

Sem vložte zadání Vaší práce.



ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE  
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ  
KATEDRA SOFTWAREVÉHO INŽENÝRSTVÍ



Bakalářská práce

## **Webová aplikace pro správu úkolů**

*Jiří Ketner*

Vedoucí práce: doc. RNDr. Ing. Marcel Jiřina, Ph.D.

7. ledna 2016



---

## Poděkování

Rád bych poděkoval vedoucímu práce doc. RNDr. Ing. Marcelu Jiřinovi, Ph.D. za cenné rady při psaní mé bakalářské práce. Rovněž bych chtěl poděkovat rodičům za jejich trpělivost a přítelkyni Bc. Kláře Bednařkové za podporu, bez které bych práci jen stěží dokončil.



---

## Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval(a) samostatně a že jsem uvedl(a) veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů, zejména skutečnost, že České vysoké učení technické v Praze má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona.

V Praze dne 7. ledna 2016

.....

České vysoké učení technické v Praze

Fakulta informačních technologií

© 2016 Jiří Ketner. Všechna práva vyhrazena.

*Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí, je nezbytný souhlas autora.*

### **Odkaz na tuto práci**

Ketner, Jiří. *Webová aplikace pro správu úkolů*. Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2016.



---

# Abstrakt

Hlavním cílem této bakalářské práce je implementace webové aplikace pro správu úkolů. Byla provedena rešerše pojmu gamifikace a průzkum současných aplikací na trhu. Získané informace byly využity při tvorbě konceptu vlastní gamifikované aplikace. Ta byla po navržení databáze a vytvoření diagramu nasazení implementována pomocí volně dostupných softwarových nástrojů. Na závěr bylo provedeno několika uživateli testování funkčnosti a uživatelské přívětivosti aplikace. Součástí práce je instalační příručka.

**Klíčová slova** webová aplikace, správa úkolů, projektové řízení, gamifikace, online hra, MEAN stack

---

# Abstract

The main objective of this Bachelor thesis is the implementation of web application for task management. Recherche of the term gamification and survey of the current applications on the market were made. Obtained information were utilized for the concept creation of own gamified application. It was implemented using freely available software tools after database design and deployment diagram creation. Function and user friendliness testing of the application were made by several users at the end. A part of the work is an installation guide.

**Keywords** web application, task management, project management, gamification, online game, MEAN stack

---

# Obsah

Úvod	1
<b>1 Cíl práce</b>	<b>3</b>
<b>2 Gamifikace</b>	<b>5</b>
2.1 Motivace hrou . . . . .	5
2.2 Definice gamifikace . . . . .	6
2.3 Historie pojmu . . . . .	7
2.4 Herní mechanismy . . . . .	7
<b>3 Analýza koncepčně odlišných webových aplikací</b>	<b>13</b>
3.1 Postup zkoumání aplikací . . . . .	13
3.2 Hlavní koncepty . . . . .	14
3.3 Shrnutí . . . . .	18
<b>4 Návrh vlastní aplikace</b>	<b>21</b>
4.1 Výběr konceptu . . . . .	21
4.2 Koncept aplikace . . . . .	21
4.3 Databázový model . . . . .	30
4.4 Diagram nasazení . . . . .	32
<b>5 Implementace</b>	<b>33</b>
5.1 MEAN.JS . . . . .	33
5.2 Administrační část aplikace . . . . .	36
5.3 Herní část aplikace . . . . .	37
5.4 Procedury spouštěné při interakci hráče . . . . .	39
5.5 Procedury spouštěné serverem automaticky . . . . .	40
<b>6 Testování</b>	<b>43</b>
6.1 Typické profily uživatelů ve skupině . . . . .	43

6.2	Testovací scénáře . . . . .	43
6.3	Testování funkčnosti . . . . .	44
6.4	Testování uživatelské přívětivosti . . . . .	45
6.5	Testování aplikace online . . . . .	46
<b>7</b>	<b>Výhled do budoucna</b>	<b>47</b>
	<b>Závěr</b>	<b>49</b>
	<b>Použité zdroje</b>	<b>51</b>
<b>A</b>	<b>Seznam použitých zkratk a pojmů</b>	<b>57</b>
<b>B</b>	<b>Instalační příručka</b>	<b>59</b>
B.1	Instalace Node.js verze 0.12.2 . . . . .	59
B.2	Instalace MongoDB verze 3.2 . . . . .	60
B.3	Import databáze . . . . .	60
B.4	Spuštění aplikace . . . . .	61
<b>C</b>	<b>Obsah příloženého CD</b>	<b>63</b>

---

## Seznam obrázků

2.1	Graf věku a pohlaví hráčů dle [1] . . . . .	5
2.2	Cyklus sociálního zapojení dle [2] . . . . .	11
4.1	Diagram návrhu databáze vytvořený v [3] . . . . .	31
4.2	Diagram nasazení vytvořený v [3] . . . . .	32
5.1	Náhled administrační i hráčské části aplikace . . . . .	36
5.2	Seznam úkolů (s hláškami o jejich stavu) v administraci . . . . .	37
5.3	Dialogové okno kasáren pro verbování jednotek . . . . .	38
5.4	Mapa světa . . . . .	39



---

# Úvod

Moderním termínem dnešní doby je prokrastinace. Lidé velmi často odkládají své úkoly a povinnosti na pozdější dobu. Místo toho vykonávají jim příjemnější, avšak méně důležité činnosti. Většinou to končí stresem a vyčerpáním z nevyspaní, když se snaží vše na poslední chvíli dohnat a dokončit. Výsledek takové práce často ani nedosahuje očekávaných kvalit.

Nabízí se otázka, jak nebo čím lidi motivovat, aby svoji práci dělali včas, precizně a efektivně. Jedním ze současných přístupů k tomuto problému je motivace hrou neboli gamifikace. Prostřednictvím gamifikace se snažíme spojit příjemné s užitečným. Vyvoláme v lidech touhy po úspěchu a vítězství, kterých dosáhnou v případě, že si splní svoje pracovní povinnosti.

Náš přístup k problému však nebude spočívat pouze v tom, že rozdáme uživatelům aplikace odznáčky a body a seřadíme je do žebříčku. Vytvoříme webovou aplikaci pro správu úkolů, ve které si uživatelé opravdu zahrají. Zahrají si reálnou počítačovou hru, ovšem pouze tehdy, budou-li dostatečně aktivní a splní-li tolik úkolů, kolik budou moci.

V bakalářské práci se nejdříve seznámíme s pojmem gamifikace a prozkoumáme několik webových aplikací pro správu úkolů. Získané informace využijeme při tvorbě konceptu vlastní aplikace. Následně provedeme návrh a implementaci a na závěr aplikaci necháme otestovat uživateli.





---

## Cíl práce

Jak již bylo zmíněno v úvodu, cílem práce je seznámit se s pojmem gamifikace, prozkoumat tento trend a navrhnout a implementovat webovou aplikaci, která bude využívat gamifikačních prvků. Aplikace si klade za cíl motivovat uživatele při vykonávání seriózních pracovních úkolů a má být implementována pomocí volně dostupných softwarových nástrojů.

Nejprve se blíže zaměříme na gamifikaci a její definice. Následně provedeme rešerši hlavních prvků, o které se gamifikace opírá, konkrétně se jedná o body, úrovně, žebříčky, odznáčky, nástup a adaptaci, výzvy a úkoly a cyklus sociálního zapojení.

V dalším kroku prozkoumáme několik koncepčně odlišných webových aplikací pro správu úkolů. Cílem práce není provést rešerši těchto aplikací, nýbrž získat přehled o použitých konceptech. Poté zvolíme jeden koncept, na jehož základě postavíme aplikaci vlastní.

Vybraný koncept rozšíříme o naši herní část a provedeme návrh databáze. Ještě než se vrhneme na implementaci aplikace, vytvoříme si diagram nasazení.

Předposledním krokem je samotná implementace aplikace. Představíme volně dostupné softwarové nástroje, které použijeme a na obrázcích ukážeme některé části aplikace. Zmíníme rovněž některé implementační postupy.

Na závěr aplikaci podrobíme důkladnému testování. Otestujeme jak funkčnost, tak uživatelskou přívětivost. Vybereme si několik testujících, kteří budou aplikaci procházet dle zadaných scénářů. Z pozorování a konzultací vyvodíme výsledky a závažné chyby aplikace následně opravíme.



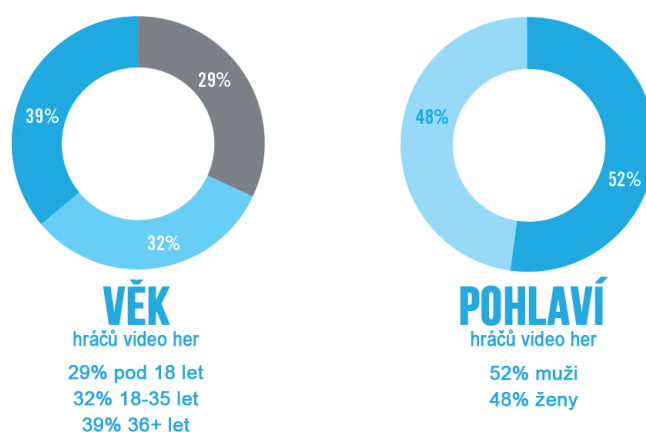
## Gamifikace

V této kapitole bakalářské práce vysvětlíme, co se skrývá pod pojmem gamifikace a zaměříme se na hlavní prvky, o které se opírá.

### 2.1 Motivace hrou

Že lze lidské zdroje motivovat k různým úkolům pomocí hry, není snad pro nikoho žádnou novinkou. Vždyť už pedagogové na základních školách se mnohdy snaží postupovat metodikou „škola hrou“.

Každý si rád hraje. V nedávném průzkumu z roku 2014 [1] se uvádí, že 59% Američanů hraje video hry. Předpokladem pro úspěšné motivování pomocí hry je samozřejmě fakt, že cílové publikum hraje hry. V průzkumu se uvádí, že průměrný věk hráče je 31 let a pohlaví je v poměru téměř 50:50. Z obrázku 2.1 tedy plyne, že aplikovat herní principy i v prostředí jiném než školním může být, co se motivace týče, správnou volbou.



Obrázek 2.1: Graf věku a pohlaví hráčů dle [1]

Dle [4] hru definují následující vlastnosti: cíl, pravidla, systém zpětné vazby a dobrovolná účast. Cíl definuje, o čem hra je, její účel. Pravidla nastavují omezení v tom, jakým způsobem mohou hráči daného cíle dosáhnout. Systém zpětné vazby udržuje hráče neustále informované o tom, jak si vedou a zobrazuje jejich pokrok při plnění cíle. A konečně dobrovolná účast znamená, že hráči rozumí a ochotně přijímají cíl hry, její pravidla a systém zpětné vazby. Při pohledu na tyto čtyři vlastnosti můžeme říci, že kombinace prvních tří vytváří vlastnost čtvrtou. Pokud hru chápeme tímto způsobem, můžeme v souladu s [4] model přenést také do řízení jakékoliv společnosti:

### **Cíle**

Všichni zaměstnanci se podílejí na stanovování cílů společnosti a mají osobní i finanční podíl na jejich splnění.

### **Pravidla**

Zaměstnanci chápou pravidla, která řídí operace společnosti; rozumí, co je a není správné a ví, jak získat body.

### **Systém zpětné vazby**

Všichni se učí číst finanční výkazy a vědí, jak si vede společnost i oni sami individuálně; vidí výsledky svých činů v reálném či téměř reálném čase.

### **Dobrovolná účast**

Zaměstnanci se účastní školení, která potřebují k odvádění dobré práce. Dle zájmu mají možnost se přesunout na jiné pozice ve společnosti podle toho, jak rozvinou své dovednosti a prokážou příslušné kompetence.

Prosadit herní myšlenky a techniky můžeme jak v práci, tak téměř v kterékoliv jiné oblasti. Uplatňováním her v neherním prostředí se zabývá gamifikace.

## **2.2 Definice gamifikace**

Gamifikaci mohou různí lidé chápat různým způsobem. Někteří tak označují vytváření her, které slouží výslovně k propagaci výrobku nebo služby. Jiní si myslí, že gamifikace s pomocí 3D virtuálních světů může ovlivňovat chování hráčů nebo je učit novým schopnostem (např. letecké či automobilové simulátory, vojenské simulátory, simulátory lidských operací). Obě myšlenky jsou správné. Gamifikace sdružuje a používá vše, co se her týká, v neherních oblastech [5].

Uvedme několik mírně se lišících definic pojmu:

- Analytická společnost Gartner definuje gamifikaci jako využívání herních mechanismů (a z nich plynoucích zážitků) ke stimulaci osob, aby dělaly žádoucí věci. Klíčem je zasáhnout lidi na úrovni emocí. Gamifikace by jim měla pomoci, aby plnění firemních cílů považovali za plnění vlastních cílů a tím se s nimi ztotožnili [6].
- Integrace mechanismů, které dělají hru zábavnou, do neherních platforem s cílem vtáhnout a zapojit uživatele do systému [7].
- Ve svém jádru je gamifikace aplikování mechanismů her do neherních aktivit za účelem změny lidského chování. Při použití v kontextu byznysu se jedná o proces integrace herních mechanismů do webové stránky, obchodní služby, online komunity, obsahového portálu nebo marketingové kampaně s cílem vtáhnout a zapojit uživatele do systému [8].

Pro náš cíl, vytvořit webovou aplikaci pro správu úkolů, si vydedukujeme definici, že gamifikace slouží k vtáhnutí a zapojení uživatelů do řešení problémů za pomoci herního myšlení a herních mechanismů.

