

Střední průmyslová škola elektrotechnická
a Vyšší odborná škola Pardubice

STŘEDNÍ PRŮMYSLOVÁ ŠKOLA ELEKTROTECHNICKÁ

MATURITNÍ PRÁCE – PROGRAMOVÁNÍ

Piškvorky

„Prohlašuji, že jsem maturitní práci vypracoval(a) samostatně a použil(a) jsem literárních pramenů, informací a obrázků, které cituji a uvádím v seznamu použité literatury a zdrojů informací a v seznamu použitých obrázků a neporušil jsem autorská práva.

Souhlasím s umístěním kompletní maturitní práce nebo její části na školní internetové stránky a s použitím jejích ukázek pro výuku.“

V Pardubicích dne

.....

podpis



ZADÁNÍ MATURITNÍ PRÁCE

Maturitní zkouška – profilová část – Maturitní projekt

Obor:	18-20-M/01 Informační technologie	Školní rok:	2020/2021
Jméno a příjmení žáka:	Jan Najman	Třída:	4.D
Téma maturitní práce:	Piškvorky		
Vedoucí maturitní práce:	RNDr. Jana Reslová		
Pracoviště vedoucího:	SPŠE a VOŠ Pardubice, Karla IV. 13, Pardubice		

Kategorie maturitní práce: PROGRAMOVÁNÍ

Téma maturitní práce (popis):

Naprogramujte aplikaci Piškvorky podle zadaných bodů.

Hlavní body – specifikace maturitní práce:

- 1) Hry se budou ukládat do databáze
- 2) Hry budou moci být hrány na tahy nebo odehrány v celku
- 3) Rozdělení uživatelů na přihlášené a nepřihlášené a administrátory
- 4) Tahy hráčů budou kontrolovány na serveru
- 5) Data odesílaná klientem nebo uživatelem budou mít co nejmenší velikost
- 6) Hry budou mít různé velikosti
- 7) Každý přihlášený uživatel bude mít statistiky

Způsob zpracování maturitní práce:

Maturitní práce musí být zpracována v souladu se zákonem č. 121/2000 Sb., autorský zákon. Maturitní práce je tvořena praktickou částí (viz téma a specifikace maturitní práce výše) a písemnou prací.

Maturitní práce je realizována žákem převážně v rámci výuky ve 4. ročníku. Lze pokračovat na projektu z nižších ročníků.

Podrobné pokyny ke zpracování maturitní práce příslušné kategorie jsou uvedeny v dokumentu *MP_Zpracovani-podrobne_pokyny*.

Zpracování písemné práce musí odpovídat požadavkům uvedeným v dokumentu *MP_Formalni_stranka_dokumentace*. Písemná práce musí být rovněž zpracována v souladu s normou pro úpravu písemností [ČSN 01 6910 (2014)], s citační normou [ČSN ISO 690], se základními typografickými pravidly, pravidly sazby, gramatickými pravidly a pravidly českého pravopisu.

Pokyny k rozsahu a obsahu maturitní práce:

Rozsah a obsah praktické části maturitní práce je určen tématem a specifikací maturitní práce, rozsah písemné části maturitní práce je minimálně 15 normostran vlastního textu. Do uvedeného rozsahu se nezapočítávají úvodní listy (titulní list, prohlášení, zadání, anotace, obsah...), závěrečné listy (seznam literatury...) a přílohy. Podrobné pokyny k rozsahu a obsahu maturitní práce příslušné kategorie jsou uvedeny v dokumentu *MP_Zpracovani-podrobne_pokyny*.

Kritéria hodnocení maturitní práce a její obhajoby:

Maturitní práce a její obhajoba u maturitní zkoušky je hodnocena bodově. Celkový dosažený počet bodů je součtem bodů přidělených vedoucím práce a oponentem (maximální dosažitelný počet je 100 bodů). Výsledná známka u maturitní zkoušky je stanovena přepočtem celkového počtu bodů na známku pomocí tabulky uvedené níže.

V případě nesplnění tématu maturitní práce a v případě plagiátorství bude práce hodnocena stupněm „nedostatečný“.

Tabulka bodového vyjádření hlavních kritérií a obhajoby:

- praktická část – zpracování tématu maturitní práce[max. 25 bodů]
- písemná část – zpracování, formální a obsahová stránka[max. 15 bodů]
- obhajoba maturitní práce před zkušební komisí[max. 10 bodů]

Podrobná kritéria pro hodnocení maturitní práce příslušné kategorie jsou uvedena v dokumentu *MP_Kriteria_hodnoceni*.

