



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA INFORMAČNÍCH TECHNOLOGIÍ**

FACULTY OF INFORMATION TECHNOLOGY

**ÚSTAV INFORMAČNÍCH SYSTÉMŮ**

DEPARTMENT OF INFORMATION SYSTEMS

**NÁSTROJ PRO PODPORU PROVÁDĚNÍ ANALÝZY  
FINANČNÍHO ZDRAVÍ FIRMY**

TOOL FOR EXECUTION OF A FIRM'S FINANCIAL HEALTHINESS ANALYSIS

**DIPLOMOVÁ PRÁCE**

MASTER'S THESIS

**AUTOR PRÁCE**

AUTHOR

**Bc. JAN KUBIŠ**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**Ing. ŠÁRKA KVĚTOŇOVÁ, Ph.D.**

**BRNO 2018**

## **Zadání diplomové práce**

Řešitel: **Kubiš Jan, Bc.**

Obor: Informační systémy

Téma: **Nástroj pro podporu provádění analýzy finančního zdraví firmy**  
**Tool for Execution of a Firm's Financial Healthiness Analysis**

Kategorie: Databáze

### **Pokyny:**

1. Seznamte se s problematikou finančního plánování a prováděním finanční analýzy firmy. Zaměřte se na hlavní finanční ukazatele, které se k finanční analýze firmy používají.
2. Prostudujte a zhodnoťte současné možnosti softwarové podpory provádění finanční analýzy firmy.
3. Navrhněte vlastní koncepci programového řešení (podpory) pro provádění analýzy finančního zdraví firmy s důrazem na hlavní finanční ukazatele (rentabilita aktiv, vlastního kapitálu, vnitřní výnosové procento a další). Zvolte vhodné implementační prostředí pro její realizaci.
4. Po konzultaci s vedoucí realizujte prototyp navrženého systému, včetně důkladného otestování.
5. Použitelnost systému demonstруйте na vhodně zvoleném vzorku dat. Zhodnoťte dosažené výsledky a diskutujte možnosti dalšího rozvoje vytvořeného produktu.

### **Literatura:**

- Sedláček, J.: Finanční analýza podniku. Computer Press, ISBN 978-80-251-1830-6.
- Růčková P.: Finanční analýza - 3. rozšířené vydání - metody, ukazatele, využití v praxi. Grada, ISBN 978-80-247-3308-1.
- Holečková, J.: Finanční analýza firmy. Aspi, ISBN 978-80-7357-392-8.
- Williams, H. E., Lane, D.: PHP a MySQL - Vytváříme webové databázové aplikace. Computer Press, 2002, 552 s. ISBN 8072267604

Při obhajobě semestrální části projektu je požadováno:

- Body 1 až 3.

Podrobné závazné pokyny pro vypracování diplomové práce naleznete na adrese

<http://www.fit.vutbr.cz/info/szz/>

Technická zpráva diplomové práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap, které byly vyřešeny v rámci dřívějších projektů (30 až 40% celkového rozsahu technické zprávy).

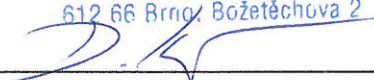
Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Květoňová Šárka, Ing., Ph.D., UIFS FIT VUT**

Datum zadání: 1. listopadu 2017

Datum odevzdání: 23. května 2018

**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**  
Fakulta informačních technologií  
Ústav informačních systémů  
602 00 Brno, Božetěchova 2

  
doc. Dr. Ing. Dušan Kolář  
vedoucí ústavu

## Abstrakt

Tématem této práce je finanční analýza podniku a investice a popis programového řešení pro její provádění, které bylo v rámci práce implementováno. V první části je popsána finanční analýza obecně – vysvětlení pojmů, shrnutí motivace, vymezení množiny vstupních dat, výčet používaných metod a ukazatelů, jejich význam a interpretace jejich hodnot. Druhá část práce obsahuje popis návrhu, implementace a fungování nástroje, jehož účelem je automatizace vybraných oblastí finanční analýzy.

## Abstract

The topic of this thesis is financial analysis of both, a company and an investment. It also contains description of software for its execution, which was implemented within the thesis. The first part describes financial analysis in general – definitions, motivation, input data set definition, enumeration of common methods and indicators, their meaning and interpretation of their values. The second part contains description of design, implementation and functionality of a tool, which purpose is automatization of chosen domains from financial analysis.

## Klíčová slova

finanční analýza, hodnocení finančního zdraví, nástroj pro finanční analýzu, finanční ukazatele

## Keywords

financial analysis, financial healthiness rating, financial analysis tool, financial indicators

## Citace

KUBIŠ, Jan. *Nástroj pro podporu provádění analýzy finančního zdraví firmy*. Brno, 2018. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Šárka Květoňová, Ph.D.

# Nástroj pro podporu provádění analýzy finančního zdraví firmy

## Prohlášení

Prohlašuji, že jsem tuto práci vypracoval samostatně pod vedením Ing. Šárky Květoňové, Ph.D. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....  
Jan Kubiš  
2. září 2018

## Poděkování

Děkuji především Ing. Šárce Květoňové, Ph.D. za odborné vedení mé diplomové práce. Děkuji také Bc. Patriku Hovorkovi za užitečné rady v oblasti finanční analýzy.

# Obsah

<b>1</b>	<b>Úvod</b>	<b>3</b>
<b>2</b>	<b>Finanční analýza</b>	<b>4</b>
2.1	Předmět, motivace a cíle . . . . .	4
2.2	Vstupní data . . . . .	4
2.3	Absolutní ukazatele . . . . .	6
2.3.1	Horizontální analýza . . . . .	6
2.3.2	Vertikální analýza . . . . .	7
2.4	Poměrové ukazatele . . . . .	7
2.4.1	Ukazatele rentability . . . . .	7
2.4.2	Ukazatele likvidity . . . . .	10
2.4.3	Ukazatele aktivity . . . . .	11
2.4.4	Ukazatele zadluženosti . . . . .	13
2.5	Ostatní . . . . .	14
2.5.1	Analýza peněžních toků . . . . .	14
2.5.2	Du Pont rozklad . . . . .	15
2.5.3	Vnitřní výnosové procento . . . . .	16
<b>3</b>	<b>Návrh řešení</b>	<b>17</b>
3.1	Existující software . . . . .	17
3.2	Vybrané metody vyhodnocování . . . . .	18
3.3	Návrh programového řešení . . . . .	20
3.3.1	Zadávání vstupních dat . . . . .	20
3.3.2	Architektura . . . . .	21
3.3.3	Klíčkové knihovny a technologie . . . . .	24
3.4	Shrnutí a další postup . . . . .	27
<b>4</b>	<b>Implementace a testování</b>	<b>29</b>
4.1	Implementace finanční analýzy . . . . .	29
4.1.1	Vnitřní reprezentace vstupních dat . . . . .	30
4.1.2	Společné položky různých struktur výkazů . . . . .	33
4.1.3	Provádění analýzy . . . . .	33
4.2	Implementace prostředí . . . . .	37
4.2.1	Datový tok . . . . .	37
4.2.2	Uživatelské rozhraní . . . . .	38
4.3	Testování . . . . .	42
4.4	Vývoj . . . . .	44
4.5	Možnosti rozšíření nástroje . . . . .	44

<b>5 Závěr</b>	<b>46</b>
<b>Literatura</b>	<b>47</b>
<b>A Obsah CD</b>	<b>48</b>

# Kapitola 1

## Úvod

Cílem této práce je popis technické realizace nástroje, který je zaměřen na provádění finanční analýzy vybraných ekonomických objektů. Součástí práce je i teorie spojená s finanční analýzou v rozsahu, který implementovaný nástroj pokrývá.

Finanční analýza je v podnikové sféře soubor činností umožňující zjistit a vyhodnotit finanční situaci podniku na základě dat o jeho hospodaření. Existuje mnoho způsobů jak analýzu provádět a objektem analýzy mohou být i jiné entity z oblasti ekonomie. Její provádění a zkoumání výsledků je jedním z předpokladů pro kvalitní a komplexní vedení podniku. Cílovou skupinu produktu tedy tvoří finanční analytici, manažeři společností i běžní podnikatelé, jimž má usnadnit provádění této činnosti.

Práce obsahuje dva tematické celky. Nejprve bude provedeno shrnutí teorie spojené s prováděním finanční analýzy – co je to finanční analýza, jaké jsou její cíle, jaká jsou vstupní data a co je výstupem, vysvětlení pojmů, používaných metod a strategií, konstrukcí a vypočtů jednotlivých ukazatelů finanční analýzy a interpretací jejich hodnot. Druhým tematickým celkem je popis nástroje, který finanční analýzu provádí a který bude v rámci této diplomové práce implementován.

Text práce je členěn do pěti kapitol. Jelikož je finanční analýza poměrně rozsáhlá oblast, celá příští kapitola je věnována prvnímu celku, teorii.

Ve třetí kapitole jsou ještě před vytvářením vlastní koncepce podrobeny analýze již existující nástroje, abychom si mohli udělat obrázek o tom, jak tuto problematiku řešili jiní. Na základě zjištěných poznatků je vypracován návrh programového řešení vlastního nástroje.

Čtvrtá kapitola se zabývá realizací návrhu, implementačními detaily a fungováním nástroje. Jsou zde zmíněny i návrhy na možná rozšíření.

V závěrečné kapitole jsou shrnuty a zhodnoceny dosažené výsledky a přínosy této práce.

## Kapitola 2

# Finanční analýza

### 2.1 Předmět, motivace a cíle

Finanční analýza podniku je metoda hodnocení finančního hospodaření podniku, při které se získaná data třídí, agregují, poměřují mezi sebou navzájem, kvantifikují se vztahy mezi nimi, hledají se mezi nimi kauzální souvislosti a určuje se jejich vývoj. Tím se zvyšuje vypovídací schopnost zpracovávaných dat, zvyšuje se jejich informační hodnota.[8]

Motivací je posouzení finanční situace podniku a její příčiny, do jisté míry i k predikci stavu podniku v budoucnu, odhalení důsledků vlastních rozhodnutí, identifikaci slabých míst, plánování, zhodnocení hospodaření, vyvozování závěrů o vlastnostech podniku, ohodnocení podniku a vyvozování dalších užitečných informací.

Mimo to se v praxi lze setkat s případy, kdy je pro dosažení podnikatelského záměru provedení finanční analýzy nezbytné. Její výsledky jsou směrodatné například při rozhodování banky mezi poskytnutím a neposkytnutím úvěru, pro vyměření výše úroků.

Při finanční analýze se používají dva navzájem propojené přístupy[3]:

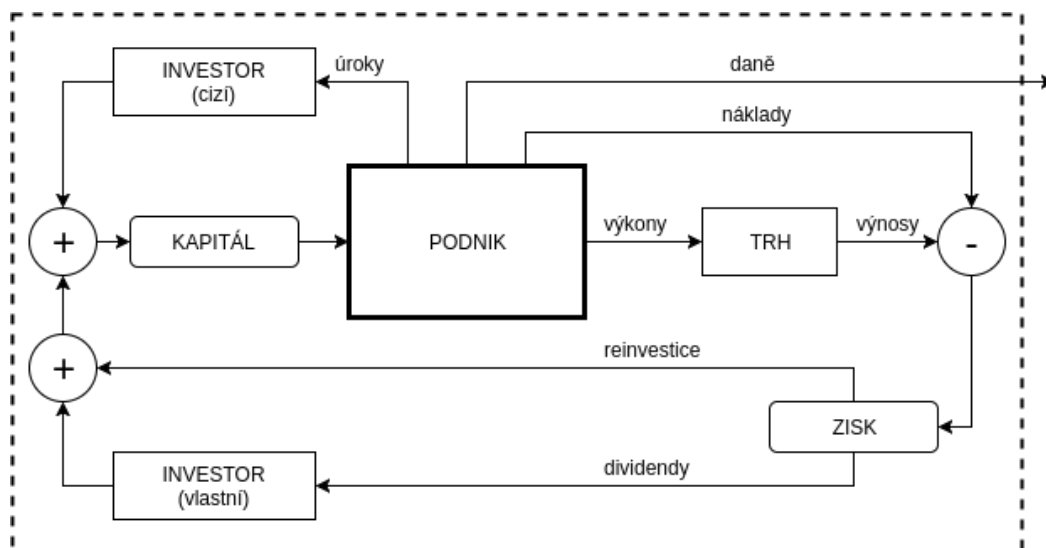
1. kvalitativní (fundamentální) analýza, založená na znalostech souvislostí mezi ekonomickými i mimoekonomickými jevy, zkušenostech odborníků a na jejich subjektivních odhadech
2. kvantitativní (technická) analýza, která využívá matematické, statistické a jiné algoritmizovatelné postupy

V této práci se budeme zabývat především kvantitativní analýzou, právě z důvodu možnosti její algoritmizace a tím pádem možnosti strojového provádění. Výstup této analýzy poté poslouží jako užitečný podklad pro analýzu kvalitativní, kterou provede samotný uživatel.

### 2.2 Vstupní data

Abychom mohli provést finanční analýzu podniku nebo investice, musíme mít data, ze kterých budeme vycházet. Množina vstupních dat není nijak omezena a může být vskutku rozsáhlá – při výpočtu a odvozování závěrů lze zohlednit nejružnější faktory, ekonomické i mimoekonomické. Pro nás budou však klíčová především data o vlastním hospodaření podniku, za jejichž evidenci je zodpovědná účetní jednotka podniku, která je v průběhu roku zapisuje do takzvané účtové osnovy.





Obrázek 2.1: Schéma finančních toků podniku [8]

Při analýze investice se vychází především z atributů investice, jako je například doba investice, úroková sazba nebo výnosy a výdaje s investicí spojené.

### Účtová osnova

Právníké osoby, fyzické osoby zapsané v obchodním rejstříku a některé další subjekty jsou ze zákona o účetnictví povinny vést podvojně účetnictví, což implikuje povinnost účtovat podle účtové osnovy. Všechny peněžní toky a prostředky musí být evidovány na příslušném účtu. Přestože je účtů nespočet, obecně se dají rozdělit do dvou skupin – rozvahové a výsledkovkové. Tyto subjekty jsou dále povinny každoročně vypracovat a zveřejnit účetní závěrku, která je tvořena dvěma finančními výkazy, jimiž jsou rozvaha a výkaz zisku a ztráty (výsledovka). Hodnoty položek ve výkazu plynou právě ze sumy hodnot jednotlivých účtů účtové osnovy (rozvahových pro rozvahu, výsledkovkových pro výsledovku).

Tyto dva výkazy jsou spolu s přehledem o peněžních tocích stěžejním výstupem účetnictví:

- výkaz zisku a ztrát – představuje finanční výkonnost podniku, zachycuje hodnoty a vztahy mezi výnosy podniku a náklady potřebných na jejich vytvoření
- rozvaha – představuje finanční pozici podniku, soupis majetku podniku v peněžním vyjádření a zdrojů jeho financování
- přehled o peněžních tocích – zachycuje schopnosti a způsoby vytváření peněžních prostředků a jejich využití

Níže se zle dočíst, že součástí strukturovaných, každoročně odevzdávaných souborů jsou pouze výkaz zisku a ztrát a rozvaha, přehled o peněžních tocích (cashflow) nikoliv. Lze jej však z uvedených výkazů spočítat pomocí tzv. cashflow analýzy. Dále bývá často do závěrek zahrnut také přehled o změnách vlastního kapitálu.



Obrázek 2.2: Výkazy účetní závěrky a jejich obsah [2]

## Výkaz zisku a ztráty

Výkaz zisku a ztráty, také výsledovka, ukazuje, jakého hospodářského výsledku bylo dosaženo v daném účetním období. Data výsledovky jsou počítány z dat výsledkovkových účtů. Jednotlivé účty jsou kategorizovány podle jejich účelu a položky výsledovky jsou sumy účtů odpovídající kategorie. Je to jakási agregace záznamů o peněžních tocích za celé účetní období, z nichž každý je buď výnosem nebo nákladem (pozn. není to to samé jako příjmy a výdaje). Výnosy i náklady jsou dále rozděleny na provozní a finanční činnost. Po sečtení výsledků za obě činnosti získáme pro mnohé nejdůležitější informaci o hospodaření podniku – výsledek hospodaření. Tím se rozumí buďto zisk (výnosy převyšují náklady) nebo ztráta (náklady převyšují výnosy). V případě zisku je navíc subjekt ze zákona povinen odvést daň ze zisku, čímž teprve získáme hodnotu čistého zisku.

## Rozvaha

Také označována termínem balance. Na rozdíl od výkazu zisku a ztráty jsou při sestavování rozvahy data vázána k jednomu určitému okamžiku, takzvanému rozvahovému dni, který je posledním dnem účetního období, zpravidla posledním dnem kalendářního roku. Data mají tedy statickou povahu.

Hlavními složkami rozvahy jsou aktiva a pasiva. Zatímco aktiva představují majetek společnosti (budovy, stroje, materiál, vozidla, . . . , ale i software nebo know-how), pasiva jsou zdrojem krytí těchto aktiv (kapitál). V průběhu roku se hodnota aktiv a pasiv liší právě o hodnotu, o kterou se liší náklady a výnosy. Při sestavování závěrky se rozdíl nákladů a výnosů po zdanění přenesne na rozvahový účet (představující pasivum) s názvem hospodářský výsledek ve schvalovacím řízení, čímž se aktiva a pasiva vyrovnají. Část zisku musí být investována do fondů (ze zákona, případně stanov společnosti), valná hromada poté rozhodne, co se zbytkem. Může navýšit vlastní jmění podniku nebo zisk rozdělit mezi vlastní investory, což je znázorněno na obrázku 2.1. Hlavním rozdělením aktiv je na krátkodobá a dlouhodobá, hlavním rozdělením pasiv je na vlastní a cizí kapitál.

## 2.3 Absolutní ukazatele

### 2.3.1 Horizontální analýza

Při horizontální analýze je určitá položka (řádek) finančního výkazu porovnávána se stejnou položkou za jiné časové období. Při výpočtu tedy porovnáváme výsledky jedné položky napříč různými obdobími - proto horizontální analýza. U jednotlivých položek zvlášť je vypočtena absolutní hodnota meziroční změny a její procentuální vyjádření. Tento postup je proti ostatním velmi ilustrativní, působivý a přímočarý. Této analýze můžeme podrobit výkaz zisku a ztrát, rozvahu i přehled peněžních toků.

Při posuzování výsledků by měl analytik zohlednit významné meziroční změny, které mají vliv na výsledek – změny v daňové soustavě, podmínek na trhu, změna politické

situace, míra inflace a podobně. Výsledek je běžně interpretován pomocí sloupcového nebo spojnicového grafu.

### 2.3.2 Vertikální analýza

Prívlastek vertikální vychází tak jako u horizontální analýzy ze způsobu zpracování finančních výkazů (opět výkazu zisku a ztrát, rozvahy i přehledu finančních toků). Dáváme tak do poměru jednotlivé položky výkazů. Na rozdíl od poměrových ukazatelů zmíněných dále se však při vertikální analýze dává do poměru část celku k celku samotnému. Dalo by se tedy říct, že vyčíslováním podílu vybrané části na celku vertikální analýza popisuje strukturální rozložení sledované entity (například aktiv nebo pasiv či jejich složek).

