

## **Abstrakt**

text

## **Summary**

text

## **Klíčová slova**

text

## **Key words**

text



**Bibliografická citace**

text

## Čestné prohlášení

Prohlašuji, že předložená diplomová práce je původní a zpracoval jsem ji samostatně.  
Prohlašuji, že citace použitých pramenů jsou úplné a že jsem ve své práci neporušil autorská práva (ve smyslu Zákona č. 121/2000 Sb., o právu autorském a o právech souvisejících s právem autorským).

V Brně dne xx. x. 2017

.....

jméno

## Poděkování

text

# Obsah

<b>ÚVOD</b>	<b>7</b>
<b>CÍL DIPLOMOVÉ PRÁCE</b>	<b>8</b>
<b>METODIKA PRÁCE</b>	<b>9</b>
Metody . . . . .	9
Postupy . . . . .	9
<b>1 TEORETICKÁ VÝCHODISKA</b>	<b>10</b>
1.1 Vývojové platformy low-code . . . . .	10
1.1.1 Příklady low-code platforem . . . . .	11
1.1.2 Cloudový výpočet . . . . .	11
1.2 Bussiness Intelligence . . . . .	14
1.2.1 Big data . . . . .	15
1.2.2 Data-mining . . . . .	15
1.2.3 Extraction, Transaction, Loading . . . . .	15
1.2.4 NoSql databáze . . . . .	15
1.3 Platforma pro pokročilou vizualizaci dat . . . . .	15
1.3.1 Single page aplikace . . . . .	15
1.3.2 Dynamická a interaktivní vizualizace dat . . . . .	15
1.3.3 Webová služba RESTful . . . . .	15
1.4 Server pro řízení přístupu a identity . . . . .	15
1.4.1 JSON Web Token . . . . .	15
<b>2 ANALYTICKÁ ČÁST</b>	<b>16</b>
<b>3 VLASTNÍ NÁVRHY</b>	<b>17</b>
<b>ZÁVĚR</b>	<b>18</b>
<b>Seznam obrázků</b>	<b>19</b>
<b>Seznam tabulek</b>	<b>20</b>
<b>Seznam použitých zdrojů</b>	<b>21</b>

# ÚVOD

tex

## CÍL DIPLOMOVÉ PRÁCE

Hlavním cílem této práce je vytvořit platformu, která bude schopná zpracovat data zadaná uživatelem, analyzovat je a na základě vnitřní logiky a informace nesené v těchto datech je uložit do logických celků. Tyto celky poté zobrazit uživateli, nechat jej dále definovat dodatečné informace, provádět reporting případně zobrazit v přehledných grafech.

Platforma se bude zaměřovat převážně na uživatelské rozhraní, tak aby její používání bylo co nejintuitivnější a nejjednodušší.



# METODIKA PRÁCE

## Metody

Nejdůležitějším zaměřením této platformy je uživatelská přívětivost a jednoduchost na používání, proto bude při vývoji kladen důraz na spokojenost uživatelů. Tohoto bude dosaženo použitím agilních metodik při vývoji, kdy bude postupně dodáváný produkt předáván úzkému kruhu uživatelů, kteří se budou vyjadřovat k uživatelskému rozhraní. Platforma bude psána jako webová aplikace, která bude přistupovat do databáze přes rozhraní napsané v jazyce Java.

V části zpracování dat bude použito několik ETL metodik a data mining technik, které povedou k získání logických informací ze zadaných informací. Platforma bude vyvíjena s možností škálovatelnosti a použití nad velkým objemem dat.

## Postupy

K vytvoření co nejpřívětivější platformy budou využity zkušenosti a knižní publikace zabývající se tímto tématem. Dále budou analyzovány jednotlivé postupy zadávání dat uživatelů do takového systému, které povedou ke zpřehlednění a zjednodušení používání.

Pro komunikaci se serverem bude použit standard REST, který usnadní komunikaci se serverem a umožní případné navázání nových aplikací. V případě že bude vytvořena mobilní aplikace pro získávání dat nebude nutné psát znovu stejnou nebo podobnou logiku.

Pro zabezpečené přihlášení do aplikace bude použit autentikační server, který bude zajišťovat vytváření a správu uživatelů spolu s jejich právy.

