

Sem vložte zadání Vaší práce.



**FAKULTA
INFORMAČNÍCH
TECHNOLGIÍ
ČVUT V PRAZE**

Diplomová práce

Trading bot pro kryptoměnu Bitcoin

Bc. Patrik Kubec

Katedra softwarového inženýrství

Vedoucí práce: Ing. Jiří Hunka

25. června 2023

Prohlášení

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

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 46 odst. 6 tohoto zákona tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou, a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla, a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené. Každá osoba, která využije výše uvedenou licenci, se však zavazuje udělit ke každému dílu, které vznikne (byť jen zčásti) na základě Díla, úpravou Díla, spojením Díla s jiným dílem, zařazením Díla do díla souborného či zpracováním Díla (včetně překladu), licenci alespoň ve výše uvedeném rozsahu a zároveň zpřístupnit zdrojový kód takového díla alespoň srovnatelným způsobem a ve srovnatelném rozsahu, jako je zpřístupněn zdrojový kód Díla.

V Praze dne 25. června 2023

.....

České vysoké učení technické v Praze

Fakulta informačních technologií

© 2023 Patrik Kubec. Všechna práva vyhrazena.

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

Odkaz na tuto práci

Kubec, Patrik. *Trading bot pro kryptoměnu Bitcoin*. Diplomová práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2023.

Abstrakt

Diplomová práce se zaměřuje na analýzu kryptoměnového prostředí a vytvořením prototypu traidovacího bota pro kryptoměnu Bitcoin. Tato práce se dále zabývá analýzou technologie Blockchain a její využití v prostředí kryptoměn. Z hlediska finančního i investičního bude kryptobot sloužit mimo jiné i jako indikátor schopnosti predikovat vývoj ceny na burze v prostředí kryptoměn a zejména Bitcoinu. Diplomová práce a technologie kryptobota má velký potenciál v oblasti leverage tradingu ve spojení s TA (technical analysis), proto se práce zabývá i analýzou možných příležitostí a s tím souvisejících rizik, které v této oblasti vznikají.

Klíčová slova Bitcoin, blockchain, burza, kryptoměna, prototyp aplikace, technická analýza, traidovací bot

Abstract

This master thesis focuses on analysis cryptocurrency environment and creation of trading bot prototype for Bitcoin cryptocurrency. This thesis further deals with analysis the blockchain technology and it's use in cryptocurrency environment. Besides the financial and investment aspect the cryptobot will work as a indication of prediction capability of stock market price development in cryptocurrency environment especially Bitcoin. This master thesis and cryptobot technology has big potential in the field of leverage trading in connection with TA (technical analysis) and that is why this thesis also deals with analysis possible opportunities and related risks which appears in this cryptocurrency area.

Keywords Bitcoin, blockchain, stock exchange, cryptocurrency, application prototype, technical analysis, trading bot

Obsah

Úvod	1
I Teoretická část	3
1 Cíl práce	5
2 Rešerše	7
2.1 Úvod do kryptoměn	7
2.1.1 Co to je kryptoměna	7
2.1.2 Peníze obecně	8
2.1.3 Základní kritéria hodnoty peněz	8
2.2 Bitcoin	9
2.2.1 Co to je Bitcoin	9
2.2.2 Výhody Bitcoinu a kryptoměn	9
2.2.3 Bitcoin a blockchain	10
2.2.3.1 Jak to přesně funguje	10
2.2.3.2 Mechanismus Proof of Work	10
2.2.3.3 Blockchain	11
2.2.3.4 Souhrn výhod Blockchainu	12
2.2.3.5 Lightning network	14
2.2.3.6 Lightning network uvnitř	15
2.2.3.7 Lightning network shrnutí	17
2.3 Obchodování s Bitcoinem	17
2.3.1 Volatilita	17
2.3.2 Klasický trh vs kryptotrž	18
2.3.3 Bitcoin a burza	18
2.3.4 Burza a směnárna	18
2.3.5 Crypto derivatives	18

2.3.6	Leverage tradig	19
2.3.7	Finanční trh (market)	19
2.4	Kryptobot	20
2.5	Technická analýza a její metody	20
2.5.1	Základní trend line	21
2.5.2	Support/rezistence	22
2.5.2.1	Trendové zóny	23
2.5.2.2	Horizontální zóny	23
2.5.2.3	Klouzavé zóny	23
2.5.2.4	Fibonacciho zóny	23
2.5.2.5	Využití odporových zón v tradingu	24
2.5.3	Breakouty	24
2.5.4	Cenové patterny	25
2.5.4.1	Trojúhelníky	25
2.5.4.2	Vlajky	26
2.5.4.3	Mnohonásobné vrcholy	26
2.5.4.4	Head and Shoulders	27
2.5.4.5	Cup and handle	28
2.5.4.6	Elliott waves	28
2.5.5	Patterny svíček	28
2.5.6	Indikátory - nástroje pro TA	30
2.5.7	Divergence	30
2.5.8	Objem obchodů	31
2.5.8.1	Objemový profil	31
2.5.9	Pivotové hladiny	31
2.5.10	Manipulace trhu	32
2.5.11	Timeframey	32
II Praktická část		33
3	Analýza	35
3.1	Obecné požadavky	35
3.2	Průzkum a popis možných technologií	36
3.2.1	Amazon Kinesis	36
3.2.2	Apache Spark	36
3.2.3	Apache Kafka	38
3.2.4	Apache Cassandra	38
3.2.5	Krátké shrnutí	39
4	Návrh a implementace	41
4.1	Infrastruktura a architektura datového toku	41
4.1.1	Websocket	41
4.1.2	Apache Kafka	42

4.1.3	Docker	43
4.1.4	Apache Spark	44
4.1.5	Cassandra	46
4.2	Monitoring a logování	47
4.2.1	Fluentd	48
4.2.2	Prometheus	49
4.2.3	Grafana	51
5	Testování a měření	53
5.1	Měření	55
	Závěr	57
	Bibliografie	59
A	Seznam použitých zkratk	65
B	Uživatelská příručka	67
B.1	Spuštění aplikace	67
C	Obsah Git repozitáře	69

Seznam obrázků

2.1	hash function	11
2.2	Blockchain	12
2.3	Lightning network (layers)	14
2.4	Lightning network (channel)	15
2.5	Alice Bob and coffee example	16
2.6	Lightning network global view	16
4.1	Spark sliding window	46
4.2	Prometheus memory available	50
4.3	Grafana metrics dashboard	51
5.1	Scala tests complete	53
5.2	Tabulka s provedenými transakcemi	56

Úvod

Svět kryptoměn zažívá v posledních letech obrovský rozmach. To se děje hlavně díky Bitcoinu, první, nejznámější a nejhodnotnější kryptoměně na světě. Cena Bitcoinu ode dne vstupu na burzu (2010), neustále stoupá. Tehdy měl Bitcoin hodnotu okolo 0,0008 \$ a k dnešnímu datu cena za jeden Bitcoin razantně vzrostla. Pohybuje se okolo 50 000 \$ za „minci“. V průběhu let se začala technologie Bitcoinu a technologie blockchainu, na níž je Bitcoin postavený dostávat více a více do povědomí. V reakci na úspěšné využití blockchainu v oblasti kryptoměn začaly vznikat další a další kryptoměny. Druhou nejznámější kryptoměnou současnosti je Ethereum, která se mnohdy nazývá jako druhý Bitcoin. Počet kryptoměn se dnes pohybuje řádově v tisících. Tyto coiny se nazývají takzvaně altcoiny, protože vznikly až po Bitcoinu.

Takovýto nárůst ceny za Bitcoin láká nejenom velké firmy, ale i retailové investory, kteří pod vidinou zisku investují nemalé částky. Velmi populární je ve sféře kryptoměn leverage trading, což je způsob obchodování s určitým násobkem svého kapitálu. Tento způsob však nese určité riziko toho, že o svůj kapitál můžete přijít, pokud nejste dostatečně zkušený. Obchodovat například s desetinásobkem svého vstupního kapitálu otevírá zkušenějším cestu a obrovské příležitosti k většímu zisku. Velké výkyvy (volatilita) ceny nejsou v krypto světě nic neobvyklého a nachází se zde prostor pro analýzu trhu a s ním spojeného sentimentu.

Právě již zmíněnou analýzou se zabývá kryptobot, který bude uskutečňovat mikrotransakce na burze. Tyto transakce budou podpořeny navíc statistickou i technickou analýzou, díky které bude možné vytvářet mnohem přesnější obchody a kryptobotovi přidá schopnost predikovat vývoj ceny v blízké budoucnosti.

Část I

Teoretická část

Cíl práce

Tato diplomová práce si ukládá za cíl navrhnout a implementovat funkční prototyp traidovacího kryptobota pro kryptoměnu Bitcoin, který bude schopný na základě statistické a technické analýzy predikovat blízký vývoj ceny na burze. Díky této analýze bude kryptobot schopný provádět mikrotransakce, jejichž výsledek bude zaznamenán pro následné vyhodnocení.

V teoretické části této práce budou shrnuty jednotlivé požadavky na kryptobota a možnosti jejich řešení. Dále se tato část práce bude zaměřovat na analýzu krypto trhu a pohyb ceny Bitcoinu na burze v čase. Kryptobot bude tyto informace shromažďovat z předních burz a následně je analyzovat. V této části práce budou shrnuty a probrány techniky takovéto analýzy. Budou probrány jednotlivé možnosti, výhody a nevýhody různých postupů a technických řešení. Jedná se také o známé strategie, které se používají při obchodování na burze. Tyto strategie budou také součástí teoretické části práce. Nedílnou součástí kapitoly je také posouzení možných rizik a příležitostí, které v tomto segmentu vznikají.

Praktická část bude zaměřena na návrh a implementaci celého prototypu Bitcoinového kryptobota. Bude se jednat zejména o zvolení vhodných technologií a frameworků pro analýzu a následné vyhodnocení dat získaných z burz. Avšak pro praktickou část je také klíčová implementace technologií samotného kryptobota a jeho transakční část. Součástí této kapitoly je i otestování hlavních částí aplikace a vyhodnocení získaných výsledků. Na základě těchto výsledků bude dále možné určit rozsah možného rozšíření, či vylepšení práce.

Rešerše

2.1 Úvod do kryptoměn

2.1.1 Co to je kryptoměna

Kryptoměna je novodobá forma digitálních peněz. Jedná se ale o jinou formu digitálních peněz, než jak je známá z bankovních účtů běžných občanů. V této době je už možné platit v některých obchodech pomocí kryptoměn. Nejčastěji se jedná o internetové obchody, nebo velké a známé firmy, jako je například Tesla, která je známá tím, že se nebojí nových věcí a technologií. Zatím se dá vypořádat, že velké firmy se spíše připravují na nástup kryptoměn jako platební metodu a proto najímají experty v této oblasti. Jedná se o firmy jako Amazon, Microsoft, nebo například Apple. Obecně se ale nedá říci, že by kryptoměny byly hojně rozšířené jako platební metoda. Avšak postupem času se k tomuto trendu přidává více a více firem, nebo dokonce států.

Prvním státem, který ustanovil Bitcoin jako národní měnu je El Salvador, což je stát ve Střední Americe. Stalo se tomu poměrně nedávno a to začátkem září 2021. To je poměrně velký krok směrem vpřed pro Bitcoin jako takový. Ať už to v El Salvadoru dopadne jakkoliv, tak Bitcoin z toho vyjde jako vítěz. V tom dobrém případě se Bitcoin osvědčí jako skvělá platební metoda, anebo se nepodaří Bitcoin plnohodnotně nasadit. Bude poměrně zajímavé sledovat dění v El Salvadoru, hlavně co se týče jeho ekonomické situace. Druhá varianta, ale nemusí nutně znamenat pro Bitcoin nic špatného. Pokud i přes to, Bitcoin neztratí svou hodnotu a lidé v něj budou nadále věřit, tak to bude skvělý *stress test* a ukázka jeho síly.

To co rozlišuje klasické peníze od kryptoměn je pozadí za tím vším. V případě kryptoměn se jedná o technologie, na kterých jsou kryptoměny postaveny, nebo technologie, které nabízejí. Název kryptoměna může čtenáře svádět k chybné úvaze, že se jedná pouze o digitální měnu určenou pro placení. Za oponou je toho ale mnohem více. Například Ethereum (ETH), stejně jako Bitcoin (BTC), je postaveno na technologii Blockchain, ale samotné Ethereum

je také *open source* platforma, která nabízí kompletní skriptovací podporu, takže je na ní možné vytvářet decentralizované aplikace. Na této platformě se jako měna používá již zmíněný ETH neboli Ether.[1] Na kryptoměny se dá také nahlížet jako na projekty, které se snaží přinést něco nového, nebo něco revolučního. To, že jedna část projektu umožňuje provádět peněžní transakce, může být bráno jako podružné. Proto by si měl každý investor pečlivě rozmyslet zda má projekt z jeho pohledu smysl, anebo se jedná pouze o *hype*, který nebude mít v budoucnu žádné využití a podle toho do projektu vkládat či nevkładat svou důvěru a finance.[2]

2.1.2 Peníze obecně

Po přečtení předchozí kapitoly může čtenář nabýt dojmu, že kryptoměny jsou pouze jakési internetové *tokeny*, nebo trochu přesněji, pár číslíček a písmenek poskládaných za sebe. Na rozdíl od „opravdových“ peněz, to nemůže mít žádnou hodnotu. Proto se tato kapitola zaměří na peníze trochu obecně a shrne základní věci pro to, aby peníze jako takové mohly vůbec fungovat.

2.1.3 Základní kritéria hodnoty peněz

Pro to, aby měly peníze hodnotu je potřeba splnit určité základní kritéria.

- musí je vlastnit dostatečně velký počet lidí
- obchodníci je musí přijmout jako formu placení
- společnost jako celek musí věřit, že mají takovou hodnotu jakou mají a že si tuto hodnotu udrží i v budoucnu

Tyto všechny body musí samozřejmě splňovat jak „normální“ peníze, tak i kryptoměny. V posledním bodě se však nachází malý zádrhel, který většinu lidí odrazuje od investice do kryptoměn.

Pokud se ohlédneme zpátky do historie, tak zjistíme, že dříve byly mince vyrobeny z drahých kovů. Váha této mince určila její hodnotu. Všechny mince samozřejmě vážily stejně, zde pouze předcházím možnému nedorozumění. To znamená, že bylo jednoduché věřit tomu, že mince měla hodnotu. Od tohoto konceptu se později upustilo, ale zachovala tato myšlenka. Veškeré peníze, které byly v oběhu byly takzvaně kryté zlatem. To znamená, že stát, který pustil do oběhu například 100 milionů dolarů, vlastnil zlato ve stejné hodnotě. Občan si tedy mohl zajít na úřad a směnit své peníze za kousek zlata.

Dnes už tomu tak ale není a to kvůli inflaci. Státy stále tisknou další a další nekryté peníze. Čím víc se takovýchto peněz nechá natisknout, tím více se daná měna devaluje a to je samozřejmě velký problém. Nejen pro danou měnu, ale i pro světovou ekonomiku.

Argument, že kryptoměny nejsou ničím podloženy, pouze vírou občanů, se jeví jako irelevantní, protože například ani dolar už dlouhou dobu není ničím

krytý a lidé tedy pouze věří, že má nějakou hodnotu. Toto téma bude více dopodrobna probrané v nadcházejících kapitolách o Bitcoinu.

2.2 Bitcoin

Tato magisterská práce se zabývá kryptoměny, ale zaměřuje se výhradně na Bitcoin. Proto se v další části pojednává hlavně o Bitcoinu.

2.2.1 Co to je Bitcoin

Jak už bylo zmíněno dříve, Bitcoin je první a nejznámější kryptoměna na světě. Datum uvedení této kryptoměny je uváděno jako 3. ledna 2009 a byla uvedena pod licencí MIT. Za vývojem Bitcoinu stojí vývojářská skupina Satoshi Nakamoto, podle které se i jmenuje dílčí jednotka BTC a to satoshi. Vedle satoshi existují i další dílčí jednotky jako jsou milibitcoin, nebo microbitcoin. Jedná se o stejný ekvivalent jako byly české halíře, nebo euro a centy.