Výhodou vertikální analýzy je, že není ovlivněna meziroční inflací a výsledky se tak dají porovnávat s výsledky za jiné časové období, mimo to i s výsledky jiných podniků.[8]

## 2.4 Poměrové ukazatele

Výpočet je opět založen na datech z rozvahy a výsledovky, existují ale i poměrové ukazatele založené na cashflow. Na rozdíl od horizontální a vertikální analýzy, při kterých je sledován vývoj jedné veličiny v čase případně vývoj ve vztahu k určité vztažné veličině (nadřazenému celku), při výpočtu poměrových ukazatelů zkoumáme vztah mezi jednotlivými veličinami navzájem, čímž získáváme zase jinou vypovídací hodnotu.

Existuje několik různých ukazatelů. Různé ukazatele slouží různým účelům, různé společnosti se řídí různými ukazateli. Podle účelu a povahy ukazatele ho lze zařadit do některé ze skupin soustavy ukazatelů. V této práci budou zmíněny základní skupiny poměrových ukazatelů – rentability, likvidity, aktivity a zadluženosti, v rámci nichž budou popsány i jednotlivé ukazatele.

Jak plyne z názvu, při výpočtu hodnoty ukazatele se obecně používá nějakého poměru, tedy zlomku. Možností, které hodnoty dát do poměru, je takřka neomezeně. Nicméně při konstrukci jednotlivých ukazatelů bychom se měli zamyslet nad tím, co při vybraných entitách vlastně vztah vyčísľuje – zda má konkrétní volba význam pro účel analýzy.[1]

### 2.4.1 Ukazatele rentability

#### *profitability ratios*

Ukazatele rentability (výnosnosti, ziskovosti, návratnosti) patří v praxi mezi nejsledovanější ukazatele. Vyjadřují míru efektivity hospodaření podniku poměřením výsledku hospodaření se zdroji, které byly na hospodaření vynaloženy.

$$\frac{\text{výsledek hospodaření}}{\text{zdroje}}$$

Ukazatele rentability jsou mezivýkazovými ukazateli, což znamená, že jsou při jejich výpočtu použity hodnoty z výsledovky i rozvahy. Jednotlivé ukazatele rentability se liší především tím, co je chápáno pod pojmem zdroje ve jmenovateli.[13] Podle volby jmenovatele je vytvořeno také následující rozdělení ukazatelů rentability.

V rámci jednoho ukazatele však existuje několik dalších variant výpočtu. Ty se vzájemně liší složkou v čitateli, která zachycuje výsledek hospodaření (zisk) v rozdílných fázích – před nebo po zdanění, odečtení úroků, provedení odpisů a amortizace. Do čitatele může být dosazen i jiný ukazatel firemní výkonosti, jako například provozní cashflow. Volíme takové hodnoty, které mají pro danou situaci největší vypovídací hodnotu.

<b>EBITDA</b>			
<b>EBIT</b>			odpisy
<b>EBT</b>		úroky	
<b>EAT</b>	daně		
čistý zisk			
zisk před zdaněním, hosp. výsledek			
zisk před zdaněním a úroky, provozní výsledek			
zisk před zdaněním, úroky a odpisy			

Obrázek 2.3: Grafická interpretace jednotlivých druhů zisku

- EAT – earnings after taxes, čistý zisk – zisk po zdanění, zúročení, odpisech a amortizaci
- EBT – earnings before taxes, hrubý zisk – zisk před zdaněním, t.j. po odečtení úroků, odpisech a amortizaci
- EBIT – earnings before interest and taxes, provozní zisk – zisk před zdaněním a zúročením. Užití této hodnoty je vhodné například při srovnávání dvou podniků s různou výší daně z příjmu a různou kapitálovou strukturou
- EBITDA – earnings before interest, taxes, depreciation and amortization – zisk před zdaněním, úroky, odpisy a amortizací
- NOPAT =  $EBIT \cdot (1 - t)$ , kde  $t$  je sazba daně z příjmů – zdaněný (čistý) zisk bez odečtení úroků, t.j. kolik by dělal čistý zisk, kdyby podnik nemusel platit úroky (kdyby nečerpal žádný úvěr)

## Rentabilita aktiv

*return on assets, ROA*

ROA je ukazatel, který do jmenovatele dosazuje celkový objem aktiv (potažmo pasiv, jelikož jsou jimi kryta). Při výpočtu se tedy nebere ohled na to, zda jsou aktiva vlastní nebo cizí, dlouhodobá nebo krátkodobá. Tím vyjadřuje, jak efektivně dokáže podnik naložit s aktivy, které se do něj vloží (odtud zřejmě plyne označení produkční síla nebo výkonnost podniku).

Nejčastějšími variantami uváděnými v literatuře jsou ty, které mají v čitateli hodnotu EBIT, tedy zisk před odečtením nákladových úroků (rentabilita celkových aktiv by neměla brát v potaz původ zdrojů) a zdaněním. Tato volba nám zároveň umožňuje porovnat podniky s rozdílnými zdroji financování a různým daňovým zatížením.

$$ROA = \frac{EBIT}{\text{celková aktiva}} \quad (\text{zdroj: [8]})$$

Další používanou variantou je s hodnotou NOPAT v čitateli. Výsledkem je rentabilita, která by plynula z čistého zisku, pokud by byly veškeré zdroje financování vlastní. Dochází zde k fiktivnímu zdanění úroků, jelikož  $EBIT \cdot (1 - t)$  se dá zapsat také jako  $EAT + \dot{U} \cdot (1 - t)$ .

$$\text{rentabilita aktiv} = \frac{EBIT \cdot (1 - t)}{\text{celková aktiva}} \quad (\text{zdroj: [8]})$$

## Rentabilita vlastního kapitálu

*return on equity, ROE*

ROE do jmenovatele dosazuje hodnotu vlastního kapitálu, čímž vyjadřuje míru jeho zhodnocení. Jinými slovy udává, kolik korun zisku připadá na 1 korunu vlastního kapitálu. Pro vlastníky (akcionáře, společníky a další investory) podniku je jedním z klíčových ukazatelů, neboť pomocí něj mohou zjistit, zda jimi vložený kapitál přináší výnos odpovídající velikosti jejich investičního rizika.[8]

Investor by se měl také zajímat, zda je procentuální hodnota ROE vyšší než bezriziková úroková míra (úroková sazba státních dluhopisů), v praxi spíše vyšší než nejvyšší úroková sazba nabízená bankami. Pokud by byla nižší, obnos investovaný do podniku by mohl raději vložit do banky a téměř bez námahy a bez rizika dosáhnout většího výnosu.

Výpočet má pochopitelně smysl zejména s čistým ziskem (EAT) v čitateli, avšak ve jmenovateli můžeme uvažovat nad odečtením fondů, ze kterých investor ve výsledku těžit nebude.

$$\text{rentabilita vl. kapitálu} = \frac{EAT}{\text{vlastní kapitál}} \quad (\text{zdroj: [8]})$$

## Rentabilita tržeb

*return on sales, ROS*

V praxi velmi často používaný ukazatel. Poměřuje zisk podniku a tržby (tržby za prodej zboží, výrobků a provedení výkonů), tedy kolik procent tržeb tvoří zisk.

Pokud by byla výsledkem relativně malá, dlouhodobě klesající hodnota (např. 2%), je vhodné upozornit. Pokud by došlo k náhlému poklesu tržeb při zachování nákladů (což je běžné), je zřejmé, že se podnik dostane do finančních potíží. Stejnětak se může dostat do potíží při zvýšení nákladů (zdražení materiálu, energií, pracovní síly) se současným zachováním objemu tržeb (zvýšení nákladů se však na tržby zpravidla promítne). V čitateli se nejčastěji používá hodnota EBIT nebo EAT

$$\text{rentabilita tržeb} = \frac{EBIT \text{ (nebo EAT)}}{\text{tržby}} \quad (\text{zdroj: [8]})$$

## Rentabilita investice

*return on investment, ROI*

Další z často používaných ukazatelů. Jak napovídá název, ukazatel vyjadřuje rentabilitu spíše z pohledu konkrétního investičního projektu, z výše sumy investic provedených během vymezeného období, nežli z celopodnikového pohledu. Oba přístupy by pojednávaly o tom samém v případě, že by se jednalo o investici do koupi podniku. Rentabilita této investice by pak byla stejná jako rentabilita vlastního kapitálu, v případě nezadluženého podniku i stejná jako rentabilita celého podniku.

$$\text{rentabilita investice} = \frac{\text{výnos investice}}{\text{náklady na investici}} \quad (\text{zdroj: [8]})$$

### Některé další ukazatele rentability:

- ROC – rentabilita nákladů – return on costs
- ROCE – rentabilita dlouhodobě investovaného kapitálu – return on capital employed

### 2.4.2 Ukazatele likvidity

*liquidity ratios* Pojem likvidita bývá nesprávně zaměňován s úzce souvisejícími pojmy likvidnost a solventnost. Likvidnost je vlastnost jednotlivých složek aktiv podniku, vyjadřující schopnost přeměny těchto složek v peněžní prostředky v co nejkratším čase a s minimálními ztrátami.[4] Přeměnou můžeme rozumět například prodej zásob nebo inkasování pohledávek.

Solventnost, neboli platební schopnost, je definována jako schopnost subjektu (v našem případě podniku) včas splácet své finanční závazky. Má-li být podnik solventní, musí kromě stálého generování zisku také zajistit, aby byla část aktiv vázána ve formě, kterou může uhradit své závazky.

Likvidita podniku dává tyto pojmy do souvislosti – ukazatele likvidity vyjadřují schopnost podniku přeměnit vybraná aktiva na peněžní prostředky (využít jejich likvidnosti) za účelem včasného uhrazení všech splatných závazků (a tím pádem být dočasně solventní).[7] Proto bývají také označovány jako ukazatele platební schopnosti.

Všechny ukazatele likvidity se počítají jako poměr toho, čím je možné platit (disponibilní prostředky), k tomu, co se musí zaplatit. Disponibilními prostředky můžeme přitom chápat různou množinu aktiv. Je samozřejmě nesmysl, aby byla zahrnuta ta aktiva, jejichž ztráta by ohrozila chod podniku.

### Běžná likvidita (likvidita III. stupně)

*current ratio, CR*

Poměr ukazuje, kolikrát pokrývají oběžná aktiva krátkodobé závazky, jinými slovy, kolikrát je podnik schopen uhradit své krátkodobé závazky z peněžních prostředků, které by získal přeměnou z oběžných aktiv.

Uváděná optimální hodnota se v literaturách různí, zřejmě se bude lišit i v závislosti na typu podniku. Hodnota ukazatele by určitě neměla být menší než 1, což by znamenalo nutnost financovat závazky z dlouhodobých zdrojů financování a jinými nevhodnými způsoby. Také berme v potaz, že po pokrytí závazků by měly zbýt prostředky pro další činnost podniku. Optimální hodnota uvedená v preferovaném zdroji je  $CR \geq 1.5$ , ale obecně lze

tvrdit, že čím stabilnější jsou příjmy a čím jistější jsou zdroje těchto příjmů, tím více se může hodnota běžné likvidity blížit jedné.[12]

$$\text{běžná likvidita} = \frac{\text{oběžná aktiva}}{\text{krátkodobé závazky}} \quad (\text{zdroj: [8]})$$

### Pohotová likvidita (likvidita II. stupně)

*quick asset ratio, QAR*

Pokud bychom zkoumali strukturu oběžných aktiv, zjistili bychom, že se likvidnost jednotlivých složek liší. Výpočet běžné likvidity zahrnuje i zásoby, nicméně ty při výpočtu pohotové likvidity nejsou zahrnuty, jelikož většinou bývají nezbytné pro zachování chodu podniku a navíc jejich likvidnost nemusí být dostačující. V čitateli tak zbyde součet objemu peněžních prostředků, ekvivalentů a pohledávek. Pro zpřesnění výsledku je vhodné odečíst nedobytné pohledávky. Dá se předpokládat, že podniky z oblasti služeb nebudou mít v zásobách mnoho aktiv a proto se bude hodnota ukazatele blížit hodnotě běžné likvidity. U výrobních podniků se naopak tyto dvě hodnoty budou spíše lišit.

$$\text{pohotová likvidita} = \frac{\text{oběžná aktiva} - \text{zásoby}}{\text{krátkodobé závazky}} \quad (\text{zdroj: [8]})$$

### Hotovostní likvidita (likvidita I. stupně)

*cash position ratio, CPR*

Při výpočtu hotovostní likvidity dosazujeme do čitatele součet jen těch nejlikvidnějších složek oběžných aktiv, kterými jsou peněžní prostředky (peníze v hotovosti a na běžných účtech) a jejich ekvivalenty (například šeky nebo obchodovatelné cenné papíry). Přestože tento ukazatel nejlépe vypovídá o platební schopnosti podniku k určitému datu, nebere v potaz strukturu krátkodobých závazků, především z pohledu jejich skutečné splatnosti.[13] Pokud bychom chtěli tuto skutečnost zohlednit, měli bychom do jmenovatele zahrnout pouze okamžitě splatné závazky. Taková varianta bývá označována jako okamžitá likvidita.

$$\text{hotovostní likvidita} = \frac{\text{peněžní prostředky} + \text{ekvivalenty}}{\text{krátkodobé závazky}} \quad (\text{zdroj: [8]})$$

### 2.4.3 Ukazatele aktivity

*activity ratios*

Další z mezivýkazových ukazatelů – vstupními daty jsou položky jak z výkazu zisku a ztrát tak z rozvahy. Ukazatele aktivity vyjadřují, jak efektivně podnik nakládá se svými aktivy. Pokud jich vlastní více, než je nutné, vznikají zbytečné náklady a dochází ke snížení zisku. Jestliže jich má nedostatek, přichází o výnosy z potenciálních zakázek, o které se kvůli nedostatku aktiv nemůže ucházet.

Výsledkem je hodnota udávající rychlost nebo dobu obratu. Rychlost obratu vyjadřuje, kolikrát se určitá složka aktiv přemění za sledované období na peněžní prostředky. Doba obratu nám říká, jak dlouho trvá přeměna.

Pokud známe hodnotu rychlosti obratu sledovaných aktiv X, dobu obratu X (výsledkem budou dny) vypočítáme jednoduše jako

$$\text{doba obratu } X = \frac{365}{\text{rychlost obratu } X}$$

## Obrat celkových aktiv

*total assets turnover*

Udává, kolikrát se obrátí celková aktiva za sledované časové období. Nevýhodou tohoto a některých dalších ukazatelů je povaha vstupních veličin: tržby jsou zachyceny jejich sumou za celé období, kdežto hodnota aktiv se vztahuje ke konkrétnímu časovému okamžiku, kdy byla rozvaha vytvořena. Přesnějšího výsledku by se dalo dosáhnout použitím průměrné hodnoty sledovaných aktiv.

$$\text{obrat celkových aktiv} = \frac{\text{tržby}}{\text{celková aktiva}} \quad (\text{zdroj: [13]})$$

## Obrat stálých aktiv

*fixed assets turnover*

Stálá aktiva jsou taková, která slouží podniku déle než jeden rok a jsou spotřebovávána postupně, nikoliv jednorázově. Konkrétně je to součet dlouhodobého hmotného, nehmotného a finančního majetku, t.j. například budovy, pozemky, vozidla, stroje, software, know-how a mnoho dalších.

Ukazatel potom vyjadřuje míru využití těchto aktiv. Pokud při analýze vyjde hodnota nižší, než je průměr v odvětví, může to znamenat nedostatečné využití stávajících fixních (dlouhodobých) aktiv a manažeři by tak měli zvážit, jak jejich využití zefektivnit namísto investování do aktiv nových.

$$\text{obrat stálých aktiv} = \frac{\text{tržby}}{\text{dlouhodobý majetek}} \quad (\text{zdroj: [13]})$$

## Obrat zásob

*inventory turnover*

Říká, kolikrát je za sledované období položka zásob prodána a opět naskladněna.

Při použití tohoto ukazatele je nutné si uvědomit, že tržby vyjadřují tržní hodnotu (za kolik se prodalo), kdežto hodnota zásob bývá vyjádřena jejich pořizovací cenou (za kolik se nakoupilo). Přesnějšího výsledku obratovosti může být dosaženo umístěním nákladů na prodané zboží do čitatele namísto tržeb.

Ukazatel pro zásoby výrobků nebo zboží je rovněž ukazatelem likvidnosti těchto zásob (udává, jak dlouho trvá přeměna zásob na finanční prostředky).

$$\text{obrat zásob} = \frac{\text{tržby}}{\text{zásoby}} \quad (\text{zdroj: [13]})$$

## Doba obratu pohledávek

V tomto případě je směřodratnější spíše doba trvání pohledávky než počet obrátů pohledávek. Doba se vypočítá jako poměr průměrného stavu pohledávek k průměrné denní tržbě na obchodní úvěr (fakturu, „sekeru“).

Hodnota ukazatele dlouhodobě vyšší než hodnota stanovená platebními podmínkami pro klienta může znamenat problémové portfolio odběratelů, kteří nejsou schopni plnit své závazky včas.

$$\text{doba obratu pohledávek} = \frac{\text{prům. stav pohledávek}}{\text{prům. denní tržby}} \quad (\text{zdroj: [13]})$$



Firma	Vlastní kapitál	Cizí kapitál	Výnosnost majetku (20%)	Úroky (8%)	Zisk nezdaněný	Daň (40%)	Zisk zdaněný	Výnosnost vlastního kapitálu
A	2000	–	400	0	400	160	240	12%
B	1000	1000	400	80	320	128	192	19,20%

Obrázek 2.4: Srovnání výnosnosti při různém poměru vlastního a cizího kapitálu

## Doba obratu závazků

Průměrná doba odkladu plnění závazků. Nabízí se porovnání s dobou obratu pohledávky.

### 2.4.4 Ukazatele zadluženosti

#### *debt ratios*

Zadluženost podniku souvisí s velikostí podílu cizích zdrojů na všech zdrojích financování podniku, jelikož cizí zdroje pro podnik zpravidla znamenají dluh, závazek. Měří tedy dopady financování cizími zdroji. Využití cizích zdrojů se vyplácí, dokud zajistí výnos vyšší, než jsou náklady na tyto zdroje – úroky.

V praxi se běžně nesetkáme s většími podniky, které by byly financovány výhradně z vlastních nebo výhradně z cizích zdrojů, jsou tedy do jisté míry zadluženy. Zadluženost obecně není negativní charakteristikou – využití cizích zdrojů vede k zesílení tzv. finančního pákového efektu, který pozitivně přispívá k rentabilitě vlastního kapitálu.[1] S rostoucím procentem vyjadřujícím zadluženost však roste i riziko věřitelů. Čím vyšší bude riziko věřitelů, tím obtížněji bude podnik další cizí zdroje získávat a tím nevýhodnější budou podmínky jeho poskytnutí.[13] Jedním z důležitých cílů finanční analýzy je nalezení ideální míry zadluženosti, respektive ideálního rozložení finanční struktury. Vzhledem k poměrně rozšířenému leasingovému financování zmíním, že by bylo vhodné takto pořízený majetek zohlednit a přičíst k cizímu kapitálu, má-li významnější objem.