Tabulka přepočtu celkového počtu bodů na známku:

[0 .. 60 bodů].....	[5]
[61 .. 70 bodů].....	[4]
[71 .. 80 bodů].....	[3]
[81 .. 90 bodů].....	[2]
[91 .. 100 bodů].....	[1]

Požadavek na počet vyhotovení maturitní práce a její odevzdání:

Kompletní maturitní práce (praktická i písemná) se odevzdává ve stanoveném termínu vedoucímu maturitní práce. Písemná práce se odevzdává v jednom tištěném vyhotovení obsahující podepsaný přenosný nosič CD/DVD/SD s písemnou prací a sadou kompletních dat v elektronické podobě dle pokynů v dokumentu *MP_Zpracovani-podrobne_pokyny*.

Termín odevzdání maturitní práce: 26. března 2021

Délka obhajoby maturitní práce před zkušební maturitní komisí: 15 minut

Pardubice 10. října 2020

.....
Mgr. Petr Mikuláš, ředitel školy

Anotace

Práce se zabývá rychle rostoucím a čím dál více populárním systémovým jazykem Rust. Ukazuje, jak je tento systémový programovací jazyk vyspělý a jaké má výhody oproti ostatním systémovým jazykům.

Tato práce konkrétně se zabývá programováním RESTful API serveru v programovacím jazyku Rust pomocí frameworku Actix Web a programováním front-endové webové aplikace pomocí frameworku Yew.

Klíčová slova: Rust, API, RESTful, Actix, Actix Web, Web, Server, Aplikace, Front-end, Front-end aplikace, programování, framework, Yew

Annotation

This work deals with the rapidly growing and increasingly popular system programming language Rust. It shows how advanced this system programming language is and what its advantages are over other system languages.

This work specifically deals with programming RESTful API server in Rust programming language using the Actix Web framework and programming a front-end web application using framework Yew.

Keywords: Rust, API, RESTful, Actix, Actix Web, Web, Server, Application, Front-end, Front-end applications, programming, framework, Yew

Obsah

Úvod	9
1 Analýza obdobných aplikací	10
1.1 turtlediary	10
1.1.1 Kladné stránky	10
1.1.2 Záporné stránky	10
1.2 Ultimate Tic Tac Toe	11
1.2.1 Kladné stránky	11
1.2.2 Záporné stránky	11
1.3 gametable	12
1.3.1 Kladné stránky	12
1.3.2 Záporné stránky	12
2 Návrh projektu	13
2.1 Obecná struktura	13
2.2 API	13
2.3 Redis	13
2.4 Databáze	13
2.5 Front-end	13
3 Zpracování praktické části	14
3.1 Back-end	14
3.1.1 Správa uživatelů	14
3.1.2 Vytváření žádostí o hru	14
3.1.3 Vytvoření hratelné hry	14
3.1.4 Hraní hry	14
3.2 Front-end	15
3.2.1 Registrace	15
3.2.2 Profil	15
3.2.3 Výpis uživatelů	15
3.2.4 Výpis her	15
3.2.5 Hraní hry	15
3.2.6 Výpis pozvánek	16
3.2.7 Vytváření pozvánky	16
3.2.8 Úprava uživatele	16

4	Závěr	17
5	Seznam použité literatury a zdrojů informací	18
6	Seznam použitých zkratk	19
7	Seznam obrázků, tabulek, příloh	20
8	Přílohy	21

Úvod

Poslední dobou se všichni pokouší optimalizovat své web servery již při jejich programování, kvůli náporu, který by nemusely stíhat. Kdo tak neprovede může toho litovat a snaží se tento problém obejít jinak. Tím že změní jazyk nebo vytvoří repliky své aplikace a dají na ně load-balancer, tuto možnost nakonec musí využít všichni, při velmi vysokém náporu.

Bohužel nepoužívanější jazyky k naprogramování web serveru jsou PHP, JS (Node.js) nebo Python (Flask, Django). Dá se v nich rychle udělat co potřebujete, ale mají spoustu nevýhod, a hlavně všechny tyto jazyky jsou tzv. interpretované jazyky. To znamená, že na pozadí běží nějaký engine (interpreter), který musí zpracovat daný kód za běhu. Jejich největším problémem oproti kompilovaným programovacím jazykům jako jsou Rust, C++, C# je jejich rychlost a možnost zjistit chybu při kompilaci.