# 1 TEORETICKÁ VÝCHODISKA

Pro plné pochopení, výběru a případném vypracování platformy je potřeba si objasnit a vysvětlit několik témat. Jsou to především *Vývojové platformy low-code*, výsledná platforma by měla splňovat tuto definici. Dále si objasníme pojmy *Business intelligence* (platforma bude z části pracovat s touto oblastí) a *Platforma pro pokročilou vizualizaci dat* – pro snadné používání uživatelského rozhraní. A vzhledem k tomu že výsledná platforma musí do určité části pracovat s uživatelskými právy a spravovat uživatele, objasníme si pojem *Server pro řízení přístupu a identity*

## 1.1 Vývojové platformy low-code

Vývojové platformy low-code jsou celkem nový pojem, tyto produkty začali vznikat, protože malé a střední podniky potřebovali vytvořit rychle a za použití menšího počtu vývojářů aplikace, které mohou být nadále rychle spravovány. [1]

Toto v podstatě znamená, že vývojáři mohou rychle měnit software na základě uživatelských požadavků, což má za následek spokojenější uživatele, uživatelsky přívětivější software a toto všechno za minimálního použití ručního programování. Takovéto platformy neeliminují programování jako takové, ale napomáhají rychlejšímu vývoji, tak že poskytují vizuální nástroje a napomáhají konfiguraci datových modulů a pomáhají eliminovat problémy spojené s datovou integrací. [2]

### Výhody low-code platforem

- **Produktivita:** Systémy mohou být vyvíjeny a nasazeny během menšího časového rozmezí, oproti klasickému programování. [3]
- **Reakční schopnost:** Vývojář může často zvolit různé druhy platforem na kterých bude výsledný produkt fungvat, od mobilních aplikací, až po webové služby. [3]
- **Spolehlivost:** Aplikace mohou být aktualizovány mnohem rychleji, což má za následek jejich stabilitu a spolehlivost. [3]
- **Úspora času a peněz:** Vývojáři mohou vytvořit mnohem více funkcionality za kratší čas, z čehož plyne že si firma může dovolit menší počet programátorů. [3]

- **Zaměření na samotný vývoj:** Zaměřením na to co má aplikace dělat, a ne jak to má dělat, programátoři se mohou zaměřit na funkcionalitu a uživatelskou spokojenost. Při vývoji je možné se zaměřit také více na uživatelské požadavky mnohem rychleji. [3]

### 1.1.1 Příklady low-code platforem

**Microsoft PowerApps** Vývojová platforma od firmy Microsoft, která dovoluje vytvořit během několika málo kliknutí aplikaci pro mobilní platformy a také jako webové služby. Při spojení této platformy a aplikace Power BI vzniká velice robustní vývojářský nástroj, díky kterému je možné rychle integrovat produkční data do aplikace, kterou budou uživatelé rádi používat. [1]

**Zoho Creator** Výhodou této platformy je využití techniky „drag-and-drop“, která umožňuje vytvářet aplikace a převážně jejich uživatelské rozhraní bez nutnosti psát jakýkoliv kód. [4]

**Rollbase** Při používání této platformy vývojář jako první definuje objekty, jejich vlastnosti a vztahy mezi těmito objekty. Po překonání tohoto kroku máme již plně funkční webovou aplikaci, která je funkční napříč všemi mobilními zařízeními. [5]