## 2.3 Historie pojmu

Podívejme se nyní stručně na historii termínu. V září 2010 překračuje pojem gamifikace kritickou hranici v počtu vyhledávání a poprvé se objevuje ve službě Google Trends [9], avšak vymyšlen byl o několik let dříve.

Pojem vymyslel v roce 2002 britský konzultant Nick Pelling jako „záměrně ošklivé slovo“ popisující použití herního uživatelského rozhraní, které zrychluje a zpříjemňuje elektronické transakce. Pellingovo pojetí gamifikace se orientovalo zcela na hardware. Kládl si za cíl transformovat elektronická zařízení do zábavných (herních) platforem. Krátce nato založil konzultační společnost Comundra Ltd., nicméně se potýkala s nezájmem zákazníků a Pelling byl nucen činnost společnosti ukončit [10].

Termín gamifikace však pád konzultační společnosti přežil, ačkoliv dnes popisuje něco zcela odlišného. Jedním z prvních, kdo se začal implementací gamifikace zabývat, byl Dennis Crowley, tvůrce aplikace Foursquare [11]. V ní jsou, stručně řečeno, uživatelé odměňováni za navštívení různých míst po celém světě.

## 2.4 Herní mechanismy

Mechanismy gamifikované aplikace jsou tvořeny z řady nástrojů, které slibují při správném způsobu použití smysluplnou odezvu od hráčů. Zaměříme se především na sedm primárních prvků dle [5]: body, úrovně, žebříčky, odznáčky, výzvy a úkoly, nástup a adaptace a cyklus sociálního zapojení.

### 2.4.1 Body

Body jsou nutnou součástí každé gamifikované aplikace, jsou ukazatelem pokroku hráčů. Podle [5] existuje několik typů bodovacích systémů:

#### **Zkušenostní body**

Z pěti druhů bodových systémů jsou nejdůležitější. Většinou se značí zkratkou XP. Znázorňují to, jak návrhář systému na hráče nahlíží, jak jejich postup hodnotí a jakým směrem je chce vést. Hráči získávají body za každou správnou akci v aplikaci. Obecně zkušenostní body neklesají a nelze si je koupit. Nikdy by neměly dosáhnout stropu.

#### **Platební body**

Jedná se o druh bodů, které mohou na účtech hráčů kolísat. Typicky to jsou různá platidla jako mince, peníze nebo suroviny. Ty lze v rámci aplikace směnit za různé jiné věci nebo ocenění. Systém platebních bodů musí být správně nastaven a otestován, aby se zabránilo inflaci nebo deflaci.

#### **Dovednostní body**

Dovednostní body jsou přiřazeny specifickým činnostem v aplikaci. Trénováním dané činnosti se v ní hráč zlepšuje a postupně se mu odemykají nové možnosti.

#### **Karma body**

Se systémem karma bodů se můžeme setkat jen zřídka. Jejich účelem je rozdat je ostatním hráčům, prokázat jim laskavost nebo jim poděkovat. Hráči nemají žádnou výhodu z jejich držení.

#### **Body reputace**

Kdykoliv je potřeba zobrazit informaci o důvěryhodnosti, kterou nelze explicitně zaručit, mezi více stranami, přichází na řadu reputační body. S reputačními body se setkáváme na různých inzertních serverech, kdy znázorňují důvěryhodnost prodejců i kupujících.

### 2.4.2 Úrovně

Úrovně ve většině aplikací symbolizují hráčův pokrok. V některých aplikacích může úroveň definovat obtížnost hraní, v jiných může sloužit pouze jako pasivní ukazatel aktivity či kvality hráče. Systém úrovní v gamifikovaném prostředí by měl být jednoduše rozšiřitelný.

### 2.4.3 Žebříčky

Žebříčky slouží k jednoduchému účelu - porovnávání hráčů.

Podle [5] se dnes můžeme setkat se dvěma typy žebříčků:

#### **Bez hráčovy pozice**

Druh žebříčku, který se v poslední době uchytil díky sociálním sítím. Hráč je umístěn na neznámé pozici někde uprostřed seznamu. Pod sebou vidí přátele s nižším skóre a nad sebou vidí nejbližšího soupeře k pokoření. Pokud je ovšem hráč na předních pozicích žebříčku (např. v nejlepší dvacíctce), je žádoucí, aby mu jeho pozice byla zobrazena.

#### **Nekonečný seznam**

Klasický žebříček všech hráčů v aplikaci, který může být rozdělen na různé sekce (dle úrovní, periody, přátel apod.).

### **2.4.4 Odznáčky**

Hráči touží po zisku různých druhů odznáčků. Mnoho z nich trpí neukojitelnou sběratelskou touhou. Čím vzácnější odznáček těmto hráčům nabídneme, tím větší motivaci k jeho získání budou mít. I ostatním hráčům uděláme radost, když jim nečekaně darujeme odznáček za nějakou jejich aktivitu v aplikaci.

Model udělování odznáčků může být jak transparentní, tak i netransparentní. V transparentním modelu každý hráč ví, za co jaký odznáček může získat. V netransparentním je tomu naopak. Hráč dopředu není obeznámen s tím, kdy a za co odznáček dostane. Druhý model činí problém právě hráčům majícím sběratelskou potřebu. Ti se mohou kvůli nevědomosti cítit frustrovaně, proto je vhodné použití druhého modelu zkombinovat s prvním.

### **2.4.5 Výzvy a úkoly**

Výzvy a úkoly říkají hráčům gamifikované aplikace, co mají dělat (poražte hráče v žebříčku nad vámi, vycvičte deset dostihových koní, přihlaste se do aplikace z místa v okolí hory Říp aj.) a jakým způsobem mají trávit čas. Většinou je úkolem seznámit hráče s principy fungování a posléze je vést k cíli. Často je vše protkáno příběhem, který hráčům dodá pocit, že jsou součástí něčeho většího, a který je zasáhne na emocionální rovině. Za splnění výzvy nebo úkolu jsou hráči právem odměněni odznáčkem nebo určitým množstvím zkušenostních bodů.

### **2.4.6 Nástup a adaptace**

Jedná se o uvedení nováčka do gamifikované aplikace. Právě v úvodu se hráč rozhoduje, zda u aplikace setrvá či nikoliv. A ačkoliv může mít systém rostoucí tendenci, co se zábavy týče, odradí-li hráče hned zpočátku, zpět ho většinou nepřiláká [12].

Základní chybou mnoha aplikací je podle [5] to, že nutí hráče k registraci dříve, než mu dají možnost vůbec něco prozkoumat. Ten jen stěží vloží osobní

informace někam, o čem ještě neví téměř nic. Důležité je mu nabídnout něco hodnotného (odznáček, virtuální věc, balíček surovin aj.), co jej nabudí k dalším krokům. Jedinou výjimkou, kdy aplikace může vyžadovat registraci, je v případě, že přihlášení probíhá prostřednictvím Facebooku.

Dále je v souladu s [5] nutné nastavit novému hráči přímočarý postup (pomocí úkolů) a nedat mu možnost žádné jiné volby. Doporučuje se následující základní vzor:

- Akce hráče
- Odměna
- Akce hráče
- Akce hráče
- Odměna
- Registrace
- Pozvání přátel

### 2.4.7 Cyklus sociálního zapojení

Návrhář gamifikované aplikace nemá na starosti sledovat pouze to, jak do ní hráči nastoupí a adaptují se, ale i to, jak ji opouštějí nebo co je přivádí zpět. Na obrázku 2.2 je znázorněn koncept cyklu, kdy motivační emoce vede k sociální výzvě k akci, ta vede k hráčově znovu-zapojení, z čehož pramení nějaká odměna.

Jednotlivé prvky cyklu si osvětlíme na příkladu aplikace Instagram:

#### **Motivační emoce**

Sdílet vlastní fotografii, která se bude líbit spoustě uživatelům.

#### **Sociální výzva k akci**

Sdílení fotografie, která vyzývá k akci ostatní uživatele (komentář, „lajk“).

#### **Znovu-zapojení hráče**

Návrat do aplikace, poděkování konkrétním uživatelům za kladné ohlasy na fotografii.

#### **Viditelný postup (odměna)**

Nárůst počtu odběratelů a „lajků“, počet kladných komentářů.





Obrázek 2.2: Cyklus sociálního zapojení dle [2]



---

## Analýza koncepčně odlišných webových aplikací

V této části bakalářské práce se podíváme na několik webových aplikací pro správu úkolů, z nichž vytvoříme přehled odlišných konceptů. Aplikace jsme vybrali na základě tipů z internetových diskuzí, výsledků vyhledávání Google a žebříčku nejoblíbenějších aplikací se štítkem „project-management“ v alternativeTo [13].

### 3.1 Postup zkoumání aplikací

Prozkoumali jsme celkem šestnáct webových aplikací pro správu úkolů. Každou z nich jsme otestovali jak v roli správce projektu, tak v roli člena projektového týmu.

Hlavní scénář každého testování v roli správce projektu byl následující:

1. Vytvořit projekt
2. Pozvat ostatní členy týmu ke spolupráci
3. Vytvořit úkoly a přiřadit je pozvaným spolupracovníkům
4. Prohlédnout si systém zpětné vazby
5. Projít si systém notifikací

Následně jsme zkoušeli všechny ostatní funkce, které aplikace umožňují, jako např. vytváření diskuzí, komentování úkolů, přidávání souborů, vytváření milníků, importování údajů, nastavování oprávnění aj.

### 3. ANALÝZA KONCEPČNĚ ODLIŠNÝCH WEBOVÝCH APLIKACÍ

---

V roli člena týmu byl hlavní scénář průchodu aplikací odlišný, zaměřený na plnění úkolů:

1. Přihlásit se do aplikace
2. Najít si vlastní úlohy
3. Splnit nějaký úkol
4. Projít si systém notifikací

Poté jsme vyzkoušeli všechny ostatní možnosti systému, které měl člen projektového týmu dostupné, např. přidávání úkolů, zaznamenávání času při práci na úlohách, přidávání komentářů, vytváření diskuzí, přidávání souborů aj.

Jakmile jsme tímto způsobem prozkoumali všechny vybrané aplikace, provedli jsme jejich rozdělení do tří skupin podle použitých konceptů:

- Robustní, složité aplikace pro projektové manažery
- Intuitivní, jednoduché aplikace pro nenáročného uživatele
- Seznamy úkolů s možností sdílení

## 3.2 Hlavní koncepty

V předchozí části jsme si dle použitých konceptů vytvořili tři skupiny aplikací pro správu úkolů. Nyní si tyto skupiny blíže osvětlíme a uvedeme na příkladech.

### 3.2.1 Aplikace pro projektové manažery

Již z názvu je patrné, na koho cílí aplikace této kategorie. Člověk se bez patřičných vědomostí ztrácí ve zmeti dostupných funkcí. Většinou jsou aplikace provozované v cloudu a jejich využívání je po ukončení zkušební doby zpoplatněno. Některé systémy obsahují i více komponent (agilní řízení projektů, sledování chyb, aj.) nebo jsou dále rozšiřitelné pomocí pluginů.

Základním stavebním kamenem je dashboard, který poskytuje informační přehled a rychlý přístup k nejdůležitějším funkcím systému. Na úrovni všech vytvořených projektů zobrazuje dashboard stručné informace o konkrétních projektech (název, počet členů v týmu, počet splněných/nesplněných úkolů, aj.) a umožňuje vytvořit nový či přejít na vybraný projekt. O úroveň níže se nám objeví kompletní informace o zvoleném projektu a množství funkcí s ním spojených.

Vytvoření nového projektu je jednoduché a intuitivní. Aplikace vyžadují vyplnit název a přizvat uživatele ke spolupráci (lze přeskočit a rozeslat pozvánky později). Volitelně můžeme vyplnit také popis, termín dokončení, štítky či status (vytvořený, probíhá, ukončený).

Po založení projektu se odemkne řada funkcí a možností, které s ním souvisí. Ve většině aplikací této kategorie lze:

- Nastavovat milníky projektu
- Vytvářet, upravovat či mazat úkoly nebo dílčí úkoly
- Přidávat komentáře k projektu či úloze, zakládat diskuze
- Zakládat osobní konverzace s uživateli
- Spravovat soubory projektu nebo úkolu
- Přizvávat nové uživatele či odstraňovat stávající
- Exportovat či importovat data
- Filtrovat zobrazení uživatelů či úloh
- Zobrazit a plánovat věci v kalendáři (včetně schůzek)

Mimoto disponují některé z aplikací také dalšími užitečnými funkcemi, které jistě mnoho projektových manažerů ocení, konkrétně zaznamenávání času, nastavení platů, správa rozpočtu projektu, přizvání klientů do systému nebo i tvorba faktur. V možnostech nastavení projektu stojí za zmínku volba oprávnění spolupracovníků, kde jim můžeme nastavit práva pro vytváření úloh, úpravu úkolů, přidávání souborů, vytváření diskuzí k projektu aj.

Člen projektového týmu má ve výchozím nastavení podobná práva jako manažer, nemůže však zakládat či upravovat projekty, spravovat uživatele ani označovat cizí úlohy za splněné. Pokud je v aplikaci implementováno sledování času, lze si stopovat dobu strávenou prací na úkolu, která se pak automaticky přepočítává na odměnu v závislosti na hodinové sazbě pracovníka. Během práce lze zaznamenávat míru splnění úkolu a následně jej zaškrtnutím označit za hotový (a podat hlášení).

Při přidávání nových úkolů je většinou povinné vyplnit pouze název, ovšem nikdy bychom tímto stylem nevyužili potenciál aplikací. Kromě názvu můžeme nastavit popis, prioritu, přílohy, termín splnění, řešitele, milník, dílčí úkoly, štítky, frekvenci opakování či zasílaná upozornění. U aplikací s možností sledování času vyplňujeme i odhadovaný čas pro vyřešení. Můžeme si následně prohlížet grafy nebo statistiky poměru odhadovaného času ku reálně strávenému času. Část aplikací v kategorii umožňuje plánovat úkoly také pomocí Ganttova diagramu.