Tabulka 1: Porovnání rychlosti jazyků

Jazyk	Pomalejší než C++ (gcc -O2)
Rust	7 %
C#	78 %
Node.js	93 %
PHP	596 %
Python 3.5	1800 %
Python 2.7	2562 %

Rozhodl jsem se ukázat, že systémový programovací jazyk Rust je na tolik vyspělý, že se nejen zvládne vše, co jiné jazyky, ale i to, že je rychlejší než konkurence.

Vybral jsem si programování RESTful API serveru a front-endové webové aplikace z důvodu, že jsou to dne nejpoužívanější technologie a mají budoucnost.

Téma Piškvorky jsem si vybral z více důvodů. Mělo by na něm jít perfektně předvést alespoň základy obou z frameworků. Tato hra by měla být každému povědomá a nemusím se zabývat vysvětlováním pravidel.

1 Analýza obdobných aplikací

Analýzu obdobných aplikací je dobré provádět, abyste získali představu, jak má vaše aplikace vypadat. Co chcete, aby uměla a v čem byla lepší než ostatní aplikace.

1.1 turtlediary

Adresa: <https://www.turtlediary.com/game/tic-tac-toe-multiplayer.html>

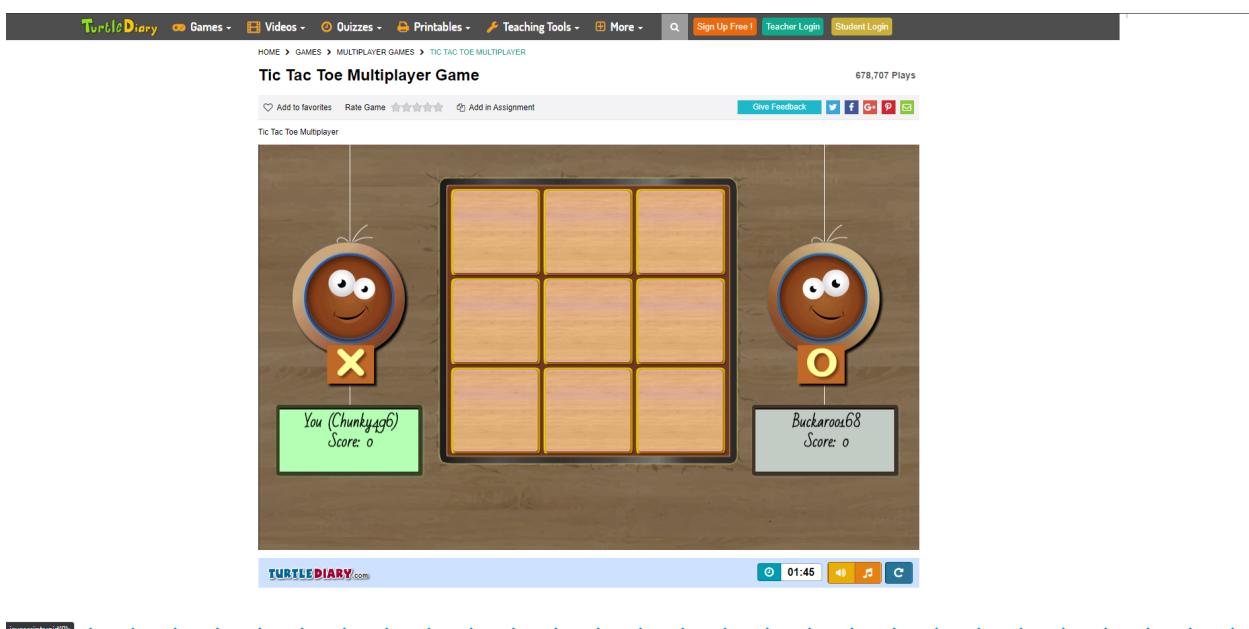
Web nabízí možnost hrát piškvorky 3x3 s náhodnými lidmi, nebo s kamarádem.