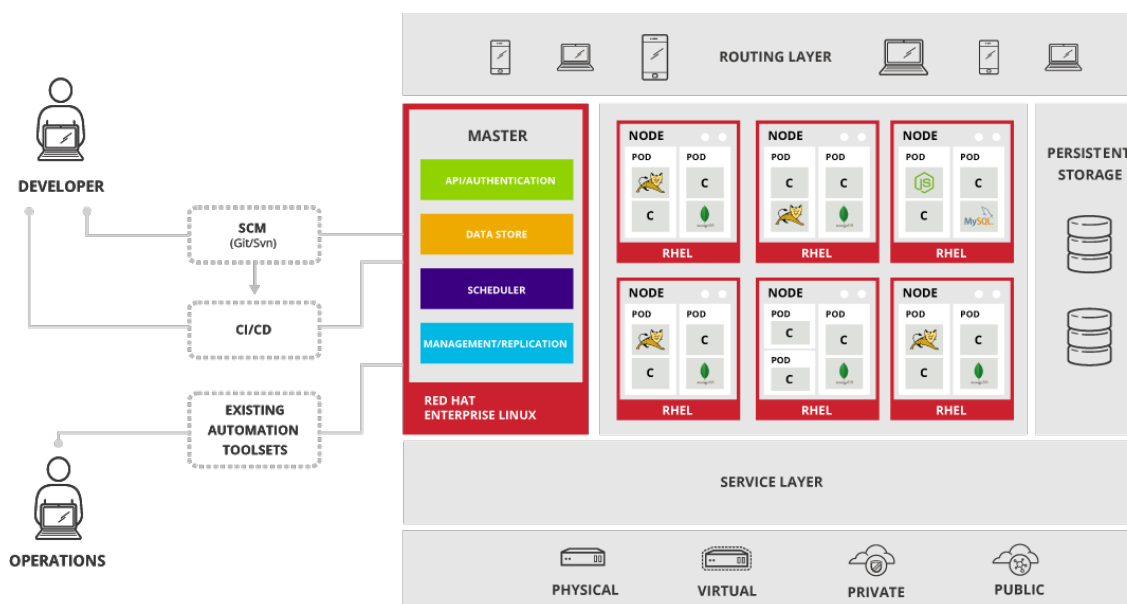
**Openshift** Platforma pro vývoj webovým a mobilních aplikací, postavená na kontejnerech, které zajišťují rychlý vývoj a možnost dedikovat vývojáře na vytvoření jednoduchých funkcionalit jako samostatné aplikace <sup>1</sup>, které za pomoci Openshiftu vytvoří velkou a komplexní aplikaci. Na Obrázku 1.1.1 můžeme vidět z jakých vrstev se skládá Openshift a můžeme vidět jednotlivé aplikace zázorněné v nodech, dále můžeme vidět jaké nástroje nabízí Openshift vývojářům (management zdrojových kódů – SCM a kontinualní integrace – CI/CD).[6]

### 1.1.2 Cloudový výpočet

Výpočet pomocí cloudu je další fáze ve vývoji internetu, cloud v tomto použití znamená že všechno potřebné pro vývoj a hostování aplikací, až po samostatné stroje je možné nabídnout jako služba kekoliv na světě se člověk nachází. I když

---

<sup>1</sup>Takovýmto aplikacím se říká Microservice <https://smartbear.com/learn/api-design/what-are-microservices/>



Obrázek 1.1: Znoznění jednotlivých vrstev v platformě openshift.

poskytovatelé cloudových řešení mají často velice robustní bezpečnostní systém je na uživateli, který má uložené data v cloudu, aby zajistil jejich bezpečnost. To znamená že pokud data uniknou z cloudu díky špatnému zabezpečení v aplikaci, která je hostovaná, není chyba poskytovatele, ale firmy, která takovou aplikaci vydala.[7]

Pporovnání ceny cloudového výpočtu a klasického datového uložště není až tak jednoduché, záleží na někkolika faktorech. Vezměme si například lokaci datového uložště, pokud například cena elektické energie v místě datového uložště je velice levná firma nemusí být tolik tlačena do cloudového řešení. ale pokud k těmto datům přistupuje velké množství uživatelů z různých koutů světa může se stát že námi poskytované služby budou neresponzivní a uživatelé mohou odejít ke konkurenci. V tomto momentě je potřeba zvážit zda se nám cloudové řešení vyplatí a kdy ne. Důležité je také uvědomit si, že 42 % nákladů na datové uložště jde do hardware a software (tyto náklady jsou rozloženy v průběhu času) a 58 % nákladů jde do topení, klimatizace, daní a samostatné práce. [7]

## Typy cloudového výpočtu

- **Veřejné** dostupné pro širokou veřejnost, jak zdarma, tak placené verze.
- **Soukromé** často používané firmami skupinami uživatelů, kteří potřebují zabezpečit data. Často velice drahé a ačasově nákladné řešení.