## Celková zadluženost

#### *debt ratio, DR*

Základním ukazatelem zadluženosti je celková zadluženost, jako podíl cizího kapitálu na celkových aktivech. S rostoucí hodnotou tohoto ukazatele roste i riziko věřitelů a cena za cizí kapitál (výše úroků). Věřitelé preferují nižší hodnoty – v případě likvidace podniku mohou být pohledávky (nejen) věřitelů uhrazeny z vlastního kapitálu podniku (takzvaný finanční polštář). Z hlediska vlastníků akcií jsou vysoké hodnoty ukazatele příznivé, dokud podnik dosahuje vyššího procenta rentability než je procento úroku na cizí kapitál, což znamená, že výsledné zhodnocení kapitálu je vyšší, než jeho pořizovací cena.[9]

$$\text{celková zadluženost} = \frac{\text{cizí kapitál}}{\text{celková aktiva}} \quad (\text{zdroj: [8]})$$

Doplňkovým ukazatelem k celkové zadluženosti je ukazatel poměru vlastního kapitálu (*equity ratio, ER*). Jejich součet by měl být  $DR+ER=1$

$$\text{poměr vlastního kapitálu} = \frac{\text{vlastní kapitál}}{\text{celková aktiva}} \quad (\text{zdroj: [8]})$$

### Koeficient (míra) zadluženosti

*debt to equity ratio, D/E*

Dá se spočítat jako poměr hodnoty cizího kapitálu k hodnotě vlastního. Stejný výsledek bychom získali poměrem předchozích dvou ukazatelů. V případě, že podnik využívá ve větším objemu leasingového financování, je vhodné tuto skutečnost zohlednit přičtením objemu k cizímu kapitálu.

$$\text{koeficient zadluženosti} = \frac{\text{cizí kapitál}}{\text{vlastní kapitál}} \quad (\text{zdroj: [8]})$$

Ve finanční analýze se používá i převrácená hodnota ukazatele, která by se dala nazvat jako míra finanční samostatnosti podniku.

### Úrokové krytí

*interest coverage, IC*

Ukazatel vyjadřuje, kolikrát podnik pokryje úroky plynoucí ze závazků věřitelům poté, co uhradí všechny náklady na provoz (tzn. z provozního zisku). Dá se tedy chápat jako dopad zadlužení na zisk. Obecně platí, že čím vyšší je jeho hodnota, tím lépe pro podnik. Pokud by byla hodnota  $IC < 1$ , znamená to, že podnik hospodaří se ztrátou (není schopen ani pokrýt úroky). Pokud by vyšla hodnota ukazatele  $IC = 1$ , znamená to, že se veškerý zisk použije k uhrazení úroků za vypůjčený kapitál a k vyplacení vlastních investorů nebo k reinvestici do kapitálu nezbudou žádné prostředky.

$$\text{úrokové krytí} = \frac{\text{EBIT}}{\text{nákladové úroky}} \quad (\text{zdroj: [8]})$$

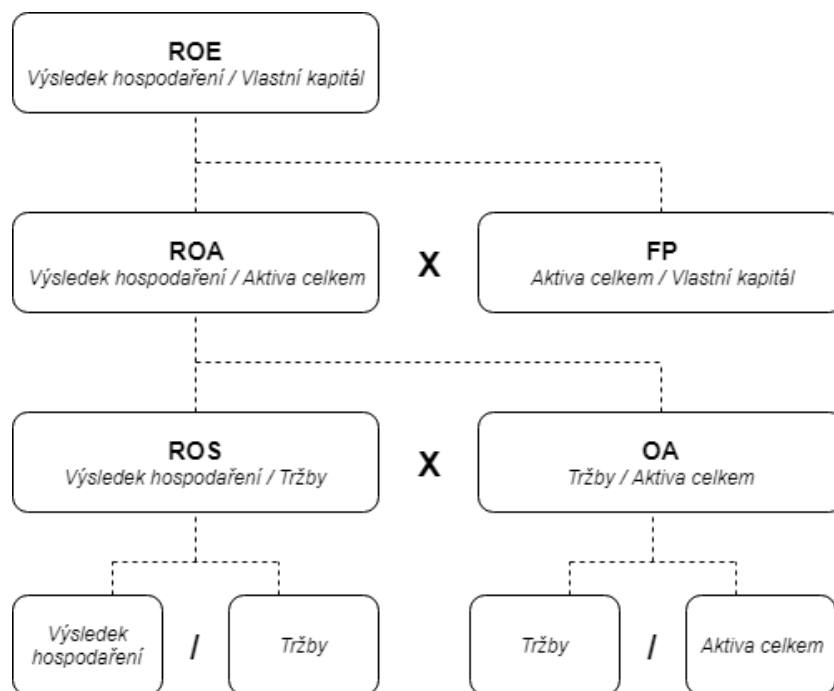
## 2.5 Ostatní

### 2.5.1 Analýza peněžních toků

Rozvaha zachycuje stav majetku a jeho krytí (kapitálu) ve chvíli jejího vytváření. Výkaz zisku a ztráty zase zaznamenává různé kategorie výnosů a nákladů a z nich plynoucího zisku či ztráty za určité období, nehledě na to, zda se jedná o reálné peněžní příjmy či výdaje. Proto se neshodují výnosy s příjmy, náklady s výdaji a zisk či ztráta se změnou stavu peněžních prostředků. Pro tento účel slouží analýza peněžních toků, jejíž podstatou je sledování změn stavu peněžních prostředků.[2] Z toho lze vyvodit, že úzce souvisí s řízením likvidity podniku.

Přehled peněžních toků jsme schopni vytvořit z výkazu zisku a ztrát a rozvahy. Peněžní toky lze rozdělit do celkem tří kategorií podle druhu činnosti podniku, pod které spadají. Jsou to:

- provozní činnost – provozní činnost podniku bez vlivu nepeněžních a mimořádných operací, obecně peněžní operace, které nelze zařadit do investiční činnosti či financování podniku



Obrázek 2.5: Rozklad Du Pont [6] (FP – finanční páka, OA – obrat aktiv)

- investiční činnost – poskytuje přehled o pohybu peněžních prostředků v souvislosti s dlouhodobými aktivy a také dalšími skupinami, které hrají významnou roli při vytváření zisků v budoucnu
- financování podniku – pohyby peněžních prostředků ovlivňující strukturu a velikost vlastního či cizího jmění podniku

### 2.5.2 Du Pont rozklad

Tato analýza se řadí mezi pyramidové soustavy ukazatelů. Zatímco výsledky všech předchozích analýz byla většinou konkrétní čísla, pyramidové soustavy ukazatelů nám dávají tato čísla do souvislostí odpovídajících realitě. Namísto čísla zachycujícího jediný rys složitého procesu zachycuje tato analýza všechny závislosti a vztahy uvnitř procesu. Jedná se o rekurzivní rozklad výpočtu finančního ukazatele na podsložky, což nám umožňuje sledovat faktory ovlivňující výsledek na všech úrovních, nebo naopak zjistit, na které podsložky se zaměřit při snaze o dosažení změny na vyšší úrovni. Zápis rozkladu může být samozřejmě vyjádřen matematicky, velmi vhodná je i jeho grafická reprezentace.

V případě analýzy Du Pont se provádí rozklad ukazatele rentability vlastního kapitálu (ROE). Schéma rozkladu je vidět na obrázku 2.5. Na nejnižší úrovni se na levé straně objevuje rentabilita tržeb, označována také jako zisková marže, která udává podíl čistého zisku na tržbách (zohlednění nákladů). Pokud tržby rostou a zisková marže klesá, je vhodné provést analýzu nákladů. Složky na pravé straně poté tuto marži dávají do souvislosti s majetkem firmy. Vynásobením obratovostí aktiv získáme jejich rentabilitu, následným zohledněním struktury kapitálu získáme rentabilitu vlastního kapitálu.

### 2.5.3 Vnitřní výnosové procento

internal rate of return

Vnitřní výnosové procento vyjadřuje míru výnosnosti investice, tedy ROI, se současným zohledněním proměnné hodnoty peněz v čase (diskontace). Ve světě patří spolu s NPV (net present value, čistá současná hodnota) mezi nejpoužívanější ukazatele pro zhodnocení a výběr investice. Podle průzkumu VŠE je v ČR preferován spíše ukazatel doby návratnosti investice.[11]

NPV slouží k přiblížení, kolik peněz nám projekt, jehož investování zvažujeme, za sledované období přinese nebo sebere. Nebere přitom nijak v úvahu výnosy a náklady nebo hodnotu společnosti, proto je nevhodný pro strategické projekty. Nevýhodou NPV a tím pádem i IRR je nutnost odhadu nejen budoucích finančních toků, které sledovaný projekt přinese, ale také výše diskontu. Odhad hodnoty lze zpřesnit různými technikami (viz doporučená literatura). Důležitá je také odhadovaná doba životnosti projektu, přidáním nebo odebráním jednoho období se může změnit ztráta na zisk i opačně.

Nejprve si uvedeme vzorec pro výpočet čisté současné hodnoty:

( $r$  = diskont,  $t$  = pořadí období,  $CF_t$  = suma toků v období,  $IN$  = vstupní investice)

$$NPV = \sum_{t=0}^n \frac{CF_t}{(1+r)^t} = \sum_{t=1}^n \frac{CF_t}{(1+r)^t} - IN \quad (\text{zdroj: [11]})$$

Matematicky lze IRR vyjádřit hodnotou diskontní sazby, při které je čistá současná hodnota (NPV) rovna nule. Dostáváme tedy následující vzorec:

$$\sum_{t=1}^n \frac{CF_t}{(1+IRR)^t} - IN = 0 \quad (\text{zdroj: [11]})$$

Z uvedeného vzorce lze usoudit, že nalezení hodnoty analytickým řešením není triviální. Rovnice se řeší numericky, iterativními výpočty.

## Kapitola 3

# Návrh řešení

V této kapitole bude popsán návrh vlastního řešení. Nejprve budou zmíněny a otestovány podobné, již existující nástroje. Poté bude vypracován návrh – vymezíme požadovanou funkcionalitu nástroje a nastíníme možné technické řešení. Budou zde popsány i technologie zahrnuté do návrhu technického řešení.

### 3.1 Existující software

Před návrhem řešení bylo vhodné prozkoumat existující řešení. Plné verze nástrojů pro tyto účely bývají zpoplatněny, závěry jsou tedy vyvozeny ze zkušebních verzí, pochopitelně s omezenou funkcionalitou, případně z náhledů a uživatelských dokumentací software.

Všechny testované nástroje jsou ve formě desktopových aplikací určených pouze pro platformu Windows. Po rozpravě s osobou z praxe bylo zjištěno, že je velmi často k účelům analýzy používán program Excel. Příčinou je nespíš široká základna uživatelů, kteří v jeho prostředí zvládají alespoň základní úkony s daty a také široké spektrum funkcionality, které zahrnuje i výpočty z oblasti finanční analýzy. Velmi často jsou v souborech určených pro tento program také uchovávána vstupní data analýzy.

#### **FAF – Finanční analýza firmy**

Software nelze bez zakoupení jakkoliv používat, nicméně na webových stránkách je dostupné instruktážní video a uživatelská příručka popisující jak nástroj vypadá, jak se s ním pracuje a co všechno dokáže. Uživatelské rozhraní vypadá na první pohled zastarale, to ale nijak neubírá na přehlednosti.

Uživatelské rozhraní kromě grafů využívá také stupnic, na které umísťuje ukazatele. Je zde obrazovka „rating“, ve které je hodnota každého z ukazatelů zobrazena za různá časová období a v konkrétním časovém období na stupnici od 1 do 10. Všechny ukazatele pohromadě lze zobrazit v tabulce pro jedno časové období.

Ukazatele jsou i na obrazovce s trendy, ta mimo ukazatele obsahuje i trendy položek výkazu zisku a ztráty a rozvahy. Je zde také zobrazen seznam rizikových faktorů. Ty se od ukazatelů liší tím, že jsou buď v pořádku nebo nikoliv, nic mezi tím (např. skutečnost, že podnik hospodaří se ztrátou).

Nástroj dále umožňuje vertikální analýzu, horizontální analýzu, analýzu cash flow a výpočet hodnoty firmy.

Funkcionalita, která mě nejvíce zaujala, je nazvaná „co když“. Spočívá v umožnění uživateli jednoduše manipulovat se vstupními daty analýzy a zároveň sledovat, jaký dopad by měla tato změna na hodnotu ukazatelů.

Hodnoty z výkazu zisku a ztráty a rozvahy jsou zadávány do tabulky. Instruktažní video ani uživatelská příručka nehovoří o tom, zda nástroj podporuje automatizované načítání výstupních dat z některých jiných nástrojů z oblasti financí (např. účetních).

Dalším nedostatkem byla sekce s vysvětlením pojmů. Všechny pojmy jsou na jedné hromadě, na konkrétní pojem se nelze prokliknout.

## **Equanta**

Zatímco FAF je pravděpodobně výtvozem jednoho nebo malé skupiny vývojářů a jsou v něm obsaženy pouze klíčové metody a ukazatele finanční analýzy (které ale ve většině případů stačí), Equanta je mnohem sofistikovanějším nástrojem, za jehož vývojem nejspíš stojí větší počet vývojářů – společnost ATLAS consulting s.r.o. . Ta má ve svém portfoliu i jiné, nejen finančně zaměřené nástroje.

Equanta má o poznání modernější rozhraní, obsahuje i více sekcí, ukazatelů a pohledů na hospodaření podniku. Obsahuje také automatizované načítání vstupu, a to nejen z výkazu zisku a ztráty a rozvahy ve formátu pro finanční úřad, ale i z účetních dat nástrojů jako je Pohoda. V nástroji je možné provádět mezipodnikové srovnání úzávěrek a ukazatelů, které se v praxi používá relativně často.

Nástroj si pamatuje naposledy otevřené dokumenty, tyto dokumenty lze seskupovat pod projekty, jsou zde zpřístupněna různá nastavení a další možnosti personalizace, což implikuje využití persistentních dat. To je proti nástroji FAF další velkou výhodou.

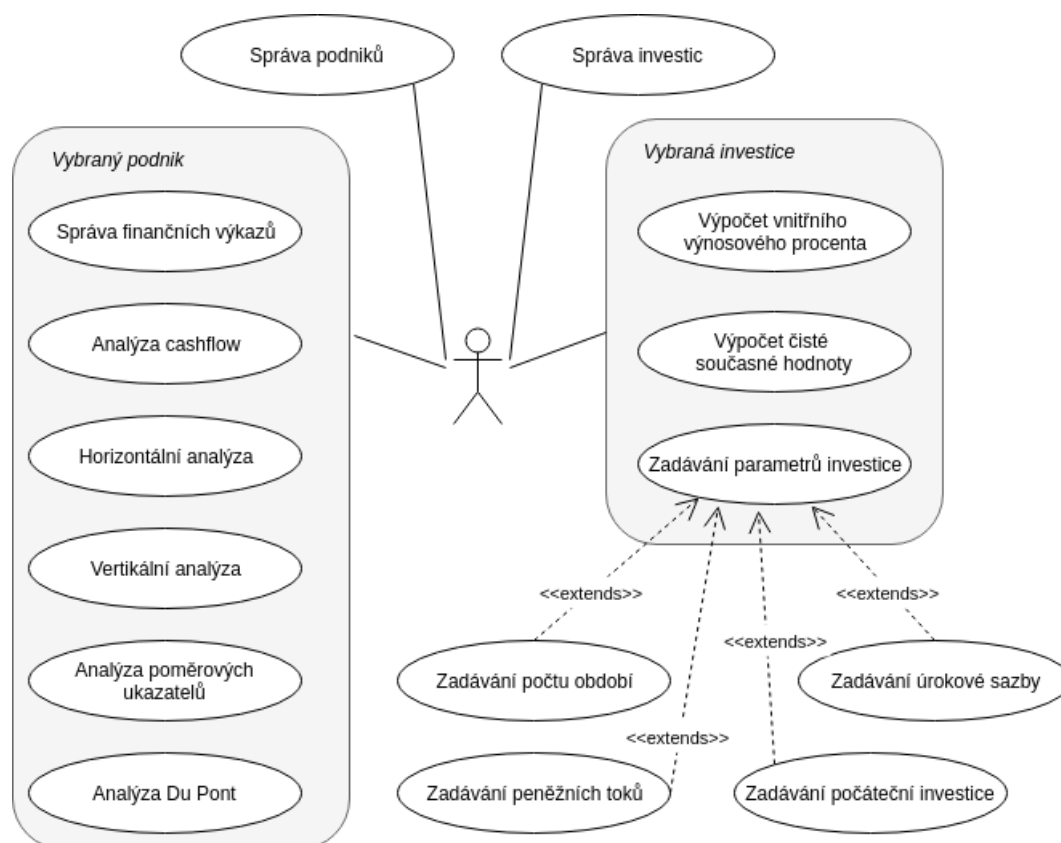
## **FinAnalysis**

Nástroj je realizován jako přednastavená šablona v tabulkovém prostředí MS Excel. Toto řešení může být rychlé a efektivní, nicméně pro naše účely naprosto nevyhovující, jelikož je jeho používání podmíněno používáním MS Excel. Nástroj je prostředím svého běhu omezen, veškerá rozšíření je nutné implementovat funkcemi Excelu.

Přesto nástroj plní svůj účel a lze se v něm inspirovat možnostmi provázení základních analýz, výpočty mnoha ukazatelů a interpretací výsledků. Jednotlivé sekce jsou realizovány jako listy v Excelu. První tři listy (list s výkazem zisku a ztráty, rozvahou a list se základními údaji jako například počet zaměstnanců) slouží k zadávání dat. Zbývající listy obsahují výsledky, jsou pouze pro čtení a odpovídají rozdělení podle jednotlivých typů analýzy. Jak již bylo zmíněno, použití programu Excel je v oblasti financí velmi časté.

## **3.2 Vybrané metody vyhodnocování**

Nástroj by měl být schopen provádět analýzu alespoň v rozsahu teoretické části této práce. Pro hodnocení podniku jsou to horizontální a vertikální analýza, analýza poměrových ukazatelů, cashflow a Du Pont rozklad. Hodnoceny budou také investice, u kterých bude počítáno vnitřní výnosové procento a čistá současná hodnota. Podoba výstupu se mění v závislosti na vybrané analýze nebo ukazateli. Výstupem může být finanční částka, absolutní nebo relativní hodnota, procentuální podíl, tabulka s těmito hodnotami v čase, graf s vývojem hodnot v čase.



Obrázek 3.1: Use case diagram navrhované aplikace

### Vertikální a horizontální analýza

Hned jako první se nabízí porovnání s hodnotami za předchozí období, tedy horizontální analýza. Při horizontální analýze jedné položky výkazu zisku a ztráty nebo rozvahy se nejčastěji používá sloupcového (v případě více sledovaných období spojnicového) grafu. Nástroj bude umožňovat i hromadnou horizontální analýzu – výsledek bude interpretován tabulkou; meziroční změna u jednotlivých položek bude vyjádřena absolutní i procentuální hodnotou.

Podobně i pro vertikální analýzu. Pokud bychom chtěli analyzovat pouze užší část výkazu zisku a ztráty nebo rozvahy, můžeme výsledky interpretovat v grafech. Výsledky hromadné analýzy budou zobrazeny v tabulce s vyjádřením změn absolutní i procentuální hodnotou.

### Poměrové ukazatele

Možnosti zobrazení poměrových ukazatelů budou podobné jako v testovaných nástrojích.