Systém zpětné vazby je oblastí, kde se aplikace v kategorii mírně liší. Za základ bychom mohli označit sled aktivit, který obsahují všechny testované systémy. Jedná se o chronologický výpis všech událostí, které se odehrály. Zaznamenává se jak u projektů, tak u úkolů či uživatelů. Samozřejmostí je přehled všech úkolů v projektu pomocí seznamu i kalendáře a přehled úkolů

daného pracovníka. Poslední společnou věcí jsou notifikace. Ty rozlišujeme dvojího druhu: zabudované a emailové. Zabudované notifikace informují členy týmu o každé události v systému, emailové pouze o událostech, které se konkrétního člena týkají. Každý pracovník má navíc přístupnou i možnost přihlásit si emailové notifikace u úloh, na kterých se jako řešitel nepodílí. Některé aplikace ještě nabízí různé grafy a statistiky, např. graf vytvořených proti vyřešeným úkolům, atributů úkolů; statistiku průměrné odpracované denní doby pracovníka, vytížení pracovníka, nákladů projektu a jiné.

Z testovaných aplikací pod tento koncept zařadíme následující: JIRA [14], Podio [15], ONLYOFFICE [16], Asana [17], Active Collab [18], GroupCamp [19], Easy Project [20] a Wrike [21].

#### 3.2.2 Aplikace pro nenáročného uživatele

Další skupina aplikací cílí na uživatele bez velkých zkušeností s řízením projektů anebo na manažery malých týmů, kteří potřebují jen rozdělit úlohy mezi členy a mít přehled o tom, kdo na čem pracuje či pracoval. Funkcionalitu některých aplikací je možné rozšířit pomocí placených pluginů. Využívání těchto systémů je taktéž po vypršení zkušební lhůty zpoplatněno.

Po přihlášení v roli manažera nás některé aplikace přeměrují na uvítací obrazovku s navigační nabídkou, ze které se můžeme proklikat až k požadovaným oblastem. Jiné nás zavedou na dashboard s vytvořenými projekty, kde můžeme vytvořit samozřejmě i nové. Členové týmu se automaticky dostanou do projektu, na kterém pracují (pokud jich není více).

Vytvoření projektu je krátké a jednoduché, stačí jen vyplnit název a nepovinně popis a následně pozvat všechny spolupracovníky, kteří se na projektu budou podílet.

Jakmile máme založený projekt, zpřístupní se nám následující možnosti:

- Nastavení seznamů úloh nebo milníků
- Vytváření, mazání či úprava úkolů nebo dílčích úkolů
- Přidávání komentářů k úkolům
- Správa souborů projektu či úloh
- Přizvání dalších členů do týmu nebo smazání stávajících
- Zobrazení a plánování věcí v kalendáři

V některých aplikacích z kategorie můžeme také vytvářet diskuze, importovat či exportovat data, filtrovat výpisy úkolů nebo archivovat projekt či jeho úlohy. Za základní funkce skupiny ovšem považujeme ty uvedené v seznamu výše.

Správci projektu je umožněno kontrolovat práva členů týmu. Ti mají ve výchozím nastavení na starost vlastní úkoly (jimi zadané nebo jim přiřazené) a nemohou přidávat nové projekty ani zasahovat do jejich nastavení. Jejich možnosti lze dále omezit a můžeme jim například i znemožnit vytváření úloh nebo diskuzí.

Pro vytvoření nové úlohy musíme povinně vyplnit název. Chceme-li úkol blíže specifikovat, nabízí se nám možnost přidat ještě popis, termín splnění, řešitele, soubory nebo štítky. Vytvořený úkol je většinou součástí nějakého seznamu úloh nebo milníku, může však zůstat i nepřirazen.

Systém zpětné vazby aplikací této kategorie je prostý. Využívají chronologické sledy událostí. Ty vypisují veškerou aktivitu členů týmu (přidal úkol, splnil úkol, přidal komentář v diskuzi, aj.) nebo všechny události projektu (byl přidán úkol, byl splněn úkol, cíl milníku naplněn, aj.) společně s datem a časem, kdy k nim došlo. Mimoto se v roli správce projektu můžeme podívat samozřejmě i na kompletní seznam všech splněných, probíhajících i nesplněných úkolů daného pracovníka nebo projektu. Důležité informace a události v projektu se členové dozívají prostřednictvím emailových notifikací, občas se setkáváme i se zabudovanými notifikacemi v systému.

Do výše popsané kategorie řádíme následující testované aplikace: Basecamp [22], hiTask [23], Flow [24] a Freedcamp [25].

### 3.2.3 Seznamy úkolů s možností sdílení

Poslední skupinu tvoří aplikace, které sází na jednoduché seznamy úkolů. Ty je následně možné sdílet a spolupracovat tak na nich s ostatními uživateli systému. Aplikace jsou, až na výjimky, zdarma s možností připlatit si za různé nadstavbové funkce. Využití najdou jak u jednotlivců, tak u menších firem a projektových týmů.

Největší rozdíl aplikací v kategorii pozorujeme v odlišném zobrazení seznamů úloh. Setkali jsme se s následujícími typy:

- Prostý seznam úloh, který lze uspořádat metodou „chyť a pusť“
- Výpis úkolů podle nadcházejících dnů
- Tabule seznamů úloh, kterou lze uspořádat metodou „chyť a pusť“

Jakmile vytvoříme nějaký seznam úloh (zadáním názvu), můžeme jej označit jako sdílený a přizvat naše spolupracovníky rozesláním pozvánek. V případě tabule se mohou účastníci podílet na kterémkoliv seznamu úloh v ní obsaženém. Aplikace umožňují následující:

- Vytvořit seznam úloh nebo tabuli seznamů
- Vytvořit, smazat či upravit úkoly nebo dílčí úkoly

- Přidávat komentáře k úkolům
- Přizvat další členy týmu nebo odstranit stávající

Některé aplikace disponují i funkcionalitou kalendáře nebo možností přidávat soubory. Žádné nastavování oprávnění v neplacených verzích aplikací neexistuje. Znamená to, že každý člen má stejná práva na vytváření, mazání či úpravu úloh nebo seznamů jako správce projektu, nemůže akorát odstranit ostatní spolupracovníky.

Zadat novou úlohu můžeme do systému vyplněním jejího názvu, nicméně je vhodné ji upřesnit pomocí dalších atributů, jako jsou řešitel, termín, přílohy nebo dílčí úkoly. Označením dílčích úkolů za splněné se u nadřazené úlohy zobrazuje postup při plnění (pomocí procent nebo zlomků).

Systém zpětné vazby se v aplikacích této kategorie různí. Buď obsahují chronologický sled událostí u tabule úloh i členů projektového týmu zvláště, nebo zaznamenávají aktivity pouze globálně u vytvořeného seznamu úkolů (hůře se dohledávají činnosti konkrétního pracovníka). Důležité události jsou členům týmu oznamovány jak prostřednictvím systémových notifikací, tak prostřednictvím emailů.

Do kategorie řadíme následující testované aplikace: Trello [26], Wunderlist [27], Any.do [28] a Remember The Milk [29].

### 3.3 Shrnutí

Cílem kapitoly bylo získat přehled o použitých konceptech v oblasti správy úkolů. Po prozkoumání vybraných aplikací jsme došli k závěru, že se konceptuálně liší v závislosti na tom, pro jakou cílovou skupinu jsou určeny.

Robustní složité aplikace, které obsahují nespočet různých funkcí, grafů a statistik, využijí převážně projektoví manažeři s patřičnými znalostmi. Obyčejný člověk nebo nepříliš zkušený lídr týmu bude mít značný problém s porozuměním všem částem systému, zvláště pak všem grafům a statistikám, které lze generovat.

Kategorie aplikací pro nenáročného uživatele ovládá správu úkolů obdobně jako předešlá skupina, rozdíl vidíme v systému zpětné vazby, který je jednoduchý a méně bohatý na informace. Ani nezkušený uživatel nenarazí na věc, které by nerozuměl. Některé aplikace, např. Basecamp nebo Freedcamp, lze pomocí placených pluginů rozšířit takovým způsobem, že bychom je poté mohli zařadit do kategorie první. Možná díky této variabilitě je Basecamp třetí nejpoužívanější aplikací vůbec [30].

Poslední kategorii tvoří prosté seznamy úkolů, které lze sdílet s ostatními uživateli systému. Uplatnění najdou v malých týmech spolupracovníků nebo u jednotlivců. Systém zpětné vazby je až příliš chudý na to, aby byly aplikace užitečné také pro projektové manažery. Za nejpoužitelnější v tomto směru



považujeme aplikaci Trello, která obsahuje sled aktivit jak u celého projektu (tabule seznamů úloh), tak u jednotlivých uživatelů.



---

## Návrh vlastní aplikace

V předešlém textu jsme se nejprve seznámili s pojmem gamifikace a následně si vytvořili přehled používaných konceptů v oblasti aplikací pro správu úkolů. Nyní si vybereme jeden z konceptů, na jehož základě postavíme naši vlastní aplikaci obohacenou o prvky gamifikace.

### 4.1 Výběr konceptu

Základem naší aplikace bude koncept druhé skupiny aplikací. Proti první kategorii zaostává v systému zpětné vazby, naopak zase vyniká jednoduchostí a srozumitelností pro informačně gramotné uživatele. Zahrneme-li fakt, že v oblasti správy úkolů v konceptu nechybí nic podstatného, existuje zde dobrý předpoklad pro možná rozšíření systému. V budoucnu tak máme cestu k projektovým manažerům otevřenou. Koncept třetí skupiny jsme nevybrali z toho důvodu, že nám vadí absence milníků projektu a až na výjimky je systém zpětné vazby informačně velmi chudý.

### 4.2 Koncept aplikace

Koncept námi vyvíjené aplikace je obohacen o zábavnou (gamifikovanou) herní část. Co se terminologie týče, budeme v našem případě označovat spolupracovníky jako hráče, správce projektu jako administrátora a projekt jako hru.

Administrátor vytváří kategorie úkolů, k nimž přiřazuje profese zadané u hráčů. Při vytváření úkolu zvolí tuto kategorii, díky čemuž je úkol zobrazován na burze či v aukci pouze hráčům dané profese. Nestane se tak, že by např. sekretářka přijmula úkol pro programátory.

Hráči plní reálné pracovní úkoly, za které jsou následně v herní části aplikace odměněni zlatáky. Bez zlatáků není šance být ve hře úspěšný. Z herních mechanismů jsou v konceptu využity body, žebříčky a odznáčky.

### 4.2.1 Uživatelské role

#### Administrátor

- Zakládá a ruší hru.
- Přidává či odebírá hráče a přiděluje jim počáteční množství zlaťáků ve hře (může být různé).
- Přiřazuje hráčům profese, dle kterých mají či nemají přístupné určité kategorie úkolů.
- Vyvěšuje úkoly na tržiště či burzu.
- Upravuje úkoly, dokud se ještě nikdo nezabývá jejich řešením.
- Stanovuje pevnou periodu vyhodnocování hry pro udělení hráčských titulů.
- Vytváří kategorie úkolů.
- S ohledem na stav hry může úkoly hráčům také přiřazovat.
- Pokud je zadavatelem úkolu, má na starost jeho vyhodnocení.

#### Hráč

- Přijímá pozvání do hry.
- Volí si přezdívku ve hře a pozici vesnice na mapě světa.
- Plní stanovené úlohy získané na burze nebo na tržišti.
- Soutěží o úkoly na burze, vybírá si z úloh na tržišti.
- Za plnění úkolů získává zlaťáky, které využívá ve hře.
- Přidává úkoly na tržiště.
- Upravuje sebou zadané úlohy, dokud se ještě nikdo nezabývá jejich řešením.
- Pokud je zadavatelem úkolu, má na starost vyhodnocení jeho splnění.
- Nastavuje si zasílání upozorňujících emailů.
- Díky získaným zlaťákům může hrát hru a ucházet se o různá ocenění.

### 4.2.2 Cíl hry

Cílem každého hráče je být nejlepší. Toho docílí jedině tehdy, pokud bude plnit dostatek úkolů a získá dostatek zlatáků na vybudování svého vojska. Body za úspěšně zabité jednotky barbarů (nepřátelé hráčů) jsou společně se získanými zlatáky měřítkem pro porovnávání v žebříčku. V administrátorem stanovených periodách jsou udělována ocenění v podobě titulů (obdoba odznáčků) a zlatáků.

### 4.2.3 Založení nové hry

Založení nové hry provádí uživatel s rolí administrátora. Administrátor může založit a spravovat i několik her současně. U každé hry je nutné:

- Vyplnit název
- Vybrat mapu světa z předdefinované nabídky
- Zvolit periodu pro pravidelné a milníky pro nepravidelné vyhodnocování hry (nutný alespoň jeden milník pro konec hry)
- Nastavit počáteční hodnoty surovinových zdrojů
- Vyplnit emailové adresy hráčů a přiřadit jim jejich profesi

Hráče i milníky lze přidat samozřejmě i v průběhu hry.