Bitcoin je také název pro *open source* platformu. V této internetové P2P síti se již zmíněný Bitcoin (BTC) používá jako měna. Celá síť je zabezpečená a pracuje na principu asymetrické šifry, která zajišťuje jak bezpečnost, ale i důvěrnost, nepopiratelnost a autenticitu zprávy, která se sítí posílá.[3]

Počet Bitcoinů je už dopředu znám a nepřesáhne 21 milionů, to zaručuje, že BTC nepodléhá a nebude podléhat inflaci. Jako pravděpodobnější se jeví spíše to, že bude podléhat malé deflaci. Například při ztrátě BTC. Bitcoin jako takový má tedy potenciál stát se v budoucnu udržitelem hodnoty a proto se o něm často mluví jako o virtuálním zlatu.

2.2.2 Výhody Bitcoinu a kryptoměn

- je plně decentralizovaný
- žádná centrální autorita
- nepodléhá žádné centrální bance
- neexistuje instituce pro regulaci BTC
- nelze s měnou manipulovat
- malé, nebo žádné poplatky
- směnitelný za „tradiční“ měny
- rychlé zpracování transakcí
- transakce jsou do určité míry anonymní
- bezpečný (asymetrická šifra)

2.2.3 Bitcoin a blockchain

Již bylo řečeno, že jedny z výhod Bitcoinu jsou anonymita a bezpečnost a také, že nepodléhá žádné centrální bance. Zde bude vysvětleno proč tomu tak je.

Na rozdíl od klasických bank, Bitcoin funguje na principu blockchainu. Blockchain je plně decentralizovaná a distribuovaná databáze. V případě Bitcoinu se může říci, že slouží jako zabezpečená účetní kniha (dále jako tabulka), kterou mezi sebou sdílí všechny uzly v blockchainové síti. V této tabulce jsou zapsány transakce, provedené uživateli sítě.[4]

2.2.3.1 Jak to přesně funguje

Řekněme, že máme účastníky transakce Alici, Boba a Charlieho. Alice pošle Bobovi 100 dolarů a zapíše si to do své tabulky. Tuto informaci dále *broadcastuje* do sítě a ostatní uzly v síti si upraví svou tabulku a přidají záznam o tom, že Alice poslala Bobovi 100 dolarů. Nastává tu však správná otázka a to, jak může Charlie a ostatní uzly v síti věřit tomu, že dostaly správnou zprávu o změně v tabulce a také, že ostatní uzly dostaly stejnou zprávu jako Charlie. Toto je základní problém, který byl zmíněn v originálním Bitcoin *white paperu*. Musel se vymyslet protokol, který

- posílá transakce broadcastem
- přijímá pouze podepsané transakce
- řeší doublespending
- řeší výše zmíněný problém popsany ve white paperu

doublespending: Problém, kdy dochází k „utracení“ nebo připsání Bitcoinu na více místech (tabulkách) naráz

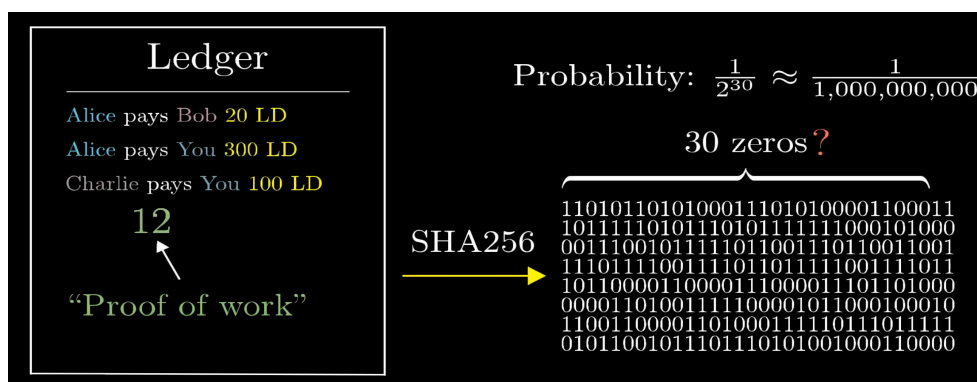
Řešení přišlo v podobě decentralizovaného mechanismu *proof of work*.

2.2.3.2 Mechanismus Proof of Work

Proof of Work je systém, ve kterém se využívá obrovského množství vynaložené práce jako prostředek k dokazování správnosti. Konkrétně u Bitcoinu se jako *proof of work* využívá hashovací funkce SHA256.

Pokud se použije tato hashovací funkce na nějaký string, výsledkem bude náhodná sekvence jedniček a nul. Pokud se tento string změní byť na jedné pozici, výsledkem bude naprosto jiná a náhodná sekvence jedniček a nul. Obrácený proces je velice výpočetně náročný. Kdyby chtěl útočník znát string, který byl na inputu, nezbylo by mu nic jiného než jen hádat a kontrolovat výsledek. Pokud by byla přidána nonce, která říká, že na začátku musí být 60 nul, tak pravděpodobnost, že nalezneme takové číslo je $1/2^{60}$. Není snad třeba dodávat, že taková pravděpodobnost je extrémně malá a že je skoro nemožné uhádnout

původní string. Čtenáře by mohlo napadnout, že by se dalo pomocí reverzního *engineeringu* pokusit nahlédnout do samotné hashovací funkce a odvodit tak původní input.[5] Nikdo však doposud nenašel způsob jak toto udělat a proto se hashovací funkce SHA256 považuje za bezpečnou. Na druhou stranu nikdo ani formálně nedokázal, že je složité zpětnou operaci udělat a přesto se kryptografické šifry používají napříč velkým množstvím zabezpečovacích systémů.



Obrázek 2.1: hash function

Proof of Work se tedy používá v síti Bitcoin jako způsob legitimizace transakcí v procesu zvaném *mining* neboli těžba. Těžení Bitcoinu je pouze termín, který se používá v tomto kontextu, ale nemá s klasickým těžením nic společného.[6]

Zjednodušeně řečeno je těžba Bitcoinu řešení složité matematické úlohy. V tomto bodě by již čtenáři mělo být jasné o jaký typ úlohy se jedná. Jde o legitimizaci transakcí, které probíhají v Bitcoinové síti.

2.2.3.3 Blockchain

Blockchain je bezesporu velmi aktuální téma zejména díky kryptoměnám. V předešlých kapitolách bylo vysvětleno, co je to hashovací funkce, mechanismus proof of work a jak funguje posílání transakcí mezi uzly (uživateli) Bitcoinové sítě.

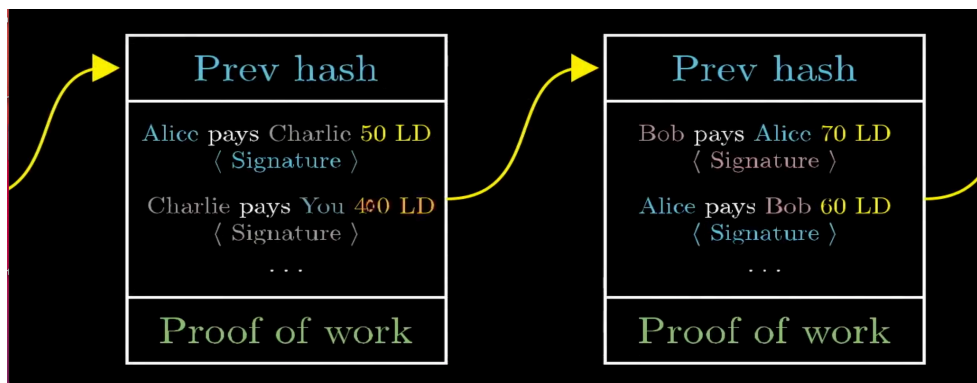
To vše je potřeba k tomu, aby mohla vzniknout technologie Blockchainu. Těžaři, nebo také uzly, které zaúčtovávají nové transakce, řeší onu matematickou úlohu. Provedené transakce se seskupí do bloku (tabulky), na jejímž základě vznikne hash. Zde začíná těžení bloku a hledání řešení matematické úlohy. Uzel, který jako první nalezne správné řešení je odměněn v podobě několika Bitcoinů. Nalezení řešení je velice výpočetně obtížné, avšak potvrzení správnosti výsledku a tím pádem potvrzení správnosti bloku, je velice snadné

a rychlé. V Bitcoinové síti obvykle stačí 6 potvrzení od různých uzlů, aby se transakce brala jako potvrzená a validní.[7]

Odměna za vyřešení bloků byla do dne 21. 11. 2012 50 Bitcoinů. Toto odměňování je způsob, jak vznikají nové Bitcoiny. Odměna se každé 4 roky půlí na polovinu. Tomu se říká *halving*. Halving nastává vždy po 4 letech, protože to je doba, za kterou se vyřeší 210 000 bloků. Takto je nastavená politika Bitcoinu. Dnes je odměna za vyřešení bloku 6,25 BTC. Díky tomu se dá snadno spočítat, že Bitcoin bude vytěžen v roce 2140. Většina BTC však již v roce 2030.

Do matematické úlohy hledání řešení bloku se přidává tzv. kryptografická nonce. Nonce je umělý matematický problém, který se přidává za účelem stabilizace délky těžení. Těžení je nastaveno tak, aby se v průměru vytěžil jeden blok (6,25 BTC) za 10 minut. S tím, jak se přidává do sítě další a další výpočetní výkon, nebo z ní naopak výpočetní výkon odchází (například restrikce v Číně), se musí uměle upravovat složitost problému, aby byl splněn průměr těžení 1 blok za 10 minut, tento čas mezi vytvořením bloku dalšího je znám jako *block time*.

Název Blockchain je odvozen od fungování takovéto sítě a je tedy velice přesný. Každý vytěžený blok má totiž svou hash, kterou získal pomocí hash funkce SHA-256. Poté v celé fázi potvrzování transakcí figuruje Proof of Work a může se zahájit těžení dalšího bloku. Ten už se však skládá nejen z transakcí, ale i z hashe bloku předchozího. Takovýmto se způsobem se bloky za sebou řetězí a vzniká pomyslný řetěz bloků. Každý blok tedy závisí na blocích předchozích. Odtud se vzdal název Blockchain.



Obrázek 2.2: Blockchain

2.2.3.4 Souhrn výhod Blockchainu

- Výhody

Immutability: Transakce uložené v bloku již nemůžou být změněny, nebo smazány a všechny mají záznam o datu a času uskutečnění (*timestamp*).

Visibility and traceability: Každý záznam je zpětně dohledatelný, i když je například schovaný za 10 předchozími bloky.

Speed: Od klasických bankovních transakcí jsou blockchainové transakce mnohonásobně rychlejší. Způsobuje to odstranění prostředníka (centrální autority), nebo neexistence žádného manuálního zásahu v celém procesu. Rychlost transakce se uvádí v jednotkách sekund, nebo dokonce méně. Vše závisí na vytíženosti sítě a na velikosti daného bloku.

Reduced costs: Blockchain a v případě Bitcoinu *mining* založený na Blockchainu, napomáhá ke snižování transakčních poplatků. V případě standardních prostředníků jako Visa nebo Mastercard jsou poplatky vysoké, protože si tyto společnosti účtují nemalé procenta z každé transakce. Kryptoměny nemají žádné transakční poplatky, jediný poplatek, který se odvádí jde samotným těžářům jako odměna za těžení a je velice malý. Například v případě *Lightning network* jde řádově o halíře, nebo desítky halířů. I kdyby to byly jednotky korun, stále se jedná o zanedbatelnou částku.[8]

Trust: Blockchain umožňuje komunikovat mezi subjekty, které mezi sebou nemají žádné jiné vztahy.

- Nevýhody

Electricity cost: Pojem Blockchain vyvolává vlny obav o životní prostředí, zejména když se se tento termín spojí s Bitcoinem, nebo jinými kryptoměnami. Kvůli procesu těžení, který už by měl čtenář znát, se zvýšila spotřeba elektrické energie. Roční spotřeba elektrické energie vynaložená k těžbě BTC je okolo 121,36 terawatthodiny (TWh). Pro srovnání, to je 2x více, než vyprodukuje jaderné elektrárny Dukovany a Temelín. V potaz se však musí vzít i fakt, že klasické banky spotřebují mnohem více energie na svůj provoz, než těžba BTC.

Private key - own bank problem: Decentralizace Blockchainu s sebou nese jednu důležitou vlastnost. Centrální autorita neexistuje a proto musí mít každý svůj privátní klíč, který nesmí nikomu sdělit. Problém nastává také při ztrátě tohoto klíče, protože uživatel se nedostane do své Bitcoinové peněženky.

Secure but not really: Bitcoinová síť založená na šifrování RSA a hashi SHA-256 je velice dobře zabezpečená. Může však dojít k takzvanému 51% útoku, kdy útočník ovládá 51% celé sítě a může měnit záznamy v blocích a provádět *double spending*. Navíc změny záznamů

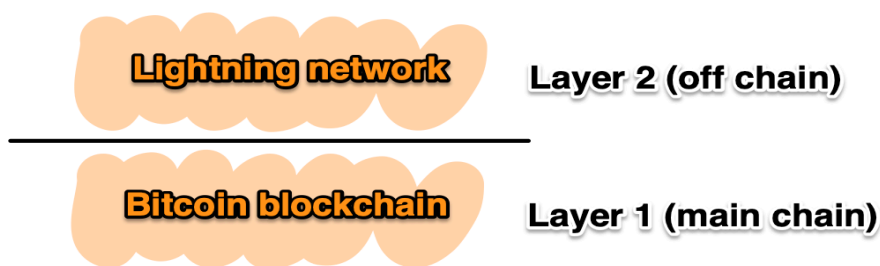
by se týkaly pouze transakcí samotného útočníka, na jiné transakce by to nemělo vliv. Takovýto útok však není proveditelný, protože Bitcoinová síť je tak obrovská, že by to stálo obrovské množství peněz a výpočetní síly, kterou nikdo na světě nemá. Pokud by se něco takového stalo, nebylo by to pro zisk bohatství, ale pouze pro poškození jména Bitcoinu.[9]

Scalability: Scalabilita je něco, s čím měla Bitcoinová síť vždy problémy. Byla a stále je terčem kritiky a taky hlavním argumentem všech odpůrců Bitcoinu. Momentální je horní hranice 7 transakcí za sekundu, což není moc a v porovnání s technologií VISA, která je schopná provádět až 65 000 transakcí za sekundu je už tato propast znatelná. Tento problém se však dá úspěšně řešit pomocí technologie zvané *Lightning network*. [10]

2.2.3.5 Lightning network

Lightning network by určitě vydal na zvláštní sekci, to ale není cílem této magisterské práce. Proto zde bude pouze nastíněno co to *Lightning network* je a jak souvisí s Bitcoinem.

Největší problém Bitcoinu je v současnosti *scalabilita* a tu úspěšně řeší technologie zvaná *Lightning network*. Jedná se o nadstavbu Bitcoinové sítě. Nebo také o tzv. druhou vrstvu. Tato vrstva odděluje odesílání a potvrzování transakcí od zbytku a využívá k tomu *micropayment channels*, díky kterým se vylepšuje škálování celého blockchainu. [11] *Lightning network* umožňuje aby byly transakce rychlejší, poplatky za provedení nižší a byly snáze potvrzovatelné, než na původní Bitcoin network.



Obrázek 2.3: Lightning network (layers)

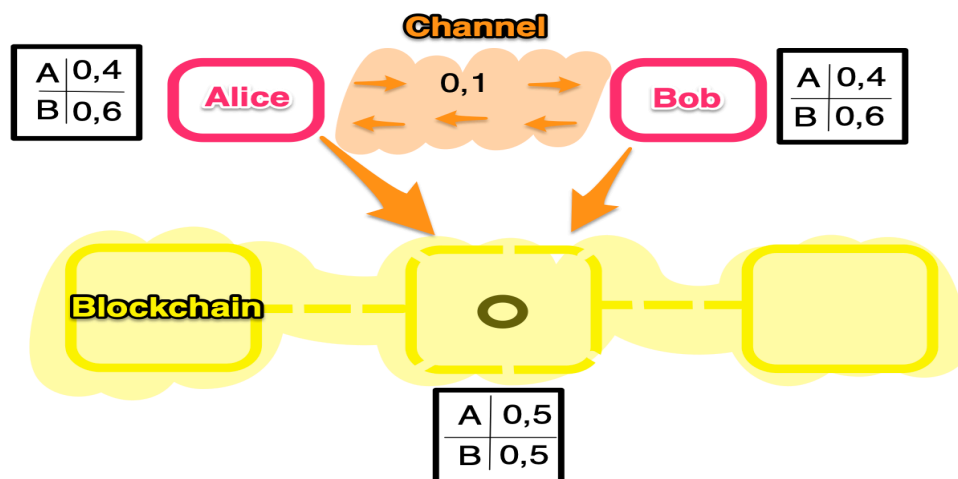
Odtržením transakční části ze zbytku hlavní blockchainové sítě tak vznikl prostor pro odlehčení transakčním poplatkům. *Lightning network* byla také navržena tak, aby nabízela více druhů transakcí. Například přímé transakce

mezi různými druhy kryptoměn bez nutnosti prostředníka. V praxi to například vypadá tak, že se dá směnit Bitcoin za Ethereum, nebo jinou kryptoměnu bez jakéhokoliv zásahu třetí strany, v tomto případě směnárny.