1.1.1 Kladné stránky

- ke hraní není potřeba registrace
- vizuální rozhraní hry

1.1.2 Záporné stránky

- web není responzivní
- hrací plocha je moc malá oproti zbylému volnému místu
- web nemá statistiky nebo výpis nejlepších hráčů
- na mobilním telefonu není hrací plocha vidět celá



Obrázek 1: turtlediary - <https://www.turtlediary.com/game/tic-tac-toe-multiplayer.html>

1.2 Ultimate Tic Tac Toe

Adresa: <https://ultimate-t3.herokuapp.com>

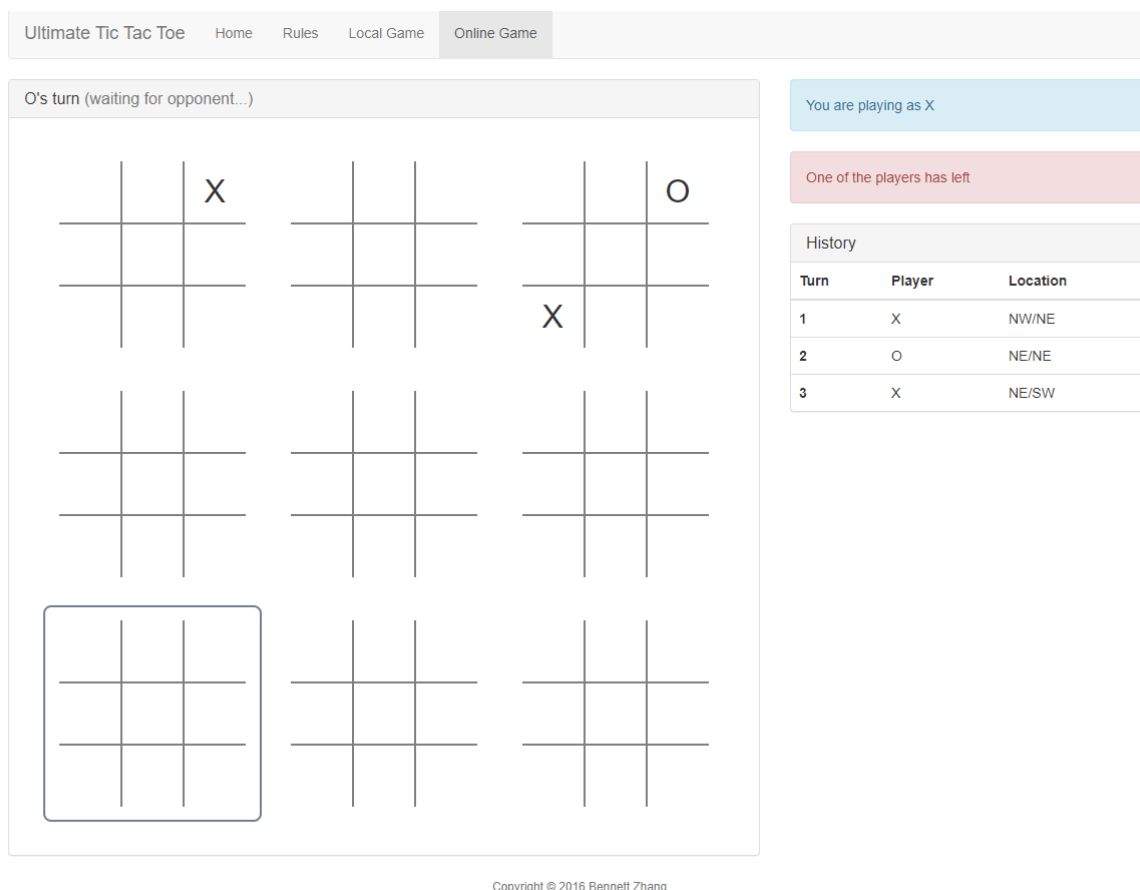
Web nabízí možnost hrát piškvorky 3x3 na více polích s kamarádem přes internet nebo lokálně.