Pro interpretaci hodnot poměrových ukazatelů budou využity:

- stupnice
- tabulka s přehledem hodnot ukazatelů
- tabulky s trendy

- grafy (sloupcové, spojnicové, koláčové a další)

Hodnoty ukazatelů půjde porovnat meziročně i mezipodnikově. Velmi vhodné by bylo také zhodnocení výstupních hodnot. V literatuře lze pro některé ukazatele nalézt referenční hodnoty, pro jiné nikoliv. Hodnoty se však v různých zdrojích liší, z principu je nelze jednoznačně stanovit. Nabízí se zobrazení a porovnání s průměrnými hodnotami v odvětví, které jsou počítány kvartálně MPO<sup>1</sup> v rámci finanční analýzy podnikové sféry<sup>2</sup>. Tato možnost bude při dalším postupu prozkoumána a zvážena.

## Hodnocení investic

Kromě analýzy podniku bude nástroj schopen analyzovat i investice, konkrétně bude možné volit mezi výpočtem vnitřního výnosového procenta investice nebo čisté současné hodnoty při zvolené úrokové sazbě. Vzhledem k rovnicím v sekci 2.5.3 můžeme usoudit, že kromě úrokové sazby bude pro oba výpočty nutné zadávat také délku trvání investice, počáteční finanční výdaj a peněžní toky za jednotlivá období. Výpočet ČSH je přímočarý, pro výpočet VVP bude využita některá z iterativních metod. Jako nejjednodušší a nejintuitivnější metoda se mi jeví metoda půlení intervalů. Pokud bude při implementaci zjištěno, že je tato metoda nevyhovující, můžeme použít jinou, například Newtonovu metodu.

## 3.3 Návrh programového řešení

Nástroj nebude sloužit vyloženě jen pro dávkové zpracování dat. Půjde o systém pro podnikatele, manažery a analytiky, ve kterém bude možné pod určitým uživatelským účtem evidovat informace o hospodaření různých podniků za různá časová období. Nástroj bude schopen načtená data uložit a svázat s daným uživatelským účtem, což umožní nejen jednoduchý přístup k těmto datům a jejich kvantitativní analýze, ale i jednoduchou správu dat pro meziroční a mezipodnikové srovnání. Díky webovému rozhraní, což činí tento nástroj v oboru jedinečným, bude možné přistupovat k systému a využívat jeho služeb kdykoliv a odkudkoliv.

### 3.3.1 Zadávání vstupních dat

Vstupem kvantitativní finanční analýzy jsou data z dokumentů výkaz zisku a ztráty a rozvaha. Při hledání vhodného způsobu automatizace načítání vstupních dat bylo zjištěno, že jsou subjekty zapsané v obchodním rejstříku ze zákona povinny každoročně tyto dva dokumenty odevzdávat pro evidenci státním orgánům. Odevzdávají se ve dvou formách. První z nich je ve formátu PDF na krajský soud, v jehož obchodním rejstříku je subjekt evidován. Formát PDF je určen spíše ke zpracování člověkem, pro strojové čtení dat není vhodný. Naštěstí pro nás jsou subjekty povinny dokumenty odevzdávat také na daňovém portálu, spolu s daňovým přiznáním, ve strukturovaném formátu XML. Přestože se v podnikové sféře používají pro účtování různé nástroje (Pohoda, MRP, ...), dá se očekávat, že většina z nich bude schopna tyto dva dokumenty vytvořit ve shodě s uváděnou specifikací s názvem DPPDP8, případně nižší verze pro starší závěrky.

Vzhledem k této skutečnosti bude v nástroji umožněno automatické načítání těchto souborů implementací odpovídajícího parseru, čímž výrazně usnadníme zadávání. Uživatel

<sup>1</sup>Ministerstvo průmyslu a obchodu České republiky

<sup>2</sup><https://www.mpo.cz/cz/rozcestnik/analyticke-materialy-a-statistiky/analyticke-materialy/>



bude samozřejmě ponechána možnost hodnotu všech položek rozvahy a výsledovky vyplnit ručně. Údaje ostatních společností jsou sice zveřejňovány, nicméně už jen ve formátu PDF, pro možnost mezipodnikového srovnání je tedy ruční zadávání dat závěrky nezbytné.

Pro jednoduchost a demonstraci bude evidence výkazů obsahovat maximálně jednu sadu výkazů na jeden rok, pro meziroční srovnání.

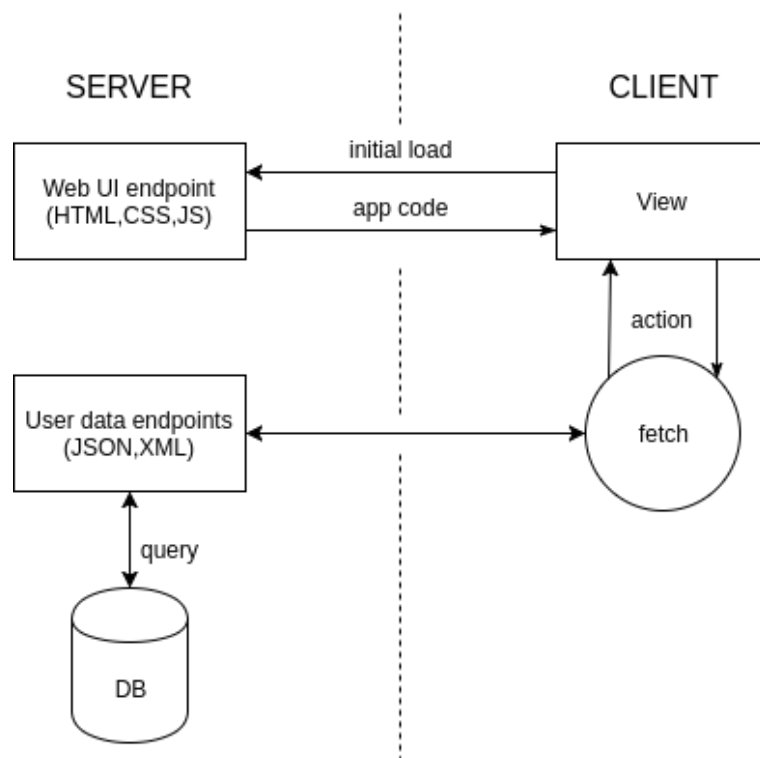
### 3.3.2 Architektura

Byla zvolena architektura klient-server, s webovým uživatelským rozhraním. V dnešní době, kdy je vysokorychlostní internet spíše pravidlem nežli výjimkou, je tato architektura využívána stále více. Motivací je také fakt, že žádný z hledaných analytických nástrojů tuto architekturu nevyužívá, přestože ve spoustě jiných oblastí se jedná o preferované řešení, které uvítají vývojáři a především uživatelé. Existují samozřejmě oblasti, pro které je tato architektura nedostačující. Po prostudování teoretického podkladu této práce si však troufám tvrdit, že v případě finanční analýzy je toto řešení realizovatelné.

Webové rozhraní má samozřejmě své výhody – uživatel může k programu přistupovat odkudkoliv (také z různých operačních systémů), používání programu není podmíněno jeho instalací a z pohledu vývojáře je mnohem jednodušší jeho údržba (nasazování nových verzí). Další výhodou jsou široké možnosti využití jazyka JavaScript – nejen v prohlížeči, ale i na serverech (Node.js) a v desktopových (Electron) i mobilních (React Native) aplikacích. To signalizuje vysokou míru znovupoužitelnosti napsaného kódu.

Nevýhodou je samozřejmě nutnost připojení k internetu, tu však částečně odbourává koncept SPA popsany v dalším odstavci a možnost uložení veškerého kódu spojeného s aplikací do lokálního úložiště, tato možnost ale nebude součástí základního návrhu. Problémem by se mohlo zdát i prostředí běhu, webový prohlížeč. Efektivita interpretace, nebo spíše překladu JavaScript kódu se však neustále zdokonaluje a skutečnost, že aplikace není v nativním kódu, je téměř nepostřehnutelná.

Nástroj bude koncipován jako single page application (SPA). To znamená, že při interakci s uživatelem bude stránka překreslována dynamicky na straně klienta, namísto opakovaného nahrávání programového kódu (webových stránek) ze serveru. Tato technika vyžaduje načtení veškerého kódu aplikace při jediném načtení stránky a dodatečné načítání uživatelských dat ve chvíli, kdy jsou potřeba. Přináší však mnohem plynulejší práci s uživatelským rozhraním, které se blíží desktopovému provedení – odezva aplikace se zdá rychlejší, uživatel nemusí čekat při akci vedoucí ke změně UI, která by normálně vedla k načtení veškerého kódu stránky ze serveru. To je také důvod, proč je SPA v dnešní době široce využíváno při implementaci rozsáhlejších aplikací nebo systémů běžících ve webovém prohlížeči.



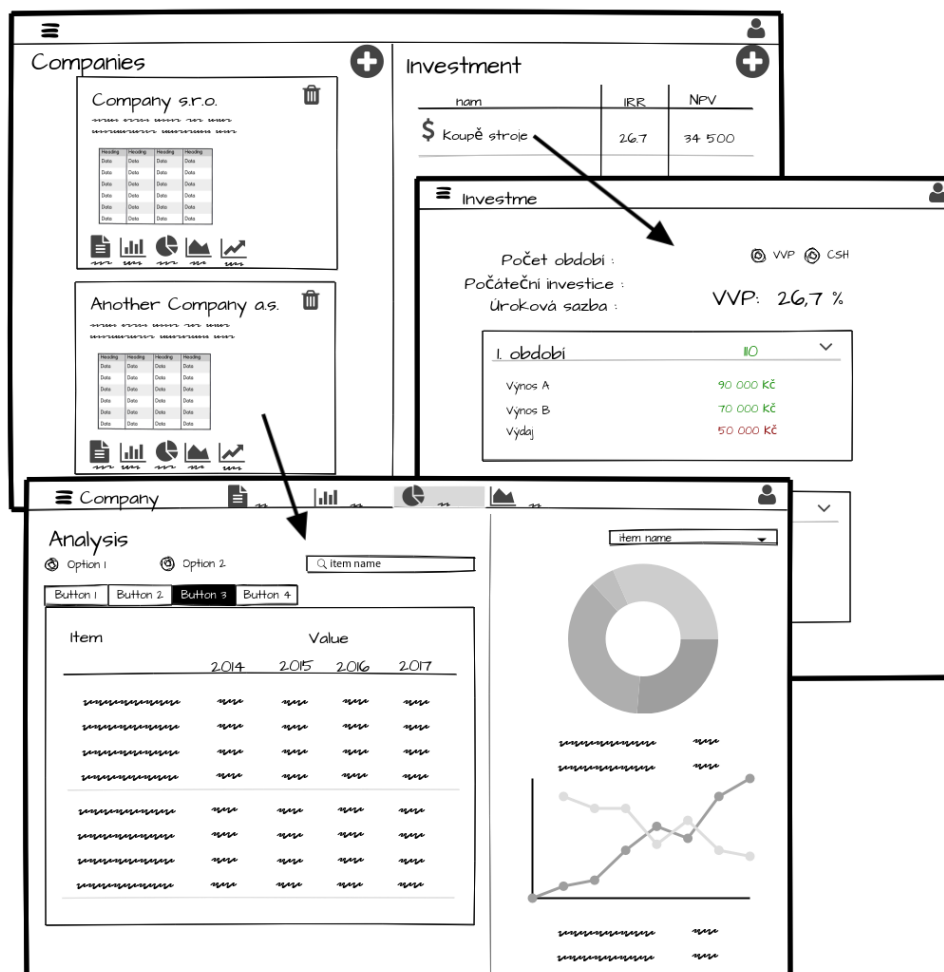
Obrázek 3.2: Schéma datového toku single page application (SPA)

### Uživatelské rozhraní

Uživateli bude umožněno vytvoření vlastního účtu pro zpřístupnění rozšířené funkcionality. Přihlášením k uživatelskému účtu systém pozná, kdo s ním pracuje. To přináší mnoho možností personalizace a možnost přizpůsobit obsah dle nastavených předvoleb. Uživatelský účet představuje osobu finančního analytika, manažera nebo podnikatele. S účtem budou svázána i data za jednotlivá časová období v minulosti, která lze nahrát a uchovat pro další použití, například porovnání nebo zpřesnění analýzy v budoucnu. Bude také zvaženo jednorázové zpracování dat pro nepřihlášeného uživatele – načtení vstupu a prohlížení výsledků, případně dodatečné upravení vstupu, který se však po skončení běhu aplikace zahazuje.

Je žádoucí, aby přihlášení přetrvalo mezi jednotlivými chody aplikace, při znovunačtení stránky i při vypnutí a zapnutí prohlížeče. K tomuto účelu bude využito lokální uložení (local storage), které umožňuje zápis limitovaného objemu dat do zařízení klienta. Ukládat se samozřejmě nebude jméno ani heslo, ale takzvaný *api token* pro komunikaci, který je uživateli udělen serverem po úspěšné autentizaci.

Domovskou stránkou pro přihlášeného uživatele bude obrazovka s přehledem podniků, které jsou pod jeho účtem evidovány. Podniky mohou být přidávány a odebírány, slouží především pro seskupování účetních závěrek a jejich následnou analýzou. Po výběru podniku se uživatel dostane na obrazovku s přehledem nahraných závěrek, přičemž závěrkou se myslí právě výkaz zisku a ztrát a rozvaha. Závěrky lze přidávat, odebírat a upravovat jejich data, v základu jedna závěrka na jedno účetní období, čili jeden rok. Při přidávání závěrek lze zvolit mezi načtením dat z XML souboru se strukturou DPPDP nebo ručním vyplněním buněk tabulek. Ty bude použita i pro editaci již nahraných nebo zadaných dat. V detailu podniku se bude dále možné přepínat mezi jednotlivými druhy analýz nad těmito závěrkami.

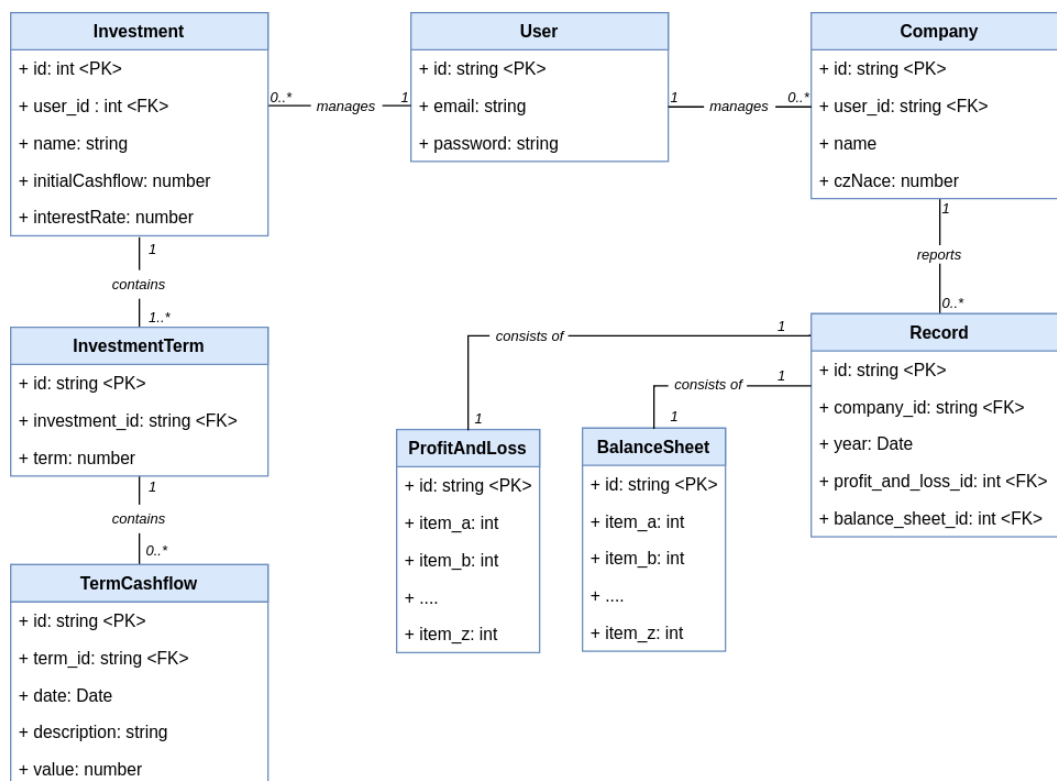


Obrázek 3.3: Domovská obrazovka, detail podniku a detail investice

Kromě podniků budou pod účtem evidovány také investice. Detail investice bude tvořit jediná obrazovka, na které budou zobrazeny výsledné hodnoty i zadané parametry, které půjde editovat.

Vzhledem k nejednotné terminologii napříč celou finanční analýzou je žádoucí, aby při zobrazování výstupu uživatel bezpečně zjistil, co se daným pojmem myslí, co daný výsledek reprezentuje. To bude zajištěno výčtem alternativních označení (a možností vložení a výběru vlastního označení), případně možností přímo přejít do sekce, kde jsou jednotlivé ukazatele detailně vysvětleny. K tomuto účelu může být použit text z teoretické části této práce.

Jelikož různé podniky sledují různé ukazatele, bylo by vhodné, aby nástroj uživateli umožnil vybrat zobrazované/preferované ukazatele. S podniky a odvětvími se liší také optimální hodnota konkrétního ukazatele, případně hodnoty hraniční. Tyto hodnoty nelze stanovit obecně, v literatuře se různí.



Obrázek 3.4: ERD na fyzické úrovni

## Serverová část

Serverová část bude sloužit k uchovávání persistentních dat a uspokojování požadavků klienta. Persistentními daty jsou vytvořené investice, data investic, podniky, v rámci podniku nahrané výkazy, rovněž je nutné vést databázi uživatelů samotných.

Dále bude server obsluhovat požadavky klienta. Vzhledem k tomu, že je nástroj koncipován jako SPA (single page application), ke stažení veškerého kódu aplikace ke klientovi bude sloužit jeden jediný endpoint. Klient tak bude moci s nástrojem pracovat bez dodatečného volání serveru, vyjma operací nad daty, jejichž uložení je na serveru. Server bude dále zpřístupňovat rozhraní pro CRUD operace nad těmito daty.

Zda se budou výpočty provádět na straně serveru nebo klienta se rozhodne až při implementaci a testování. Preferovaný je výpočet na straně klienta, při většině metod analýzy se neprovádějí náročné výpočty a objem zpracovávaných dat je relativně malý. Přesto je však potřeba dbát na to, aby zátěž u klienta nebyla znát.

### 3.3.3 Klíčkové knihovny a technologie

K implementaci bude použita kombinace programovacích, skriptovacích, značkovacích a dotazovacích jazyků. Pro klientskou část jsou to HTML, CSS a Javascript (přesněji TypeScript). Kód serveru bude v jazyce PHP s připojením na MySQL databázi.

## Typescript

Jedná se o nastavbu jazyka JavaScript vyvíjenou společností Microsoft, jejíž hlavním přínosem je statické typování a další techniky známé z objektově orientovaného programování – třídy (ačkoliv ty jsou podle nejnovější specifikace ECMAScript součástí samotného JavaScriptu), rozhraní, moduly a podobně.

Jelikož se jedná o nastavbu, lze používat i čistý JavaScript bez použití typů, což nám umožní použít i knihovny, které jsou implementovány v čistém JavaScriptu. Otypování kódu však zvyšuje jeho přehlednost, snižuje chybovost, otevírá nové možnosti pro vývojové prostředí (našeptávání, refaktorizace kódu) a tím zkvalitňuje vývoj. Proto budeme vyvíjet striktně typovaný kód.