2.2.3.6 Lightning network uvnitř

Jak technologie fungují se nejlépe vysvětluje na příkladech s doprovodnými obrázky. Vezměme si tedy učebnicové příklady ze světa IT, Alici a Boba.

Dříve bylo zmíněno, že Lightning network funguje na principu *channelů*. Tyto channely se vytváří mezi jednotlivými koncovými uživateli sítě. Nesmí se ale opomenout fakt, že Lightning network je nadstavba hlavního Bitcoinového blockchainu. To vše bude zachyceno na příkladu níže.



Obrázek 2.4: Lightning network (channel)

Alice a Bob chtějí mezi sebou provést transakci. Alice chce poslat Bobovi 0,1 BTC. Vše probíhá nad Bitcoinovým hlavním blockchainem. Nejprve musí Alice i Bob vložit do tzv. trezoru pomyslnou jistinu, neboli vstupní *deposit*. Toto není nic víc než záznam v tabulce s hodnotami kdo a kolik BTC vložil. Tento záznam musí být oběma stranami podepsaný jejich privátními klíči. Pokud je tato podmínka splněna, záznam se zapíše do hlavního blockchainu a mezi Alicí a Bobem se otevře transakční kanál ve kterém budou probíhat veškeré transakce mezi těmito dvěma účastníky. Tentokrát už bez účasti hlavního blockchainu. Mezi Alicí a Bobem může probíhat tisíce transakcí a teprve až jedna strana bude chtít kanál ukončit, nebo pokud dojdou jedné straně finance z depositu, tak se konečný záznam propíše do hlavního blockchainu, kde se zkontrolují všechny validace a digitální podpisy. Veškeré transakce tedy probíhají takzvaně *off-chain*. Nutno zmínit, že každá strana si uchovává kopii

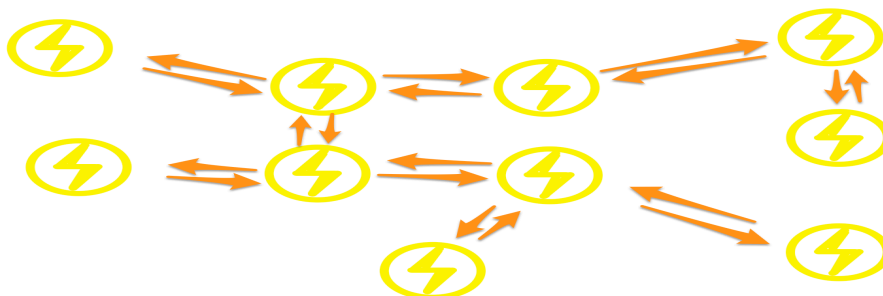
transakční tabulky mezi sebou a druhou stranou a že při každé transakci se podepisují pomocí privátních klíčů. Takovéto transakce prováděné v channelu jsou *instantní* a bez zbytečných poplatků.[12]

Takovýto pohled na Lightning network je však naprosto základní model. Celá síť funguje mnohem více efektivně a komplexně. Již zmíněné transakční channely jdou totiž otevírat mezi více uživateli naráz. Řekněme, že Bob má svou oblíbenou kavárnu a často v ní nakupuje. Má tedy otevřený stálý channel s touto kavárnou. Zároveň má Alice otevřený channel s Bobem a tím pádem může provádět transakce i s Bobovou oblíbenou kavárnou. Vše se uchovává s lokálních transakčních tabulkách do doby, kdy jedna ze stran ukončí channel a propíše tabulky do hlavního blockchainu.



Obrázek 2.5: Alice Bob and coffee example

Tímto způsobem může být otevřeno velké množství channelů napříč mnohými uživateli a transakce mohou být prováděny bez nutnosti otevření přímých nových channelů. Takto vytvořená druhá vrstva nad Bitcoinovým blockchainem se tedy nazývá *Lightning network*.



Obrázek 2.6: Lightning network global view

2.2.3.7 Lightning network shrnutí

- Co si odnést

Scalability solve: Lightning network je technologické řešení pomalého transakčního zpracování Bitcoinového blockchainu.

Decentralized: Stejně jako hlavní Bitcoinovej blockchain je Lightning network zbavena centrální autority.

Not a patch: I přes to, že jsou vynálezci Lightning network Joseph Poon a Thaddeus Dryja, kteří tuto technologii zmínili v roce 2015, tak první zmínky byly už v původním *white paperu* od Satoshi Nakamoto. Tudíž se dá Lightning network brát jako původní záměr a součást celého Bitcoinu a ne jako *patch*.

2.3 Obchodování s Bitcoinem

V předchozí části práce bylo vysvětleno, co je to kryptoměna, co je to Bitcoin, jejich výhody a případné nevýhody oproti klasickým penězům. Jak Bitcoin a přesněji blockchain funguje a na jakém mechanismu je založen. To by mělo dát čtenáři jakousi jistotu v tom, že Bitcoin není pouze *scam* a že do budoucna má Bitcoin smysl.

S tím úzce souvisí i fakt, že díky své rostoucí popularitě a stále se zlepšující platební technologii nachází Bitcoin a ostatní kryptoměny uplatnění v odvětví obchodu a v oblasti zhodnocování peněz. To platí jak pro každodenní obchodování, tak pro dlouhodobé investice.

Vzrůstající cena láká nové investory a svádí k dalším investicím již stávajících investorů. Větší zájem podporuje zvyšování ceny. Tento trend se točí v cyklu, ale samozřejmě nemůže být nekonečný.

2.3.1 Volatilita

Slovem volatilita se označuje jak moc určité aktivum kolísá. Často se uvádí jako směrodatná odchylka této míry kolísání za určitý časový úsek.

Trh s Bitcoinem a kryptoměnama je extrémně volatilní a právě tato vysoká volatilita je další faktor toho, proč je o obchodování s kryptoměnama takový zájem.[13]

Cena Bitcoinu může během dne vyskočit, ale i klesnout o několik jednotek až desítek procent. Právě v tom je ukrytý obrovský investiční potenciál. Vysoká volatilita trhu je však dvousečná zbraň, protože přináší i obrovské riziko. Za krátký časový úsek lze svůj kapitál několikanásobně zvýšit anebo o všechn přijít.

2.3.2 Klasický trh vs kryptotrž

Na rozdíl od trhu s klasickými měnami, který je otevřen pouze v obchodních hodinách je trh s kryptoměnami otevřen neustále. Likvidita je však o poznání menší o víkendu, kdy jsou méně aktivní i obchodníci. Jinak řečeno o víkendu klesá nabídka i poptávka.[14]

2.3.3 Bitcoin a burza

Kryptoměnová burza funguje na stejném principu jako klasické burzy s klasickými měnami. V případě, že je o danou měnu zájem, čili je poptávka, musí být v danou chvíli i nabídka (Někdo ji musí nabízet). Samozřejmě musí platit i princip opačný.

2.3.4 Burza a směnárna

V případě směnárny se věci malinko zjednodušují. Ve směnárně se kupuje za kurz, který si směnárna určí. Směnárna na tom chce něco vydělat, proto bude cena vždy o něco vyšší. Zde se však platí za pohodlí, protože směna je velice rychlá. Příklady takovýchto směnárén jsou Bybit, PheMex, Binance, Coinbase, FTX nebo PrimeXBT.

2.3.5 Crypto derivatives

Co jsou to tzv. *crypto derivatives* je velice důležité vědět, protože tato práce se zabývá vytvořením kryptobota, který bude s těmito *derivatives* obchodovat.

Pojem derivatives může být trochu složitější na pochopení, nejedná se o žádný fyzický derivát Bitcoinu, nebo jiné kryptoměny. Vše je více virtuální. Jedná se o druh kontraktu mezi dvěma nebo více stranami. Hodnota je podložena na nějakém dlouhodobém finančním aktivu. Mohou to být akcie, cenné papíry, dluhopisy, nebo například kryptoměny.

Derivatives jsou synteticky vytvořené produkty, které mají a mění hodnotu na základě své předlohy, Bitcoinu, Etherea a dalších. Derivát není skutečný Bitcoin a nemůže být použit v blockchainu, avšak deriváty mohou být směněny za skutečné peníze, případně za již skutečný Bitcoin. Deriváty ale musí zůstat pod kontrolou společnosti, která je vydala. To taky znamená, že pokud společnost zbankrotuje, nebo se dostane do insolvence, deriváty ztratí svou hodnotu. To je ale riziko, které obchodníci musí podstoupit, pokud chtějí s deriváty obchodovat.[15]

Výhoda derivátů tkví v tom, že jsou dostupnější a lépe získatelné než přímé vlastnictví Bitcoinu. Deriváty se často dají obchodovat jako pár, například BTCUSD, BTC/LTC a podobně.

2.3.6 Leverage tradig

Leverage trading, nebo také obchodování na páku je dnes poměrně populární téma. Jedná se o způsob obchodování s násobkem svého kapitálu. Objem transakcí tak může být mnohonásobně větší. Pokud se použije páka 1:15, může být zisk, ale i ztráta z obchodu patnáctkrát větší. Je to tedy takové obchodování s půjčkou. Pro kryptoměny se většinou používá velmi malá páka a to kvůli vysoké volatilitě trhu. Vysoká páka může při takové volatilitě snadno způsobit vynulování celého účtu, tedy ztrátu veškerého kapitálu.

Výhoda obchodování s deriváty ve spojení s pákou je více než lákavá. Díky tomu, že derivát je druh kontraktu mezi více stranami je možné vydělávat i když cena Bitcoinu klesá. Pokud si vezmeme modelový příklad, tak kupující koupí Bitcoin za určité množství peněz (lze koupit jen část Bitcoinu). Počká až cena Bitcoinu vzroste a poté dokončí transakci a Bitcoin prodá. Zisk je tedy procentuální rozdíl mezi původní hodnotou a novou hodnotou Bitcoinu přepočítaná například na dolary minus daň, která činí v České republice 15%. Daň z kryptoměn se platí až poté, co se peníze převedou například z krypto směnární na bankovní účet majitele. Pokud máte veškerý svůj získaný kapitál na směnárně, nemusíte nic platit. Toto byla ukázka klasické výdělečné transakce, kdy jedna strana benefituje z nárůstu ceny. V případě derivatives kontraktů se dá vydělávat i když cena klesá. Nezáleží totiž na pořadí nákupu a prodeje. Dá se tedy nejprve prodat určité množství kontraktů za nějakou cenu, poté co cena Bitcoinu klesne, lze tyto kontrakty koupit za cenu nižší. Obchodovat se tedy dá oběma směry, ten největší problém samozřejmě tkví v odhadu, jakým směrem se bude cena ubírat v následujících dnech, hodinách, případně minutách.[16]

2.3.7 Finanční trh (market)

K tomu aby bylo možné obchodovat s Bitcoinem, je dobré porozumět alespoň základním pojmům, které ve své podstatě tvoří trh. Od toho se dále odvíjí i trh s kryptoměnami. Poté je možné obchody uskutečňovat a dále je automatizovat.

Finanční trh je místo, nebo můžeme zůstat u označení trh, kde strany obchodují s cennými papíry nebo jakýmkoliv jiným finančním aktivem. Dříve se obchodovalo hlavně osobně, ale dnes se transakce uskutečňují online. Pokud se toto vztáhne na Bitcoin a kryptoměny, tak kryptoměnový trh je místo spojující prodávající a kupující s Bitcoinem.[17]

Existují dvě základní možnosti jak vytvořit objednávku (dále jako order) nehledě na to, zda je člověk v pozici kupujícího, nebo prodávajícího. První možnost je pomocí *Market order*. Market order zajistí to, že strana koupí/-prodá měnu za nejlepší cenu, která je momentálně dostupná. Jedná se o okamžitý order, který se hned vyplní.

Ta druhá možnost je vytvořit objednávku pomocí *Limit orderu*. Limit order je výhodná zejména pro stranu, která zadává tento order. U tohoto

způsobu se platí velmi malé procento *fee*, zatímco u Market orderu je toto procento mnohem vyšší. Limit order funguje v principu tak, že strana která chce provádět transakci tímto způsobem si nastaví cenu, za kterou je ochotná koupit/prodat a čeká do té doby, než se cena aktiva vyhoupne, nebo klesne na požadovanou hodnotu.

2.4 Kryptobot

Kryptobot je aplikace, která dokáže uzavírat obchody na kryptoměnové burze bez přímé asistence člověka. Vše se děje automaticky na základě naprogramovaného algoritmu a know-how. Cíl každého kryptobota je generovat zisk. Pokud je zisk nedostatečný, nebo dokonce minusový (ztráta), tak je dobré uvažovat o vylepšení, nebo změně strategie takového bota. Kryptobot využívá celou paletu funkcí a indikátorů zkombinovanou s vloženým know-how a obchodní strategií.

Mnoho obchodníků se spoléhá na takzvanou technickou analýzu a na nic jiného. To je samozřejmě jedna z možností, jak se dá obchodovat, ale taková strategie by nepřinesla uspokojivé a konstantní výsledky.

Pro správné a hlavně bezpečnější obchodování a nakládání s penězi musí mít v sobě člověk nastavenou hranici, kam až je schopen zajít a jaké riziko už mu nestojí za vidinou zisku. Tomuto chování, nebo vzorci chování se říká *risk management*(RM). V klasickém obchodování, kdy člověk sedí za počítačem, kouká na monitor na křivku ceny Bitcoinu, kouká na volatilitu a na všechny obchody, které se právě uskutečňují je velice obtížné udržet si chladnou hlavu a vynechat ze svých manažerských rozhodnutí veškeré emoce. Člověk má dokonce tendenci porušovat svá předem nastavená pravidla. Právě naprogramováním určité obchodní strategie a automatizováním se obchod odděluje od zbytečných emocí a porušováním vlastního RM. To je nesporná a hlavní výhoda kryptobota opomineme-li například fakt, že kryptobot obchoduje neustále, zatímco člověk nikoliv.

2.5 Technická analýza a její metody

Technická analýza(TA) je doslova způsob pohledu na graf. Graf může znázorňovat jakékoliv hodnoty a dá se nad ním udělat technická analýza. V případě kryptoměn, komodit, kurzů akcií a Bitcoinu se jedná o graf vývoje ceny. Technická analýza se snaží na základě vyhodnocení grafů odhadnout budoucí vývoj ceny. Obchodníci a především retailoví investoři rádi tuto TA používají především díky jejímu snadnému uchopení a dobrým výsledkům. Pojem dobré výsledky je spíše zavádějící, protože u obchodování hraje hodně faktorů. Traider si může udělat správnou TA, ale navzdory tomu dosahovat špatných výsledků. Tato práce kombinuje více inputů pro určení budoucí ceny. Tyto inputy zahrnují TA, indikátory, oscilátory, historii, ML, risk management a další.

Pokud se v práci používají různé metody, je dobré vědět proč se používají a proč fungují. V pozadí TA se skrývá známá hláška „historie se opakuje“. Technická analýza je analýza grafu vývoje ceny.[18] Cenu určují lidé svými činy a svými obchody. Jak už bylo řečeno dříve, lidé se rozhodují na základě emocí a pouze pár profesionálů si umí udržet chladnou hlavu. Lidé obecně tedy reagují na stejné zprávy stejně. Pokud cena vzroste, lidé paradoxně zareagují nakoupením. Pokud cena klesne, nebo klesne razantně hodně, lidé zpanikaří a prodají své akcie/Bitcoinů a zareagují vždy stejně. Vždy se ale najde přibližně stejná skupina lidí, kteří na takovýto propad trhu zareagují nakoupením tzv. „za levno“. Obecně lze konstatovat, že určité chování lidí v tržním systému se dá do jisté míry předpovídat. Nelze to však říci bezpodmínečně, protože pokud by to bylo jednoduché, tak by každý obchodník vykazoval pouze zisk a to se neděje.