1.2.1 Kladné stránky

- ke hraní není potřeba registrace
- čistý interface

1.2.2 Záporné stránky

- více hracích ploch
- web nemá statistiky nebo výpis nejlepších hráčů



Obrázek 2: Ultimate Tic Tac Toe - <https://ultimate-t3.herokuapp.com>

1.3 gametable

Adresa: <https://gametable.org/games/tic-tac-toe>

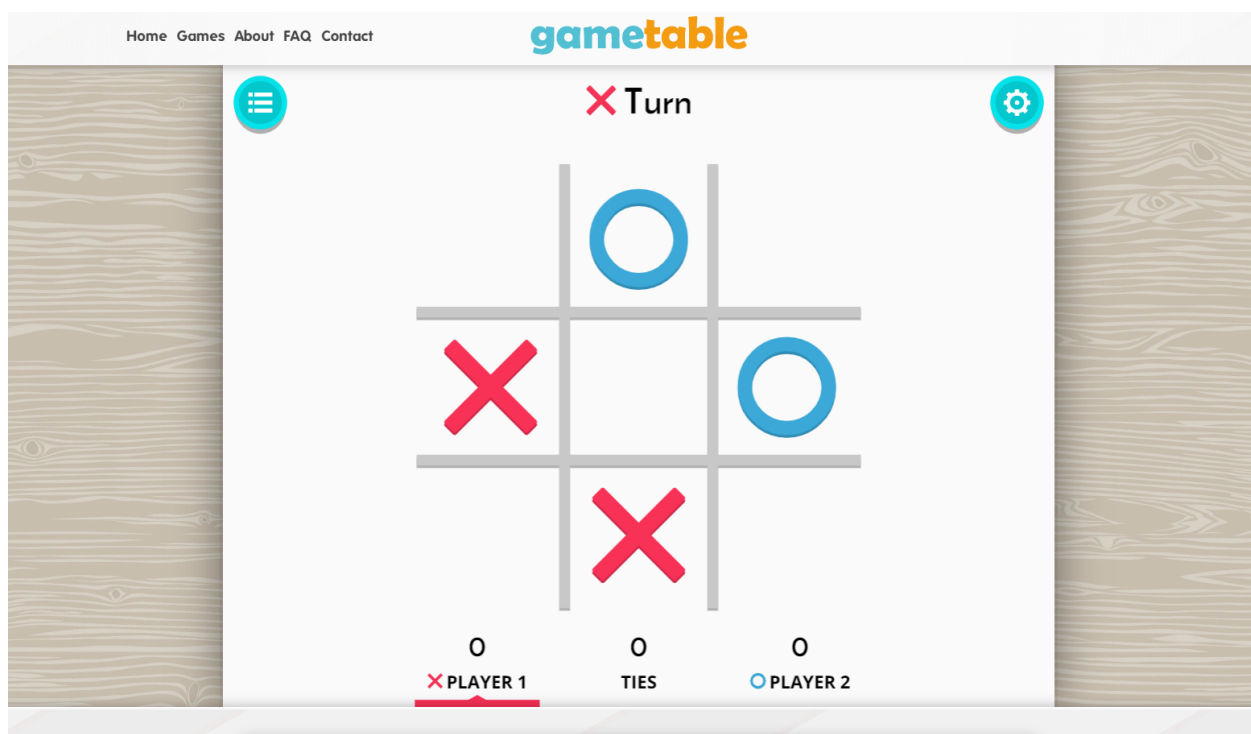
Web nabízí možnost hrát piškvorky 3x3 s kamarádem lokálně nebo proti AI.

1.3.1 Kladné stránky

- ke hraní není potřeba registrace
- čistý interface
- hra má vysvětleny pravidla pod hrací plochou

1.3.2 Záporné stránky

- není možnost hrát s někým přes internet
- web nemá statistiky nebo výpis nejlepších hráčů



Obrázek 3: gametable - <https://gametable.org/games/tic-tac-toe>

2 Návrh projektu

2.1 Obecná struktura

Veškeré požadavky, které nebudou odkazovat na front-end zodpovídá API. API je poté napojené na databázi a redis.

Front-end nemá sám o sobě žádný přístup k databázi nebo redisu.

2.2 API

API spojuje vše dohromady. Poskytuje veškeré informace front-endu a zpracovává veškeré příchozí informace.

2.3 Redis

V redisu jsou ukládány uživatelské relace, aby k nim byl rychlý přístup a jejich ověření.

2.4 Databáze

Databáze ukládá informace o uživateli a hrách pro dlouhodobé uložení dat.

Diagram fig. 4

2.5 Front-end

Slouží jako grafické zobrazení dat a druhořadá kontrola dat. Zobrazuje informace o uživateli a hrách.

3 Zpracování praktické části

3.1 Back-end

3.1.1 Správa uživatelů

Uživatelé jsou umístěni v tabulce *users*. Role jsou uloženy v tabulce *roles* a jsou k uživatelům přiřazovány skrz tabulku *roles_to_users*.