### 4.2.4 Přidání úkolu

Poté, co je hra vytvořena, může do ní administrátor přidávat úkoly. U každého úkolu je nutné:

- Vyplnit název
- Zvolit či vytvořit kategorii úkolu, dle které bude viditelný pro konkrétní skupinu hráčů
- Vyplnit detailní popis
- Vyplnit měřitelné výstupy úlohy, dle kterých bude vyhodnoceno její splnění či nesplnění
- Zadat pevnou nebo vyvolávací cenu (tržiště nebo burza)
- Přiřadit úkol k milníku (volitelně)
- Zadat termín pro splnění úlohy
- Vytvořit podmínky penalizace za pozdní odevzdání
- Zvolit dobu nutnou pro vyhodnocení (po odevzdání splněné úlohy)

Přidávat úkoly mohou také hráči, ale pouze v případě, že mají aktivní nějaký jiný úkol, na kterém pracují (obdoba dílčích úkolů). Zadavateli je po vytvoření umožněno úkol upravovat až do doby, než jej jiný hráč přijme. Administrátor může dle svého uvážení vytvořený úkol přiřadit konkrétnímu uživateli, aby nebyl ohrožen průběh hry. O vyhodnocení odevzdávaného úkolu (mimo prostředí aplikace) se stará zadavatel, který má na to předem stanovenou dobu. Za úspěšné splnění je hráči připsána odměna ve formě zlaťáků.

### 4.2.5 Systém zpětné vazby

Administrátor má kompletní přehled o všech aktivitách v systému prostřednictvím časového sledu událostí, který je zaznamenáván jak u hry, tak u každého uživatele.

U hráčů se sledují a vypisují následující aktivity:

- Zaregistrování se do hry
- Přijmutí, výhra, přiřazení úkolu k vypracování
- Odevzdání, neodevzdání zadaného úkolu
- Splnění, nesplnění úkolu

U hry se kromě výše uvedených aktivit vypisují navíc ještě následující:

- Vytvořená, ukončená hra
- Vytvořený, expirovaný, smazaný úkol
- Zaregistrovaný, smazaný hráč
- Vytvořený, smazaný, splněný, nesplněný milník hry

Administrátor si ke všem uvedeným událostem může nastavit také emailovou notifikaci. Hráčům chodí emaily zcela odlišné, týkající se jejich aktivit v herní části aplikace:

- Zaregistrování nového hráče do hry
- Přidání nového úkolu do aukce či na burzu
- Překonání hráčova příhozu v aukci
- Výhra úkolu v aukci
- Vyhodnocení odevzdaného úkolu
- Připomínání neodevzdaného úkolu s blížícím se datem odevzdání

- Expirování úkolu
- Notifikace k vyhodnocení hráčem zadaného úkolu
- Hlášení o příchozím útoku na vesnici hráče
- Výsledky vyhodnocení hry po každé periodě
- Konec hry

### 4.2.6 Herní logika

Hráč si po obdržení emailu s pozvánkou zvolí svoji herní přezdívku a vybere si pozici na mapě pro svoji vesnici. Hraje se po periodách, jejichž délku stanovuje administrátor. Během periody si hráči vybírají úkoly na tržišti či burze, plní úkoly mimo prostředí hry, splněné úkoly odevzdávají zadavateli a od něj pak případně získávají zlatáky, za které mohou verbovat vojenské jednotky. Po každé periodě dochází k vyhodnocení a udělování titulů nejlepším hráčům.

#### 4.2.6.1 Vesnice

Na začátku hry dostane každý hráč svoji vlastní vesnici. Od okolního světa bude chráněna hradbami 1. úrovně a uvnitř budou defaultně všechny budovy na počátečních úrovních. Při kliknutí na vesnici se hráči zobrazí informace o daném hráči.

#### 4.2.6.2 Budovy

Ve vesnici lze vylepšovat různé typy budov, které slouží k produkci a uskladnění surovin, výcviku vojenských jednotek nebo ke zvýšení obrany vesnice.

#### Hradby

Poskytují základní ochranu vesnice před nepřáteli. Přidávají procentuální bonus k obraně.

- Maximální úroveň: 10
- Cena: (50 dřeva, 50 kamene) · (úroveň · 2)
- Bonus k obraně:  $1,05^{\text{úroveň}}$

#### Hrad

Hráčovo sídlo, ústřední prvek vesnice. Slouží k nastavování ohlašovacích emailů a sám o sobě zajišťuje produkci 2 jednotky dřeva i kamene za hodinu. Hrad v prvotní fázi hry supluje sklad surovin, kdy je jeho kapacita 400 jednotek.

##### **Tržiště**

Je místo, kde si hráči vybírají pracovní úkoly, za jejichž splnění pak získávají odměny ve formě zlaťáků (u každého úkolu je uvedena odměna). Úlohu získá hráč, který ji přijme jako první.

##### **Burza**

Na burze je seznam pracovních úkolů stejně jako na tržišti, avšak s tím rozdílem, že se o ně hráči ucházejí prostřednictvím holandské dražby. Příhozy se počítají automaticky, jsou vždy o 5% nižší než aktuální cena.

##### **Dřevorubec (2 budovy)**

Slouží k produkci dřeva ve hře. Vylepšováním budovy stoupá její efektivita.

- Maximální úroveň: 10
- Cena: (40 dřeva, 30 kamene) · (úroveň · 2) pro úroveň > 0, jinak 40 dřeva, 30 kamene
- Produkce: 10 jednotek/hod na 1. úrovni (+ 2 každá další úroveň)

##### **Kamenolom (2 budovy)**

Slouží k produkci kamene ve hře. Vylepšováním budovy stoupá její efektivita.

- Maximální úroveň: 10
- Cena: (30 dřeva, 40 kamene) · (úroveň · 2) pro úroveň > 0, jinak 30 dřeva, 40 kamene
- Produkce: 10 jednotek/hod na 1. úrovni (+ 2 každá další úroveň)

##### **Kasárny**

V kasárnách se cvičí vojenské jednotky. Výcvik jednotek nezabírá žádný čas. Jednotky se dělí na útočné a obranné.

##### **Sklad surovin**

Množství uskladněných surovin je dáno kapacitou skladu, který lze vylepšovat. Pokud je sklad surovin plný, suroviny přetékají a hráč o ně přichází.

- Maximální úroveň: 5
- Cena: (120 dřeva, 120 kamene) · (úroveň · 3) pro úroveň > 0, jinak 120 dřeva, 120 kamene
- Kapacita:  $800 \cdot (\text{úroveň} \cdot 2,5)$  pro úroveň > 0, jinak 800

##### **4.2.6.3 Mapa světa**

Na mapě světa jsou zobrazeny hrady ostatních hráčů a hrady barbarů. Na hrady barbarů lze vést útoky. Vzdálenost na mapě určuje dobu přesunu vojska.



#### 4.2.6.4 Vojenské jednotky

Existují 2 typy vojenských jednotek. Jejich výcvik nelze provádět jen za samotné suroviny, jsou potřeba i zlaťáky. Jednotku charakterizují 3 atributy: síla útoku, síla obrany a cena. Oba typy jednotek se pohybují rychlostí 10 polí za hodinu a unesou 10 jednotek surovin.

**Válečník** - jednotka vynikající především v obraně

- Útok: 5
- Obrana: 30
- Cena: 18 dřeva, 20 kamene, 11 zlaťáků

**Rytíř** - jednotka vhodná pro útok na barbary

- Útok: 30
- Obrana: 5
- Cena: 20 dřeva, 18 kamene, 11 zlaťáků

#### 4.2.6.5 Útok

Útoky lze vést na hrady barbarů. Za úspěšnou výpravu uloupí vojáci nějaké suroviny, občas zlaťáky. Doba cesty vojáků závisí na vzdálenosti obou hradů na mapě světa. Pokaždé, když hráč porazí hrad nějakého barbara, zvýší se barbarova úroveň a tím pádem i obtížnost. Naplánovaný útok může hráč zrušit, dokud k němu nedojde (může vojsko stáhnout z cesty).

Kořist se určuje na základě nosnosti přeživších jednotek následujícím způsobem:

- Dřevo: suma nosnosti · náhodné číslo (0,1)
- Kámen: zbytek kapacity
- Zlaťáky: 5% šance na zisk 100 zlaťáků

#### 4.2.6.6 Obrana

Barbaři vedou na vesnice hráčů náhodné útoky. Jejich síla je závislá na úrovni barbara, který útok vede. Hráč je o příchozím útoku informován notifikačním emailem a zobrazí se mu čas příchodu vojska. Pokud barbaři porazí obranu hráče, budou mu uloupeny suroviny podle kapacity přeživších jednotek barbarů.

- Frekvence útoků: 0 až (počet hráčů / 5) denně
  - Pokud je počet hráčů menší než 5, pak je frekvence útoků 0 až 1 denně.
  - První 2 dny hry barbaři neútočí.
- Síla útoku barbara je dána jeho úrovní.

##### 4.2.6.7 Výpočet boje

Při výpočtu výsledku bitvy záleží na útočné síle napadajících jednotek a obranné síle bránících jednotek. Pokud se navíc bitva odehrává v hráčově vesnici, bere se v potaz i ochranný bonus hradeb. Výpočet vysvětlíme nejlépe na konkrétních příkladech.

##### Útok hráče na barbara

Uvažujme napadení barbara hráčem s armádou o 30 rytířích a 10 válečnicích. Vesnici barbara stráží celkem 25 válečníků. Spočteme výslednou útočnou (AP) a obrannou (DP) sílu.

$$\begin{aligned}30 \cdot 30 + 10 \cdot 5 &= 950 \text{ AP} \\25 \cdot 30 &= 750 \text{ DP}\end{aligned}$$

Bitva probíhá, dokud není jedna strana kompletně zničena. Porovnáním útočné a obranné síly obou vojsk určíme poraženého. Je zřejmé, že obranná armáda napadeného barona padne. Zbývá nám tedy dopočítat ztráty na jednotkách útočícího hráče. Procento ztrát zjistíme pomocí následujícího vzorce:

$$100\% \cdot (\text{LP} / \text{WP})^K,$$

kde LP je síla poraženého, WP síla vítěze a K je konstanta zohledňující početní převahu vojska. Konstantu K nastavíme na hodnotu 2, abychom hráčům nečinili při útocích příliš vysoké škody na vojsku.

$$\begin{aligned}100\% \cdot (750 / 950)^2 &\doteq 100\% \cdot 0,6233 = 62,33\% \\30 \cdot 0,6233 &= 18,699 \doteq 18 \\10 \cdot 0,6233 &= 6,233 \doteq 6\end{aligned}$$

Útočník porazí barbara se ztrátou 18 rytířů a 6 válečniců. Přeživší jednotky uloupí surovinové zdroje dle své kapacity. Hráč má navíc 5% šanci na zisk 100 zlatáků.

##### Útok barbara na hráče

Uvažujme napadení hráče barbaram s armádou o 40 šermířích. Vesnici hráče stráží celkem 20 válečniců a 15 šermířů. Procentuální bonus k obraně přidávají hradby 4. úrovně. Spočteme výslednou AP a DP.

$$40 \cdot 30 = 1200 \text{ AP}$$

$$20 \cdot 30 + 15 \cdot 5 = 675 \text{ DP}$$

$$\text{Bonus hradeb: } 1,05^4 \doteq 1,2155 = 21,55\%$$

$$\text{Celková DP: } 675 \cdot 1,2155 \doteq 820 \text{ DP}$$

Obranné vojsko hráče podlehne nájezdu barbarů. Hráč ztratí všechny jednotky ve vesnici a budou mu uloupeny suroviny dle kapacity přeživších šermířů barbara.

#### 4.2.6.8 Nepřátelé

Nepřátelé hráčů jsou proradní a loupeživí barbaři. Boj mezi hráči a barbary je nekonečný, nelze totiž zničit hrady barbarů ani hráčů. Hrady barbarů si nelze přivlastnit. Úroveň barbarů roste pokaždé, když je některý z hráčů porazí.

- Počet barbarů na mapě:  $4 \cdot$  celkový počet hráčů hry
- Počáteční úroveň: 1
- Vojsko: při obraně disponuje 1 válečníkem, při útoku 1 šermířem
- S každou další úrovní má 50% šanci na zvýšení stavu svého vojska

Pokud hráč zaútočí na barbara a prohraje, dojde k oslabení armády barbara. S novou úrovní se mu však jednotky obnoví. Při útocích barbarům jednotky neubývají, stále disponují počtem dle jejich úrovně. Barbar je natolik schopný, že při nájezdech hráčů dokáže uhnout se svojí útočnou armádou.

#### 4.2.6.9 Surovinové zdroje

Dřevo a kámen jsou základní suroviny pro stavbu budov i trénování armády ve hře. Prémiovou surovinou jsou zlaťáky, bez kterých nelze provádět výcvik vojenských jednotek. Dřevo a kámen produkují budovy dřevorubců a kameníků. Zlaťáky mohou hráči získat:

- Plněním pracovních úkolů
- Bonusem pro nejlepší hráče hry
- Útočením na barbary

#### 4.2.6.10 Nejlepší hráči a bonusy

Po předem stanovených periodách se vyhodnocují nejlepší hráči hry. Ti dostanou před svoji přezdívku jeden z následujících titulů (vždy ten nejvyšší, seřazeno sestupně):

- **Veliký:** titul za nejvíce získaných zlaťáků, odměna 50 zlaťáků

- **Dobyvatel:** titul za nejvíce zabitých jednotek při útocích, odměna 40 zlaťáků
- **Neústupný:** titul za nejvíce zabitých jednotek při obraně, odměna 30 zlaťáků

Periodické i celkové statistiky výše uvedených veličin jsou dostupné jen hráčům ve hře. Pokud na stejný titul dosáhnou dva a více hráčů, získají jej všichni včetně bonusu. V případě, že hráč dosáhne na více titulů současně, je na něj aplikován pouze ten nejceněnější.