Technická analýza se nezabývá vnějšími zprávami, ale pouze samotným grafem. Zkoumá různé patterny, trend liney, velikosti a tvary svíček, zóny se zvýšenou aktivitou atd. Avšak TA může vykazovat mnoho falešných signálů. Zde se skrývá potenciální kámen úrazu při obchodování. Pro zvýšení úspěšnosti obchodů je žádoucí vyfiltrovat tyto falešné signály a spoléhat se na ty pravdivé. Jak to udělat je v rukou samotného traidera, jeho zkušeností a know-how.

Technická analýza jako taková je velice obsáhlé téma a pro účely této diplomové práce naprosto zásadní. V následující sekci bude TA probrána od od nejjednoduššího základu až po komplexnější metody využívající například Fibonacciho posloupnost.

2.5.1 Základní trend line

Základním stavebním kamenem traidingu a TA je rozpoznání trendu a zakreslení trendových čar na svíčkovém grafu. Obchodování na trhu, který má daný trend je mnohem jednodušší, než obchodovat na nestabilním trhu bez trendu. [19]

Existují dva způsoby, jak se může svíčkový graf pohybovat. První způsob pohybu je stagnace ceny na jedné cenové hladině. Graf se pohybuje tzv. do strany. To ovšem neznamená, že by se cena držela přesně na dolar na jedné hodnotě. Cena může v této fázi stoupat i klesat, ale vždy se vrátí zpátky do původní hladiny, tomu se říká výkyvy.

Druhým způsobem pohybu grafu je trendový pohyb. Cena se pohybuje buď nahoru anebo dolů, jedním směrem. Trend se obvykle určuje na svíčkách s větším časovým horizontem, například hodinovém, nebo čtyřhodinovém. V rámci tohoto trendu se však může trh vrátit (pullback) a poté znovu pokračovat dál. Právě tyto pullbacky jsou vyhledávanými vstupními oblastmi pro vstup do pozice. Slovem pozice se v traidingovém světě označuje nákup aktiva.

Pozice může být otevřená, uzavřená, nebo neotevřená. Otevřít pozici znamená nákup, nebo spekulaci, pokud se obchoduje na páku. Uzavřít pozici

znamená prodej, prodej může být ziskový, či ztrátový. Pojmem neotevřená pozice znamená objednávku. Tato pozice se otevře po splnění určitých předem nastavených kritérií.

Způsob jakým se určuje trend je v principu jednoduchý. Na grafu se vytvářejí lokální maxima a lokální minima, neboli nárůst ceny a posléze její propad. Pokud trend roste, tak se v grafu vytvářejí vždy vyšší a vyšší lokální maxima i minima. Pokud trend klesá, děje se opačný jev. Vytvářejí se vždy menší lokální minima i maxima. Ve stagnujícím trhu se nevytváří žádná vyšší maxima či minima, ale jsou možné odchylky.

Trend lze do grafu znázornit přímkou. Spojením po sobě jdoucích maxim se určí horní trendová přímka a spojením po sobě jdoucích minim se získá spodní trendová přímka. Spolu tyto linie tvoří stoupající nebo klesající channel ve kterém se cena pohybuje do chvíle, dokud nedojde k porušení tohoto pomyslného channelu. Může se jednat o odchylku anebo o proražení trendové linie. V tomto okamžiku by měl každý obchodník zpozornět a očekávat změnu trendu, případně stagnaci. Toto platí jak pro porušení trendu směrem nahoru, tak i směrem dolů.

Definice trendových čar je poměrně volná a benevolentní a proto zakreslení těchto čar do grafu se může zdát složité. Existuje přísné pravidlo pro zakreslení těchto čar. Spojují se vždy dva knoty svíce, jejichž minimum je menší, než knoty svíce na levé i pravé straně. Tato definice je velmi striktní ale zároveň nejvíc korektní. V průběhu času, jak trh reaguje se tyto trendové čáry dají překreslit podle aktuálního sentimentu. Trh tvoří lidé a lidé mají na trhu určitou náladu. To se projevuje v grafu a analytik na tuto náladu musí reagovat. V tomto případě překreslením trendové čáry.

2.5.2 Support/rezistence

Supporty a rezistence jsou zóny, ve kterých se cena většinou odrazí, nebo zastaví. Rozdíl mezi těmito dvěma termíny je ve směru pohybu ceny. Support je pod aktuální cenou a rezistence zase nad aktuální cenou. Tyto zóny jsou klíčové pro TA a bez jejich znalostí nelze obchodovat.[20]

Nejedná se o přesné hodnoty ceny, nýbrž o zóny. Příkladem může být zóna mezi cenou 50 500 \$ - 52 000 \$. V minulosti tato zóna mnohokrát působila jako support a rezistence na grafu s jednodenními svíčkami. Psychologie za supporty a rezistencemi je taková, že na supportu je vyšší kupní síla, zatímco na rezistenci zase síla prodejní. Proč se takovéto zóny tvoří je zase dáno psychologií lidí, kteří vytvářejí obchody a celý trh. Jako demonstrativní příklad může být cena 20 000 \$, nebo 30 000 \$. Tyto ceny mají společnou jednu věc, jsou to tzv. kulaté čísla. Lidé mají tendenci nastavovat svoje limit order tam, kde si myslí, že je mohou mít i ostatní lidé a to jsou ve většině případů takovéto „hezké“ levely.

Existuje více druhů supportů a rezistencí.

- Trendové
- Horizontální
- Klouzavé průměry
- Fibonacciho úrovně

2.5.2.1 Trendové zóny

Trendové supporty a rezistence jsou zóny okolo trendových čar, které byli probírány v předešlé kapitole. Horní trendová přímka působí jako rezistence a spodní jako support. Někdy se jim také říká support trend line nebo resistance trend line.

2.5.2.2 Horizontální zóny

Horizontální zóny odporu se hledají na grafu pomocí chování ceny v dané horizontální hladině. Pokud se v minulosti cena zastavila na dané hladině, je pravděpodobné, že se tak stane znovu v budoucnosti. Z pozorování a z analýzy dat v minulosti se dá říci, že čím více knotů svíček spadne do vyznačené oblasti, tím je tato zóna silnější a je vyšší pravděpodobnost, že se cena odrazí a půjde opačným směrem po nějakou dobu. Pokud se support/rezistence prorazí, je pravděpodobné, že se tato zóna otočí (z rezistence se stane support a naopak).[21]

2.5.2.3 Klouzavé zóny

Klouzavé průměry, nebo taky moving averages (MA) jsou průměrné ceny za určitý čas vyobrazené do křivky. Nejčastějším časovým úsekem bývá 50, 100 a 200 uzavřených svíček na grafu. Tyto křivky se následně stávají zónou supportu, nebo rezistence.

2.5.2.4 Fibonacciho zóny

Poslední druh supportů a rezistencí jsou Fibonacciho úrovně. Jsou to také horizontální zóny odporu, ale mnohem přesnější. Jedná se zejména o úrovně 0.618, 0.5, 0.382 a 0.236. Tyto čísla dostaneme, pokud podělíme dvě po sobě jdoucí čísla Fibonacciho posloupnosti. V zásadě je jedno, jaké čísla to jsou, vždy vyjde stejný výsledek 0.618.

- $34/55 = 0.618$
- $55/89 = 0.618$
- $89/144 = 0.618$

Číslo 0.382 dostaneme následným dopočítáním do jedničky. Pro trading a technickou analýzu grafu se dále počítá s polovinami a násobky. Kupříkladu úroveň 0.236 dostaneme podělením 34/144. V trading světě se termín jako Fibonaccioho posloupnost neobjevuje, ale pracuje se s tzv. Fibonaccioho retracement modelem, který znázorňuje právě výše zmíněné úrovně a zobrazuje je do grafu. Tento model/nástroj potřebuje počáteční dva body. Bod 0 a bod 1, nebo také 0% a 100%. Tyto body určují oblast, ve které se budou Fibonaccioho úrovně nacházet. Fibonaccioho retracement je nástroj, který určuje na jaké úrovni se vrátí cena při *pullbacku*. Stejný model funguje i na opačnou stranu a nazývá se Fibonaccioho extension model. Ten udává na jaké úrovni se vyšplhá cena po dokončení *pullbacku*. Tyto úrovně jsou často zóny, ve kterých tradeři berou profit a ukončují dosavadní obchody.[22]

2.5.2.5 Využití odporových zón v tradingu

Při obchodování se tyto oblasti využívají pro nákup a prodej. Z pohledu obchodní strategie se tyto zóny dají použít dvěma způsoby. První způsob je nakupovat v support zóně a prodávat v rezistenční zóně. Při obchodování na páku se dá v rezistenční oblasti aktivum tzv. *shortnout*, což je spekulování o poklesu ceny a na tomto poklesu vydělat.

Druhý způsob je počkat až cena prorazí jednu z odporových zón, ať už support či rezistenci a využít toho, že poté bývá velmi dynamický pohyb ceny. V případě proražení supportu se dá očekávat prudký pokles ceny a v případě překonání rezistence zase výrazný nárůst ceny. Vždy je však dobré mít na paměti, že proražení takovéto odporové zóny může být pouze odchylka a cena se může vrátit zpátky. Jak poznat takovéto odchylky, případně jak snížit riziko obchodu bude vysvětleno dále.[20]

2.5.3 Breakouty

Traider by se měl nejčastěji držet trendu, protože to nejčastěji přináší největší jistotu. Platí fráze „Trend is your friend“. Často se k ní přidává dovětek „untill the end“ a tím je myšlen onen breakout. (změna trendu)

Termínem breakout se v tradingu označuje již dříve zmíněné proražení supportu. Stále častějším jevem jsou tzv. falešné breakouty, které dělají obchodníkům potíže. Tyto falešné signály lze mnohdy minimalizovat, nikdy však eliminovat, jelikož trh je v zásadě nepředvídatelný.

Způsob jakým se dají poznat falešné breakouty je jednoduše počkat na další vývoj trhu a neukvapovat se s otevíráním pozice. Jak dlouho čekat a také na co konkrétně čekat je už na samotném obchodníkovi, způsobů může být více.

Nabízí se metoda spodního a horního supportu. V grafu se vyznačí horní mez pro support a pod ní se nakreslí spodní mez pro support. Jak velká mez by měla být není striktně dáno a každý si ji může zvolit jak uzná za vhodné. Zde se obchodníci řídí vlastními zkušenostmi a intuicí.

Nyní nastává fáze čekání na signál. Pro někoho to může být uzavření svíčky pod spodní mezí support zóny, pro jiného uzavření druhé potvrzující svíčky pod touto zónou. Další strategie může být stanovit si o kolik % se musí cena vzdálit od supportu. Po splnění těchto podmínek je možné určit, zda šlo o falešný breakout, či nikoliv. Falešný průraz prorazí support, ale po chvíli se vrátí do proražené oblasti a pokračuje opačným směrem.[23]

V případě legitimního breakoutu se proražená zóna změní. Support na rezistenci a naopak. Velice často cena po breakoutu skočí zpátky do této oblasti, aby se breakout potvrdil a poté pokračuje stanoveným směrem. Počkání na tento jev může být taktéž jedna z možných strategií, která by se dala využít pro otevření obchodu.

Nastává zde otázka, zda se dá předpovědět breakout, nebo alespoň určit pravděpodobnost proražení. K tomu se dá využít celkový peněžní objem, který do dané měny přitéká (objem se zvětšuje), nebo z dané měny odtéká (objem se snižuje). Z burzy se dají vytáhnout tyto data v aktuálním čase a tyto informace použít jako indikátor, který může usnadnit rozhodování při určení zda support vydrží, či se prorazí.

2.5.4 Cenové patterny

Cenové patterny, nebo také vzory se dají pozorovat na všech trzích (Forex, kryptoměnový, komodity). Tyto vzory se tvoří jako odraz nálady a sentimentu ve společnosti. Další vzory se tvoří například kvůli manipulaci trhu velkými hráči. V kryptoměnovém prostředí se jim říká velryby (lidé s obrovskými finančními prostředky), zatímco normální obchodníci jsou malé rybky. Tito velcí hráči umí manipulovat trh svými enormními objemy peněz. Trh na tyto částky reaguje a velryby také. Tím se v grafu tvoří stejné vzory, protože postup těchto velryb je vždy velmi podobný. Více o velkých hráčích a velkých objemech peněz bude rozebráno v kapitole o manipulaci trhu.

Prozatím čtenáři postačí, že ve svíčkovém grafu se tvoří stejné vzory, kterých se dá využít pro otevření pozice. Člověk si musí uvědomit, že všichni vidí stejný graf a spekulují, jaký další cenový vývoj nastane. Právě díky tomuto důvodu cenové patterny fungují a mají vysokou úspěšnost při tradingu. Ovšem stejně jak při tvorbě těchto vzorů hrají velkou roli *velryby*, tak těchto vytvořených vzorů i zneužívají. Celkově však tyto vzory fungují a díky nim se s určitou mírou pravděpodobnosti dá odhadnout další pohyb na trhu.[24]

2.5.4.1 Trojúhelníky

Trojúhelník je klasickým cenovým vzorem, který se dá pozorovat zejména na vyšších timeframech a zároveň je univerzálním vzorem, který se objevuje ve všech trzích. S trojúhelníky jsou úzce spjaté vlajky, protože tyto vzory mohou být tvořeny malým trojúhelníkem. Definice trojúhelníku na grafu v oblasti TA je jednoduchá. Jedná se o vývoj ceny, která směřuje do jednoho bodu a svůj

cenový rozptyl postupně zmenšuje. V technické analýze se vyskytují tři typy trojúhelníků. Rostoucí, klesající a symetrický.

V případě rostoucího a klesajícího trojúhelníku je vždy jedno rameno horizontální a tvoří support/rezistenci. Rostoucí trojúhelník má horní rameno horizontální a klesající trojúhelník má dolní rameno horizontální. Oproti tomu symetrický trojúhelník nemá horizontální žádné rameno. Časem se cena dostane do bodu, kdy dosáhne konce tohoto vzoru. Jedná se přibližně o posledních 5 – 10 %. To indikuje, že nastává čas zlomu a cena prolomí tento pattern. Nelze s jistotou říci, jakým směrem se průraz uskuteční. Bylo řečeno, že trojúhelník je univerzální vzor a dá se podle něj obchodovat na všech trzích, nicméně směr prolomení trojúhelníku je silně ovlivněn daným trhem. Krypto trh a konkrétně každá kryptoměna může mít odlišný směr. Velice záleží na sentimentu a na náladě v daném trhu. Kupříkladu 5 měsíců za sebou může mít trader jistotu, že klesající trojúhelník breakne směrem nahoru a od jisté chvíle, kdy se změní nálada trhu se ten stejný vzor může začít prolamovat směrem dolů. Toto vše je při obchodování dobré mít na paměti. Proto není dobré slepě obchodovat čistě na základě TA. Rozhodnutí by se měla dělat pomocí kombinace TA a dalších důležitých aspektů, které musí trader znát. Jedná se zejména o čtení tzv. *Price action*. Pokud má být obchodník úspěšný, tak čtení price action je naprosto esenciální znalost. To jakým způsobem se dá číst price action bude zmíněno v samostatné kapitole navazující na část o cenových vzorech.

2.5.4.2 Vlajky

Vlajky jsou vzorem, který obchodníci v grafu velmi rádi vidí. To jakým směrem se cena bude pohybovat po dokončení vlajky můžeme s určitou pravděpodobností předpovědět. Jedná se o pravděpodobnost větší než 60 % a to už je pro tradera velice přívětivé. Vše musí být samozřejmě spojeno s dobrým RM a *Risk/Reward ratio*. Vlajky mají často podobu dvou rovnoběžek, ale mohou se vyskytnout i vlajky s podobou malého trojúhelníku. Tento pattern značí krátkou konsolidaci trhu, aby následně po této konsolidaci trh pokračoval v předešlém trendu. Pro vlajky je typické, že se objevují v trhu se silným trendem.[25]

2.5.4.3 Mnohonásobné vrcholy

Jako v předchozích případech můžeme vícenásobné vrcholy pozorovat jak na forex trzích, tak i na krypto trhu. Jedná se o vzor, který nastává pokud cena nedokáže překonat určitý support, odrazí se zpět a poté to zkouší znovu. Pokud cena ani napodruhé nepřekoná tento support a odrazí se zpět, může se jednat o tzv. *double bottom*. Opakem double bottom je *double top*. Někdy se tyto vzory označují jako *W* (double bottom) a *M* (double top). Tyto patterny ve většině případů obrací trend. Na jak dlouho se trend otočí záleží

na timeframu, ve kterém daný pattern vznikne. Tyto patterny musí být vždy kompletní a potvrzené. Potvrzené jsou ve chvíli, kdy cena vytvoří vzor dvou vrcholů a poté překročí dřívější lokální maximum/minimum, záleží na kontextu. Dvojitě vrcholy jsou poměrně častým vzorem, ale mohou se vytvořit například i trojitě vrcholy. Z tohoto důvodu je důležité potvrzení vzoru.