- **Komunitní** podobné soukromým, ale rozšířené mezi větší skupiny lidí.
- **Hybridní** vytvořené z jednoho a více druhů, privatního a nebo veřejného cloudu. Mezi jejich portfoliočastopatří zálohakekritickýmslužbám.
- **DaaS** data jako služba, pouze data uložené v cloudu.
- **PaaS** pro vývoj a hostování celého vývojového cyklu, často včetně možnosti nasazení výsledné aplikace.
- **IaaS** infrastruktura jako služba, opravdové, nebo virtuální počítače nabízené uživatelům. [8]

### Platforma jako služba – PaaS

Platforma jako služba tento pojem označuje službu, která zahrnuje kompletní škálu nástrojů sloužících pro vývoj aplikací. Od databází, přes aplikační rámce a testovací nástroje až po nasazení a překlad aplikace. Výhoda této služby je převážně v tom, že všechno je přístupné přes internet a často jako webová aplikace, takže není nutné kupovat často velice drahé nástroje. Někteří poskytovatelé nabízí také vyrovnaní zatížení, to znamená že pokud je výsledná aplikace pod vysokým náparem uživatelů, automaticky se přiřadí prostředky, aby uživatelé nezaznamenali pád aplikace a bez nutnosti zasáhnout do nastavení služby.[9]

### Výhody PaaS

- **Méně kódu** - Díky možnosti projení několika menších aplikací dohromady není potřeba psát stejný kus kódu pořád dokola.
- **Nové možnosti bez potřeby nabírání nových lidí** - Díky PaaS dostane tým do rukou sofistikovanější nástroje.
- **Vývoj pro více platform** - Výhodou mnoha poskytovatelů služby PaaS je možnost překladu aplikací pro různé platformy (několik mobilních platforem a webová aplikace).
- **Propojení geograficky nesourodých týmů** - Pokud je tým rozdělen po různých částech světa služby PaaS dovolují takto rozděleným týmům pracovat efektivněji.
- **Efektivní životní cyklus aplikace** - V rámci integrovaného prostředí se často nachází funkce pro podporu životního cyklu aplikace (sestavení, nasazení,

otestování, správa, aktualizace...). [10]

## 1.2 Bussiness Intelligence

Nástroje pro bussiness Intelligence zahrnují jak samostatná data, tak časovou jednotku, takže můžeme nad těmito daty provádět predikci pomocí sofistikovaných nástrojů a výpočtů. Ze začátku bylo jednoduché provádět takové výpočty, protože jednoduše firma nesbírala takové množství dat. Aktuálně však není v lidských silách provádět takové výpočty nad tak obrovským množstvím dat.

Pravděpodobně nejvíce rozšířeným aplikováním data-miningu je marketing – sledování nakupování a chování zákazníků. V této oblasti je možné vybrat si každého zákazníka, a v případě že máme dostatek dat, cílit na něj lépe reklamu.

Pro plné využití BI nástrojů potřebujeme sledované subjekty rozdělit do několika skupin, tyto skupiny musí být co nejvíce **Heterogenní** (rozdílné) vůči sobě, a subjekty v rámci jedné skupiny musí být na druhou stranu co nejvíce **Homogenní** (stejně). Pokud sledované subjekty spadají do více skupin (toto se může hodit z mnoha důvodů) můžeme použít takzvané **Clustery**. Což jsou skupiny objektů, které si jsou co nejvíce podobné, ale objekty v jiné skupině se jim budou co nejvíce lišit.

Samostatná data jako taková však nejsou to nejdůležitější, pokud nebudeme schopni vhodně interpretovat a využít data, která jsme nashromáždili jsou nám k ničemu, z toho důvodu vzniklo strojové učení. V podstatě to znamená, že pokud předáme stroji dostatečně velké množství dat a nastavíme správně parametry, tak nám mohou stroje umožnit rychlejší interpretaci takových dat. Ale na předpovědi je nutno nahlížet s odstupem a nebrat je příliš vážně a přesně, naštěstí například pro úspěšný prodej není potřeba přesných předpovědí, stačí pouze vědět kdy a komu poslat přesně cílenou reklamu. Pokud se systém trefí do těhotot kritérií je velká pravděpodobnost že si námi zacílený zákazník pořídí produkt, který se mu snažíme prodat.