Pokud má uživatel roli *Admin* tak mohou upravovat kohokoli údaje bez omezení včetně rolí a hesla.

Pokud má uživatel roli *Banned* tak nemůže vytvářet pozvánky, ani nemůže být zahrnut do pozvánky jiným uživatelem.

3.1.2 Vytváření žádostí o hru

Přihlášení uživatelé mají možnost vytvářet nové hry s různými parametry.

Nejprve se vytvoří žádost o hru, která se nachází v tabulce *game_requests*. K dané žádosti na hru se přiřadí uživatelé skrz tabulku *users_to_game_requests*.

Pozvánky nemohou vytvářet uživatelé s rolí *Banned*.

3.1.3 Vytvoření hratelné hry

Po vytvoření žádosti o hru jí musí všichni hráči potvrdit a hra bude vytvořena, nebo někdo z pozvaných hráčů odmítne žádost a žádost o hru bude vymazána.

Jakmile je účast všech hráčů potvrzena, tak se vytvoří nová hra v tabulce *games*, přiřadí se k ní uživatelé skrz tabulku *games_to_users* a žádost o hru je poté vymazána.

3.1.4 Hraní hry

Hrát můžete jen když jste na tahu a pokud hrané políčko ještě nebylo použito. Vyhraní hry se kontroluje na front-endu, pokud front-end usoudí, že hráč vyhrál tak výhru oznámí back-endu a ten výhru zkontroluje.

Back-end kontroluje, jestli je hráč na tahu, jestli hra neskončila, nebo jestli jeho tah je validní. V případě, že hráč ohlásí výhru, ale server zjistí, že lže, tak daný tah

zahodí a odpoví chybou.

3.2 Front-end

3.2.1 Registrace

Všechna pole jsou kontrolována. Pokud nějaké pole není validní, tak se nepošlou data na back-end.

Back-end data zkontroluje a pokud zjistí, že nejsou validní, tak žádost zahodí a vrátí chybu.

Pro hashování hesla se používá 128 znaková sůl a algoritmus Argon2.

3.2.2 Profil

Zobrazuje informace o uživateli a hry, ve kterých se nachází.

Zobrazuje také počet výher, proher a remíz. Z těchto dat poté vypočítá winrate (výhry / prohry).

Pokud je uživatel na svém profilu, nebo pokud má uživatel roli *Admin*, tak se mu také zobrazí tlačítko na upravení profilu.

3.2.3 Výpis uživatelů

Zobrazuje všechny registrované uživatele a pár informací o nich.

Uživatelům s rolí *Admin* se navíc zobrazuje tlačítko upravení profilu.

3.2.4 Výpis her

Zobrazuje všechny rozehrané, nebo dohrané hry s jejich hráči.

3.2.5 Hraní hry

Hry jsou hrány na síti 30x30.

Uživatelé jsou zobrazováni s jejich symbolem a za jménem je napsáno, jestli jsou na tahu.

Hrát můžou jen uživatelé, kteří jsou v dané hře, ale dívat se může kdokoli.

Tahy uživatelů jsou kontrolovány, jestli jsou validní a jestli nastala výhra nebo remíza.

3.2.6 Výpis pozvánek

Zobrazuje název pozvánky (později název hry), počet tahů k vítězství.

Uživatel může pozvánku přijmout, nebo odmítnout.

3.2.7 Vytváření pozvánky

Při vytváření pozvánky jsou skoro všechna pole kontrolována.

Uživatelé s rolí *Admin* mají práva na vypnutí skoro všech kontrol.

3.2.8 Úprava uživatele

Uživatel může upravovat vše, kromě jeho rolí.

Uživatel s rolí *Admin* může upravovat vše a má možnost vypnout kontrolu, která je vyžadována po ostatních uživateli.