Každý obchodník má svou strategii, které se drží. V případě patternů M a W se nabízejí rovnou dvě. První z nich je počkat na druhý odraz a ihned otevřít pozici s úzkým stoplosem. Tento postup se nazývá spekulace, protože pattern zatím není kompletní. Člověk tedy pouze sází na to, že pattern vznikne a zároveň se chrání tím, že si nastaví potřebný stoplos v případě omylu. Celé se to dá shrnout pod risk management. Druhá strategie je čekání na potvrzení daného patternu a otevření pozice až po tomto potvrzení. Otevření pozice může nastat ihned, může nastat po malém pullbacku, vždy záleží na konkrétním traderovi.

2.5.4.4 Head and Shoulders

Hlava a ramena je cenový vzor, který s velkou pravděpodobností dokáže také otočit trend. Z tohoto důvodu je velmi populární. Mnoho obchodníků na něm zakládá své obchodní strategie také protože šance na dokončení tohoto patternu je z historického hlediska větší než 80 %. Jakákoliv strategie, která je založená na pravděpodobnosti větší než 60 % je v tomto byznysu velmi lákavá a z dlouhodobého hlediska opravdu přináší dobré výsledky.

Podle názvu vzoru může být čtenáři jasné, o jaké schéma se jedná. Vzor se skládá ze čtyř částí, levého ramena, hlavy, pravého ramena a *neckline*. Všechny tyto části až na *neckline* tvoří vrcholy. Nejprve se vytvoří levé rameno (vrchol), poté se vytváří hlava, která musí být výš než rameno a nakonec pravé rameno. *Neckline* je důležitá hranice, na kterou se musí cena pokaždé vrátit, aby byl vzor kompletní. Pokud se cena opravdu vrátí na úroveň *neckline* a prorazí tuto úroveň, tak se cena vrátí zpátky na úroveň, kde začal celý vzor. Cena však nemusí *neckline* prorazit vždy a může se odrazit od tohoto supportu.

Již bylo řečeno, že H&S může působit na trh jako reverzní pattern, ale je třeba zmínit, že tento vzor může být i dobrým *continuation* patternem. Příkladem může být trendový *bear* market ve kterém se vyskytne pattern H&S ve své základní podobě. To indikuje posílení trendu a další pokles ceny.[26]

Tento vzor může být někdy trochu hůře rozeznatelný a to protože může mít více variant. Tou nejjednodušší je pouze celý vzor převrácený a *neckline* netvoří support, ale rezistenci. U tohoto vzoru nemusí být *neckline* vždy vodorovně, může být i nakloněná a to už se v grafu hůře pozoruje. Také ramena nemusí být pokaždé stejně velká a stále se to z pohledu TA bere jako platný pattern. Nejtěžší na pozorování je kombinace těchto dvou faktorů, nakloněné *neckline* a nesymetrických ramen. Tato kombinace nemusí být vždy dobře čitelná pro člověka.

2.5.4.5 Cup and handle

Tento vzor připomíná svým tvarem hrníček s uchem. Nejedná se o žádný náhodný vzor, který se objevuje v grafech a proto má své logické opodstatnění. Zprvu se v trhu vytvoří maximum (vrchol), to je začátek patternu. Cena nedokáže překonat toto maximum a trh časem ztrácí na síle. Děje se tak postupně a pomalu. To indikuje, že se jedná pouze o akumulativní fázi, nebo také korekci. Trh se dostane na lokální minimum a od tohoto minima se postupně začíná vzpamatovávat a cena opět stoupne až k původnímu maximu. Toto maximum je rezistentní zóna a zde se očekává, že cena tuto rezistenci nepřekoná na poprvé. Odrazí se na krátkou dobu zpět a vytvoří tak pomyslné ucho.[27]

Tento vzor se dá obchodovat znovu více způsoby. Více strážlivý přístup je počkat na dokončení celého vzoru i s uchem a otevřít pozici pro nákup až poté. Agresivnější způsob je otevřít pozici hned po malém pullbacku zpět (odraz od rezistence). Vždy je třeba mít na paměti, že jakákoliv strategie je pouze obchodování s pravděpodobnostmi a proto je třeba mít dobrý RM.

2.5.4.6 Elliott waves

Elliott pattern a Elliot teorie říká, že market se pohybuje v pravidelných cyklech, které se dají předpovídat. Úspěšnost této předpovědi nejvíc záleží na zkušenostech, protože se nedá určit přesně.

Podle teorie Eliotových vln se market, který je v trendu pohybuje v osmi vlnách a dvou fázích. První fáze je ve směru trendu a druhá fáze je korektivní. Tyto fáze jsou rozděleny do vln, první fáze má celkem pět vln a druhá fáze tři vlny. Dále se vždy vždy v rámci těchto fází střídají vlny impulzivní a korektivní.

Celý pattern začíná první impulzivní vlnou ve směru trendu a následuje druhá vlna, která je korektivní (proti trendu). Tyto vlny jsou vždy kratší. Nejdelší ze všech vln je vlna třetí (impulzivní), která vyzvedne cenu výše. Následuje znovu menší korektivní vlna, která by neměla cenu posunout níže než k poslednímu *lower low*. Nakonec se cena znovu odrazí v poslední impulzivní vlně nad poslední *higher high*. Zde končí první fáze Eliotových vln. Druhá fáze začíná znovu impulzivní vlnou, tentokrát v proti směru trendu a tato vlna je následována dalšími dvěma vlnami korektivní a impulzivní v tomto pořadí.[28]

V tom jak poznat hloubku korekce může pomoci Fibonaccioho retracement. Pokud obchodník správně určí začátek první impulzivní vlny, může zkombinovat Eliotovu teorii vln s Fibonaccioho retracement nástrojem a lépe tak odhadnout do jakých úrovní se může cena vrátit.

2.5.5 Patterny svíček

V grafu se dají číst různé patterny a obrazy. Tyto vzorce chování trhu jsou složeny z cenových svíček a mohou pomoci při určení dalšího cenového vývoje.

Tyto vzorce však nejsou jediným nástrojem, který má obchodník k dispozici. Samotná cenová svíčka v sobě obsahuje mnoho užitečných informací a každý trader by je měl dobře znát. Jednotlivá svíčka může mít více podob a různé tvary a podle těchto tvarů se dá také odhadnout budoucí vývoj. Vždy však záleží na okolnostech a kontextu v grafu. Existuje mnoho svíčkových patternů, avšak zde budou pro jednoduchost popsány pouze některé. Dále je potřeba zmínit, že tyto typy a formace svíček se vztahují k *japanese candlestick* grafu.[29]

Dříve než budou vysvětleny jednotlivé svíčkové patterny, tak se podíváme na základní svíčku a čím je tvořena. Svíčka v japonském grafu se skládá z těla, knotu a barvy. Knot může být na obou stranách těla a udává rozsah ceny za daný časový úsek. Svíčka se otevře na určité hodnotě a v průběhu daného časového horizontu se může cena pohybovat směrem nahoru i dolů. To vše je zachyceno v jedné svíčce. Pokud se cena uzavře níže, než byla svíčka otevřena, tak barva svíčky je standardně červená, pokud se uzavře výše, tak je barva zelená. Samotné zbarvení svíčky udává poměrně důležitou informaci a to, že cena v daném časovém úseku vzrostla. Důležitou roli hraje v TA také počet svíček stejné barvy jdoucí za sebou. To indikuje silný trend ať už směrem nahoru, nebo dolů. Další důležitým faktorem při analýze svíčky je knot, někdy nazývaný stín svíčky. Velikost knotu, případně žádný knot indikuje mnoho věcí, které budou vysvětleny u jednotlivých typů svíček.

- Základní typy svíček

Morning star: Jedná se o složení tří svíček dohromady a je charakteristická pro *down trend*. První svíčka je dlouhá a tvoří ji z 75 % tělo, zároveň je ve směru klesajícího trendu. Druhá svíčka má delší knot a poměrně malé tělo. Třetí svíčka je proti směru trendu a musí se uzavřít alespoň nad polovinou první svíčky.

Evening star: Tato formace svíček je naprosto totožná s Morning start až na to, že je ve směru rostoucího trendu.

Doji: Doji se nazývá svíčka, která má velmi malé tělo. Jinak řečeno je cena při otevření svíčky a při uzavření svíčky skoro stejná.

Hammer: Tato svíčka má tvar kladiva. Má malé tělo a velmi dlouhý knot směrem dolů. Značí změnu klesajícího trendu na stoupající trend.

Marubozu: Marubozu je svíčka tvořená pouze tělem bez knotů. Barva svíčky indikuje jakým směrem se trh ubírá.

Toto byly základní typy svíček a jejich základní formace. Existuje celá řada svíčkových vzorů, které jsou sice důležité, ale už nejsou tolik běžné. Tyto svíčkové vzory budou zobrazeny na obrázku pod odstavcem.

2.5.6 Indikátory - nástroje pro TA

Technická analýza (TA) je nástroj pro analýzu finančních trhů, který se zaměřuje na studium historických tržních dat. TA se snaží identifikovat opakující se vzorce a trendové signály, které by mohly naznačovat budoucí pohyby na trhu.

Mezi nejdůležitější nástroje TA patří indikátory, což jsou matematické funkce, které umožňují vizualizovat různé aspekty tržního chování, jako je například trend, síla trendu, překoupenost, nebo přeprodanost. Tyto indikátory lze použít pro analýzu trhu, identifikaci trendů a korekcí a také pro rozhodování o obchodních signálech.

V oblasti kryptoměn, které jsou obecně velmi volatilní a často se setkávají s prudkými pohyby cen, jsou indikátory TA velmi užitečné. Pomáhají investorům analyzovat trh, porozumět vývoji cen a uskutečňovat obchody s větší pravděpodobností úspěchu. Kromě toho mohou být indikátory TA použity pro tvorbu algoritmů pro automatické obchodování.

Mezi nejčastěji používané indikátory TA patří například:

- Základní typy indikátorů

Moving Average: jednoduchý průměr posledních X hodnot. Tyto hodnoty poté slouží jako ukazatel trendu

Relative Strength Index (RSI): ukazuje, zda je aktivum takzvané překoupené nebo přeprodané na základě cenových pohybů v určitém období

Bollinger Bands: ukazuje volatilitu cenového pohybu a pomáhá identifikovat výkyvy ceny vzhledem k průměrnému cenovému pohybu.

Average Convergence Divergence (MACD): ukazuje rozdíl mezi krátkodobým a dlouhodobým MA a slouží k identifikaci trendu a jeho změn

Fibonacci retracement: ukazuje potenciální úrovně podpory a odporu na základě matematického poměru mezi cenami a Fibonacciho čísly

Tyto indikátory jsou jen několik příkladů z mnoha indikátorů, které jsou k dispozici pro technickou analýzu trhů. Každý z těchto indikátorů má své vlastní výhody a omezení. V dalších podkapitolách budou podrobněji vysvětleny další, o něco více složitější indikátory, které jsou také součástí analýzy trhu.[30]

2.5.7 Divergence

Divergence je jeden z nejvýznamnějších technických indikátorů pro analýzu finančních trhů včetně trhu s kryptoměny. Tento indikátor se zaměřuje

na analýzu vztahu mezi cenou a oscilátorem, což je ukazatel, který se pohybuje mezi dvěma extrémy a poukazuje na situaci, kdy trh je překoupený nebo přeprodáný. Divergence ukazuje odchylku mezi pohybem cen a oscilátorem a může být použita k identifikaci možných změn v trendu. Existují dvě základní formy divergence - *bullish divergence* a *bearish divergence*. *Bullish divergence* signalizuje, že trh je připraven na vzestupný trend, zatímco *bearish divergence* signalizuje naopak pokles. Identifikace divergence může být užitečným nástrojem při analyzování budoucího trendu na trhu. Při použití tohoto indikátoru je však třeba mít na paměti, že výsledky analýzy mohou být ovlivněny mnoha faktory a ne vždy jsou přesné.[31]

2.5.8 Objem obchodů

Cenový objem (Price-Volume) je jeden z nejoblíbenějších technických indikátorů používaných v technické analýze kryptoměn. Tento indikátor se zaměřuje na vztah mezi pohybem cen a objemem obchodů na trhu. Zatímco ceny mohou být ovlivněny mnoha faktory, objem obchodů se obvykle zvyšuje, když se trh pohybuje v určitém směru.[32] Cenový objem může pomoci identifikovat růstový nebo klesající trend na trhu s kryptoměnami a také předpovědět možné změny ve stávajícím trendu.

2.5.8.1 Objemový profil

Indikátor objemového profilu je nástroj používaný pro analýzu objemu obchodování s kryptoměnami v určitém časovém období. Tento indikátor zobrazuje vertikální histogram objemu obchodů na různých cenových úrovních, ukazuje úrovně, na kterých došlo k největšímu počtu obchodů. Objemový profil může pomoci při identifikování důležitých úrovní supportu a rezistence.[33]

2.5.9 Pivotové hladiny

Pivotové hladiny jsou často používané v oblasti kryptoměn. Jedná se o úrovně supportu a rezistence, které jsou určeny podle předchozích výkyvů ceny. Pivotové hladiny mohou být použity k identifikaci klíčových cenových hladin, při kterých se očekává zvrát trendu. Existuje několik způsobů výpočtu pivotových hladin, přičemž nejčastější jsou standardní pivotové body, Fibonacciho pivotové body a Camarilla pivotové body.

Standardní pivotové body se vypočítávají pomocí předchozího dne nebo týdne a zahrnují hlavní pivotový bod (PP) a tři úrovně supportu a tři úrovně rezistence. Fibonacciho pivotové body jsou založeny na Fibonacciho číslech a poskytují více úrovní podpory a rezistence. Camarilla pivotové body jsou založeny na předchozím denním obchodování a též obsahují více úrovní podpory a rezistence.[34][35]

Pivotové hladiny jsou užitečným nástrojem pro obchodníky, kteří hledají možnosti vstupu a výstupu na trhu, neboť jim poskytují jasná očekávání o tom,

kam by se cena mohla pohybovat. Navíc jsou často používány jako místo pro umístění stop losu a take profitu. Použití pivotových hladin jako nástroje pro technickou analýzu trhu by mělo být prováděno ve spojení s dalšími indikátory, tak jak je běžné u ostatních indikátorů.

2.5.10 Manipulace trhu

Manipulace trhu v oblasti kryptoměn se vyskytuje v podobě různých praktik, které mají za cíl ovlivnit cenu kryptoměn v určitém směru. Mezi nejčastější formy patří tzv. "pump and dump" skupiny, které se snaží uměle navýšit cenu kryptoměny, aby ji poté prodaly s vysokým ziskem.[36] Další formou je tzv. "spoofing", kdy manipulátor vytváří falešné objednávky k nákupu nebo prodeji, aby vytvořil dojem silné poptávky nebo nabídky na trhu. Tato praktika může vést k umělému navýšení cen kryptoměn a způsobit ztráty pro ostatní obchodníky. Manipulace trhu je v oblasti kryptoměn nelegální. InSTITUTE regulující kryptoměny se snaží tuto praxi co nejvíce omezit a potrestat viníky.

2.5.11 Timeframy

Timeframem se rozumí časový rámec, v němž jsou zobrazeny data o cenách kryptoměn. Trh se může v krátkých časových intervalech chovat jinak než v dlouhodobých, a proto je důležité zvolit správný časový rámec pro analýzu.[37] Typické timeframy jsou například 1 minutový, 5 minutový, 15 minutový, 1hodinový, 4hodinový, denní nebo týdenní. Kratší timeframy jsou vhodné pro každodenní obchodování, zatímco delší pro dlouhodobou investici. Je důležité vybírat timeframy, které odpovídají obchodní strategii. Při kombinaci více timeframů lze získat lepší přehled o trendu trhu.