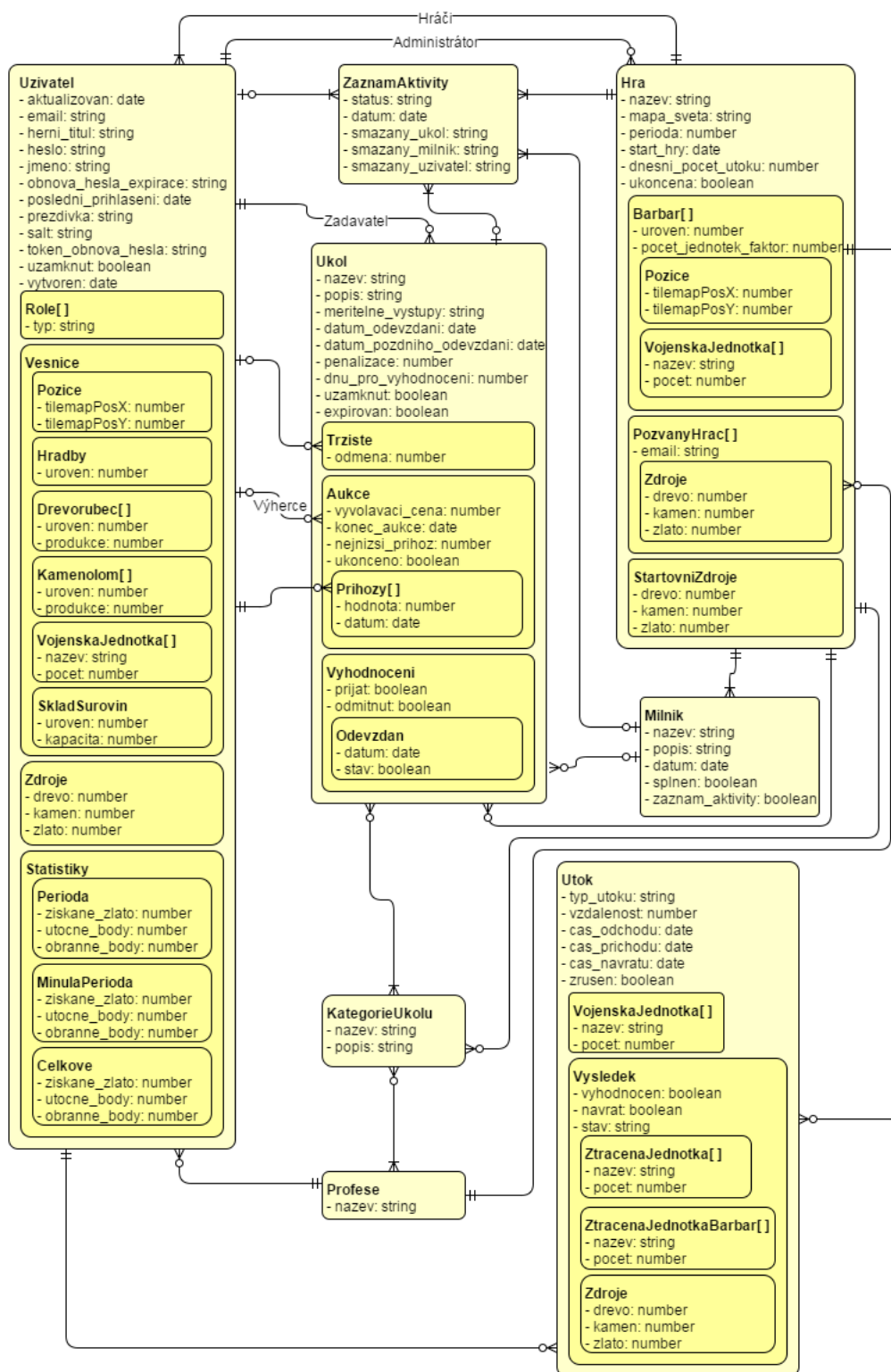
#### 4.2.7 Konec hry

Za konec hry se považuje datum posledního milníku projektu. Po ukončení hry je hráčům ve hře přístupný pouze žebříček se statistikami.

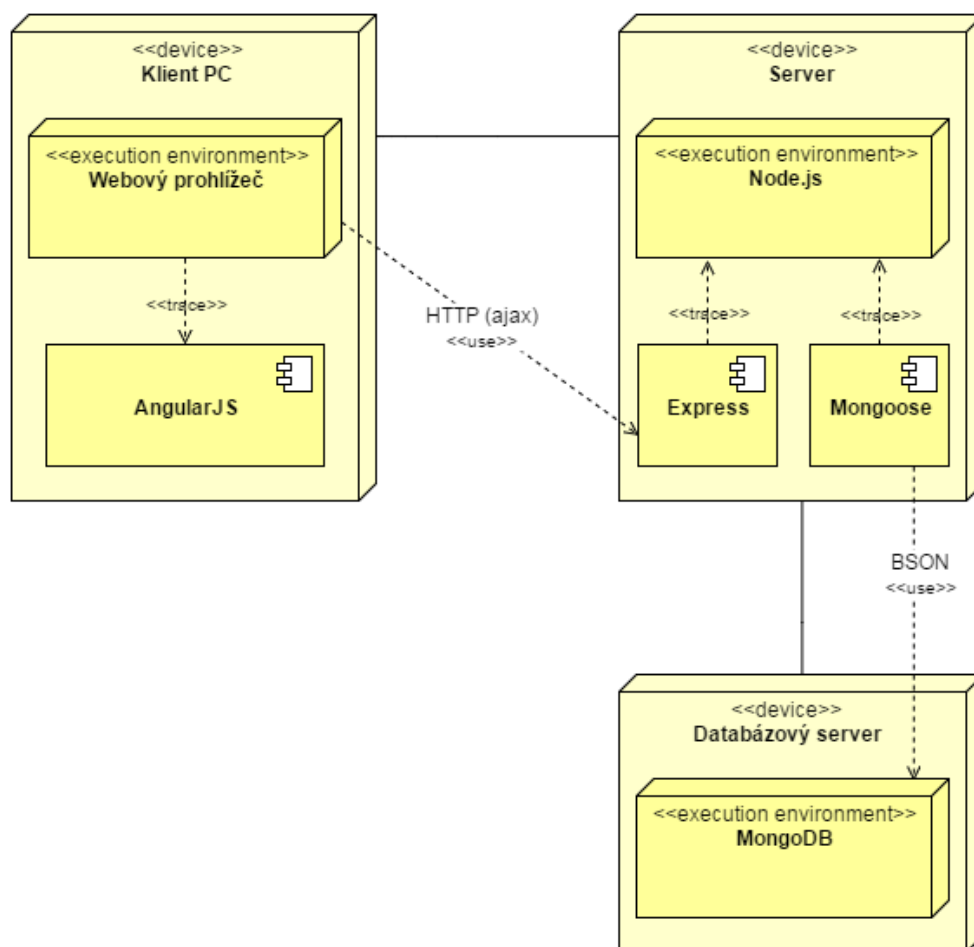
### 4.3 Databázový model

Databáze aplikace je navržena pro MongoDB [31], což je SŘBD uchovávající data jako dokumenty ve formátu BSON (podobný JSON). V MongoDB neexistují relace, nýbrž reference, kdy je v jednom dokumentu uloženo ID jiného dokumentu jako odkaz. Abychom znázornili kardinalitu a parcialitu vztahů, používáme v diagramu 4.1 spojnice známé z E-R diagramu.

### 4.3. Databázový model



Obrázek 4.1: Diagram návrhu databáze vytvořený v [3]



Obrázek 4.2: Diagram nasazení vytvořený v [3]

#### 4.4 Diagram nasazení

Diagram nasazení 4.2 popisuje umístění jednotlivých softwarových komponent na fyzická zařízení a jejich vzájemnou spolupráci. K zajištění funkčnosti aplikace potřebujeme databázový server MongoDB a server s frameworkem Node.js [32] s nainstalovanými komponentami Express [33] a Mongoose [34]. Aplikace je přístupná ze zařízení s webovým prohlížečem, který podporuje a nemá zakázaný javascript.

---

# Implementace

V předchozí kapitole jsme provedli návrh aplikace, kterou nyní naimplementujeme. Zvolíme si a představíme volně dostupné softwarové nástroje, které použijeme. Na obrázcích ukážeme některé části aplikace a zmíníme zajímavé části systému. Z důvodu rozsáhlosti a tím spjaté časové náročnosti implementace nebyla naprogramována funkčnost související s emaily a přidávání úkolů hráčem v prostředí hry.

## 5.1 MEAN.JS

Z diagramu nasazení 4.2 je zřejmé, že jsme pro implementaci aplikace vybrali databázi MongoDB, server postavený na Node.js s frameworkem Express a frontend MVW framework AngularJS [35]. Pro vyjmenované technologie existuje balík MEAN.JS [36], který integruje potřebné knihovny pro práci s nimi.

### 5.1.1 Node.js

Javascript je interpretovaný jazyk, který potřebuje nějaké běhové prostředí starající se o provedení uživatelského skriptu. Téměř raketový vzestup znamenal engine V8 [37] od Googlu, vytvořený pro prohlížeč Chrome. Ten obrátil do té doby poměrně vyrovnané žebříčky rychlosti jednotlivých interpretů a svou rychlostí nechal veškerou konkurenci daleko za sebou. Node.js je vlastně engine V8, rozšířený o funkce, které umožňují prováděným skriptům přistupovat k souborům či síťovým funkcím [38].

Node.js dokáže zpracovávat vstupně-výstupní operace asynchronně, díky V8 enginu je velmi rychlý a sbírka modulů npm [39] je dle [32] největší open-source knihovna na světě. Protože Node.js nekončí zpracování požadavku, ale pokračuje dál, nachází uplatnění v real-time aplikacích, např. chat, webové hry, streamovací služby, aukce a další. Nijak nás to však neomezuje v tom, abychom Node.js nemohli využít i pro standardní webové aplikace [40].

### 5.1.2 Express

Express je framework na Node.js, který pomáhá organizovat webovou aplikaci do MVC struktury na straně serveru. Framework v podstatě obstarává vše, od routování přes vyřizování požadavků až po renderování views neboli pohledů.

### 5.1.3 MongoDB

MongoDB je představitelem dokumentově-orientované databáze, která se řadí mezi NoSQL. Každý záznam v databázi je reprezentován jako dokument ve formátu BSON, což je vlastně JSON v binární podobě. Databáze podporuje velké množství datových typů, kromě těch běžných také odkaz na jiný dokument (tzv. referenci) nebo binární data. Struktura dokumentu je velmi flexibilní, každý prvek může obsahovat více prvků či dokonce další datové struktury.

Dotazovací jazyk MongoDB je unikátní a v některých možnostech dokonce překonává tradiční SQL. V dotazech můžeme používat množinové operace, regulární výrazy, javascriptové podmínky nebo přímo spustit javascriptovou metodu na straně serveru. Databáze se hodí pro provozování velkých nebo zatížených aplikací [41].

Za zmínku stojí ještě následující funkce [42]:

#### Replikace

Sady replik poskytují redundanci a vysokou dostupnost. Jsou základem pro všechny produkční nasazení.

#### Sharding

Typ horizontálního škálování, jedná se o proces ukládání dat na větším počtu hardware.

#### GridFS

Specifikace pro ukládání a načítání souborů, které přesahují 16 MB omezení velikosti BSON dokumentu. Namísto uložení souboru v jediném dokumentu jej GridFS rozdělí na části, které ukládá jako samostatné dokumenty.

#### Agregace

Agregační operace shlukují hodnoty z více dokumentů. Na seskupených datech se může následně provádět celá řada operací. Agregací operace umožňují uživatelům získat stejný druh výsledků jako SQL GROUP BY příkaz v relačních databázích.

### 5.1.4 AngularJS

AngularJS je frontend javascriptový MVW framework navržený pro vytváření webových aplikací. Velmi často se v Node.js vytvářejí aplikace, které používají hodně klientského javascriptu a komunikují se serverem přes API. Pro tyto



případy je vhodné pro zjednodušení práce využít např. zmíněný AngularJS, který programátora odstíní od rutinních úkolů a zjednoduší mu práci.

Mezi nejužitečnější koncepty frameworku dle [43] se obvykle řadí:

### **Two Way Data-Binding**

Řeší synchronizaci stavů mezi modelem a view.

### **Implementace Dependency Injection**

Jedná se o návrhový vzor, který řeší závislost mezi jednotlivými komponentami programu.

### **Testovatelnost**

Jedna z oblastí, na kterou je framework zaměřen, je testovatelnost kódu. Přispívá k tomu implementace DI, ale také nástroje pro testování, které díky frameworku vznikly.

### **Direktivy**

Direktivy představují způsob, jak naučit HTML novým trikům. Přes direktivy můžeme např. udělat to, že po kliknutí na nějaký prvek jej změním na textové pole, do kterého uživatel něco napíše a po kliknutí jinam data odešleme na server.

## **5.1.5 Další knihovny v balíku MEAN.JS**

Součástí balíku MEAN.JS je ještě několik dalších knihoven, které jsou nápomocné při vytváření webových aplikací. Patří mezi ně:

### **Bower [44]**

Jestliže npm je balíčkovací systém pro serverovou část, pak Bower je v podstatě to samé pro část klientskou.

### **Grunt [45]**

Neboli „JavaScript Task Runner“ není nic jiného než robot, který automatizuje opakující se nebo nudné úkoly. Hlídá změny v souborech a po jejich provedení vyvolá nějaké akce.

### **Lodash [46]**

Knihovna javascriptových funkcí, která nabízí čisté a výkonnější metody pro manipulaci s objekty, kolekcemi, řetězci aj.

### **Mongoose**

Nástroj pro modelování a práci s MongoDB objekty.