## React

React je populární frontendová knihovna v jazyce JavaScript vyvíjená společností Facebook. Jejím jediným cílem je tvorba uživatelského rozhraní pomocí komponent, jejichž kód je zapsán deklarativně – spíše definujeme, co se má udělat (jak má komponenta vypadat) a ne jak se to má udělat (to za nás řeší životní cyklus komponent). Při změně modelu, který je uložen v JavaScriptu, React na základě definice komponent pozná, které komponenty (a na pozadí tedy i uzly DOM) to ovlivní, a překreslí pouze ty. Velkou výhodou je tedy oddělení modelu od prezentační vrstvy (v případě jQuery je model uložen přímo v HTML) se současným použitím deklarativního zápisu. Komponentě poté už jen dodáváme stále nová data.

Systém komponent nás kompletně odstíní od manipulace s DOM, což bývá v případě komplexních UI, i s pomocí jQuery, velmi nepohodlné. Aplikace manipulující přímo s DOM jsou také velmi náročné na údržbu (změna na jednom místě vede k změnám na mnoha jiných místech) a její části hůře znovupoužitelné.

Aplikace se tedy skládá z komponent. Komponenty jsou do sebe hierarchicky vnořeny, vzniká strom komponent. Pro zápis rozhraní, zanoření i vzhledu komponent se používá jazyk JSX, což je jakási kombinace JavaScriptu a HTML. Do šablon s definicí vzhledu za použití HTML jsou interaktivně doplňována data pomocí různých JavaScriptových konstrukcí (iterace, selekce, předávání parametrů apod.). Kromě toho má každá komponenta svůj stav uložený v proměnné a běžně obsahuje i metody pro obsluhu komponenty, nejčastěji změnu stavu. My se použití stavu komponent vyhneme, namísto toho budeme pracovat s jedním globálním stavem společným pro všechny komponenty, což je popsáno v další sekci.

## Redux

Velkým trendem současných aplikací napsaných s využitím knihovny React je současné použití knihovny React Redux, což je knihovna Redux přizpůsobená pro integraci do React aplikace. Hned na začátku zmíním, že React a Redux jsou dvě zcela odlišné knihovny, které mohou fungovat jedna bez druhé. Zatímco React je zaměřen na jednoduchou tvorbu komponent uživatelského rozhraní a jejich hierarchizaci v aplikaci, Redux se spíše zaměřuje na správu dat na aplikační úrovni. Tato knihovna elegantně řeší uložení a práci s globálním stavem aplikace na základě tří principů:

1. **princip stavu** – stavový strom celé aplikace je uložen v jednom *store*<sup>3</sup>:
  - jeden store pro celou aplikaci, rozdělení na úrovni reducerů (viz dále)

---

<sup>3</sup>český ekvivalent by mohl být kontejner

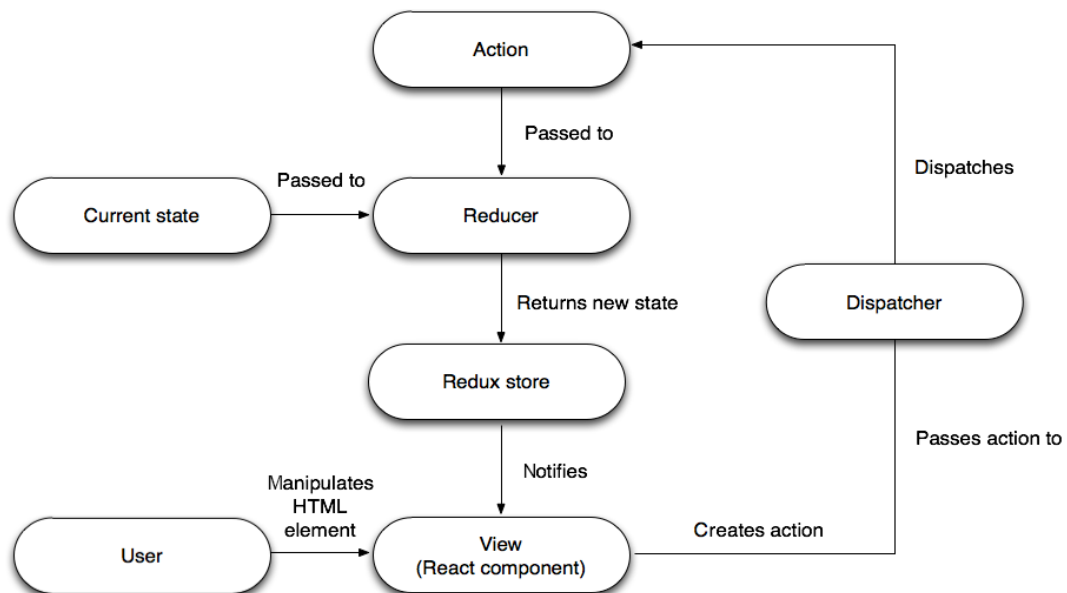
- stav je imutabilní – nelze modifikovat, pouze vytvořit nový
- lze registrovat funkci pro zavolání při změně stavu (*subscribe*)

2. **princip akcí** – stav aplikace lze měnit jen a pouze vyvoláním *akce*:

- akce je vyvolána pomocí funkce *dispatch*
- akce má formu Plain JavaScript Object (asociativní pole)
- akce obsahuje atribut *typ* a obsahuje také data (pro zpracování reducerem)

3. **princip reducerů** – změna stavu aplikace v závislosti na akci je popsána v *reducerech*:

- reducer je čistá funkce (výstup je závislý pouze na vstupních argumentech)
- vstupem jsou dva argumenty: současný stav aplikace a vyvolaná akce
- výstupem je nový stav aplikace (nemodifikuje současný stav)



Obrázek 3.5: Diagram změny stavu s použitím knihovny Redux (převzato z internetu)

Pojmem stav aplikace jsou myšlena jak data ze serveru, tak i data lokálního charakteru, například uživatelské vstupy, ve větších aplikacích také veškerá nastavení ovlivňující vykreslování komponent, jelikož s rostoucí aplikací se tato nastavení zesložitují. Díky přístupu Reduxu lze se stavem aplikace velmi pohodlně pracovat, za největší výhodu považuji jeho dlouhodobou udržitelnost. Uložení stavu na jedno jediné místo je ulehčena jeho inspekce a ladění, díky němu je také možné jednoduchým způsobem uchovat stav aplikace pro další spuštění, což významně urychluje vývojářský cyklus, či porovnávat jednotlivé stavy mezi sebou pro sledování změn.

Princip akcí zase zajišťuje to, že žádný kus kódu nevyvolá změnu stavu jinak než pomocí funkce *dispatch*, které předá objekt akce popisující požadovanou změnu. Tím je zajištěna centralizace provádění změn a lze tak snadno sledovat postupně vyvolané požadavky na změnu stavu. Jelikož jsou akce popsány pomocí Plain JavaScript Object, jsou snadno serializovatelné a mohou tak být zalogovány či uloženy pro pozdější inspekci.

Při tvorbě reducerů si musíme dát pozor, abychom nemodifikovali stav přímo a nezpůsobovali tak tzv. vedlejší efekty (což bychom nejspíš ve spojitosti s imperativním programováním intuitivně udělali), ale namísto toho vraceli stav nový, s provedenými modifikacemi. Pokud je tato zásada dodržena při zpracovávání všech akcí, je zajištěno, že pro každou stejnou dvojici argumentů (*stav, akce*) bude mít funkce stejnou návratovou hodnotu. To je opět výhodné při ladění aplikace, zejména pokud chceme vyzkoušet vyvolání určité akce ve chvíli, kdy je aplikace v určitém stavu.

## React-Redux

Pomocí samotného Reactu, bez výhod zmíněných v předchozí sekci, bychom udržovali stav naší aplikace velmi těžko a zesložili si tak i vývoj. Knihovna zmíněná v nadpise nám přináší pohodlné napojení Reduxu do React aplikace, čímž zpřístupní všechny jeho výhody. Pro jeho použití stačí knihovnu nainstalovat a obalit vybraný podstrom JSX kódu (obvykle ale kód celé aplikace) elementem, který představuje propojení všech jeho obsažených komponent s kontejnerem stavu (*store*). Tento element přijímá právě objekt typu *Store*, ve kterém je naše definice stavu pro tento podstrom. Přesněji se jedná o možnost propojení. Tento úmysl musí být poté ještě u komponenty explicitně vyjádřen pomocí funkce *connect*. Ta má běžně jeden parametr – deklarativní popis toho, kterou část kontejneru (které proměnné) chce komponenta sledovat. Propojené komponenty jsou poté automaticky upozorňovány na změny a pomocí deklarativního popisu React komponent se změna stavu odpovídajícím způsobem promítne ve změně uživatelského rozhraní.

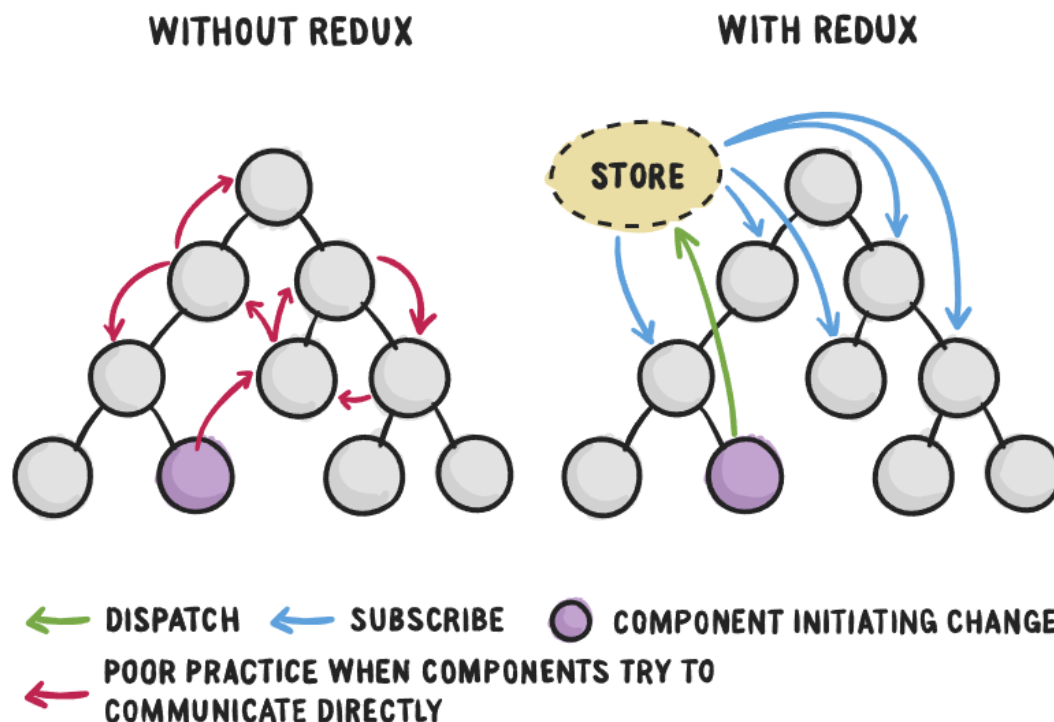
Na obrázku 3.6 je tok dat aplikací při změně zobrazen graficky. Vzhledem ke stromové hierarchii komponent je tok dat v běžné React aplikaci jednosměrný, data jsou předávána z rodičovského uzlu na své potomky. To s sebou nese jisté nevýhody. Komponenty aplikace jsou sice logicky členěné, pokud však jedna z komponent iniciuje změnu dat, aktualizace ostatních komponent závislých na těchto datech je prováděna velmi složitou cestou, která musí být v kódu specifikována. Tento řetězec změn je velmi neprůhledný a může vést k cyklickým závislostem. Napravo je zobrazen tok dat s využitím reduxu. Nejprve je pomocí funkce *dispatch* vyvolána změna stavu, na změny jsou poté upozorněny všechny komponenty, kterých se tato změna dotkne.

## Ostatní

- balík **react-scripts-ts** – startovní balík pro urychlení vývoje, obsahuje skripty pro spouštění, ladění, testování a jednoduché sestavení produkční verze aplikace
- **Material UI** – sada prvků uživatelského rozhraní, usnadní nám design uživatelského rozhraní, separátně taky sada ikon
- **React Router** – bude zajišťovat navigaci v aplikaci, na webové platformě je stav navigace uložen v URL adrese
- **Chart.js** – knihovna pro práci s grafy

## 3.4 Shrnutí a další postup

V dosavadním textu byly zmíněny základy finanční analýzy podniku – její definice, praktický i teoretický význam, byla popsána také vstupní data analýzy. Dále byly zmíněny



Obrázek 3.6: Změna stavu react aplikace – rozdíl bez a s využitím knihovny Redux [10]

a popsány pojmy horizontální a vertikální analýza a výběr těch nejzákladnějších poměrových ukazatelů spolu se vzorcem výpočtu a popisem jejich významu. Z oblasti podnikové analýzy byla zmíněna také analýza peněžních toků a Du Pont rozklad. Z oblasti investic bylo popsáno vnitřní výnosové procento a sním související čistá současná hodnota.

Ještě před návrhem vlastního programového řešení byl proveden průzkum současných možností v této oblasti. Na jeho základě byl vypracován vlastní návrh, zohledňující přednosti i nedostatky vybraných nástrojů. Hlavním rozdílem je architektura aplikace (klient-server) a prostředí běhu (webový prohlížeč). Tento výběr následuje trend moderních aplikací a činí nástroj dostupnějším. Dalším významným rozdílem je výběr technologií, které učiní kód strukturovaným, lehce rozšiřitelným, dlouhodobě udržitelným a vysoce znovupoužitelným. Co do oblasti finanční analýzy, hlavním rozdílem bude vlastní přístup k ohodnocení výsledných poměrových ukazatelů i její technické řešení umožňující jednoduchou parametrizovatelnost a rozšiřitelnost analýzy.

Při dalším postupu bude ze všeho nejdříve připraveno vývojové prostředí (devstack). To obnáší instalaci, zprovoznění a nastavení IDE, serveru, databáze, instalace správce balíčků a balíčků samotných, vytvoření repozitáře, obstarání testovacích sad. Následně může začít implementace klienta, zcela nezávisle na implementaci serveru. Detaily implementace budou uvedeny dále v této zprávě.



## Kapitola 4

# Implementace a testování

V této kapitole bude popsáno technické řešení nástroje pro provádění finanční analýzy. Nejprve bude probrána implementace činností spojených s finanční analýzou, poté implementace kódu tvořící kostru nástroje a jejich společná integrace do výsledné aplikace.

### 4.1 Implementace finanční analýzy

Nástroj bude podporovat většinu typů analýzy zmíněných v teoretické části. Jmenovitě to jsou:

- **podnik** (čerpá data z výkazů účetní závěrky)
  - analýza finančních toků (cashflow)
  - horizontální analýza
  - vertikální analýza
  - analýza poměrových ukazatelů
  - Du Pont rozklad rentability vlastního kapitálu
- **investice** (počáteční investice, délka, peněžní toky, diskontní sazba)
  - výpočet vnitřního výnosového procenta
  - výpočet čisté současné hodnoty

Tento výběr nám ukládá pracovat se dvěma druhy vstupních dat. Vstupními daty pro analýzu podniku jsou výkazy účetní závěrky podle specifikace daňového portálu, pro investici je to vlastní výběr nejdůležitějších aspektů investice.

Nejprve tedy specifikujeme struktury pro popis vstupních dat. Implementaci jejich načítání a uložení můžeme odložit na později, z počátku si vystačíme s uměle vytvořenými daty. Jakmile jsou struktury specifikovány, můžeme začít s implementací modulu pro provádění analýzy, jejichž vstupem jsou tyto struktury.

Vzhledem k rozsahu vzorku podnikových dat, na kterém byla aplikace po celou dobu vývoje testována, jsou podporovány roky 2014 až 2017. Po prozkoumání specifikace struktury z těchto let si lze všimnout, že nejsou zdaleka identické. Mapování rozdílných struktur finančních výkazů na strukturu společnou pro všechny podporované roky bylo jednou z největších výzev tohoto projektu. Problém se však podařilo vyřešit, což je popsáno v následujícím textu. Řešení tak umožňuje sledovat vývoj výsledných položek analýzy podniku napříč

všemi podporovanými roky. Po zběžném prozkoumání specifikace starších struktur výkazů jsem došel k závěru, že díky zvolenému řešení by rozšíření podpory pro tyto struktury nemělo být problémem, stejně tak jako rozšíření pro podporu nových struktur v budoucnu.

#### 4.1.1 Vnitřní reprezentace vstupních dat

Jelikož jsou vstupními daty finanční analýzy podniku účetní data podniku, která mají oporu v legislativě, budeme dodržovat legislativou stanovený formát. To uživatelům nejen zpřehlední jejich zadávání, ale také umožní implementaci jejich automatizovaného načítání dle specifikace DPPDP8. Co se týče investic, zde žádný rozšířený standard není, atributy investice budou zadávány ručně.

Vnitřní reprezentace vstupních dat nepředstavuje žádnou složitou strukturu. Naopak, všechny potřebné hodnoty, informace a závislosti jsou uloženy jako několik struktur s jejich výčtem. Vzhledem k nespočtu změn a výjimek ve specifikaci výkazů z různých let by uložení a popis všech těchto vlastností jednou sofistikovanou strukturou tvořilo tuto strukturu opravdu velmi komplexní. I kdyby se nám podařilo veškerou sémantiku výkazů do struktury zakódovat, takové řešení by nejspíš bylo při další změně legislativy, která je velmi pravděpodobná, velmi těžké modifikovat.

Namísto toho si definujeme pro každou ze specifikací několik jednoduchých struktur pro různé účely a k nim funkce s těmito strukturami pracující. Funkce se strukturou modelují pouze určitou část specifikace v daném roce a tak je možné v nich dělat jednoduše změny, případně implementovat další. Soubor těchto struktur pro určitou specifikaci nazýváme *mapa*. V kódu je mapa reprezentován čistým JavaScript objektem, který lze přirovnat k asociativnímu poli klíč-hodnota. Každý uživatelem uložený výkaz potom obsahuje kromě hodnot položek také odkaz na mapu (jsou spojeny přes ročník výkazu), která definuje, jak se má s výkazem pracovat. Mapa obsahuje následující položky (pro aktiva, pasiva a výkaz zisku a ztráty zvlášť):

- pořadí řádků ve výkazu (reprezentováno polem)
  - pro správné pořadí při výpisu
  - využito také při automatizovaném načítání dat
- označení a plný název položky s daným číslem řádku (asociativní pole)
  - vizuální informace při výpisu
  - klíčem je číslo řádku výkazu
- strom zanoření položek (asociativní pole)
  - zaznamenává vztahy mezi položkami výkazu
  - klíčem je číslo řádku výkazu
  - přepočít souvisejících položek při editaci výkazu
- mapa pro výpočet položek společných pro všechny struktury výkazů (asociativní pole)
  - klíčová při provádění analýzy
  - obsahuje informace o tom, jak z výkazu v daném roce získat hodnotu položek společných pro všechny roky

– klíčem je označení společné položky

Všechna data spojená s položkami výkazů jsou uložena v objektu, jehož klíče tvoří čísla řádků těchto položek. To umožňuje rychlý přístup – pokud máme k dispozici číslo řádku, s nímž chceme pracovat, data spjatá s tímto řádkem lze získat přímým přístupem do objektu přes daný klíč. Pokud by byly řádky a s nimi spojená data uloženy v poli, což se ve spojitosti s řádky určitě nabízí, pokaždé bychom museli iterovat celým polem, dokud bychom nenarazili na data odpovídající vybranému řádku.