Část II

Praktická část

Analýza

3.1 Obecné požadavky

V rámci diplomové práce se chci zaměřit na vytvoření trading bota pro kryptoměny, který bude schopen samostatně analyzovat trhy a uskutečňovat, případně simulovat obchody. Pro tento účel je nutné provést analýzu různých technologií a vybrat ty nejvhodnější.

Výsledná aplikace by měla stahovat data v reálném čase z předních burz/burzy, nebo využívat volně vystavené API, které tyto data nabízí. Nad těmito daty následně provádět analýzu a na základě této analýzy/strategie vyhodnotit výsledek. Výsledkem by měl být signál pro nákup nebo prodej aktiv.

Z důvodu požadavku na vyhodnocení dat v reálném čase je vhodné pro stahování dat použít formu streamu. Streamovacích knihoven a API je k dispozici velké množství a proto bude nutné udělat průzkum a jednu z knihoven vybrat. Další z mnoha požadavků na aplikaci je, aby byly data dostupná pro následné zpracování. Je dobré mít na paměti, že dat z krypto marketu je nespočetné množství. Zde je dobré otevřít téma Big Data a použít technologie, které toto odvětví nabízí. Pro paralelní zpracování obrovského množství dat se nabízejí hned dvě Big datí technologie, které spolu skvěle pracují. Prvotně se jedná o technologii Kafka z rodiny Apache. Kafka dokáže velmi spolehlivě a rychle přenášet velké množství dat mezi jednotlivými systémy. Druhá technologie, která koresponduje s požadavky na danou aplikaci je také z rodiny Apache a to konkrétně Apache Spark. Spark je výpočetní systém, který zvládá v reálném čase zpracovávat a počítat s velkým množstvím dat.

Jedním z dalších požadavků na aplikaci je její škálovatelnost. Z počátku může být výkon pro výpočet dostatečný, ale může se časem ukázat, že bude potřeba výkon navýšit. Důvodů může být více, například rozšíření aplikace o jiné kryptoměny, nebo větší množství zpracovávaných údajů. Zde se bude jednat o aplikaci, která zpracovává reálná data, proto je žádoucí, aby byla

aplikace tzv. odolná vůči výpadkům a zároveň aby byla *high available* (vysoce dostupná). Výsledná data upravená podle vybrané strategie je nutné perzistentně ukládat. Nároky na ukládání nejsou nijak vysoké, každý DB stroj je schopen kvalitně a rychle ukládat data. Problém, který zde nastává je čtení. Čtení by mělo být dostatečně rychlé a pokud to bude DB poskytovat, tak i paralelní. Na tyto požadavky bude brána zřetel při analyzování a výběru technologií, které budou popsány v následujících kapitolách.

3.2 Průzkum a popis možných technologií

V této kapitole bude provedený průzkum a důkladný popis Big Datích technologií, které připadají v úvahu pro potřeby výsledné aplikace. Na konci kapitoly bude shrnut výsledek průzkumu a uveden seznam se zvolenými technologiemi.

3.2.1 Amazon Kinesis

Amazon Kinesis je jedna z mnoha služeb poskytovaných od společnosti Amazon, konkrétně Amazon Web Services (AWS). Služba je určená pro rychlé stahování dat pomocí streamu a to z různých zdrojů. Dále umožňuje tyto data zpracovat, transformovat a ukládat v různých formách. Toto zpracování je umožněno ve velkých objemech dat. Služba je záštitou pro další služby, které se starají právě o tyto zpracování.

- Přidružené služby

Amazon Kinesis Data Streams: služba zaměřená na sběr a ukládání velkých datových objemů v reálném čase pomocí streamu. slouží jako ukazatel trendu

Amazon Kinesis Data Firehose: služba umožňuje přesměrování dat do různých platforem jako ElasticSearch a další

Amazon Kinesis Data Analytics: služba, která poskytuje možnost provádět analýzu streamu v reálném čase pomocí jazyka SQL a zároveň vytvářet vizualizace ze získaných dat

3.2.2 Apache Spark

Apache Spark je framework pro distribuované zpracování dat. Řadí se do rodiny Apache, kam dnes patří velká škála technologií. Spark umožňuje paralelní zpracování velkých objemů dat a disponuje širokou škálou funkcí pro následnou analýzu, transformaci a manipulaci s daty. Apache Spark je vysoce škálovatelný a odolný vůči výpadkům a chybám.

Technologie Apache Spark se od ostatních technologií liší ve způsobu zpracování. Umožňuje data zpracovávat streamově, tak jak přicházejí ze streamu, anebo je zpracovává v batchi. Tyto batche mohou být různě velké, záleží na

konfiguraci. Pokud se jedná o velmi malé batche, označují se správně jako micro batche.

Právě již zmiňovanou výhodou a silnou stránkou je toto batchové zpracování, pro které je Spark nejlépe optimalizovaný. Speciální Struktury jako jsou distribuované Dataframy a Datasets, které Spark nabízí a nad kterými zvládne dělat různé optimalizace, pouze zvyšuje efektivitu tohoto batchového zpracování.

S technologií Apache Spark lze pracovat například v pomoci Pythonu. Tyto dvě technologie se pak nazývají PySpark a tvoří jednotný celek. Se sparkem lze pracovat také v jazyce Scala, který vychází a je plně kompatibilní z jazyka Java, který je mi velmi blízký, protože s Javou mám dlouholeté zkušenosti.

- Spark komponenty

Spark Core: Jedná se o základní modul, který poskytuje obecné funkce pro distribuované zpracování, včetně plánování úloh, paměťového řízení a komunikace mezi uzly v clusteru. Spark Core je také zodpovědný za správu distribuovaného prostředí a výkonného

Spark SQL: Spark SQL je modul pro práci s daty ve formátu SQL. Tento modul poskytuje pohodlnou práci s různými komponenty a podporuje také integraci s externími datovými zdroji, jako jsou relační databáze, Hadoop Distributed File System (HDFS) nebo Apache Hive a Apache Cassandra.

Spark Streaming: tato komponenta zpracovává data v reálném čase. Spark Streaming umožňuje přijímat a zpracovávat data v malých časových intervalech (mikro-batche) a poskytuje podobné rozhraní jako Spark Core a Spark SQL. V jednom frameworku se tedy skrývá služba pro zpracování dat i pro jejich analýzu.

MLLIB(Machine Learning Library) jedná se o knihovnu pro strojové učení, která je dostupná v rámci Sparku. MLLIB poskytuje algoritmy a nástroje pro různé typy úloh jako jsou klasifikace, regrese, agregace, více úrovněvé filtrování a další. V oblasti strojového učení využívá MLLib výhod distribuovaného prostředí Sparku pro efektivní a škálovatelné výpočty

V předchozích odstavcích byly stručně popsány a shrnuty asi dvě nejnámější technologie v oblasti Big Data pro rychlé a distribuované zpracování dat. V tuto chvíli se jeví Apache Spark jako správný nástroj a to zejména díky optimalizovanému batchovému zpracování a jednoduché integraci do okolního systému. Další výhodou může být fakt, že je kompatibilní s jazykem Scala. Jako další benefit vidím v tom, že Spark má vlastní machine learning knihovnu, která by se dala v budoucnu využít pro rozšíření aplikace o strojové učení.

Kombinace jazyku Scala a technologie Apache Spark dává obrovský smysl neboť Scala je kompilovaný jazyk (stejně jako Java), který se překládá do bytcode JVM. V praxi to znamená, že Scalovský kód může být spuštěn přímo na Spark uzlech. Scala je na rozdíl od Javy funkcionální jazyk a pracuje se s ním jinak. Právě zde leží výhoda kombinace Sparku a Scaly, protože se využívá výkonnosti Sparku pro distribuované zpracování a zároveň se využívá výhod jazyka Scala pro elegantní a efektivní programování. Funkcionální přístup umožňuje snadnou paralelizaci a distribuci úloh. Existuje mnoho knihoven a frameworků, které poskytují Scala rozhraní právě pro Apache Spark.

Po uvážení všech nejdůležitějších požadavků na aplikaci a zvážení výhod Apache Spark engine, byl zvolen právě Spark jako stěžejní technologie, na kterou se bude dále nabalovat zbytek celkové infrastruktury. Jedním z požadavků je právě již zmíněné stahování dat ze streamu. Takovýchto dat může být v oblasti kryptoměn velké množství a s dalším požadavkem na škálovatelnost aplikace se jeví jako logický krok sáhnout po další technologii, která umožňuje rychlý a paralelní přenos dat. Mluvím zde o technologii Apache Kafka. Kafka jako další patří do skupiny Apache, právě to je jeden z důvodů, proč se často používá v kombinaci se Sparkem. Tím dalším je již zmíněná kompatibilita. Spark může efektivně konzumovat data z Kafky pomocí *topiců*. Dokonce je schopný tyto data konzumovat z více *pipeline* a to samé umí i Kafka. Kafka je taktéž schopna přijímat data z více *pipeline* zároveň. Pokud je to možné a pokud zvolené technologie odpovídají případu užití, tak rád volím nástroje od stejné společnosti. Výsledná aplikace si poté zachovává ucelený obrys.

3.2.3 Apache Kafka

V této podkapitole bude krátce shrnut a popsán nástroj Kafka. Jelikož tato technologie již byla zmíněná v předchozích odstavcích, proto zde budou shrnuty pouze její vlastnosti a přednosti. Důvod, proč tato technologie byla zvolena již zazněl. Jednou z předností Kafky je její škálovatelnost, velice snadno lze do clusteru přidávat další a další brokery, což může pomoci zejména při zvýšené zátěži. Zpracování dat probíhá v reálném čase a bez vysoké latence. Díky několika brokerům má Kafka možnost replikovat data. Je tedy odolná proti chybám. Jako poslední vlastnost a nespornou výhodou je možnost uchování historie zpráv, takže pokud vypadne na nějakou dobu aplikace konzumující data z Kafky, není to velký problém, protože data v Kafce zůstanou uložena na nezbytně nutnou dobu.

3.2.4 Apache Cassandra

V tomto bodě jsou již zvolené tři hlavní nástroje, které budou spolu komunikovat, přeposílat si mezi sebou data a tyto data v určitém bodě transformovat, nebo nad nimi provádět určitou logiku. Z těch velkých a stěžejních technologií to jsou tedy Apache Spark, Apache Kafka, Apache Cassandra a funkcionální

jazyk Scala podporující paralelní programování a distribuci úloh v jednoduché formě.

3.2.5 Krátké shrnutí

Zvolením nástroje Apache Spark jako základní kámen pro celou Aplikaci byl tranzitivně odstartován řetězec dalších technologií, které na Spark dále navazují a jejíž zvolení vidím jako logický krok.

V dalších kapitolách a podkapitolách bude detailně popsána implementace aplikace a její celkový návrh. Nástroje, které byly použity a důvody proč tomu tak bylo. Zatím známe pouze tři až čtyři použité technologie, které tvoří základ aplikace, to ale zdaleka nejsou všechny. Zbylé technologie budou postupně představeny spolu s praktickou ukázkou implementace z již hotové aplikace. V této kapitole nebyly představeny, protože netvoří jádro programu. Tyto další technologie bych nazval jako periferie a v podstatě by se dala použít jakákoliv varianta, nebo podobný nástroj místo nich.

Návrh a implementace

V této části práce je prakticky popsáno a názorně předvedeno, jakým způsobem je výsledná aplikace navržena a implementována. Jednotlivé komponenty jsou vypíchnuty a k nim je přidána ukázka kódu a použití v aplikaci. Čtenář by si tak měl lépe udělat obrázek o tom, jak je aplikace propojena a jak jednotlivé komponenty fungují a k čemu jsou potřeba.

4.1 Infrastruktura a architektura datového toku

Aplikace jako taková (*Spark stream bot*) je rozdělena do jednotlivých dílčích aplikací, které dohromady tvoří celek. Každá z těchto aplikací má svůj úkol. Tímto se docílilo efektu, že aplikace je distribuovaná. Každá část může běžet jako samostatný objekt. Celkem byla aplikace rozdělena na tři menší aplikace psané v jazyce Scala. První aplikace s názvem *CryptoPriceTracker* zasílá získaná data přes *websocket* do již zmíněné Kafky. *WebSocket* je protokol spojený s HTTP, protože nejprve se zašle požadavek přes HTTP, který obsahuje request pro přechod na protokol *websocket*. Můžeme tedy tyto dvě technologie zahrnout do seznamu použitých nástrojů.

4.1.1 WebSocket

Technologie *websocket* byla vyvinuta pro obousměrnou komunikaci mezi klientem a serverem. Komunikace začíná již zmíněným zasláním HTTP *requestu* pro přechod na *websocket*. Tomuto navázání spojení se říká *handshake*. Po tom, co je *handshake* úspěšný je vytvořen kanál mezi klientem a serverem, po kterém se mohou posílat data, dokud jedna ze stran tento kanál neuzavře. Tento způsob komunikace je dobrý například pro chat v reálném čase anebo pro zasílání dat jako stream, což je přesně ten důvod, proč byla tato technologie zvolena.

V ukázce pod textem je zaznamenáno vytvoření *websocketu* pomocí extendování *WebSocketClient* (`new URI(endpoint)`)

```
import org.java_websocket.client.WebSocketClient
import org.java_websocket.handshake.ServerHandshake

object CryptoPriceTracker extends App with KafkaSchema {

  val bitcoinEndpoint = "wss://stream.binance.com:9443/ws/
  btcusdt@aggTrade"
  val ethereumEndpoint = "wss://stream.binance.com:9443/ws/
  ethusdt@aggTrade"

  val bitcoinClient = new ReconnectingWebSocketClient(bitcoinEndpoint,
  "BTC", "BTC_TOPIC")
  val ethereumClient = new ReconnectingWebSocketClient(ethereumEndpoint,
  "ETH", "ETH_TOPIC")

  bitcoinClient.connect()
  ethereumClient.connect()

}

class ReconnectingWebSocketClient(endpoint: String, symbol: String,
  topic: String) extends WebSocketClient(new URI(endpoint)) {

  override def onOpen(handshakedata: ServerHandshake): Unit = {
    println(s"Connected to $symbol websocket")
  }
}
```

Zdrojový kód 1: Ukázka navázání spojení přes websocket

4.1.2 Apache Kafka

Jako streamová technologie pro přeposílání dat a zároveň jako prostředník mezi producerem a konzumerem je použita Apache Kafka. Kafka pracuje na principu topiců. Topic je skupina, do které se zasláná data roztrídí a na kterých jsou zase k dispozici pro další zpracování. Tyto topicy se musí nejdříve vytvořit, aby byla Kafka schopná data přijmat. Pod textem je ukázka vytvoření topiců, zaslání dat do Kafky z producer aplikace a také získání těchto dat z konzumer aplikace.


```
docker exec -it kafka kafka-topics.sh --create --topic btc-topic
--partitions 1 --replication-factor 1 --if-not-exists
--bootstrap-server localhost:9092
```

Zdrojový kód 2: Ukázka vytvoření topiců v Kafce

4.1.3 Docker

Z ukázky kódu číslo 2 je vidět, že pro vytvoření topiců byl v příkazu zmíněn Docker. Docker je open-source platforma určená pro spouštění aplikací v tzv. kontejneru. Tento kontejner je lightweight prostředí, které obsahuje veškeré potřebné knihovny, kód a nástroje pro spuštění dané aplikace/technologie. Software se tak stává lépe spravovaným, přenositelným a spustitelným na jakékoli platformě(i cloudové) bez ohledu na OS. Docker umožňuje spustit aplikace přes tzv. image. Definice image a konfigurace Dockeru je uložena v souboru s příponou `yml`. Tento soubor `docker-compose.yml` se mimo jiné stará o orchestraci celé aplikace. V ukázce je zachycen příklad takové konfigurace.

```
version: '3'

services:
  zookeeper:
    image: wurstmeister/zookeeper
    container_name: zookeeper
    ports:
      - "2181:2181"
  kafka:
    image: wurstmeister/kafka
    container_name: kafka
    ports:
      - "9092:9092"
    environment:
      KAFKA_ADVERTISED_HOST_NAME: localhost
      KAFKA_ZOOKEEPER_CONNECT: zookeeper:2181
```

Zdrojový kód 3: Ukázka konfigurace dockeru

4.1.4 Apache Spark

V této praktické kapitole bude ukázáno jádro celého programu, nástroj Spark použití jeho dominantních funkcí a objektů. Připomeneme si, že Spark má výhodu v rychlém batchovém zpracování. Tyto batche mohou mít flexibilní velikost, od mikro batche až po batche o velikosti například 100 000 záznamů. V praxi toto číslo může být mnohem vyšší. Záleží na několika faktorech jako například povaha dat, hardwarové možnosti, velikost samotné zprávy, nebo latence zpracování. Pokud se klade nárok na rychlé zpracování dat a poskytnutí výsledku takřka v reálném čase, je vhodné zvolit velikost batche menší, pokud se klade důraz na vysoký výkon zpracování a je potřeba zpracovat obrovské množství dat, volí se velikost batche spíše větší.