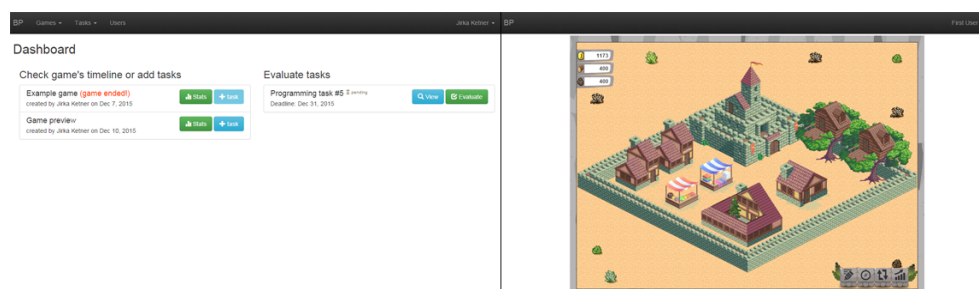
### **Nodemailer [47]**

Nástroj pro odesílání emailů z Node.js serveru.

### **Passport [48]**

Autentifikační prostředí fungující s frameworkem Express, které slouží pro ověřování uživatelů pomocí hesla, Facebook účtu, Twitteru a dalších.

## 5. IMPLEMENTACE



Obrázek 5.1: Náhled administrační i hráčské části aplikace

### 5.1.6 Proč MEAN.JS?

Vybrali jsme si MEAN.JS hned z několika důvodů. Prvním z nich je, že se jedná o moderní přístup k webovým aplikacím a všechny hlavní komponenty akronymu MEAN (MongoDB, Express, AngularJS, Node.js) procházejí pilným vývojem. Svědčí o tom pravidelné aktualizace kódu na GitHubu [49].

Dalším důvodem je rychlost serveru Node.js. Ten je v porovnání se serverem Apache „superrychlý“! Dokazuje to množství benchmarků [50], z nichž vyplývá, že Node.js zvládne mnohem více požadavků za sekundu a má vyšší přenosovou rychlost s mnohem nižším počtem selhání požadavků.

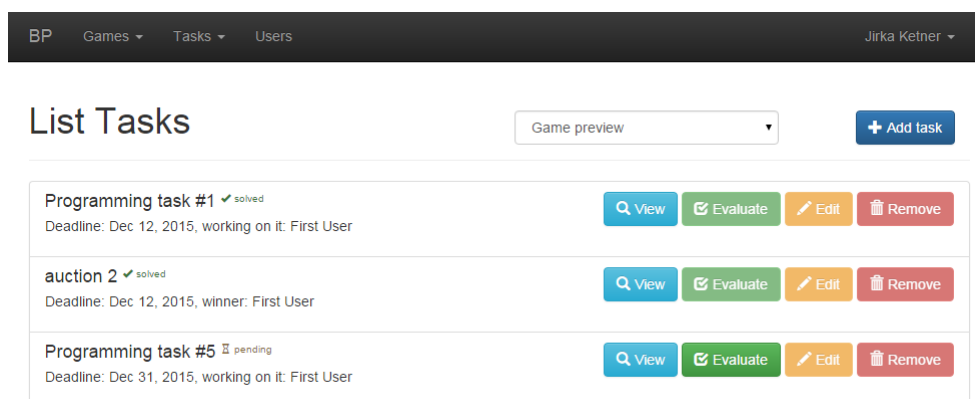
Databáze MongoDB je flexibilní a můžeme v ní uchovávat strukturované objekty. Nejsme omezeni pouze na tabulky s hodnotami. Pokud uchováváme data správně v dokumentech a nesnažíme se udělat relační databázi, pak je MongoDB velmi rychlý databázový systém. [51].

Posledním důvodem, proč jsme se vydali cestou MEAN-stacku je to, že k napsání aplikace potřebujeme znát (mimo HTML a CSS) jeden jediný jazyk - javascript. Od dotazování se na databázi, přes backendovou logiku po frontendovou část v AngularJS, to vše programujeme v javascriptu.

## 5.2 Administrační část aplikace

Aplikace se skládá ze dvou částí. První, administrační, je přístupná administrátorovi her (projektů), druhá část je herní, zcela odlišná a přístupná pouze pozvaným hráčům. Vzhled obou částí je vyobrazen na 5.1. Grafická stránka administrace je vytvořena za pomoci knihovny Bootstrap [52], zatímco na herní část jsou použity prvky vlastní tvorby a obrázky z OpenGameArt.org [53].

Celá administrační část je protkána různými seznamy (hráčů, úkolů 5.2, her, aj.), kde u jednotlivých položek jsou tlačítka zpřístupněné akce pro zobrazení, úpravu (případně vyhodnocení) či smazání. U seznamu hráčů a her je navíc tlačítko pro zobrazení sledu aktivit dané položky. Vytváření nových položek se provádí prostřednictvím formulářů.



Obrázek 5.2: Seznam úkolů (s hláškami o jejich stavu) v administraci

### 5.2.1 Dashboard

Na dashboardu jsou k dispozici nejčastější akce, které v systému administrátor provádí. Máme zde seznam her, u kterých jsou tlačítka pro zobrazení sledu událostí a tlačítka pro přidání úkolů do hry. Další seznam je určen pro odevzdané úkoly, které jsou ve frontě na vyhodnocení. U každého úkolu v seznamu je umístěno tlačítko pro zobrazení a vyhodnocení úlohy. Pokud neexistují žádné úlohy k vyhodnocení, je seznam prázdný.

## 5.3 Herní část aplikace

Prostředí hry je tvořeno dvěma hlavními pohledy - na hráčovu vesnici a na mapu světa. Oba pohledy obsahují několik dialogových oken, které se vyvolávají po kliku na různé prvky ve hře, a mají společné hlavní menu.

### 5.3.1 Hráčova vesnice

Všechny budovy ve vesnici včetně hradeb vyvolávají po kliknutí příslušné dialogové okno. Budovy, které lze dle navrhovaného konceptu vylepšovat, zobrazují potřebné suroviny pro posun do další úrovně. Pokud má hráč na kontě dostatek surovin (vyobrazeny v levém horním rohu), může vylepšení provést. Vojenské jednotky, nutné pro obranu nebo útok, verbujeme v kasárnách 5.3. V okně tržiště či aukce je výpis zadaných úkolů, které jsou přiřazeny hráčově profesi a které může plnit.

Pokud hráč přijme nebo vyhraje nějaký úkol, stává se jeho řešitelem. Hráčovy úkoly jsou dostupné z nabídky hlavního menu, kde je rovněž může označit za splněné.

Hlavní menu sestává ze čtyř položek, které otevírají mapu světa, přehled válečných přesunů, statistiky hry a, jak již bylo zmíněno, hráčovy úkoly. Okno



Obrázek 5.3: Dialogové okno kasáren pro verbování jednotek

válečných přesunů obsahuje seznam příchozích a odchozích útoků s časovými údaji o příchodech, ve statistikách najdeme seřazený seznam všech hráčů podle tří veličin - získané zlaťáky, útočné body a obranné body.

### 5.3.2 Mapa světa

Mapa světa je implementována v direktivě Angularu pomocí HTML5 canvasu. V programu Tiled [54] se vytvoří z grafických dlaždic předloha, která se vyexportuje do souboru typu JSON. Ten se poté v direktivě zpracuje a vykreslí se díky němu mapa v canvasu. Pohybovat mapou se můžeme prostřednictvím šipek na klávesnici.

Mapu světa 5.4 osídlují tři typy klikatelných hradů:

- Bílý - hrad přihlášeného hráče, po kliknutí zobrazí informace o hráči.
- Šedý - hrad jiného hráče, po kliknutí zobrazí informace o hráči.
- Černý - hrad barbara, po kliknutí zobrazí dialogové okno s možností útoku.

### 5.3.3 Kontrola HTTP požadavků uživatele

Každý hráč má možnost nahlédnout do zdrojového kódu frontendu. Z něj se dá zjistit, co se na jakou adresu při různých požadavcích posílá. Zkušenější hráč



Obrázek 5.4: Mapa světa

potom nemá problém poslat HTTP požadavek s podvodnými daty. Mohl by si tím přijít např. na neomezený počet vojenských jednotek, vylepšení budov zdarma či dokonce na neomezené množství surovin. Proto je nutné každý požadavek na straně serveru ověřovat.

### 5.4 Procedury spouštěné při interakci hráče

V aplikaci jsou implementovány tři procedury, které se spouští při různých interakcích hráče.

#### 5.4.1 Vyhodnocování útoků

V závislosti na čase je potřeba vyhodnocovat útoky ve hře. Musíme si dát pozor na pořadí, ve kterém výpočty provádíme, jinak by docházelo k chybným vyhodnocením (např. bychom nevrátili hráčovy jednotky z cesty předtím, než by došlo k útoku barbara). Útoky vyhodnocujeme podle času návratu jednotek z cesty, kdy nájezdy barbarů mají nastaven čas návratu stejně jako čas útoku. Protože Node.js pracuje asynchronně, musíme zajistit synchronní vyhodnocování útoků sami za pomoci modulu Async.js [55].

Další problémy s vyhodnocením útoků plynou z různých situací ve hře, např. by hráč nakoupil jednotky v čase po proběhnuté obraně proti barba-

rům, ale výpočet by se spustil až později. Díky tomu by vesnici bránilo více vojenských jednotek, než by mělo. Proto vyhodnocování probíhá při každém:

- Přihlášení uživatele
- Vyvolání dialogového okna s vojenskými přesuny
- Vyslání útoku na barbara
- Nákupu jednotek v kasárnách

### 5.4.2 Vyhodnocování úkolů

U úloh je potřeba řešit expiraci na tržišti, konec aukce, odevzdání v termínu a vyhodnocení v termínu. Pokud úloha expiruje nebo v aukci nikdo nepřihodí, přestane se hráčům zobrazovat. Administrátor ji však může změnou termínů obnovit. Pokud hráč neodevzdá úlohu ani v pozdním termínu, zablokuje se mu tlačítko pro odevzdání a úkol dostane status „selhání“. Nedojde-li ze strany zadavatele k vyhodnocení úlohy včas, bude automaticky označena za splněnou a hráčovi připadne odměna.

U vyhodnocování úloh není třeba dbát na nějakém pořadí, nevádí nám ani asynchronní volání ze strany Node.js. Vyhodnocování úloh provádíme při každém:

- Přihlášení uživatele
- Vyvolání dialogového okna tržiště
- Vyvolání dialogového okna aukční síně
- Vyvolání dialogového okna s hráčovy úkoly

### 5.4.3 Výpočet surovinových zdrojů

Při přihlášení hráče si ukládáme aktuální datum a čas a porovnáváme jej s posledním přihlášením. Z rozdílu nám potom vyplývá, jak máme v závislosti na produkci navýšit stavy surovinových zdrojů. Aktualizování surovin probíhá také při každém nákupu ve hře. Po dobu nečinnosti hráče provádíme aktualizaci pouze na straně klienta.

## 5.5 Procedury spouštěné serverem automaticky

K volání různých procedur automaticky na straně serveru nám slouží crony. Jejich nastavení v prostředí Node.js umožňuje modul node-cron [56]. Naše aplikace má implementovány celkem dva crony, první se spouští každých deset minut, druhý se spouští každý den jednou, v ranních hodinách.

## 5.5. Procedury spouštěné serverem automaticky

---

První cron obsahuje volání jediné procedury, která řeší útoky loupeživých barbarů. Při každém spuštění existuje šance na provedení útoku barbarami. Pakliže se má útok provést, vybere se náhodně jeden hráč, na kterého pošle náhodný barbar všechny své útočné jednotky.

Druhý, denní cron, obsahuje několik procedur, které vykonávají následující úlohy:

- Smazání zrušených nebo starých útoků
- Vynulování denního počítadla útoků barbarů
- Vyhodnocení proběhlých period her
- Vyhodnocení proběhlých milníků
- Vyhodnocení úkolů
- Ukončení proběhlých her





---

# Testování

V následující kapitole se vrhneme na testování námi implementované aplikace. Nejdříve představíme profily uživatelů, ti následně dle zadaných scénářů aplikaci vyzkouší. Z přímého pozorování pak vyvodíme závěry o funkčnosti a uživatelské přívětivosti aplikace.

## 6.1 Typické profily uživatelů ve skupině

Všichni uživatelé v desetičlenné skupině mají minimálně středoškolské vzdělání, pokročilou znalost angličtiny a umí pracovat a orientovat se ve webovém prohlížeči. Dále se nám rozdělují do tří následujících profilů:

### 1. profil (2 uživatelé)

Středoškolské vzdělání v IT oboru, znalost počítače na úrovni kódování webů v HTML a CSS, velmi dobrá znalost počítačových her, žádné zkušenosti s projektovým řízením ani aplikacemi pro správu vlastních úkolů

### 2. profil (3 uživatelé)

Vysokoškolské vzdělání mimo obor IT, znalost počítače na úrovni práce s textovým či tabulkovým editorem, zkušenost s online hrami v prohlížeči, neví téměř nic o projektovém řízení, zná aplikace pro správu vlastních úkolů

### 3. profil (5 uživatelů)

Vysokoškolské vzdělání v oboru IT, znalost počítače na úrovni programování, velmi dobrá zkušenost s počítačovými hrami, částečná teoretická znalost projektového řízení, ovládá aplikace pro správu vlastních úkolů

## 6.2 Testovací scénáře

Každý z uživatelů testoval aplikaci prostřednictvím následujících scénářů, které na sebe navzájem navazují. Scénáře svým rozsahem pokrývají téměř

## 6. TESTOVÁNÍ

---

všechny možnosti aplikace. Úkony jsou v nich seřazeny chronologicky tak, jak je má uživatel splnit.