Při práci s položkami, například jejich výpisu do tabulky, je žádoucí dodržet pořadí stanovené na stránkách daňového protálu<sup>1</sup>, jehož číselná řada však není konzistentní. Například u tabulky pasiv je běžné, že se mezi prvními položkami s čísly řádků tvořícími posloupnost přirozených čísel najednou objeví položka s číslem řádku o několik desítek vyšším. To je evidentně způsobeno dodatečným přidáváním nových položek. Ty jsou umísťovány k souvisejícím položkám (nikoliv na konec), nicméně čísla řádků následujících položek zůstávají a tak je nová položka označena úplně novým, nejvyšším číslem řádku. Jinými slovy, atribut *číslo řádku* položky ve specifikaci obecně neodpovídá pořadí položky tamtéž a tím pádem ani pořadí v poli.

Mapa s označením a plnými názvy položek sice obsahuje všechna čísla řádků, jelikož jsou to ale klíče objektu, pro jejich iteraci by se muselo použít konstrukce `for ... in` (případně `Object.keys()`), která však v prohlížeči Chrome nezachovává pořadí, v jakém jsou klíče zadány. Standard ECMAScript pořadí klíčů při jejich výčtu nijak nespecifikuje [5], to je tak ponecháno na vývojářích. Abychom byli schopni vypsát řádky nejen všechny, ale i ve správném pořadí, obsahuje mapa také pole čísel představující toto pořadí. Na rozdíl od objektu, pole pořadí svých položek zachovává. Nástroj dále využívá pole s pořadím řádku ve výkazu při automatizovaném načítání jednotlivých výkazů a jejich uložení do vnitřní reprezentace, což je popsáno v následující podsekcí.

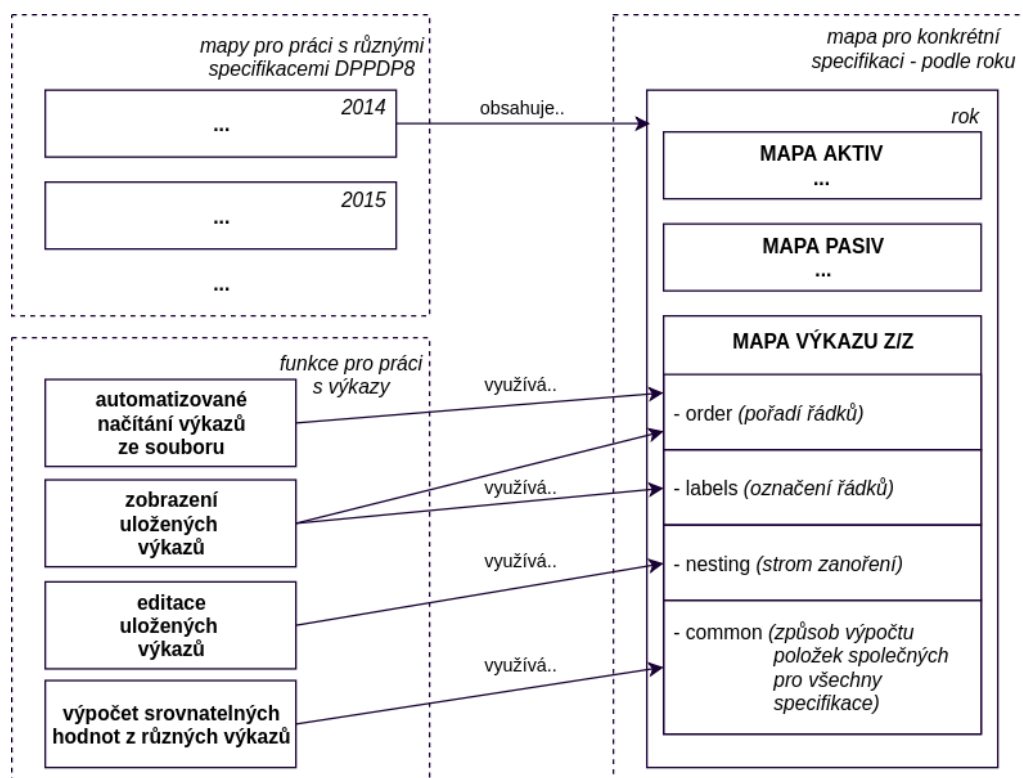
Pro zajištění konzistence položek při ručním vyplňování nově vytvořených výkazů (t.j. výkazů s nulovými hodnotami) či editaci již stávajících výkazů je zase důležitý zmíněný strom zanoření, což je opět popsáno v příslušné podsekcí.

Lze vydedukovat, že rozdílná struktura výkazů v různých letech bude mít velký dopad na porovnatelnost položek a tedy výpočet významově identických hodnot pro všechny podporované roky. Tento fakt byl jednou z největších překážek při implementaci. Byla potřeba především nastudovat legislativu k pochopení těchto změn, vymyslet způsob, jakým vytvářet srovnatelné položky a nakonec jej přepsat do zdrojového kódu. Řešení je popsáno v sekci 4.1.2. Mapa s informacemi, jak společné položky vytvářet, je poslední částí mapy výkazu.

## Načítání souborů podle specifikace DPPDP8

Pro automatizované načítání souborů se syntaxí DPPDP8 pro účely analýzy podniku byl implementován parser. Nejprve je pomocí knihovny `fast-xml-parser` řetězec s obsahem XML souboru přetransformován na odpovídající JSON řetězec. Podle tohoto řetězce je vestavěnou funkcí `JSON.parse()` sestaven odpovídající JavaScript objekt, se kterým můžeme dále pohodlně pracovat tak, jak jsme v JavaScriptu zvyklí. Jelikož je před zpracováním výkazu známý jeho ročník (zadáva uživatel), můžeme si z příslušné mapy načíst pole představující pořadí řádků ve výkazu a podle tohoto pole poté načíst položky ze vstupního souboru.

<sup>1</sup>[https://adisepo.mfcr.cz/adistc/adis/idpr\\_pub/hlib/uv\\_info/uv\\_info.faces](https://adisepo.mfcr.cz/adistc/adis/idpr_pub/hlib/uv_info/uv_info.faces)



Obrázek 4.1: Mapa pro práci s různou strukturou výkazů a její využití

## Ruční zadávání dat

Pro případ, že uživatel nemá možnost dodat výkazová data ve formátu XML, je mu umožněno také ruční zadávání. Hodnoty položek vyplňuje do předchystané tabulky, avšak její naplňování a ukládání hodnot není tak přímočaré, jak by se mohlo zdát. Jelikož jsou výkazy určeny primárně pro zpracování člověkem, jsou v nich i v jejich XML reprezentaci uložena redundantní data, zejména hodnoty položek s hodnotou součtů těchto položek představujících nadřazené položky.

Po úvaze byla do kódu uložena pro každou ze struktur výkazů také jejich hierarchie, schéma vzájemného zanoření položek. Toto řešení mi přijde vhodné, jelikož při editaci položky by se měla logicky postupně přepočítat i hodnota všech nadřazených položek. Pokud by naopak uživatel editoval neterminální položku, pro udržení konzistence výkazu by bylo nutné navýšit některou z podpoložek – neznámo kterou. Do tabulky je tím pádem možné zapisovat pouze listové položky, přičemž dojde k automatickému přepočtu všech nadřazených položek.

Při nahrávání výkazů není tato technika aplikována, řádky jsou do struktury v aplikaci nahrány z XML jedna ku jedné. Informace o struktuře výkazů a funkce pro výpočet hodnoty nadřazené položky by se v tomto případě mohly využít pro zjišťování a následném signalizování nekonzistencí ve výkazu, které by měl uživatel ve vlastním zájmu eliminovat.

Stejně jako musely být všechna čísla řádků, označení a názvy položek uloženy explicitně, tak i strom musí být zapsán v kódu kompletním výčtem. Při pohledu na finanční výkazy by se dalo usoudit, že je možné jeho automatické vygenerování na základě označení položek. To je však bohužel možné jen do jisté míry, kvůli nekonzistencím tento způsob rozhodně nefunguje pro všechny položky výkazů. Nejsme schopni z nich odvodit všechny potřebné

souvislosti, tj. jaké jsou všechny podpoložky dané položky a zda jsou jednotlivé podpoložky kladné či záporné.

#### 4.1.2 Společené položky různých struktur výkazů

Mění se struktura finančních výkazů v čase ve specifikaci DPPDP8 byla vzhledem k striktnímu dodržení formátu vstupních dat podle této specifikace jedním z největších problémů týkajících se finanční analýzy, se kterými jsem se při implementaci setkal. Abychom mohli cokoliv analyzovat a porovnávat výsledky analýzy za různá časová období, je nutné nejprve v každém z těchto období získat odpovídající, srovnatelná data.

Do roku 2015 včetně byly změny struktur minimální, zejména se jednalo o rozpad nějaké ze stávajících položek na více podpoložek. V roce 2016 však došlo k významným změnám – k přesunům celých položek a především jejich částí pod jiné položky. Některé položky zůstaly nezměněny, u některých došlo ke změnám, které lze pomocí výpočtu eliminovat a získat tak srovnatelná data pro všechna období, u ostatních položek změna znemožňuje jejich použití.

Samotné výkazy nám situaci neulehčují, v mnoha případech jsou totiž položky, u kterých došlo ke změně, pojmenovány podobně nebo dokonce stejně jako před změnou. Nalezení změny tedy spočívá v porovnání seznamu účtů účetní osnovy použité k tvorbě dané položky. Komplettní procházení všech výkazů a následné kontroly na změnu nám ulehčí převodový můstek vypracovaný v Českém účetním standardu pro podnikatele, který lze nalézt přímo na stránkách Ministerstva Financí ČR<sup>2</sup>. Je to co nejpodrobnější popis změn ve specifikaci, ale pouze na úrovni účetní závěrky. Ke každé z položek z nové specifikace je uvedena, tam kde existuje, srovnatelná položka ze specifikace předchozí. Tento zápis změn je tak pro naše použití ideální.

Na řádcích převodového můstku, kde je u odpovídající položky z minulých let napsáno kurzívou „(část položky)“, máme bohužel smůlu. Změna je totiž na úrovni účetnictví a z dodaných výkazů není možné vypočítat, o jak velkou část položky se jedná. Tyto položky tedy musely být vynechány také. V případě ostatních položek byly vybrány či vytvořeny takové, jejichž hodnota půjde v každém z roků nějakým způsobem získat.

Technickým řešením bylo tedy tyto položky stanovit a poté zapsat jejich výpočet pro každou specifikaci zvlášť. Výpočet je realizován opět funkcí a strukturou. Ve struktuře je ke každé z těchto vybraných společných položek uloženo pole kladných a pole záporných položek dané specifikace. Výsledek tedy získáme sčítáním a odečítáním položek – vzájemného rozdílu sumy hodnot obou polí.

Tyto společné položky se používají pro všechny druhy analýzy, původní výčet položek v jednotlivých letech je využit zejména při ukládání, zobrazování a editaci účetních výkazů a pro účely demonstrace také při horizontální analýze. Cílem bylo vytvořit maximální počet společných položek s postupným rozpadem na menší a menší části, čímž se maximalizuje počet možných výpočtů a tím pádem počet výsledků analýzy.

#### 4.1.3 Provádění analýzy

Podařilo se implementovat všechny typy analýz zmíněné na začátku této kapitoly. Navíc bylo implementováno také mezipodnikové srovnání. To sice není omezeno počtem podniků, ale způsob prezentace výsledků srovnání většího počtu podniků v delším časovém období

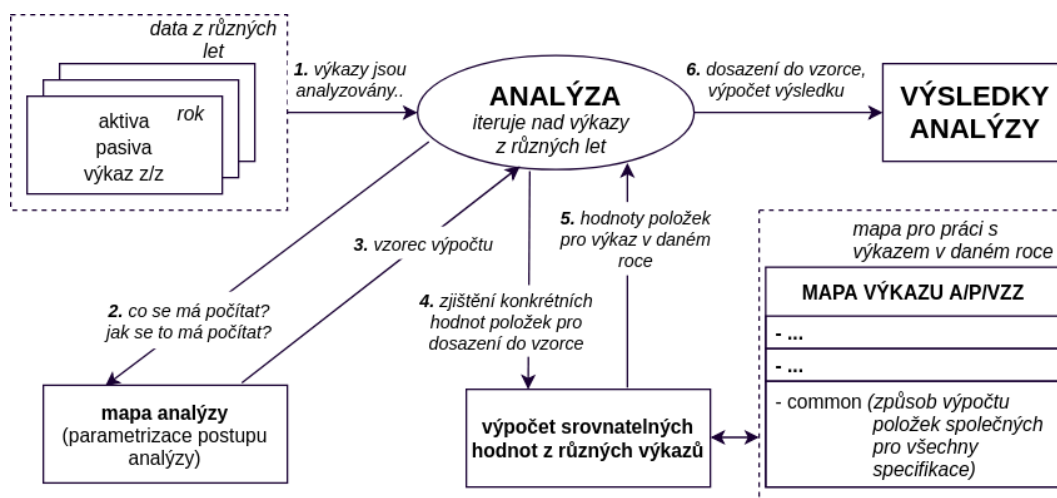
---

<sup>2</sup>[https://www.mfcr.cz/assets/cs/media/Ucetnictvi\\_2016\\_Ceske-ucetni-standardy-pro-500-2002.pdf](https://www.mfcr.cz/assets/cs/media/Ucetnictvi_2016_Ceske-ucetni-standardy-pro-500-2002.pdf)

zatím není vyřešen. Mezipodnikové srovnání lze provést v rámci horizontální analýzy a analýzy poměrových ukazatelů, implementace je však v tomto směru snadno rozšiřitelná i na ostatní typy analýzy díky způsobu uložení výsledků.

Výsledek není uložen v běžném (indexovaném) poli, ale v poli asociativním. Nevýhodou je sice nutnost explicitní iterace nad jednotlivými výsledky analýzy, tento způsob uložení je však velmi příznivý při znovupoužití vypočítaných výsledků – například při výpočtu rozdílu hodnot jedné položky v různých letech nebo srovnání podniků v rámci některé z analýz. Nejprve se vypočítají výsledky pro všechny srovnávané podniky, poté je nutné podnikové výsledky nahrát do jedné společné struktury. Pokud by byly výsledky pro podnik uloženy v běžném poli, museli bychom pro každou z položek respektive pro každý z analyzovaných roků procházet všechny položky respektive roky a hledat mezi nimi ten, který se právě chystáme zpracovat. Díky asociativnímu poli můžeme na hledanou hodnotu přistoupit přímo, přes klíč položky a následně klíč roku, neprobíhá tedy žádné vyhledávání na úrovni naší implementace.

Všechny podporované analýzy jsou implementovány podobným principem, který byl použit i při práci se vstupními daty. Každá z analýz má svůj vlastní popis – *mapu* (objekt), ve které jsou uloženy všechny potřebné informace: seskupování a hierarchie výsledků, pořadí výsledků, způsob jejich výpočtu, označení a jednotka. Pro práci s mapou je implementována funkce vykonávající analýzu nad konkrétními daty. Funkce odpovídající mapu systematicky prochází a čerpá z ní informace pro výpočet, do kterého dosazuje data získané ze svého jediného parametru – finančních výkazů. Výsledek uloží do struktury opět podle informací z mapy. Tento způsob implementace považuji za flexibilní a velmi dobře parametrizovatelný, zároveň zredukuje vazby mezi funkcí provádějící celkovou analýzu jistého typu a analýzou jedné konkrétní položky.



Obrázek 4.2: Diagram provádění analýzy

## Horizontální analýza

Horizontální analýza je prováděna nad položkami výkazu zisku a ztráty, aktivy i pasivy, výsledné hodnoty jsou rozděleny stejným způsobem. Na nejvyšší úrovni je tato analýza prováděna dvěma způsoby. Ve zdrojovém kódu je uložena informace o tom, které finanční výkazy mají stejnou strukturu (podle roku). Díky tomu lze uložené výkazy seskupit a pro-

vést nad těmito skupinami separátně horizontální analýzu na detailní úrovni – pro výkazy se stejnou strukturou je analyzována každá z položek těchto výkazů; potřebné informace, jako jsou názvy, označení a pořadí položek jsou uloženy přímo v mapě struktury těchto výkazů. Analýza určitého počtu skupin se stejnou strukturou přináší stejný počet struktur s výsledky. Je to, pro demonstraci, jediná z analýz, ve které je zohledněna různá struktura výkazů z různých let (tato skutečnost je zohledněna samozřejmě také v editoru výkazů).

Tímto způsobem však nejsme schopni analyzovat delší časové období, vzhledem k podporovaným letem v implementovaném nástroji jsou to maximálně 2 roky. Abychom mohli analyzovat výkazy v delším časovém horizontu, využijeme mapy společných položek napříč různými strukturami výkazů. Pro účely horizontální analýzy je nutné definovat v rámci každého z výkazů pořadí těchto společných položek. Při výpočtu konkrétních hodnot položek tak stačí nad uloženým pořadím iterovat a přes odpovídající funkce zjistit název položky a vypočítat její hodnotu pro daný rok.

Kromě konkrétních hodnot položek v daných letech jsou vypočítány i meziroční rozdíly hodnot – absolutní a relativní (procentuální). Ze vzorce pro výpočet relativního rozdílu vyplývá nutnost explicitně ošetřit případ, kdy je hodnota položky v předchozím roce nulová.

$$d_{ABS} = h_n - h_{n-1}$$

$$d_{REL} = (h_n - h_{n-1})/h_{n-1}$$

V rámci horizontální analýzy bylo implementováno také mezipodnikové srovnání, je tak možné porovnávat atributy podniků a jejich hospodaření plynoucí přímo z finančních výkazů.

## Vertikální analýza

Při vertikální analýze se nezohledňuje různá struktura výkazů z různých let, používají se pouze společné položky. Jedna položka vertikální analýzy obsahuje vždy vztahnou veličinu a pole částí, z nichž je složena a jejichž procentuální zastoupení chceme vyjádřit. Tyto položky jsou obsaženy pro každý z výkazů – pro aktiva, pasiva i výkaz zisku a ztrát. Při výpočtu algoritmus iteruje nad výkazy, v rámci výkazů dále iteruje nad položkami. U každé položky nejprve spočte hodnotu všech jejích částí, poté hodnotu položky samotné (vztahné veličiny), z nichž lze vypočítat podíl částí na celku.

Při vertikální analýze je potřeba dbát na to, aby byly vybrané položky vzájemně disjunktní – počítáme procentuální zastoupení částí na celku a proto by se části neměly překrývat. Jinak by se mohlo stát, že suma procentuálního zastoupení všech částí by přesáhla 100 %. Kvůli různým strukturám výkazů a problému nastíněném v sekci 4.1.2 je však situace spíš opačná – při rozložení některých celků na části nejsme schopni celek pokrýt.