Další výhodou spark engine jsou jeho datové struktury speciálně přizpůsobené pro práci s velkými daty, nad kterými Spark dokáže provádět optimalizace. Spark umí optimalizovat celkem tři druhy plánů.

- Spark komponenty

Logický plán: jedná se o abstraktní reprezentaci operací v kódu. Tento plán říká co se má stát s daty, ale neříká jak se výsledku docílí.

Fyzický plán: fyzický plán je konkrétní implementace operací z logického plánu. Dochází k přeformátování operací, paralelizaci, nebo k optimalizaci paměti.

Exekuční plán: poslední plán, který vychází z fyzického plánu a je v něm zapsáno přesné vykonání kroků nad daty.

```
object BtcProfile extends Session {
  def main(args: Array[String]): Unit = {
    val btcProfile = new BtcProfileTransform()
    val streamDataset = kafka.readKafkaStreamTopic(
      (appConfig.getString("btcProfile.btcTopic"))

    for ((windowDuration, index) <- windowDurations.zipWithIndex){
      executor.execute(() => { // Delay execution based on index
        TimeUnit.SECONDS.sleep(index * 2)
        btcProfile.processStream(streamDataset, windowDuration,
          triggerInterval)
      })
    }
    executor.shutdown()
  }
}
```

Zdrojový kód 4: Ukázka čtení topicu a paralelizace procesu

Datové struktury jako *Dataframe* a *Dataset* jsou příklady optimalizovaných objektů a mají strukturu tabulky se sloupci. Jak již bylo několikrát zmíněno, Spark je určený pro rychlou práci s velkými daty a proto se dají tyto celé datové struktury číst z různých platforem, a nebo do nich zapisovat. V ukázce pod textem je znázorněno použití *Dataframe* pro každý příchozí batch z Kafky. Zároveň je vidět, že pro každý batch je zavolána funkce *IndicatorsCalculation*, která počítá indikátory pro následné vyhodnocení a použití ve zvolené strategii. Funkce nejen počítá, ale i transformuje *Dataframe* a tento transformovaný *Dataframe* je dále poslán do Cassandra pro perzistentní uložení.

```
val query = windowedStream
    .writeStream
    .outputMode("append")
    .trigger(Trigger.ProcessingTime(triggerInterval))
    .foreachBatch { (batchDF: DataFrame, batchId: Long) =>
        // Acquire the lock before processing the batch
        lock.synchronized {
            // Call indicator calculation method
            IndicatorsCalculation(batchDF, windowDuration)
        }
    }
    .start()
query.awaitTermination()
```

Zdrojový kód 5: Ukázka zpracování po batchi (.foreachBatch)

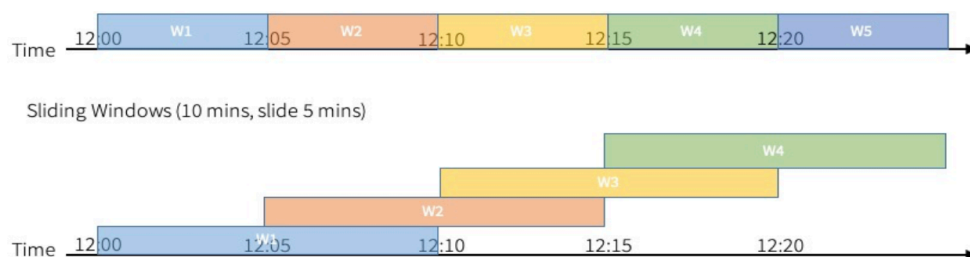
V rámci vysvětlení datového toku a transformace dat je dobré zmínit, že současná aplikace využívá Spark Windowed funkci, která je schopná po určité konfiguraci číst příchozí data ze streamu provádět nad těmito daty výpočty v různých časových intervalech. Tento způsob je také znám pod pojmem *sliding window*. Tohoto způsobu je využito pro zpracovávání ceny bitcoinu v určitém timeframu. V praxi je použito více vláken a každé vlákno má na starosti jeden timeframe, nad kterým provádí výpočty. Po každém výpočtu je posunut tento timeframe o tzv. *sliding duration*. Toto okno může přímo navazovat na předchozí timeframe, nebo ho překrývat. Pro lepší pochopení této sliding window techniky je přiložen obrázek. Dále je možnost vidět náhled do aktuální implementace.

```
object SlideWindowConfig {
  val windowDurations: Seq[String] = List("5 minutes",
    "10 minutes", "15 minutes", "30 minutes", "60 minutes")
  val triggerInterval = "5 minutes"
  val slideDuration = "5 minutes"
}
```

Zdrojový kód 6: Ukázka konfigurace sliding window

```
val windowedStream = processedStream
  .withWatermark("timestamp", "1 seconds")
  .groupBy(window($"timestamp", windowDuration,
    slideDuration).alias("window"))
  .agg(collect_list(struct(kafkaSchema.fields.map(
    field => col(field.name)):_*).alias("batchData"))
  .select(
    col("window").as(s"window"),
    explode($"batchData").alias("batchData")
  )
```

Zdrojový kód 7: Ukázka použití sliding window



Obrázek 4.1: Spark sliding window

4.1.5 Cassandra

V předchozích kapitolách byla popsána databáze Apache Cassandra, zdůrazněny její výhody a kompatibilita se Sparkem. Jedna z těchto výhod byla možnost použití Spark/Cassandra konektoru. Díky tomuto konektoru je umožněno dotazování nad Cassandra přímo ze Sparku a to pomocí SQL nebo CQL. V následujících ukázkách je vidět použití dvou různých způsobů dotazování.

```
def writeToCassandra(dataFrame: DataFrame, table: String,
  keyspace: String): Unit = {
  dataFrame.write
    .format(appConfig.getString("cassandra.format"))
    .options(Map("table" -> table, "keyspace" -> keyspace))
    .mode(SaveMode.Append)
    .save()
```

Zdrojový kód 8: Ukázka zápisu celého Dataframe do Cassandra

```
def readLastRecordForSymbol(symbol: String, table: String,
  keyspace: String): DataFrame = {
  val query = s"SELECT * FROM $keyspace.$table WHERE
    symbol = '$symbol' LIMIT 1"
  val result = spark.read
    .format("org.apache.spark.sql.cassandra")
    .option("query", query)
    .option("keyspace", keyspace)
    .option("table", table)
    .load()

  result
}
```

Zdrojový kód 9: Ukázka použití CQL query pro zápis

4.2 Monitoring a logování

Doposud byla popsána architektura trading bota z pohledu výměny dat mezi jednotlivými aplikacemi, které spolu komunikují a posílají si data. V této kapitole bude navázáno na popis podpůrných nástrojů, které byly použity za účelem pozorování, logování a pro lepší udržitelnost aplikace jako celku. Zde je nutno říci, že tato sada celkem třech technologií nebyla zvolena na začátku projektu jako hlavní, ale postupem času se projekt dostal do fáze, kdy bylo potřeba zvolit jinak.

Z počátku měl monitoring stát na triu technologií, které spolu dobře komunikují a je zaručena kompatibilita. Jedná se o nástroje z sady ELK (Elastic search, logstash, Kibana). S těmito technologiemi jsem získal zkušenosti v předešlých projektech, proto bylo zvoleno právě jejich použití. Bohužel poslední verze ELK přešla pod jinou licenci, která není open-source. Z tohoto důvodu byla nalezena alternativa v podobě technologií Fluentd (Td-agent), Prometheus a Grafany. Nástroje budou postupně představeny a ukázány v praxi.

4.2.1 Fluentd

Fluentd nebo také td-agent je unifikovaný nástroj pro sbírání dat z různých systémů a platforem. Přidáním Fluentd do aplikace se zajistí, že působí jako centrální vrstva postavená nad všemi komponenty, která sbírá odeslaná data. V případě traiding bota se jedná o sběr logovaných událostí, zkráceně logů. V podstatě se jedná o pipeline, schopnu nejen data sbírat, ale i parsovat, transformovat a přeposílat do mnoha dalších systémů. Tato pipeline je obdoba Logstahe, který měl být původně použit. Výsledná aplikace tedy používá Fluentd, konkrétně td-agent, pro logování. Td-agent je odnož Fluentd, která má různé možnosti konfigurace.

Po představení této sběrnice už zbývají pouze dva monitorovací prvky, které byly ve spojení právě s Fluentd použity a které sbírají data, pro výsledné zobrazení. Jedná se o monitorovací platformu Prometheus a zobrazovací rozhraní Grafana.

Ukázka pod textem se zaměřuje na konkrétní implementaci Logovací třídy pomocí této pipeline. Dále je vidět část z konfigurace.

```
<match stream_bot.**>
  @type copy
  <store>
    @type file
    @id output_file
    path /var/log/td-agent/myapp.log
  </store>
  <store>
    @type prometheus
    <metric>
      name stream_bot_logs_count
      type counter
      desc Count of log events
      label_keys stream_bot.logs
    </metric>
  </store>
</match>
```

Zdrojový kód 10: Ukázka konfigurace Fluentd

```

class Logger {
  private val fluentLogger = FluentLogger.getLogger("stream_bot")
  def write(log: Map[String, Object]): Unit = {
    fluentLogger.log("dataCount", convertToJavaMap(log))
    println("Logs wrote")
  }
  def convertToJavaMap[K, V](map: Map[K, V]):
    java.util.Map[K, V] = {
    map.asJava
  }
  def close(): Unit = {
    fluentLogger.close()
  }
}

```

Zdrojový kód 11: Ukázka použití Fluentd pro logování

4.2.2 Prometheus

Prometheus je platforma poskytující rozsáhlý *monitoring* a *alerting* systém a zároveň možnost analyzovat získané metriky a data. Prometheus pracuje na základě známého *pull-based* modelu, který *scrapuje* data přímo z cílových systémů, servis, nebo aplikací. Podmínkou je, aby cílová platforma měla možnost vystavit požadovaná data ve správném formátu na kompatibilní endpoint. Ve většině případů se tohoto dá docílit pomocí Prometheus-client knihovny, nebo exportéru, který sbírá data ze systémů a vystavuje je na cílový endpoint. Protože se o vystavení stará knihovna přímo pro Prometheus, data jsou vystavena ve formátu, který umí Prometheus přijmout.

Způsob jakým se *scrapují* data je založen na periodickém zasílání GET requestů na vystavený endpoint, který je nakonfigurován v souboru *prometheus.yml*. Konkrétní konfigurace aplikace je k nalezení v ukázce číslo 12.

```

global:
  scrape_interval: 5s

scrape_configs:
  - job_name: "prometheus"
    static_configs:
      - targets: ["localhost:9090"]

```

Zdrojový kód 12: Ukázka konfigurace Prometheus

V příloženém kódu je vidět, že screpovací endpointy jsou celkem čtyři a jeden je samotný Prometheus. To je z důvodu, aby byly dostupné metriky jak

4. NÁVRH A IMPLEMENTACE

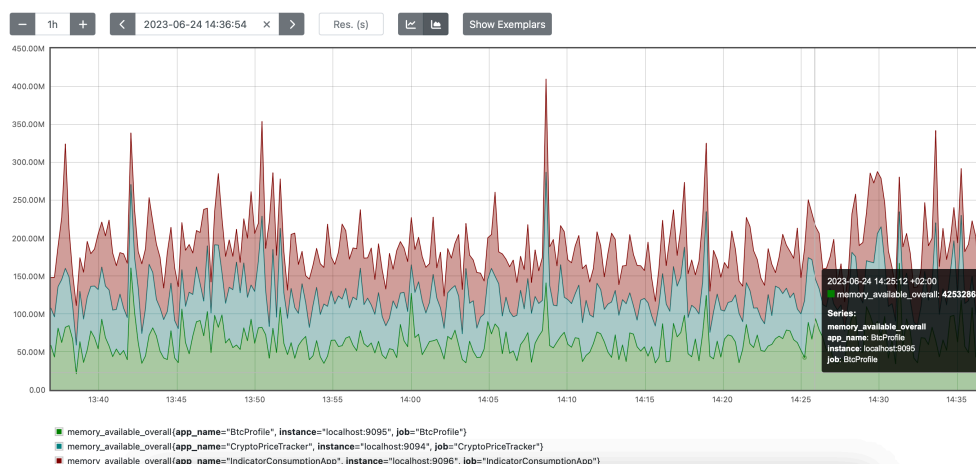
```
- job_name: "CryptoPriceTracker"
  static_configs:
    - targets: ["localhost:9094"]

    - job_name: "BtcProfile"
  static_configs:
    - targets: ["localhost:9095"]

- job_name: "IndicatorConsumptionApp"
  static_configs:
    - targets: ["localhost:9096"]
```

Zdrojový kód 13: Pokračování konfigurace Prometheus

ze všech aplikací, tak z monitorovacího systému. Metriky pro monitorování mohou být různé. Může se jednat například o logování počtu zaslaných zpráv, timestampů, nebo počtu dat v tabulce. Také se dají měřit metriky samotné aplikace. Toho je využito v této práci a použitý Prometheus se zaměřuje na screpování výkonnostních metrik aplikace jako jsou počty běžících vláken, počty CPU, celkové využití CPU, velikost využité paměti, nebo velikost zbývající paměti. Je třeba připomenout, že platforma Prometheus je nejvíce zaměřena na *monitoring* a *alerting*, takže možnost vizualizace získaných metrik je značně omezena. K vizualizaci slouží poslední z použitých nástrojů a tím je Grafana. Částečnou vizualizaci z Prometheus lze vidět na přiloženém obrázku.



Obrázek 4.2: Prometheus memory available

4.2.3 Grafana

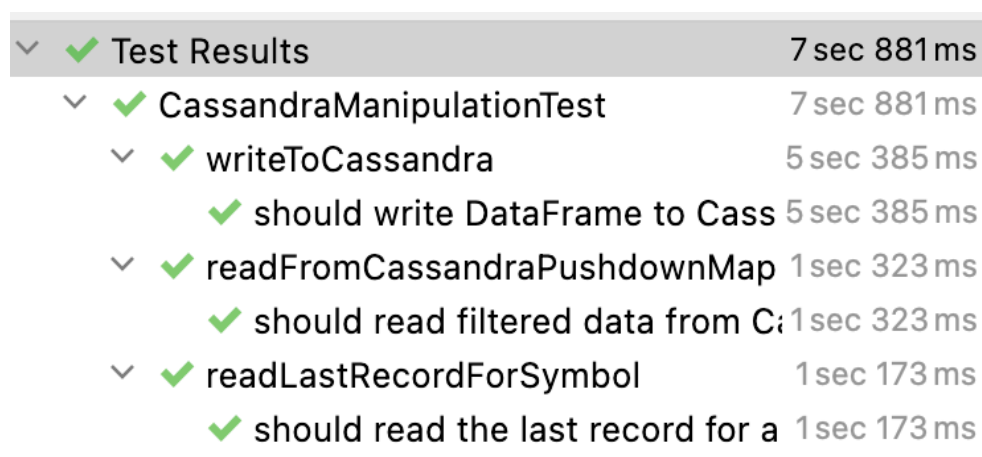
Grafana je nástroj primárně určena pro vizualizaci získaných dat ve formě grafů a dashboardů. Jako zdroj dat má Grafana k dispozici širokou paletu systémů. Mezi takové patří mimo jiné Prometheus a Apache Cassandra. Kromě surové vizualizace dat dokáže Grafana nad těmito daty provádět i různé typy agregací, pohledů a může provádět mnoho funkcí. Výměna dat mezi Grafanou jejími zdroji už neprobíhá pomocí screpování, ale díky vystavenému API, ke kterému přistupuje. Grafana slouží v celkové hierarchii aplikací jako nejvyšší zobrazovací vrstva.