### 1. scénář, v roli administrátora

Vytvoření nové hry, přidání dalšího milníku do hry, změna vyhodnocovací periody, smazání milníku, vytvoření kategorie úkolů, zadání úkolu do tržiště, zadání úkolu do aukce

### 2. scénář, v roli hráče

Zaregistrování pozvaného hráče, zvolení místa na mapě, vylepšení budovy dřevorubce, přijetí úkolu na tržišti, vytvoření příhozu v aukci, odevzdání přijatého úkolu, verbování jednotek v kasárnách

### 3. scénář, v roli administrátora

Vyhodnocení odevzdaného úkolu, zhlédnutí sledu aktivit hry, zhlédnutí sledu aktivit hráče, pozvání dalšího hráče do hry, úprava profese hráče, zobrazení milníků hry, smazání nějaké jiné hry

### 4. scénář, v roli hráče

Přihlášení hráče, zobrazení vlastních úkolů, přechod na mapu světa, vyslání útoku na hrad barbara, zobrazení vojenských přesunů, zobrazení statistik

Před zahájením testování na uživateli jsme všechny seznámili s účelem aplikace. Uživatelům, radícím se do první a druhé profilové skupiny, jsme nastínili také základní principy projektového manažerství.

## 6.3 Testování funkčnosti

Testování funkčnosti je nedílnou součástí procesu vývoje aplikace. Během implementace jsme jich provedli nespočet, vždy po vytvoření funkčně celistvého bloku kódu. Při objevení chyby byla ihned provedena náprava.

I přesto se několik nedostatků aplikace projevilo až při testování uživateli:

- Při vyhodnocení úkolu z dashboardu se do herního sledu vytvořila aktivita s „undefined“ hráčem.
- Nefungovala validace číselných formulářových polí, administrátor mohl zadat nesmyslné údaje (záporné či příliš vysoké číslo), což vedlo k chybným výpočtům.
- V prohlížeči Mozilla Firefox [57] nefungovala mapa herního světa kvůli přístupu do neexistujícího prvku pole.

- Stejný prohlížeč nemá implicitně naimplementovaný vyskakovací kalendář pro volbu datumů. Tato chyba se projevuje při vytváření milníků, u vytváření úkolů používáme knihovnu Bootstrap 3 Datepicker [58] kvůli volbě času. Řešením je použít knihovnu také u milníků nebo vypsát datum v prohlížeči Mozilla Firefox ručně.
- Administrátor musí dbát opatrnosti při volbě datumů ve formulářových polích. Není implementována žádná validace. Může tedy vytvořit již propadlý úkol či milník. Vše však lze následně upravit.

První tři zjištěné chyby byly dodatečně po testování v rámci bakalářské práce opraveny. Poslední dva nedostatky nemají zásadní dopad na funkčnost, nicméně v rámci výhledu do budoucna by bylo žádoucí provést nápravu.

## 6.4 Testování uživatelské přívětivosti

Uživatele jsme při plnění scénářů bedlivě pozorovali a po každém kroku jsme spolu konzultovali, zda-li se nevyskytl nějaký problém. Napříč všemi profilovými skupinami jsme vyvodili následující seznam nedostatků:

- Vytvoření nového milníku prostřednictvím symbolu „+“ bylo pro většinu testujících neintuitivní. Proto jsme symbol doplnili o slovíčko „Přidat“.
- Více než polovina uživatelů přiřadila k úkolům kategorii až po systémovém upozornění. Existence kategorií v pravé části obrazovky si totiž nevšimli. Box jsme zvýraznili červenou barvou na pozadí.
- Polovina testujících chtěla ze sledu aktivit hry přejít na sled aktivit hráče kliknutím na jeho jméno. Nyní tedy funguje jako odkaz.
- Převážná část uživatelů měla problém najít milníky ve hře. Nakonec se to všem podařilo, ale stálo by za zvážení vytvořit položku v menu a milníky zobrazovat stejně jako seznamy hráčů, úkolů nebo her. Prozatím jsme tlačítko pro úpravu hry, přes které se k milníkům administrátor dostane, přidali také na dashboard aplikace.
- Smazání hry by bylo vhodné doplnit o potvrzovací dialogové okno.

Další nedostatky se týkaly pouze první a druhé profilové skupiny uživatelů. První skupina by uvítala u formulářových polí vyskakovací okna s nápovědou, tzv. tooltips. U druhé skupiny jsme vyzorovali, že se uživatelé snažili klikat i na zakázaná tlačítka. Rovněž jim trvalo delší dobu, než zjistili, jak se pohybovat po mapě světa či jak vylepšovat budovy ve vesnici.

Všichni uživatelé dokázali projít aplikaci dle zadaných scénářů bez našeho zásahu. Někteří se dotazovali pouze na věci týkající se projektového manažerství, což pramení z jejich neznalosti problematiky. Ovládání herní části

aplikace označili všichni testující za intuitivní a uživatelsky přívětivé. V administraci by někteří ocenili, kdyby milníky nebyly „ukryté“ v editaci hry.

Všem uživatelům se vytvořená aplikace líbila a dokázali by si představit její reálné uplatnění ve firmách.

### 6.5 Testování aplikace online

Umístili jsme aplikaci na cloudovou platformu Pivotal Web Services [59], která nabízí šedesát dní zdarma na vyzkoušení. Následně jsme opět prověřili její funkčnost. Po otestování jsme došli k závěru, že je do produkční verze třeba naimplementovat nějaký indikátor stavu načítání či zpracovávání uživatelského požadavku, např. Angular Loading Bar [60].

---

## Výhled do budoucna

Aplikace by si v budoucnu zasloužila další vývoj. Kromě doděláním emailových notifikací a přidávání úkolů hráči je potřeba vyřešit také registraci a schvalování dalších uživatelů s rolí administrátora. Zajímavé by bylo aplikaci doplnit o nějaký interní způsob komunikace (chat, komentáře k úkolům, aj.) a zpřístupnit ji také pro mobilní zařízení. Při testování jiných aplikací pro správu úkolů nám přišla velmi vhod funkcionalita plánovacího kalendáře. Během produkčního nasazení aplikace je nutné vyřešit rovněž stránkování záznamů.

V kapitole o gamifikaci jsme probrali důležité herní mechanismy. Systém úrovní by hráče mohl nabudit k větší soutěživosti. Vylepšování budov poté může být přístupné od určitých úrovní. K ucelení a sjednocení týmu je skvělé hráče postavit před společnou výzvu, např. obranu aliančního divu světa. Trénování jednotek a vylepšování budov by mělo trvat nějaký čas.



---

# Závěr

Cílem této práce bylo navrhnout a implementovat webovou aplikaci pro správu úkolů. Na začátku jsme zjistili, co se skrývá pod pojmem gamifikace a provedli jsme rešerši jejích stěžejních prvků. Zjistili jsme, že se opírá hlavně o body, úrovně, žebříčky, odznáčky, nástup a adaptaci, výzvy a úkoly a cyklus sociálního zapojení. Následně jsme prozkoumali několik webových aplikací zaměřujících se na správu úloh. Vyplývalo nám, že je můžeme koncepčně rozdělit na tři odlišné skupiny:

- Aplikace pro projektové manažery
- Aplikace pro nenáročného uživatele
- Seznamy úkolů s možností sdílení

Dalším krokem bylo vybrat si jeden koncept, upravit jej dle potřeb a obohatit ho o prvky gamifikace. Zvolili jsme koncept aplikací pro nenáročného uživatele. Rozhodli jsme se však, že nám nestačí přidat body, odznáčky a žebříčky a prohlásit aplikaci za gamifikovanou. Chtěli jsme vytvořit reálnou hru, kterou si uživatelé za odměnu zahrají, která v nich probudí jejich soutěživost a touhu po úspěchu. Vznikl tedy koncept, dle kterého jsme navrhli model databáze a vytvořili diagram nasazení aplikace.

Pak přišla řada na samotnou implementaci navržené aplikace. Pro tuto činnost jsme si vybrali javascriptový balík MEAN.JS skládající se z komponent MongoDB, Node.js, Express a AngularJS. Povedlo se nám naprogramovat téměř veškerou funkcionalitu dle konceptu, chybí emailové notifikace a přidávání úkolů hráči. Vytvořená aplikace umožňuje administrátorovi zadávat pracovní úkoly, o které se hráči v prostředí hry uchází. Za jejich splnění pak obdrží prémiové platidlo, díky kterému mohou nakoupit jednotky a bránit tím svoji vesnici nebo útočit na cizí.

Na závěr jsme nechali aplikaci otestovat několika uživateli. Při testování jsme je bedlivě pozorovali a po každém kroku jsme s nimi průběh konzultovali. Závažnější funkční chyby jsme opravili. Všichni testující zvládli projít

## ZÁVĚR

---

aplikací dle zadaných scénářů bez větších obtíží, i tak jsme se pokusili vylepšit uživatelskou přívětivost aplikace kosmetickými změnami. Každý z dotázaných uživatelů by si dokázal představit reálné použití implementované aplikace.



---

## Použité zdroje

- [1] ENTERTAINMENT SOFTWARE ASSOCIATION: *Essential Facts About the Computer and Video Game Industry* [online]. ©2014, [cit 2015-5-18]. Dostupné z: [http://www.theesa.com/wp-content/uploads/2014/10/ESA\\_EF\\_2014.pdf](http://www.theesa.com/wp-content/uploads/2014/10/ESA_EF_2014.pdf)
- [2] HEE, C. W.: *Important things you have to remember when study gamification* [online]. Oct 30, 2014 [cit 2015-5-22]. Dostupné z: <http://www.slideshare.net/whla/what-is-the-gamification>
- [3] JGraph Ltd: *Draw.io* [online]. [přístup 18. prosince 2015]. Dostupné z: <https://www.draw.io/>
- [4] HUGOS, M.: *Enterprise Games: Using Game Mechanics to Build a Better Business*. Sebastopol, CA: O'Reilly Media, Inc., první vydání, ©2012, ISBN 978-1-449-31956-4.
- [5] ZICHERMANN, G.; CUNNINGHAM, C.: *Gamification by Design: Implementing Game Mechanics in Web and Mobile Apps*. Sebastopol, CA: O'Reilly Media, Inc., první vydání, ©2011, ISBN 978-1-449-39767-8.
- [6] MANDÍK, P.: *Gamifikace, to je prý Zaměstnání hrou*. In: *BusinessIT* [online]. Květen, 2014 [cit 2015-5-20]. Dostupné z: <http://www.businessit.cz/cz/gamifikace-to-je-pry-zamestnani-hrou.php>
- [7] FINDLAY, K.; ALBERTS, K.: *Gamification: How Effective Is It?* [online]. Sep 26, 2011 [cit 2015-5-20]. Dostupné z: <http://www.slideshare.net/ervler/gamification-how-effective-is-it>
- [8] BUNCHBALL, Inc.: *Gamification 101: An Introduction to the Use of Game Dynamics to Influence Behavior* [online]. ©2010, [cit 2015-5-20]. Dostupné z: <http://www.bunchball.com/sites/default/files/downloads/gamification101.pdf>

- [9] GOOGLE, Inc.: *Google Trends* [online]. [přístup 20. května 2015]. Dostupné z: <http://www.google.com/trends/explore#q=gamification>
- [10] PELLING, N.: *The (short) prehistory of „gamification“*. In: *Funding Startups (& other impossibilities)* [online]. Aug 9, 2011 [cit 2015-5-20]. Dostupné z: <https://nanodome.wordpress.com/2011/08/09/the-short-prehistory-of-gamification/>
- [11] FOURSQUARE LABS, Inc.: *Foursquare* [online]. [přístup 20. května 2015]. Dostupné z: <https://foursquare.com/>
- [12] CHEUNG, G.; ZIMMERMANN, T.; NAGAPPAN, N.: *The First Hour Experience: How the Initial Play can Engage (or Lose) New Players* [online]. ©2014, [cit 2015-5-22]. Dostupné z: <http://research.microsoft.com/pubs/226763/First%20Hour%20-%20CHIPlay%202014%20-%20preprint2.pdf>
- [13] JOHANSSON, O.; OLAUSSON, M.: *AlternativeTo* [online]. [přístup 28. května 2015]. Dostupné z: <http://alternativeto.net/tag/project-management/>
- [14] ATLISSIAN, Inc.: *JIRA* [online]. [přístup 31. května 2015]. Dostupné z: <https://www.atlassian.com/software/jira>
- [15] CITRIX SYSTEMS, Inc.: *Podio* [online]. [přístup 31. května 2015]. Dostupné z: <https://podio.com/>
- [16] ASCENSIO SYSTEM, SIA: *ONLYOFFICE* [online]. [přístup 31. května 2015]. Dostupné z: <https://www.onlyoffice.com/>
- [17] ASANA, Inc.: *Asana* [online]. [přístup 31. května 2015]. Dostupné z: <https://asana.com/>
- [18] ACTIVE COLLAB: *Active Collab* [online]. [přístup 31. května 2015]. Dostupné z: <https://www.activecollab.com/>
- [19] GROUPOCAMP: *GroupCamp* [online]. [přístup 31. května 2015]. Dostupné z: <http://www.groupcamp.cz/>
- [20] EASY SOFTWARE s.r.o.: *Easy Project* [online]. [přístup 31. května 2015]. Dostupné z: <http://www.easyproject.cz/>
- [21] WRIKE, Inc.: *Wrike* [online]. [přístup 31. května 2015]. Dostupné z: <https://www.wrike.com/>
- [22] BASECAMP, LLC: *Basecamp* [online]. [přístup 2. června 2015]. Dostupné z: <https://basecamp.com/>