## Analýza poměrových ukazatelů

Tak jako u ostatních analýz, i výpočet poměrových ukazatelů je realizován pomocí funkce, která se řídí mapou pro provádění analýzy a podle této mapy získává hodnoty z dodaných výkazů. Mapa na nejvyšší úrovni obsahuje skupiny, či spíše druhy poměrových ukazatelů (rentabilita, likvidita atd.). V rámci každé skupiny je uloženo pole prvků výčtového typu. Tyto prvky představují konkrétní ukazatele, přesněji jsou to klíče, přes něž lze k datům spojeným s daným ukazatelem přistupovat – momentálně jsou to název a způsob výpočtu ukazatele. Stejný klíč je využíván i ke zjištění průměrné hodnoty ukazatele v odvětví. Uložení klíčů v poli je stanoveno jejich pořadí při analýze.

V rámci analýzy poměrových ukazatelů, která tvoří stěžejní oblast finanční analýzy podniku, bylo implementováno také mezipodnikové srovnání.

### **Ohodnocení poměrových ukazatelů**

Pro některé ukazatele, například ukazatele likvidity, lze najít doporučené hodnoty ve většině zdrojů. Pakliže porovnáme hodnoty z těchto zdrojů, zjistíme, že se různí. Pro jiné ukazatele není optimální hodnota uváděna vůbec. V žádném případě nelze pouze z jeho hodnoty, která za sebou skrývá mnoho souvislostí, bez lidského faktoru vyvozovat závěry a výslednou hodnotu jednoznačně ohodnotit. V praxi toto platí dvojnásob – hodnoty ukazatelů mohou nabývat extrémních odchylek, přičemž to nemusí nutně signalizovat špatnou situaci podniku. Vyvozování závěrů tak necháme na uživateli, přičemž k podpoře této činnosti se pokusíme dodat co nejrelevantnější informace.

Za velmi užitečné referenční hodnoty pro tento účel považuji výsledky finanční analýzy podnikové sféry, prováděné kvartálně Ministerstvem průmyslu a obchodu. Tato volba nám také stanovuje přesný vzorec výpočtu jednotlivých ukazatelů z mnoha variant – aby byly hodnoty porovnatelné, při výpočtu se budeme řídit metodikou MPO. Pro většinu ukazatelů použitých v mé implementaci jsou ve výsledcích celorepublikové analýzy přímo hodnoty, pro jiné jsou zde hodnoty potřebné pro jejich výpočet a pro ostatní nelze průměr zjistit nijak.

V analýze podnikové sféry se statistiky dělí podle tzv. klasifikace ekonomických činností „CZ-NACE“, která nabývá číselných hodnot – každé odvětví má své číselné označení. V kódu aplikace jsou tyto statistiky uloženy jako asociativní pole, jehož klíče tvoří čísla CZ-NACE. Číslo CZ-NACE je uloženo i u podniku. Při jeho analýze tak víme, pod jakým klíčem ve statistikách hledat. Statistiky pro vybrané odvětví jsou opět reprezentovány asociativním polem, tentokrát kvůli rozdělení statistik na hodnoty jednotlivých ukazatelů.

### **Du Pont rozklad**

Pro implementaci byl z pyramidových rozkladů vybrán rozklad rentability vlastního kapitálu. Většina prvků rozkladu jsou hodnoty již zmíněných poměrových ukazatelů (ukazatele představují uzly diagramu) či položek finančních výkazů (položky představují listy diagramu). Přesto byl pro tuto analýzu vytvořen separátní modul. V něm jsou doplněny především metody pro výpočet ukazatelů, které nebyly implementovány v rámci analýzy poměrových ukazatelů. Modul funguje také jako mezivrstva – pokud bychom chtěli výslednou hodnotu ukazatele upravit speciálně pro účely Du Pont analýzy, nebo ji doplnit o další informace, můžeme to udělat zde.

Jelikož jsou při grafické interpretaci pyramidového rozkladu zobrazeny také matematické vzorce pro výpočet jednotlivých uzlů, v modulu jsou uloženy i tyto vzorce. Rovnice a funkce vyčísľující výsledek jsou však uloženy zvlášť, lepším řešením by mohlo být uložení vzorce do abstraktního syntaktického stromu, což by umožnilo čerpat vzorec i výsledek výpočtu z jednoho zdroje. Více se o tomto řešení zmiňuji v sekci možných rozšíření.

### **Analýza cashflow**

Jelikož tato implementace dále s výsledky cashflow analýzy nepracuje, jsou pro usnadnění práce při jejich vykreslování uloženy v běžném poli. Mnoho vzorců z oblasti poměrových ukazatelů lze však změnit dosazením výsledků analýzy cashflow namísto stávajících členů, čímž získáme zase další užitečnou vypovídající hodnotu. Po přepracování struktury výsledků



cashflow analýzy, zejména jednoznačným označením jednotlivých položek, lze nástroj rozšířit o další typy ukazatelů.

Dosavadní implementace umožní analytikovi sledovat peněžní toky za různá období. Tok nese kromě hodnoty také informaci o tom, proč vůbec vznikl a dále je také zařazen mezi jednu ze tří klíčových činností. Samozřejmě je vyčíslena i suma toků pro každou z činností a celková suma toků.

## **Analýza investice**

Vstupem pro všechny doposud zmíněné analýzy jsou roční výkazy podniku – rozvaha a výkaz zisku a ztrát. Pro výpočet vnitřního výnosového procenta určité investice či projektu se však používají spíše peněžní toky výhradně s ním spojené. Prvním peněžním tokem obvykle bývá počáteční investice, která je intuitivně evidována zvlášť. Zbývající peněžní toky je nutné začlenit do období, ve kterém proběhnou. Dalším vstupním parametrem je tedy délka trvání investičního projektu. Tyto toky mají také finanční hodnotu (kladnou či zápornou) a pro potřeby uživatele může být textem specifikován jejich účel.

Při analýze investice budeme zjišťovat hodnotu její čisté současné hodnoty a hodnotu vnitřního výnosového procenta, které spolu úzce souvisí. Výpočet ČSH je přímočarý, spočívá v pouhém dosazení do vzorce při dodatečné specifikaci úrokové sazby. Analytické řešení výpočtu VVP však není známo, výpočet provedeme numericky. Pro tento účel jsem zvolil metodu půlení intervalu, která mi přijde ze všech numerických řešení nejnázornější a nejjednodušší na implementaci.

Vhodným vylepšením této analýzy by byl výpočet ČSH i VVP v průběhu celé investice, tedy jejich hodnot na konci každého z období. To by pomohlo investorovi zvážit změnu délky trvání investice.

## **4.2 Implementace prostředí**

Doposud byla popsána implementace modulů provádějících činnosti spojené s finanční analýzou, v této sekci bude popsána implementace nástroje jako celku.

### **4.2.1 Datový tok**

Při implementaci architektury a persistence dat jsem se mírně odchýlil od návrhu. Původním záměrem byla implementace vlastní serverové části s napojením na MySQL databázi. Po přezkoumání možností mi přišlo výhodnější využít služby Firebase poskytované společností Google. Činnosti jako instalace, konfigurace, zprovoznění a implementace vlastního serveru a jeho další úpravy při změnách na straně klienta by přinesly poměrně velké množství práce navíc, přičemž způsob technického řešení nástroje není předmětem zadání. Zvolená databáze je objektová, její schéma zhruba odpovídá ERD diagramu v kapitole návrhu. Vystupují zde ty samé entity s těmi samými atributy, data jsou pouze jinak strukturovaná.

Jak je vidět na obrázku 3.2, serverová část v mém návrhu měla sloužit pouze pro stažení kódu aplikace při jejím spuštění a dále už jen pro manipulaci s uloženými uživatelskými daty. Přesně toto nám umožňuje služba Firebase, která poskytuje jak hostování kódu aplikace, tak přímý přístup na databázi. Přímým přístupem se zde myslí rozhraní na úrovni aplikace, které umožňuje pokládat dotazy nad daty (strukturami a kolekcemi). Z uvedeného popisu vyplývá, že mezi klientskou částí aplikace a databází není žádná programovatelná vrstva – server, na kterém bychom chtěli spouštět svůj vlastní kód. I pro tento problém má Firebase

řešení, službu Cloud Functions. My ji však nevyužijeme, jelikož se ukázalo, že provádění veškerých implementovaných operací u klienta nepředstavuje výkonnostní zátěž.

Další velkou výhodou služby Firebase je správa uživatelů aplikace a zajištění procesu autentizace. Přes funkce poskytované knihovnou je tak možné velmi jednoduše vytvářet nové uživatelské účty či zajistit celý proces automatického přihlášení, tedy i jeho persistence. Tato služba v oblasti autentizace a autorizace nabízí mnohem více, my se však omezíme na funkcionalitu, kterou jsem zmínil. Posledním důležitým prvkem je jednoznačný uživatelský identifikátor, který je automaticky vygenerován při vytvoření uživatelského účtu. Ten slouží jako klíč v databázi, pod kterým jsou uložena veškerá data spojená s tímto účtem – podniky a investice.

Při přihlášení uživatele dojde ke stažení těchto dat do aplikace, uživateli jsou zpřístupněny evidované podniky a investice, které může analyzovat a editovat. Při jejich editaci dojde nejprve k zápisu změny do databáze, poté jsou z ní znovu načteny všechny podniky či investice. Řešení je sice velmi neúsporné s ohledem na nadbytečné využití prostředků, je to však nejjednodušší způsob synchronizace databázových a aplikačních dat. Pokud by měl být nástroj nasazen, bylo by vhodné objem přenesených dat minimalizovat.

Podle konceptu knihovny Redux, popsáném v sekci [3.3.3](#), jsou všechna data na aplikační vrstvě ukládána do reducerů, jediných zdrojů informací o stavu aplikace. Při spuštění jsou nahrána data tří reducerů:

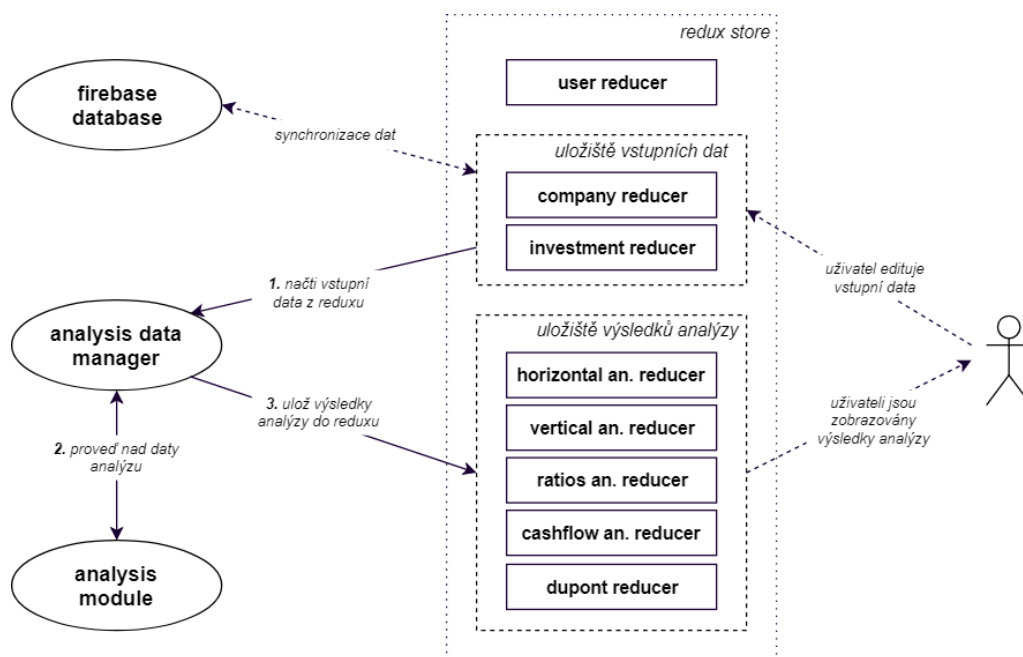
- *user reducer* – uživatelská data
- *company reducer* – data spojená s podniky
- *investment reducer* – data spojená s investicemi

Ostatní implementované reducery slouží pro uložení výsledků jednotlivých analýz. Ty čerpají vstupní data z *company reduceru* a *investment reduceru*. Schéma datového toku při výpočtu a zpracování výsledků je téměř u všech analýz stejný, liší se pouze způsobem výpočtu. Celý proces výpočtu a adekvátní změny globálního stavu je prováděn speciálním objektem, který jsem nazval *data manager*. Výsledkem analýzy jsou buď data, nebo popis problému, kvůli kterému se analýza nezdařila. Schéma je znázorněno na obrázku [4.3](#).

#### 4.2.2 Uživatelské rozhraní

Prostředím běhu aplikace je webový prohlížeč, aplikace byla vyvíjena primárně pro Google Chrome. Vzhled aplikace definují jazyky HTML a CSS, kód je však psán v jazyce JSX. Ten je syntaktickým rozšířením jazyka JavaScript umožňujícím v kódu pracovat s šablonou jako s proměnnou Javascriptu, přičemž šablona má podobu kódu HTML ve kterém je možné použít JavaScriptové konstrukce pro iteraci, selekci a podobně. Před vykreslením je JavaScriptový kód interpretován a konečný vzhled odpovídajícím způsobem sestaven.

Použití samotných prvků HTML bez dodatečného nastýlování je k tvorbě vzhledných aplikací nedostačující, což je nejspíše způsobeno původním záměrem tohoto jazyka. Design webových stránek nebo aplikací je však poměrně rozsáhlá oblast, bylo proto vhodné sáhnout po knihovně hotových, nastýlovaných prvků. Z počátku byla zvažována knihovna Bootstrap, nakonec jsem se rozhodl pro modernější knihovnu Material UI. Knihovna programátorovi poskytuje nejen nastýlované komponenty, ale také řešení rozložení těchto komponent na stránce či sadu ikon. Díky této knihovně bylo možné poměrně snadno vytvořit webovou aplikaci, která vypadá obstojně i po designové stránce.



Obrázek 4.3: Tok dat v aplikaci

Výsledkem většiny analýz jsou tabulární data, sledujeme vývoj souboru položek analýzy v různých letech. Při výběru některé z položek je zobrazen její detail, který obsahuje vhodný graf zobrazující současnou situaci či vývoj položky. Pro vykreslování a práci s grafy byla použita knihovna Chart.js.

## Navigace

Aplikace je v souladu s návrhem koncipována jako single page application. Pokud uživatel přistoupí na kořenovou adresu aplikace s jakýmkoliv suffixem, server vždy vrátí kód představující celou aplikaci, což vyžaduje jeho konfiguraci. Jelikož je tento typ aplikací mezi webovými aplikacemi poměrně rozšířený, možnost tohoto nastavení je zpřístupněna v samotném průvodci konfigurací firebase. Suffix adresy tak není zpracováván serverem, nýbrž aplikací na straně klienta, jedná se tedy o client side routing.

Pro routování na straně klienta, tedy uspořádání obrazovek a pohyb mezi nimi, byla použita knihovna React Router. Části URL adresy jsou pro uživatele popisné. Lze podle nich zjistit, s jakými entitami se pracuje a co se s nimi provádí. URL adresa svým tvarem v podstatě udává stav aplikace – při jejím přečtení víme, kterou obrazovku zobrazit a jakými daty ji vyplnit. Do URL se dá uložit v podstatě jakákoliv informace, my se omezíme na ukládání obrazovek a identifikátorů entit.

## Obrazovky

Po spuštění aplikace se uživatel musí přihlásit, zobrazeno je pouze modální dialogové okno pro vyplnění přihlašovacích údajů nebo registraci. Při úspěšném přihlášení je přesměrován na hlavní obrazovku a je mu zpřístupněna fixní horní lišta aplikace, která slouží jako navigace. V liště je také tlačítko pro odhlášení. Na domovské stránce se lze pomocí horní lišty přepínat mezi dvěma obrazovkami – seznamem podniků a seznamem investic.

### Přihlášení

Email

xkubis13@email.com

Heslo

\*\*\*\*\*

[Vytvořit nový účet..](#) [PŘIHLÁSIT](#)

### Registrace

Email

\_\_\_\_\_

Heslo

\_\_\_\_\_

Heslo znovu

\_\_\_\_\_

[Přihlásit se k existujícímu účtu..](#) [REGISTROVAT](#)

Obrázek 4.4: Dialog pro přihlášení a registraci

Po výběru investice ze seznamu je uživatel přesměrován do detailu investice, kde je možné nastavit její parametry – počáteční rok, velikost počáteční investice a při výpočtu čisté současné hodnoty i hodnotu úrokové sazby. Nezbytnými parametry jsou také peněžní toky v jednotlivých časových obdobích, uživateli je tak umožněno přidávat další roky trvání investice, v rámci nichž zadává libovolně pojmenované výnosy a výdaje. Pokud jsou zadány nevyhovující parametry, je uživatel upozorněn. Pokud jsou k dispozici výsledky, může přepínat mezi zobrazením hodnoty čisté současné hodnoty a vnitřního výnosového procenta.

Start [rok]: 2001

+

–

Počáteční investice [Kč]: 20 000

☒ VWP ☐ NPV

**VVP: 16.72 %**

#### Peněžní toky v roce 2001

Průběžná dotace	5 000	
Ušetření na mzdách	3 000	
<a href="#">+</a>		

Obrázek 4.5: Rozhraní pro práci s investicí

Na obrazovce se seznamem podniků jsou vykresleny karty s jednotlivými podniky a tlačítko pro možnost přidání nového podniku. V rámci konkrétního podniku lze pokračovat na seznam jeho evidovaných výkazů, jednu z obrazovek s analýzou nebo na obrazovku srovnání s jiným podnikem.

Na obrazovce se seznamem výkazů se lze přepínat mezi výkazy z různých let, případně výkaz smazat nebo přidat nový, což je nezbytný krok i při importu dat. Na kartě s výkazem lze ručně vyplnit údaje nebo použít tlačítko pro import. Výkazem se zde rozumí trojice dokumentů – výkaz zisku a ztrát, tabulka aktiv, tabulka pasiv.

Pokud jsou nahrány potřebné výkazy, lze se přepínat mezi obrazovkami s analýzou. Výsledkem všech analýz, až na rozklad Du Pont, jsou tabulární data. U horizontální a

VÝKAZ ZISKU A ZTRÁTY		AKTIVA	PASIVA
č. řádku	označení	název	Kč
1		AKTIVA CELKEM	10 511 599
2	A.	Pohledávky za upsaný základní kapitál	0
3	B.	Dlouhodobý majetek	3 395 961
4	B.I.	Dlouhodobý nehmotný majetek	0
5	B.I.1.	Zřizovací výdaje	0
6	B.I.2.	Nehmotné výsledky výzkumu a vývoje	0
7	B.I.3	Software	0

Obrázek 4.6: Zobrazení výkazu dle specifikace a možnost jeho editace

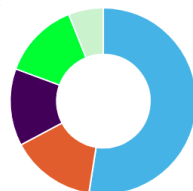
vertikální analýzy jsou to tři sady tabulárních dat podle rozdělení na výkaz zisku a ztrát, aktiva a pasiva, přepínat se lze pomocí tabů. Při výběru položky horizontální analýzy je zobrazen sloupcový graf s hodnotami ve všech zaznamenaných letech. Při výběru položky na obrazovce vertikální analýzy je zaznamenan celek, do kterého položka patří, a také rok, který nás zajímá. Zobrazen je koláčový graf ilustrující rozložení celku na části v daném roce. Obrazovka s analýzou poměrových ukazatelů obsahuje všechny skupiny ukazatelů i jejich položky v jedné tabulce. Pro vybranou položku je zobrazen sloupcový graf. U položek, jejichž průměrná hodnota v odvětví je známa a zadána, jsou tyto průměrné hodnoty zobrazovány vedle hodnot vypočítaných pro vybraný podnik.

