4. | funkce 5, 6 |
| 8. 5. | předání projektu |

### Funkce projektu

|  |  |  |  |
| --- | --- | --- | --- |
| Č. | Název funkce | Aplikační část (Tomáš Gabriel) | Databázová část (Adam Lukáš) |
| 1. | Zobrazení beden | Bedny budou zobrazeny (upřesnit, jak budou zobrazeny) | Zobrazení definovaných beden |
| 2. | Zobrazení obsahu bedny | Po kliknutí se zobrazí seznam předmětů | Pro konkrétní bednu zobrazí seznam předmětů |
| 3. | Otevření bedny | Přehraje se animace a vygeneruje se náhodný předmět, který bedna obsahuje | Získaný předmět bude umístěn do inventáře |
| 4. | Navbar | Bude vytvořen pomocí React Router | ----- |
| 5. | Upgrader | Uživatel může obětovat předmět a může získat předmět s vyšší hodnotou nebo o něj přijít s určitou pravděpodobností (pravděpodobnost určíme) | Odstranění předmětu z inventáře a případné přidaní nového předmětu |
| 6. | Kontrakty | Uživatel obětuje 10 předmětů z inventáře, získá za ně předmět větší vzácnosti | Výběr 10 předmětu z inventáře a jejich odstranění a přidání nového předmětu |
| 7. | Doplnění tokenů | Po kliknutí na tlačítko plus v navbaru se otevře okno na přidání peněz | Z databáze se načtou tokeny uživatele a přidá se k nim daná hodnota |
| 8. | Inventář | Zobrazí předměty, které uživatel vlastní, bude mít možnost je: prodat, použít v bodu 5 nebo 6 | Výpis seznamu inventáře a funkce z bodu 5 nebo 6 |

# Zpráva o testování

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Jméno testera | Jak se ti projekt líbí? (grafická stránka) | Na jaké chyby jsi narazil? | Jak bys projekt zlepšil / co přidat? | Náročnost instalace | Chceš ještě něco dodat? |
| Petr Bederka | pěkný | prodávání skinů, | automatické refrešování, animace točení, contracts | Jednoduchá |  |
| Michal Brožovič | po grafické stránce je projekt velice hezky zpracovaný. | Při svém testování jsem na žádné chyby nenarazil | Projekt se mi líbí takový jaký je | Střední | Myslím si že projekt má velký potenciál a jde vidět že vývojáři si dali záležet. Už se těším na další verze projektu. |
| Matěj Krajsa | Velice krásně graficky zpracováno, příjemné pro oči, a dost se mi líbí ikony. | Na žádnou chybu jsem nenarazil. | Rozhodně je tady místo na přidávání věcí třeba více beden na výběr, ale aktuální počet beden je dostačující. | Jednoduchá |  |

Zprávy od testerů **jsou** v originálním znění

## Server

### Index.js

Graphical user interface, text

Description automatically generated

Obrázek připojení k databázi

Slouží pro připojení k databázi

Graphical user interface, text, application

Description automatically generated

Obrázek přeposílání proměnných z webu na databázi

Aplikace si vezme proměnné z URL a pošle je přes proceduru do databáze

Logo, company name

Description automatically generated

Obrázek čekání na dotazy

API čeká na dotazy pod <http://localhost:3030/>

## Client

### App.jsx

Text

Description automatically generated

Obrázek Ukázka useState hooku

Ukázka useState hooku a načítání proměnné z úložiště v prohlížeči

Text, letter

Description automatically generated

Obrázek Router, Context

React Router, cesty k jednotlivým elementům

AppContext pošle všem elementům, které jsou pod ním jeho proměnné

### CaseDetail.jsx

Graphical user interface, application

Description automatically generated

Obrázek Komunikace s API -URL, useEffect

Příklad komunikace s API – posílání proměnné v URL, využití useEffect hooku

Text

Description automatically generated with medium confidence

Obrázek Ukázka využití .map funkce

Pro výpisu dat z arraye lze využít .map funkce. Do ní vypíšeme co kam chceme vypsat. Vypisuje se to jako for loop, s tím že se opakuje tolikrát, kolik objektů se nachází v arrayi.

# Vlastní hodnocení

Myslím si, že se mi projekt vcelku vydařil, vzhledem k tomu že jsem se ReactJS v tuto dobu teprve učil. Jde tam určitě hodně věcí zlepšit a například jednotlivé karty převést do komponentů a do nich posílat props. Tento projekt mě naučil mnoho věcí mezi které patří Tailwind CSS a React hooky.

Grafické zpracování bych řekl že je také velmi pěkně udělané.

Projekt má ale vcelku velké chyby. Například u prodávání skinů může uživatel prodávat stejný předmět do doby než si znova načte stránku. Jak to tak ale vypadá, tak si této chyby ani jiných chyb testeři nevšimli.

# Závěr

## Splněno:

* Jde si zobrazit stránku s truhlama
* Jde si otevřít truhlu. Po otevření se ukáže, co uživatel dostal.
* Upgrader funguje (bohužel jen někdy)
* Jde zobrazit inventář a prodávat z něj skiny
* Navbar funguje a ukazuje aktuálního uživatele a počet tokenů. Lze v něm také přidat tokeny

## Nesplněno:

* Při otevření truhly se nepřehraje animace, ale ukáže se jen okno se skinem, co uživatel dostal
* Kontrakty

## Věci navíc:

* Základní správa účtů
* Synchronizace tokenů s databází
* API server pro komunikaci s databází
* Pěkný design

# Zdroje

Playlist CRUD tutorial. *YouTube - PedroTech* [online]. [cit. 2022-06-05]. Dostupné z: https://www.youtube.com/playlist?list=PLpPqplz6dKxW42HMj6iuiA45d4xqP9ICq

ReactJS dokumentace. *ReactJS dokumentace* [online]. [cit. 2022-06-05]. Dostupné z: https://reactjs.org/docs/getting-started.html

React Router DOM dokumentace. *React Router DOM dokumentace* [online]. [cit. 2022-06-05]. Dostupné z: https://reactrouter.com/docs/en/v6/getting-started/overview

ReactJS hooks tutorial. *ReactJS hooks tutorial* [online]. [cit. 2022-06-05]. Dostupné z: https://www.youtube.com/watch?v=LlvBzyy-558&t=4235s

Vite dokumentace. *Vite dokumentace* [online]. [cit. 2022-06-05]. Dostupné z: https://vitejs.dev/guide/#trying-vite-online

TailwindCSS instalace. *TailwindCSS instalace* [online]. [cit. 2022-06-05]. Dostupné z: https://tailwindcss.com/docs/guides/create-react-app

TailwindCSS dokumentace. *TailwindCSS dokumentace* [online]. [cit. 2022-06-05]. Dostupné z: https://tailwindcss.com/

Seznam obrázků

[Obrázek 1 Wireframe - Cases 6](#_Toc105351789)

[Obrázek 2 Wireframe - Upgrader 6](#_Toc105351790)

[Obrázek 3 Wireframe - kontrakty 7](#_Toc105351791)

[Obrázek 4 připojení k databázi 10](#_Toc105351792)

[Obrázek 5 přeposílání proměnných z webu na databázi 10](#_Toc105351793)

[Obrázek 6 čekání na dotazy 10](#_Toc105351794)

[Obrázek 7 Ukázka useState hooku 11](#_Toc105351795)

[Obrázek 8 Router, Context 11](#_Toc105351796)