- 
- [23] HUMAN COMPUTER LLC: *hiTask* [online]. [přístup 2. června 2015]. Dostupné z: <https://hitask.com/>
- [24] METALAB DESIGN Ltd: *Flow* [online]. [přístup 2. června 2015]. Dostupné z: <https://www.getflow.com/>
- [25] FREEDCAMP, Inc.: *Freedcamp* [online]. [přístup 2. června 2015]. Dostupné z: <https://freedcamp.com/>
- [26] TRELLO, Inc.: *Trello* [online]. [přístup 5. června 2015]. Dostupné z: <https://trello.com/>
- [27] 6 WUNDERKINDER GmbH: *Wunderlist* [online]. [přístup 5. června 2015]. Dostupné z: <https://www.wunderlist.com/>
- [28] ANY.DO, Inc.: *Any.do* [online]. [přístup 5. června 2015]. Dostupné z: <http://www.any.do/>
- [29] REMEMBER THE MILK Inc.: *Remember The Milk* [online]. [přístup 5. června 2015]. Dostupné z: <https://www.rememberthemilk.com/>
- [30] BARRISH, J.: *The 20 Most Popular Project Management Software Products*. In: *Capterra Project Management Blog* [online]. Nov 13, 2013 [cit 2015-6-6]. Dostupné z: <http://blog.capterra.com/20-popular-project-management-software-products-infographic/>
- [31] MongoDB, Inc.: *MongoDB v. 3.2* [software]. [přístup 18. prosince 2015]. Dostupné z: <https://www.mongodb.org/>
- [32] Node.js Foundation: *Node.js v. 0.12.2* [software]. [přístup 18. prosince 2015]. Dostupné z: <https://nodejs.org/en/>
- [33] StrongLoop, Inc.: *Express v. 4.10.11* [software]. [přístup 18. prosince 2015]. Dostupné z: <http://expressjs.com/en/index.html>
- [34] LearnBoost, Inc.: *Mongoose v. 3.8.8* [software]. [přístup 18. prosince 2015]. Dostupné z: <http://mongoosejs.com/>
- [35] GOOGLE, Inc.: *AngularJS v. 1.2 - Superheroic javascript MVW framework* [software]. [přístup 21. prosince 2015]. Dostupné z: <https://angularjs.org/>
- [36] HAVIV, A.: *MEAN.JS v. 0.3.0* [software]. [přístup 20. prosince 2015]. Dostupné z: <http://meanjs.org/>
- [37] GOOGLE, Inc.: *Chrome V8* [online]. [přístup 20. prosince 2015]. Dostupné z: <https://developers.google.com/v8/>

- [38] MALÝ, M.: *Node.js – s JavaScriptem na server*. In: *Zdroják* [online]. Sep 7, 2010 [cit 2015-12-20]. Dostupné z: <https://www.zdrojak.cz/clanky/node-js-s-javascriptem-na-server/>
- [39] npm, Inc.: *Node.js package manager* [online]. [přístup 20. prosince 2015]. Dostupné z: <https://www.npmjs.com/>
- [40] JŮNA, J.: *Základní vlastnosti a výhody Node.JS*. In: *Node.js - Tvorba aplikací pro platformu Node.js* [online]. Nov 17, 2013 [cit 2015-12-20]. Dostupné z: <http://www.nodejs.cz/vlastnosti-a-vyhody-nodejs/>
- [41] KLOBÁSA, P.: *Letmý technologický pohled na MongoDB*. In: *Vsadě na Javu.cz* [online]. Jan 23, 2012 [cit 2015-12-21]. Dostupné z: <http://vsadnajavu.cz/2012-01/odborne/database/letmy-technologicky-pohled-na-mongodb/>
- [42] MongoDB, Inc.: *The MongoDB 3.2 Manual* [online]. [cit 2015-12-21]. Dostupné z: <https://docs.mongodb.org/manual/>
- [43] MROZEK, J.: *Začínáme s AngularJS*. In: *Zdroják* [online]. Nov 30, 2012 [cit 2015-12-22]. Dostupné z: <https://www.zdrojak.cz/clanky/zaciname-s-angularjs/>
- [44] Twitter Inc.: *Bower v. 1.3.8 - A package manager for the web* [software]. [přístup 22. prosince 2015]. Dostupné z: <http://bower.io/>
- [45] Bocoup: *GRUNT v. 0.1.13 - The JavaScript Task Runner* [software]. [přístup 22. prosince 2015]. Dostupné z: <http://gruntjs.com/>
- [46] The Dojo Foundation: *Lodash v. 3.10.1* [software]. [přístup 22. prosince 2015]. Dostupné z: <https://lodash.com/>
- [47] REINMAN, A.: *Nodemailer v. 1.3.0* [software]. [přístup 22. prosince 2015]. Dostupné z: <https://github.com/andris9/Nodemailer>
- [48] HANSON, J.: *Passport v. 0.2.0* [software]. [přístup 22. prosince 2015]. Dostupné z: <http://passportjs.org/>
- [49] GitHub, Inc.: *GitHub* [online]. [přístup 22. prosince 2015]. Dostupné z: <https://github.com/>
- [50] ZGADZAJ, M.: *Benchmarking Node.js - basic performance tests against Apache + PHP*. In: *changeblog-7.x-2.x-dev* [online]. Aug 19, 2010 [cit 2015-12-22]. Dostupné z: <http://zgadzaj.com/benchmarking-nodejs-basic-performance-tests-against-apache-php>
- [51] SEVEN, Y.: *The Pros and Cons of MongoDB*. In: *Valhalla* [online]. Aug 1, 2014 [cit 2015-12-22]. Dostupné z: <http://halls-of-valhalla.org/beta/articles/the-pros-and-cons-of-mongodb,45/>

- [52] Twitter, Inc: *Bootstrap v. 3* [software]. [přístup 22. prosince 2015]. Dostupné z: <http://getbootstrap.com/>
- [53] KELSEY, B.: *OpenGameArt* [online]. [přístup 22. prosince 2015]. Dostupné z: <http://opengameart.org/>
- [54] LINDEIJER, T.: *Tiled Map Editor v. 0.11.0* [software]. [přístup 23. prosince 2015]. Dostupné z: <http://www.mapeditor.org/>
- [55] McMAHON, C.: *Async.js v. 1.5.0* [software]. [přístup 23. prosince 2015]. Dostupné z: <https://github.com/caolan/async>
- [56] *Node-cron v. 1.0.1* [software]. [přístup 23. prosince 2015]. Dostupné z: <https://github.com/ncb000gt/node-cron>
- [57] Mozilla: *Mozilla Firefox v. 41.0.2* [software]. [přístup 29. prosince 2015]. Dostupné z: <https://www.mozilla.org/en-US/>
- [58] PETERSON, J.: *Bootstrap 3 Datepicker v. 4* [software]. [přístup 31. prosince 2015]. Dostupné z: <https://eonasdan.github.io/bootstrap-datetimerangepicker/>
- [59] Pivotal Software, Inc.: *Pivotal Web Services* [online]. [přístup 29. prosince 2015]. Dostupné z: <https://run.pivotal.io>
- [60] CRUVER, W.: *Angular Loading Bar v. 0.8.0* [software]. [přístup 29. prosince 2015]. Dostupné z: <http://chieffancypants.github.io/angular-loading-bar/>



---

## Seznam použitých zkratk a pojmů

**3D** Mající 3 rozměry, třírozměrný

**API** Rozhraní pro programování aplikací, jedná se o sbírku procedur, funkcí, tříd či protokolů, které může programátor využívat.

**Benchmark** Metoda srovnávání hodnot určité veličiny

**Cloud computing** V principu se jedná o jakýkoliv program nebo službu, která je dostupná z jakéhokoliv počítače připojeného k internetu, protože ukládá data nebo celou svou aplikační logiku na server.

**Cron** Softwarový démon, který v operačních systémech automatizovaně spouští v určitý čas nějaký příkaz resp. proces (skript, program apod.).

**CSS** Kaskádové styly, jazyk pro popis zobrazení elementů na webových stránkách

**Dashboard** Informační a ovládací panel aplikace

**DI** Dependency Injection

**Framework** Softwarová struktura, která usnadňuje programování jiných softwarových projektů.

**Frontend** Část webu, která je přístupná běžnému návštěvníkovi.

**Ganttův diagram** Diagram graficky znázorňuje naplánování posloupnosti činností v čase.

**HTML** HyperText Markup Language je název značkovacího jazyka používaného pro tvorbu webových stránek.

## A. SEZNAM POUŽITÝCH ZKRATEK A POJMŮ

---

**HTML5 canvas** Slouží k dynamickému vykreslování bitmap a grafických primitiv.

**JSON** JavaScript Object Notation

**MVC** Softwarová architektura Model-View-Controller

**MVW** Softwarová architektura Model-View-Whatever, kde Whatever znamená „co vám funguje“.

**NoSQL** Databázový koncept, ve kterém datové úložiště i zpracování dat používají jiné prostředky než tabulková schémata tradiční relační databáze.

**Plugin** Doplňkový modul aplikace, který rozšiřuje její funkčnost.

**SŘBD** Systém řízení báze dat

**XP** Zkušební body



---

## Instalační příručka

Před spuštěním aplikace je třeba nainstalovat server Node.js a databázi MongoDB. Uvedeme postup pro operační systémy Windows a různé distribuce Linuxu.

### B.1 Instalace Node.js verze 0.12.2

Pro správný chod aplikace je třeba zprovoznit server Node.js ve verzi 0.12.2. Instalace na operačním systému Windows je jednoduchá. Na přiloženém CD ve složce `software/nodejs/Windows/x64` nebo `x86` najdete spustitelný instalační soubor, kterým Node.js nainstalujete.

Před instalací na linuxových distribucích nejdříve zkontrolujte, že systém obsahuje následující položky:

- GCC 4.2 nebo novější
- G++ 4.2 nebo novější
- Python 2.6 nebo 2.7
- GNU Make 3.81 nebo novější
- libexecinfo (pouze FreeBSD a OpenBSD)

Pokud některou z podmínek nespĺňujete, proveďte nápravu a poté pokračujte následujícím postupem:

1. Zkopírovat soubor `software/nodejs/Linux/node-v0.12.2.tar.gz` z přiloženého CD do složky ve vlastním adresáři
2. Otevřít terminál a přepnout se do složky se zkopírovaným souborem
3. Rozbalit archiv příkazem `tar -xvzf node-v0.12.2.tar.gz`

4. Přejít do nově rozbalené složky příkazem `cd` a následně spustit příkaz `./configure`
5. Spustit příkaz `make`
6. Spustit příkaz `[sudo] make install`

### B.2 Instalace MongoDB verze 3.2

Na přiloženém CD ve složce `software/mongodb/windows` najdete spustitelný instalační soubor, kterým nainstalujete databázi MongoDB. Pro data databáze je třeba vytvořit na disku počítače adresářovou strukturu `C:/data/db`.

Instalace MongoDB na linuxových operačních systémech je opět poněkud složitější:

1. Soubor v adresáři `software/mongodb/Linux` zkopírovat do adresáře na disku
2. Otevřít terminál a přepnout se do složky se zkopírovaným souborem
3. Rozbalit archiv: `tar -zxvf mongodb-linux-x86_x64-3.2.0.tgz`
4. Obsah zkopírovat do složky, ze které se bude MongoDB spouštět
5. Řádek `export PATH=<mongodb-cesta>/bin:$PATH` přidat do souboru `~/.bashrc`
6. Příkazem `mkdir -p /data/db` vytvořit složku pro data databáze a nastavit oprávnění pomocí `sudo chmod 777 /data/db`

### B.3 Import databáze

Pro aplikaci jsou připravena vzorová data (včetně účtu administrátora), která je potřeba naimportovat do databáze. Nejprve spusťte databázový server:

- V příkazové řádce Windows pomocí "`C:/Program Files/MongoDB/Server/3.2/bin/mongod.exe`"
- V terminálu linuxových operačních systémů pomocí `mongod`

Na přiloženém CD se ve složce `src/database_dump` nachází soubory potřebné pro import databáze. V novém okně příkazové řádky Windows přejděte příkazem `cd` do složky "`C:/Program Files/MongoDB/Server/3.2/bin`" a odtud spusťte `mongorestore -db=bp-app cd-disk/src/database_dump/`. U systémů založených na Linuxu stačí v novém okně terminálu spustit zmíněný příkaz `mongorestore` s totožnou syntaxí (za předpokladu, že jste upravili soubor `~/.bashrc`).

## B.4 Spuštění aplikace

Zdrojové soubory aplikace, které se nachází v adresáři `src/implementation` na příloženém CD, zkopírujte do složky na disku. Budete ji odtud následně spouštět. V okně příkazové řádky nebo terminálu, kde nám neběží databázový server, přejděte do složky se zkopírovanou aplikací. Proveďte zde následující sadu příkazů (sudo připisujte na začátek v linuxových systémech):

1. `[sudo] npm install -g bower`
2. `[sudo] npm install -g grunt-cli`
3. `[sudo] npm install`
4. `set NODE_ENV=production` ve Windows,  
`export NODE_ENV=production` v linuxových operačních systémech
5. `node server.js`

Aplikace by v tuto chvíli měla být spuštěná a přístupná v prohlížeči na adrese `localhost:3000`. Přihlásit se do ní můžete pomocí uživatelů: `admin`, `first`, `third`, `fifth`. Všichni mají definováno stejné heslo `bachelor123`. Pokud přihlášení nefunguje, buď nemáte nainportovanou databázi nebo jste nenastavili proměnnou `NODE_ENV` na hodnotu `production`. Do vytvořené hry lze zaregistrovat ještě další hráče pod emailovými adresami `second@player.cz` a `fourth@player.cz`.



---

## Obsah přiloženého CD

readme.txt.....	stručný popis obsahu CD
software .....	instalační soubory webového a databázového serveru
├─ nodejs.....	instalační soubory Node.js
├─ mongodb.....	instalační soubory MongoDB
src	
├─ database_dump.....	soubory pro import databáze
├─ implementation.....	zdrojové kódy implementace
│   ├─ app .....	serverová část aplikace
│   ├─ config.....	konfigurační soubory
│   ├─ public .....	klientská část aplikace
├─ thesis .....	zdrojová forma práce ve formátu $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$
text .....	text práce
├─ thesis.pdf .....	text práce ve formátu PDF