4 Závěr

5 Seznam použité literatury a zdrojů informací

6 Seznam použitých zkratek

Tabulka 2: Seznam použitých zkratek

Zkratka	Význam
API	Application Programming Interface
REST	Representational State Transfer
SPŠE	Střední průmyslová škola elektrotechnická
VOŠ	Vysoká odborná škola

7 Seznam obrázků, tabulek, příloh

Seznam obrázků

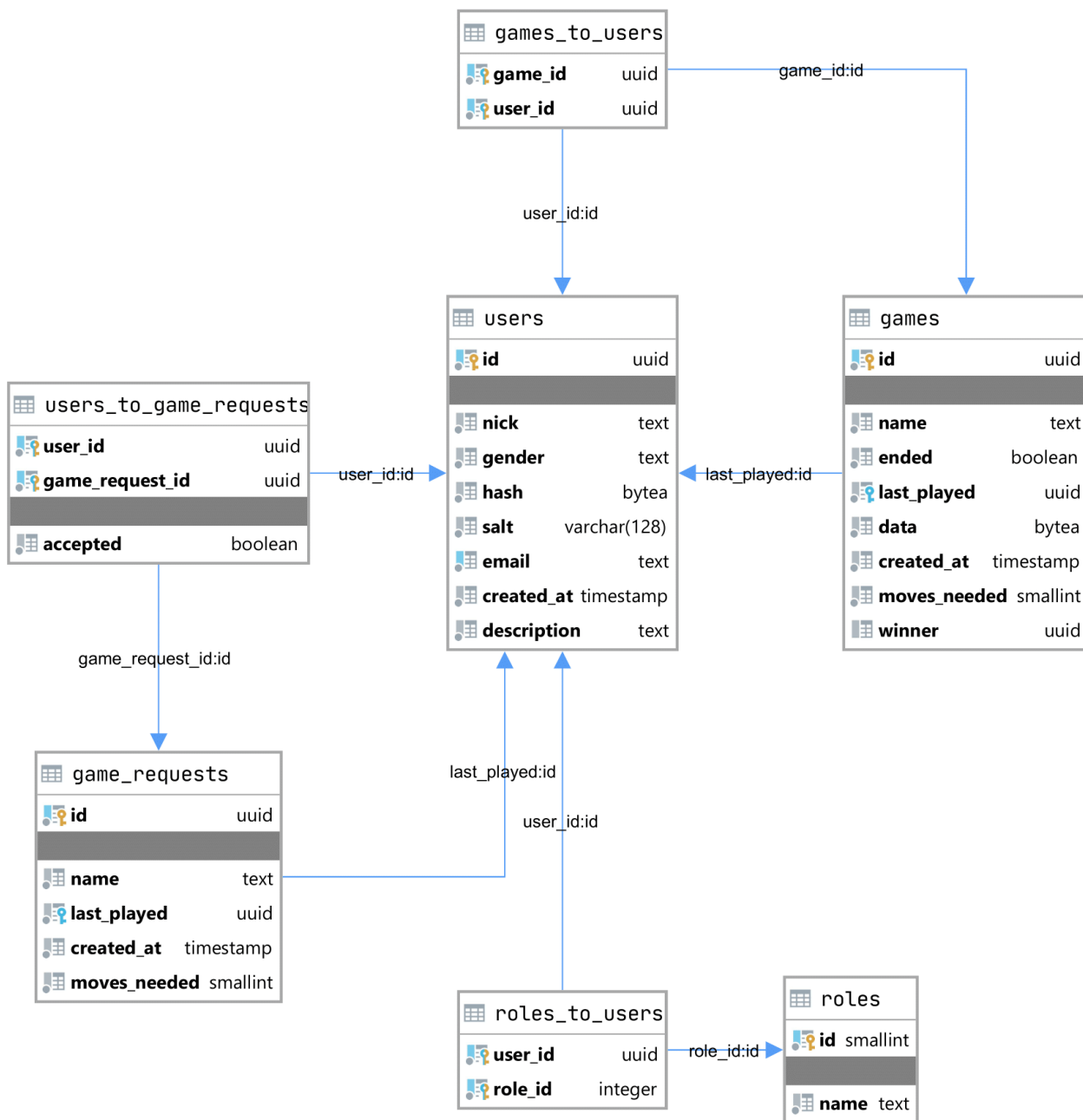
1	turtlediary - https://www.turtlediary.com/game/tic-tac-toe-multiplayer.html	10
2	Ultimate Tic Tac Toe - https://ultimate-t3.herokuapp.com	11
3	gametable - https://gametable.org/games/tic-tac-toe	12
4	ER Diagram	21

Seznam tabulek

1	Porovnání rychlosti jazyků	9
2	Seznam použitých zkratk	19

Seznam příloh

8 Přílohy



Obrázek 4: ER Diagram