#### Vertikální analýza

VÝKAZ ZISKU A ZTRÁTY	AKTIVA		PASIVA	
Položka	Hodnota [%]			
	2014	2015	2016	2017
Provozní výnosy				
Tržby za prodej zboží	16	19.2	17.5	11.6
Tržby z prodeje výrobků a služeb	83.7	80.2	82.3	88.1
Tržby z prodaného materiálu	0.2	0.2	0.2	0.1
Tržby z prodaného dlouhodobého majetku	0	0	0	0.1
Provozní náklady				
Náklady vynaložené na prodané zboží	15.6	18.7	18.6	12.4
Spotřeba materiálu a energie	41.4	39.4	44	48.7
Služby	22.1	18.3	13.7	12.4
Mzdové náklady	14.1	14.9	13.5	13.7
Náklady na soc. zabezpečení, zdrav. pojištění a ostatní	5.2	5.2	5.1	5.6

#### Provozní náklady

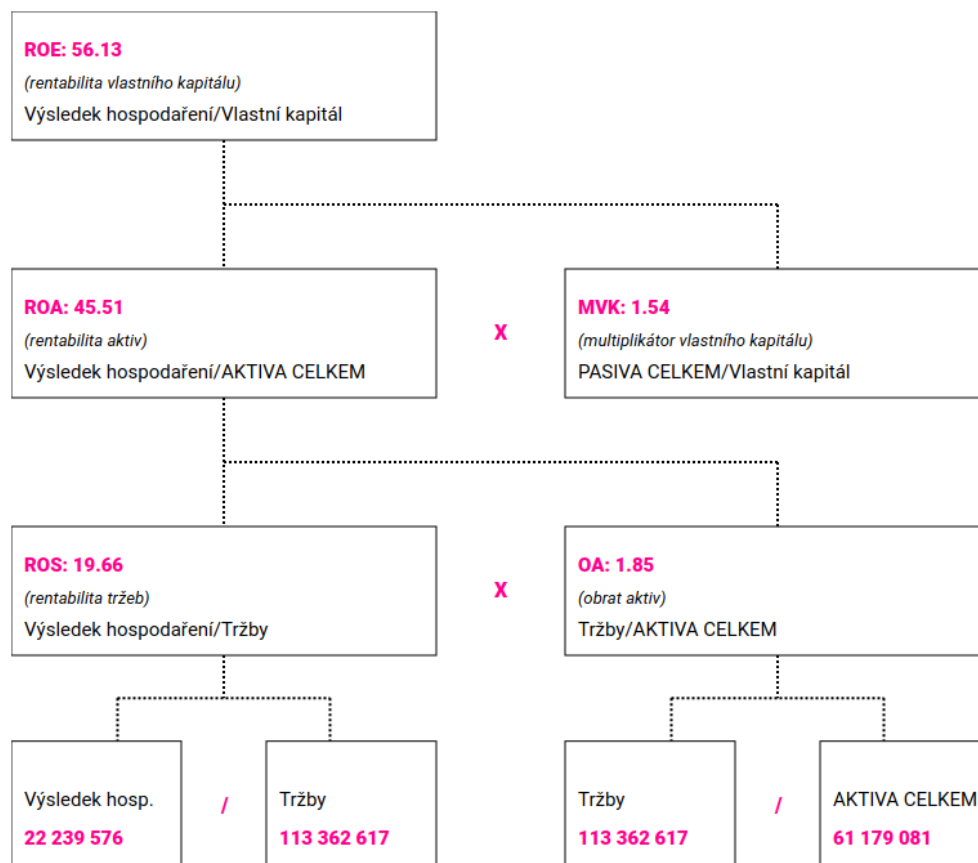
2017  
Údaje pro vybraný rok



Položka	%
Spotřeba materiálu a energie	48.7
Mzdové náklady	13.7
Služby	12.4
Náklady vynaložené na prodané zboží	12.4
Náklady na soc. zabezpečení, zdrav. pojištění a ostatní	5.6

Obrázek 4.7: Obrazovka vertikální analýzy

Na obrazovce s analýzou Du Pont jsou data reprezentována jako vizuální rozklad. Výstup analýzy by mohl být reprezentován i tabulkou či rovnicí, zvolený způsob je ale dle mého názoru mnohem názornější a je také použit v mnoha studovaných zdrojích. Vizualizací stromu rozkladu jsou obdélníky s prvky rozkladu, terminálními i neterminálními, propojené čarami. Vizualizace je realizována pomocí jazyka SVG, který je ve většině webových prohlížečů podporován. Proporce vizuálních prvků jsou uloženy v proměnných, ty jsou na sobě relativně závislé. To umožňuje jednoduchou škálovatelnost výsledného zobrazení.



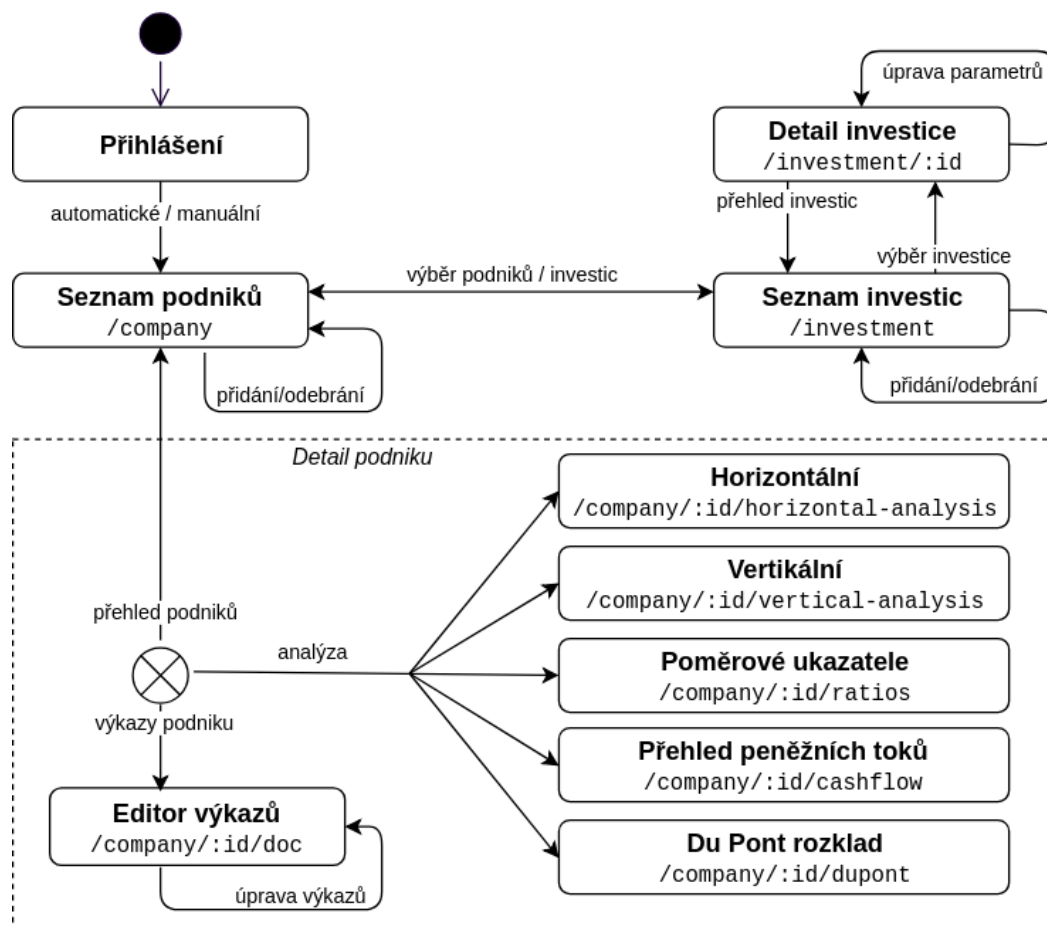
Obrázek 4.8: Grafická reprezentace Du Pont rozkladu

### 4.3 Testování

Kód nástroje je plně testovatelný, testování je prováděno knihovnou Jest. Pro demonstraci byla implementována sada unit testů.

Unit testy pokrývají testování správnosti výsledků většiny analýz. Jako vstupní data jsou použita data konkrétního podniku, pro který byly vyčísleny hodnoty, kterých by měly jednotlivé typy analýzy dosahovat. Po provedení analýzy je ověřována jejich shodnost s těmi vypočítanými.

Jelikož se jedná o nástroj, který je schopen více než jen dávkového zpracování dat, bylo by v rámci rozšíření vhodné implementovat také testy integrační. S kombinací knihoven Jest a React lze integrační testy psát velmi pohodlně. Testy by simulovaly různé scénáře interakce uživatele s nástrojem.



Obrázek 4.9: Diagram návaznosti obrazovek aplikace

Scénáře testů plynou z use-case diagramu:

- testy analýzy
  - zobrazení výsledků analýzy validních vstupních dat
  - zobrazení chybového hlášení při analýze nevalidních vstupních dat
  - signalizace nekonzistence dat načtených ze souboru
- testy správy uživatelů
  - test registrace uživatele
  - test autentizace uživatele
- testy nástroje
  - test persistence změn při modifikaci dat
  - test synchronizace mezi aplikací a databází při modifikaci dat
  - automatická aktualizace výsledků analýzy při modifikaci dat

## 4.4 Vývoj

Pro ověření správnosti navrženého řešení byl v každé z oblastí finanční analýzy nejprve implementován prototyp se zúženým výběrem analyzovaných položek a omezenou funkcionalitou. Úplné doplnění položek a funkcionality proběhlo až v dalších fázích implementace, stejně tak jako opakovaná refaktorizace. Tento postup byl zvolen i při implementaci persistence dat.

Ve zdrojovém kódu je možnost volby mezi čerpáním uživatelských dat z databáze, nebo jejich podvržením pomocí definice jednotného rozhraní těchto dat. Pokud zrovna neimplementujeme modul zajišťující persistenci dat, využití podvržených dat je velmi výhodné z důvodu urychlení vývojářského cyklu a možnosti vyvíjet bez připojení k internetu.

Jako vývojové prostředí bylo použito IDE WebStorm od společnosti JetBrains, jehož licence je poskytována studentům zdarma. Kromě mnoha dalších, nástroj má podporu pro všechny technologie využité při implementaci a jejím usnadnění, zejména jazyky JavaScript, TypeScript a JSX. Celý projekt byl průběžně verzován pomocí nástroje Git, jež má ve zmíněném IDE grafické rozhraní.

- `npm run install` – instalace závislostí (jsou uvedeny v souboru *package.json*)
- `npm run start` – spuštění aplikace pro vývoj
- `npm run test` – spuštění testů
- `npm run build` – zkompilování kódu do balíčku pro nasazení
- `firebase deploy` – nasazení aplikace do provozu na veřejné adrese

Při spuštění aplikace pro vývoj zůstává na pozadí běžet kontrola na změny souborů se zdrojovými kódy, tzv. watcher. Při změně je TypeScript kód automaticky překompilován na JavaScript kód, poté je kód dále transformován, například pro funkčnost v co nejvíce prohlížečích či jejich verzích. Po doběhnutí dojde ke znovunačtení aplikace v prohlížeči – aktualizace stránky. Díky knihovně React Router, která pro navigaci využívá URL adresu, se při aktualizaci stránky stav navigace uchová a aplikace se spustí na původní obrazovce, pouze si znovu načte vykreslovaná data. Watcher funguje i při testování, stačí tedy uložit zdrojové kódy a příslušné testy se automaticky spustí znovu.

## 4.5 Možnosti rozšíření nástroje

V předchozím textu byla popsána některá možná rozšíření nástroje. Zde se je pokusím shrnout. Kromě implementace rozšíření z vlastní iniciativy by bylo vhodné v budoucnu implementovat také podporu nových specifikací stanovených legislativou.

### Uložení sémantiky výpočtu

Vzhledem k tomu, že se způsoby výpočtu poměrových ukazatelů v různých literaturách liší, bylo by vhodné uživateli zobrazit způsob výpočtu, který je použit v tomto nástroji. Při současné implementaci se jeví jako nejjednodušší řešení ke každé metodě provádějící výpočet přiřadit také popis výpočtu, například ve formě řetězce. Popis výpočtu a výpočet samotný by však nevycházely z jednotného zdroje informace a nebyly by sémanticky propojeny. Vývojář by tak musel při každé změně algoritmu pro výpočet adekvátně měnit popis.



Vzhledem k tomu, že se jedná o víceméně jednoduché matematické vzorce, nabízí se mnohem lepší řešení – uložit vzorec pro výpočet ukazatele do odpovídajícího abstraktního syntaktického stromu (AST). Tato struktura by umožnila obojí – vypočítat výsledek i vygenerovat popis výpočtu a jeho složek – na základě jednotného zdroje. V konečném důsledku by tato struktura umožnila rozklad výpočtu až na konkrétní položky výkazů, přičemž by toto automaticky umožnila nad všemi pojmy v nástroji. Pomocí vhodného rozhraní by také mohl uživatel editovat vzorce existujících ukazatelů podle svých potřeb či vytvářet úplně nové.

Rozšíření by učinilo nástroj v tomto směru unikátním.

### **Zkvalitnění provádění analýzy**

- analýza různých časových období (momentálně jen meziročně)
- výpočet dalších ukazatelů
  - minimálně těch, jejichž průměrné hodnoty jsou uváděny ve statistikách
  - ukazatele na bázi cashflow
- signalizace evidentních rizikových faktorů (např hospodaření se ztrátou)
- implementace dalších typů analýzy, podniku i investice
- zavedení uživatelských nastavení
- podpora změn v legislativě v budoucnu

Z oblasti finanční analýzy se dále nabízí rozšíření importu o zkrácené výkazy, účelového členění výkazu zisku a ztrát nebo možnost čerpat zdrojová data přímo z účetnictví (nikoliv z účetní závěrky), pokud existuje nějaký standardizovaný formát těchto dat. To by umožnilo ještě detailnější analýzu. Nebo naopak, umožnit uživateli vyplnit pouze data nezbytná k výpočtu ukazatelů a nenutit jej tak nahrávat celé výkazy, které ani nemusí mít k dispozici. Samozřejmostí je také důkladnější ošetření uživatelských vstupů.

Co do výkonu nástroje, bylo by vhodné ukládat výsledky analýzy do databáze, aby nemusely být znovu počítány při každém spuštění aplikace. Výkon by vylepšila také lepší správa dat na aplikační úrovni – méně časté dotazy na data a menší objem přenesených dat.

## Kapitola 5

### Závěr

V rámci této diplomové práce se podařilo implementovat prakticky využitelný nástroj umožňující provádět vybrané úkony z oblasti finanční analýzy. Implementaci předcházelo studium problematiky finanční analýzy i průzkum současných softwarových možností jejího provádění. Na základě zjištěných poznatků byl vypracován návrh vlastní programové koncepce, který byl poté realizován. Nástroj se zaměřuje na analýzu dvou entit, podniku a investice, přičemž se snaží k vytvoření závěrů o situaci analyzovaných entit poskytnout co nejúžitečnější informace.

Nad podnikem umožňuje provádět analýzu horizontální a vertikální, analýzu poměrových ukazatelů, analýzu peněžních toků a rozklad Du Pont. Pro vybrané analýzy umožňuje také mezipodnikové srovnání. Ve stěžejní oblasti, za kterou považuji analýzu poměrových ukazatelů, byla realizována možnost porovnání výsledků s celorepublikovými průměry v odvětví. Pro investici umožňuje nástroj výpočet její čisté současné hodnoty a vnitřního výnosového procenta.

Cílovou skupinu nástroje tvoří podnikatelé a finanční analytici, jimž se snaží práci maximálně usnadnit. Prostředím běhu je webový prohlížeč, což umožňuje nástroj používat odkudkoliv, bez nutnosti jeho instalace. Dodržování požadavků plynoucích z legislativy ČR zajišťuje jednoznačnost při zadávání vstupních dat i možnost jejich automatizovaného načítání. Díky podpoře uživatelských účtů je možné nahraná data s uživatelským účtem spojit, uložit a k analýze se opakovaně vracet.

Řešení je kvalitní nejen z pohledu provádění finanční analýzy a uživatelské přívětivosti, ale i z pohledu vývojáře. Díky implementaci testů, použitým technologiím a způsobu implementace je zdrojový kód především robustní ke změnám a tím pádem dlouhodobě udržitelný. To je pro budoucí vývoj, zejména v oblasti finanční analýzy, obrovskou výhodou.

# Literatura

- [1] Kislingerová, E.; Hnilica, J.: *Finanční analýza krok za krokem*. C. H. Beck, s.r.o., 2005, ISBN 80-7179-321-3.
- [2] Knápková, A.; Pavelková, D.; Šteker, K.: *Finanční analýza – Komplexní průvodce s příklady – 2., rozšířené vydání*. Grada publishing, a.s., 2013, ISBN 978-80-247-4456-8.
- [3] Kovanicová, D.; Kovanic, P.: *Poklady skryté v účetnictví*. Polygon, 1995, ISBN 80-901778-4-0.
- [4] Nývltová, R.; Marinič, P.: *Finanční řízení podniku: Moderní metody a trendy*. Grada Publishing, a.s., 2010, ISBN 978-80-247-3158-2.
- [5] Resig, J.: *JavaScript in Chrome*. [Online; navštíveno 17.01.2018].  
URL <https://johnresig.com/blog/javascript-in-chrome/>
- [6] Růčková, P.: *Finanční analýza – 5. aktualizované vydání*. Grada publishing, a.s., 2015, ISBN 978-80-247-5534-2.
- [7] Scholleová, H.: *Ekonomické a finanční řízení pro neekonomy*. Grada Publishing, a.s., 2008, ISBN 978-80-247-2424-9.
- [8] Sedláček, J.: *Finanční analýza podniku*. Computer Press, a.s., 2009, ISBN 978-80-251-1830-6.
- [9] Suchánek, P.: *Podnikohospodářská analýza*. Masarykova univerzita, 2006, ISBN 80-210-3985-X.
- [10] Westfall, B.: *Leveling Up with React: Redux*. [Online; navštíveno 17.01.2018].  
URL <https://css-tricks.com/learning-react-redux>
- [11] Zikmund, M.: *Hodnocení investic: Vnitřní výnosové procento (IRR)*. [Online; navštíveno 17.01.2018].  
URL <http://www.businessvize.cz/rizeni-a-optimalizace/hodnoceni-investic-vnitрни-vynosove-procento-irr>
- [12] Zikmund, M.: *Ukazatelé likvidity*. [Online; navštíveno 17.01.2018].  
URL <http://www.businessvize.cz/financni-analyza/ukazatele-likvidity>
- [13] Živělová, I.; Svobodová, E.: *Podnikové finance - Finanční analýza*. [Online; navštíveno 17.01.2018].  
URL <http://is.mendelu.cz/eknihovna/opory/index.pl?opora=5324>

## Příloha A

### Obsah CD

- složka `tool` - zdrojové kódy implementovaného nástroje
- složka `text` - tento text a zdrojové kódy pro jeho vygenerování