### **1.2.1 Big data**

### **1.2.2 Data-mining**

### **1.2.3 Extraction, Transaction, Loading**

### **1.2.4 NoSql databáze**

[15]

## **1.3 Platforma pro pokročilou vizualizaci dat**

### **1.3.1 Single page aplikace**

### **1.3.2 Dynamická a interaktivní vizualizace dat**

### **1.3.3 Webová služba RESTful**

Restful web APIs

## **1.4 Server pro řízení přístupu a identity**

[http://www.sersc.org/journals/IJMUE/vol9\\_no9\\_2014/9.pdf](http://www.sersc.org/journals/IJMUE/vol9_no9_2014/9.pdf)

### **1.4.1 JSON Web Token**

<https://tools.ietf.org/html/rfc7519> <https://scotch.io/tutorials/the-anatomy-of-a-json-web-token>

## 2 ANALYTICKÁ ČÁST

text



### **3 VLASTNÍ NÁVRHY**

text[?][?]

## ZÁVĚR

## Seznam obrázků

1.1	Znázornění jednotlivých vrstev v platformě openshift. . . . .	12
-----	---	----

## Seznam tabulek

## Literatura

- [1] MARVIN, Rob. Building an App With No Coding: Myth or Reality. *Pcmag*[online]. Ziff Davis, LLC. PCMag Digital Group, 2016 [cit. 2017-03-11]. Dostupné z: <http://www.pcmag.com/article/345661/building-an-app-with-no-coding-myth-or-reality>
- [2] RUBENS, Paul. Use Low-Code Platforms to Develop the Apps Customers Want. *CIO* [online]. IDG Communications, 2014 [cit. 2017-03-11]. Dostupné z: <http://www.cio.com/article/2845378/development-tools/use-low-code-platforms-to-develop-the-apps-customers-want.html>
- [3] MARVIN, Rob. How low-code development seeks to accelerate software delivery. *SD Times* [online]. BZ Media LLC., 2014 [cit. 2017-03-11]. Dostupné z: <http://sdtimes.com/low-code-development-seeks-accelerate-software-delivery/>
- [4] Zoho Creator REVIEW. *Finances Online* [online]. [cit. 2017-03-11]. Dostupné z: <https://reviews.financesonline.com/p/zoho-creator/>
- [5] CIOT, Thierry. What is a Low-Code Platform? *Progress* [online]. 2016 [cit. 2017-03-11]. Dostupné z: <https://www.progress.com/blogs/what-is-a-low-code-platform>
- [6] OpenShift Origin Overview. *OpenShift Origin* [online]. [cit. 2017-03-11]. Dostupné z: <https://docs.openshift.org/latest/architecture/index.html>
- [7] HURWITZ, Judith. *Cloud computing for dummies*. Hoboken, NJ: Wiley Pub., c2010. ISBN 978-0470484708.
- [8] ROME, C. H. *The cloud computing Book: The ultimate guide to mastering cloud computing*. Fifth edition. Bernemouth: Imagine Publishing, 2015.
- [9] CHANDRASEKARAN, K. *Essentials of Cloud Computing*. Maiami: CRC Press, 2014. ISBN 978-1482205435.
- [10] Co je PaaS?: Platforma jako služba. *Microsoft Azure* [online]. [cit. 2017-03-12]. Dostupné z: <https://azure.microsoft.com/cs-cz/overview/what-is-paas/>
- [11] HARTSON, H. Rex. a Pardha S. PYLA. *The UX Book: process and guidelines for ensuring a quality user experience*. Boston: Elsevier, c2012. ISBN 978-

0123852410.

- [12] MARZ, Nathan a James WARREN. *Big data: principles and best practices of scalable real-time data systems*. ISBN 978-1617290343.
- [13] ALLAMARAJU, Subrahmanyam. *RESTful Web services cookbook*. Sebastopol, CA.: O'Reilly, c2010. ISBN 978-0596801687.
- [14] HAN, Jiawei, Micheline KAMBER a Jian PEI *Data mining: concepts and techniques*. 3rd ed. Haryana, India ; Burlington, MA: Elsevier, 2012. ISBN 9789380931913.
- [15] SADALAGE, Pramod J. a Martin FOWLER *NoSQL distilled: a brief guide to the emerging world of polyglot persistence*. Upper Saddle River, NJ: Addison-Wesley, c2013. ISBN 978-0321826626.