Obrázek 4.3: Grafana metrics dashboard

Testování a měření

Testování aplikace jako celku proběhlo ve více krocích. Nejprve se otestovaly hlavní a stěžejní komponenty pro provoz celé infrastruktury a komunikace mezi aplikacemi a teprve poté se testovala zvolená strategie kryptobota. Mezi hlavní prvky komunikace patří výměna dat mezi Kafkou a *consumer* aplikací. Bez potřebných dat by nemohl nastat výpočet. Proto se testovalo čtení z pipeline, zda docházely potřebná data a hlavně jejich formát. Pro tyto druhy testů byla použita *scalatest* knihovna a *mockito* knihovna pro mockování struktur. Všechny testy jsou provedeny v prostředí IntelliJ Idea, ale dají se spustit i příslušným příkazem z terminálu. Pro nahlédnutí do Scalovských testů je čtenáři k dispozici ukázka kódu.



✓ Test Results	7 sec 881 ms
✓ CassandraManipulationTest	7 sec 881 ms
✓ writeToCassandra	5 sec 385 ms
✓ should write DataFrame to Cass	5 sec 385 ms
✓ readFromCassandraPushdownMap	1 sec 323 ms
✓ should read filtered data from C	1 sec 323 ms
✓ readLastRecordForSymbol	1 sec 173 ms
✓ should read the last record for a	1 sec 173 ms

Obrázek 5.1: Scala tests complete

Jako další klíčová komunikace k testování je zde Apache Cassandra da-

5. TESTOVÁNÍ A MĚŘENÍ

```
val result: DataFrame = kafkaManipulation
  .readKafkaStreamTopic(topic)

  // Start a streaming query to read from the Kafka
  // topic for 2 seconds
val queryOutput = "query_output"
val streamingQuery = result
  .writeStream
  .format("memory")
  .queryName(queryOutput)
  .outputMode(OutputMode.Append())
  .trigger(Trigger.ProcessingTime("2 seconds"))
  .start()
  // Wait for the streaming query to finish processing
streamingQuery.awaitTermination(5000) // Wait for 5 seconds
  // Stop the streaming query
streamingQuery.stop()
  // Retrieve the DataFrame representing the output of the
  // streaming query
val queryResult: DataFrame = spark.table(queryOutput)
  // Assert that the DataFrame has at least one row
queryResult.count() should be > 0L
  // Assert the DataFrame schema
queryResult.schema.fieldNames should contain allOf ("symbol",
  "price", "volume", "timestamp")
```

Zdrojový kód 14: Ukázka testování Kafka pipeline

tabáze. Testuje se nejen připojení, správný *key space*, ale i všechny použité funkce pro zápis a čtení. Tyto testy proběhly taktéž v pořádku, jak je vidět z příloženého obrázku číslo 5.1 na předchozí straně. Při testování zápisu a čtení se dá otestovat i samotné CQL, například pomocí příkazu truncate, který byl použit pro testování Cassandra.

```
private def truncateTable(): Unit = {
  CassandraConnector(spark.sparkContext.getConf).withSessionDo { session =>
    session.execute(s"TRUNCATE $keyspace.$table")
  }
}
```

Zdrojový kód 15: Ukázka testování připojení do Cassandra

5.1 Měření

Po úspěšném otestování hlavních částí aplikace je možné přejít k samotnému měření. Pro měření a monitoring je využívána hlavně Grafana společně s Prometheusem. Z průměrných naměřených hodnot, které jsou k dispozici Grafaně lze usoudit, že aplikace jako celek má jisté rezervy ve využívání paměti. Memory management je zajisté věc, ke které by se mělo přistoupit prioritně při rozšiřování programu. Dokonce z příloženého obrázku u odstavce popisující zobrazení Grafany lze vyčíst, že paměti zbývá pouze málo z celkově přidělené.

Pokud se jedná o využití CPU, tak z naměřených hodnot a z časového horizontu se dá vyčíst, že použití CPU roste lineárně s časem, avšak přibližně u 66 % se růst zastaví a zůstává konstantní. To je s největší pravděpodobností způsobeno tím, že celá aplikace je spuštěna na localhostu a klientský stroj nemá dostatek prostředků a výpočetního výkonu pro lepší využití CPU. Tento výpočetní problém se dá vyřešit nasazením celé aplikace na cloud jako je například Microsoft Azure a jiné.

Poslední měření v aplikaci se týká samotné logiky trading bota. Práce si jako jeden z cílů kladla naprogramovat prototyp trading bota, který bude shromažďovat obchodní metriky a počítat z těchto metrik signál pro nákup a prodej za pomoci trading strategie. Na začátku práce bylo vysvětleno, co to jsou indikátory a k čemu slouží. Po úspěšném výpočtu dvanácti indikátorů z nich bylo pár vybráno a spočítána strategie zaměřující se na moving average ceny. Tato strategie byla obohacena o další dva separátní indikátory, konkrétně o Bollinger Bounds a Stochastic oscillator. Trigger hodnoty všech indikátorů byly nastaveny podle vlastního uvážení a podle zkušeností, které doposud mám. Avšak testováním této strategie se ukázalo, že tyto trigger hodnoty byly nastaveny velmi přísně, protože signál k vykonání obchodu byl zachycen v průměru jednou za den. Do celkové statistiky se musí započítat fakt, že měření neprobíhalo nepřetržitě, ale s velmi vysokou granularitou. To je znovu způsobeno tím, že aplikace pracuje na localhostu. Toto se dá znovu vyřešit vystavením krypto bota na jednu z cloudových platform. Důvod proč aplikace nebyla vystavena je ten, že tyto platformy jsou placené pomocí *subscribe* a za využívání a navyšování výpočetních zdrojů se připlácí taktéž. V příložené tabulce pod textem může čtenář najít doposud provedené transakce.

5. TESTOVÁNÍ A MĚŘENÍ

Obrázek 5.2: Tabulka s provedenými transakcemi

symbol	timestamp	closePrice	closeTime	entryPrice	openTime	result
BTC	2023-06-24 15:33:27	30282.08	2023-06-24 15:33:27	30766.55	2023-06-24 14:01:59	stop-loss
BTC	2023-06-22 13:26:00	30254.27	2023-06-22 13:26:00	30161.66	2023-06-22 13:15:40	take-profit
BTC	2023-06-20 23:54:11	28392.6	2023-06-20 23:54:11	28306.9	2023-06-20 23:43:31	take-profit
BTC	2023-06-17 17:45:03	26332	2023-06-17 17:45:03	26719.65	2023-06-17 07:32:03	stop-loss
BTC	2023-06-14 21:07:03	25947.13	2023-06-14 21:07:03	26027.4.55	2023-06-14 19:34:03	stop-loss

Závěr

Práce si kladla za cíl, provést analýzu kryptoměnového prostředí a implementovat prototyp traidovacího kryptobota pro kryptoměnu Bitcoin. Zároveň měla čtenáře více vtáhnout do světa kryptoměn, blockchainu a vysvětlit možné příležitosti a rizika. Součástí analýzy bylo provést průzkum vývoje Bitcoinu a jeho pohyb na trhu. Kryptobot měl dále za požadavky shromažďovat data z předních burz/burzy pomocí streamu a následně nad těmito daty provádět analýzu. Výsledkem takovéto analýzy měl být signál, který aplikace generuje a zároveň vyhodnocuje v reálném čase pomocí zvolené a implementované strategie. Podle provedené analýzy nástrojů z Big Data sféry, pohybu ceny, výpočtu indikátorů byla zvolena a úspěšně implementována tzv. *Moving Average* strategie obohacená o Stochastic oscillator a Bollinger Bounds. Všechny vypočtené indikátory jsou perzistentně uloženy v databázi Cassandra. Tyto data jsou následně přečtená a použita jednou z aplikací pro výpočet signálu a kontrolování zvolené strategie.

Otestování hlavních komunikačních částí aplikace byl také jeden z cílů této práce. Za účelem testování a monitorování byla vytvořena sada testů v jazyce Scala s podporou knihovny `scalaTest` a zároveň vytvořena logovací a monitorovací infrastruktura, díky které bylo možno zjistit některé nedostatky implementovaného Sparkového enginu. Měření zvolené strategie ukázalo, že podmínky pro zaslání signálu jsou nastaveny velmi přísně a je potřeba se této části dále věnovat.

Celková komplexita a variace použitých systémů je velmi široká. Práce zahrnovala nový funkcionální jazyk Scala a technologie ze světa Big Data (zejména Apache Spark), které jsem se musel v průběhu naučit. Výsledná práce a implementovaná distribuovaná aplikace splňuje všechny cíle, které si projekt na začátku kladl.

Z výsledků měření a chování systému lze určit další vývoj. Prostor je především pro úpravu použité strategie a přidání strategie nové. Inovace může být například vylepšení logovacího systému o alerty, podporující Prometheus, nebo použití Spark MLLib knihovny pro implementaci strojového učení.

Bibliografie

1. INVESTPLUS. *Ethereum* [[online]]. Dostupné také z: <https://investplus.cz/kurzy/aktualni-kurz-ethereum-online-graf-kde-koupit-tezba-kryptomeny-cena-hodnota/>. [cit. 2021-08-26].
2. DANIAL, Kiana. *What Is Cryptocurrency?* [[online]]. Dostupné také z: <https://www.dummies.com/personal-finance/what-is-cryptocurrency/>. [cit. 2021-08-25].
3. INVESTPLUS. *Bitcoin* [[online]]. Dostupné také z: <https://investplus.cz/kurzy/aktualni-kurz-bitcoin-online-graf-kde-koupit-tezba-kryptomeny-cena-hodnota/>. [cit. 2021-08-27].
4. LAURENCE, Tiana. *Blockchain For Dummies Cheat Sheet* [[online]]. Dostupné také z: <https://www.dummies.com/personal-finance/blockchain-dummies-cheat-sheet/>. [cit. 2021-08-27].
5. FRANKENFIELD, Jake. *Proof of Work (PoW)* [[online]]. Dostupné také z: <https://www.investopedia.com/terms/p/proof-work.asp>. [cit. 2021-08-28].
6. VENCL, Jiří. *Proof of Work nebo Proof of Stake? Jaké typy těžby kryptoměn existují?* [[online]]. Dostupné také z: <https://finex.cz/kryptomeny-proof-of-work-proof-of-stake/>. [cit. 2021-08-28].
7. BITCOIN.ORG. *Balances - block chain* [[online]]. Dostupné také z: <https://bitcoin.org/en/how-it-works>. [cit. 2021-08-28].
8. PRATT, Mary K. *Top 10 benefits of blockchain technology for business* [[online]]. Dostupné také z: <https://searchcio.techtarget.com/feature/Top-10-benefits-of-blockchain-technology-for-business>. [cit. 2021-08-29].
9. IREDALE, Gwyneth. *Top Disadvantages Of Blockchain Technology* [[online]]. Dostupné také z: <https://101blockchains.com/disadvantages-of-blockchain/>. [cit. 2021-08-29].

10. IEEE. *Blockchain and Scalability* [[online]]. Dostupné také z: <https://ieeexplore.ieee.org/abstract/document/8431962>. [cit. 2021-08-30].
11. FRANKENFIELD, Jake. *Lightning Network* [[online]]. Dostupné také z: <https://www.investopedia.com/terms/l/lightning-network.asp>. [cit. 2021-08-30].
12. KANÁL, Bitcoinovej. *Bitcoin Lightning Network* [[online]]. Dostupné také z: <https://www.youtube.com/watch?v=pZdRxi-zi1Y>. [cit. 2021-09-02].
13. COINMARKETCAP. *What Is Volatility?* [[online]]. Dostupné také z: <https://coinmarketcap.com/alexandria/glossary/volatility>. [cit. 2023-05-04].
14. GENÇ, Ekin. *Is There a 'Best' Time to Trade Crypto? Here's What the Data Says* [[online]]. Dostupné také z: <https://www.coindesk.com/learn/is-there-a-best-time-to-trade-crypto-heres-what-the-data-says/>. [cit. 2022-05-04].
15. INTERMEDIATE. *Introduction to Crypto Derivatives, Options, and Futures* [[online]]. Dostupné také z: <https://crypto.com/university/what-are-crypto-derivatives-options-futures>. [cit. 2023-04-12].
16. LIOUDIS, NICK. *Forex Leverage: A Double-Edged Sword* [[online]]. Dostupné také z: https://www.investopedia.com/articles/forex/07/forex_leverage.asp. [cit. 2023-04-12].
17. RIVERFINANCIAL. *Financial Markets* [[online]]. Dostupné také z: <https://river.com/learn/what-is-market-making/>. [cit. 2021-09-10].
18. VENCL, Jiří. *Technická analýza* [[online]]. Dostupné také z: <https://finex.cz>. [cit. 2022-01-14].
19. CHEN, James. *Trendline: What It Is, How To Use It in Investing, With Examples* [[online]]. Dostupné také z: <https://www.investopedia.com/terms/t/trendline.asp>. [cit. 2023-04-12].
20. MURPHY, Casey. *Support and Resistance Basics* [[online]]. Dostupné také z: <https://www.investopedia.com/trading/support-and-resistance-basics/>. [cit. 2023-04-12].
21. FOLGER, Jean. *The Psychology of Support and Resistance Zones* [[online]]. Dostupné také z: <https://www.investopedia.com/articles/technical/02/061802.asp>. [cit. 2023-04-12].
22. MITCHEL, cory. *What Are Fibonacci Time Zones? How the Indicator Works in Trading* [[online]]. Dostupné také z: <https://www.investopedia.com/terms/f/fibonaccitimezones.asp>. [cit. 2023-04-12].

23. CHEN, James. *The Anatomy of Trading Breakouts* [[online]]. Dostupné také z: <https://www.investopedia.com/articles/trading/08/trading-breakouts.asp>. [cit. 2023-04-13].
24. DOWNEY, Lucas. *Pattern* [[online]]. Dostupné také z: <https://www.investopedia.com/terms/p/pattern.asp>. [cit. 2023-04-13].
25. SCOTT, Gordon. *Flag* [[online]]. Dostupné také z: <https://www.investopedia.com/terms/f/flag.asp>. [cit. 2023-04-13].
26. MITCHELL, Cory. *How to Trade the Head and Shoulders Pattern* [[online]]. Dostupné také z: <https://www.investopedia.com/articles/technical/121201.asp>. [cit. 2023-04-13].
27. CHEN, James. *Cup and Handle Pattern: How to Trade and Target with an Example* [[online]]. Dostupné také z: <https://www.investopedia.com/terms/c/cupandhandle.asp>. [cit. 2023-04-13].
28. CHEN, James. *Elliott Wave Theory: What It Is and How to Use It* [[online]]. Dostupné také z: <https://www.investopedia.com/terms/e/elliottwavetheory.asp>. [cit. 2023-04-13].
29. FARLEY, Alan. *What Is a Candlestick Pattern?* [[online]]. Dostupné také z: <https://www.investopedia.com/articles/active-trading/092315/5-most-powerful-candlestick-patterns.asp>. [cit. 2023-05-13].
30. BYBIT. *The 11 Best Technical Indicators for Cryptocurrency Trading (2023)* [[online]]. Dostupné také z: <https://learn.bybit.com/indicators/best-technical-indicators/>. [cit. 2023-06-05].
31. MITCHELL, Cory. *What Is Divergence in Technical Analysis and Trading?* [[online]]. Dostupné také z: <https://www.investopedia.com/terms/d/divergence.asp>. [cit. 2022-02-20].
32. CHEN, James. *What Is the Volume Price Trend (VPT) Trading Indicator?* [[online]]. Dostupné také z: <https://www.investopedia.com/terms/v/vptindicator.asp>. [cit. 2022-02-18].
33. CRYPTO, XC. *Using the Volume Profile in Crypto* [[online]]. Dostupné také z: <https://medium.com/@xanxsells/using-the-volume-profile-in-crypto-b88461efb2fc>. [cit. 2022-02-17].
34. TRADINGSTRATEGYGUIDES. *Camarilla Pivot Trading Strategy* [[online]]. Dostupné také z: <https://tradingstrategyguides.com/camarilla-pivot-trading-strategy/>. [cit. 2022-02-11].
35. TRADINGSTRATEGYGUIDES. *How to Trade with Pivot Points* [[online]]. Dostupné také z: <https://tradingstrategyguides.com/how-to-trade-with-pivot-points/>. [cit. 2022-02-11].

BIBLIOGRAFIE

36. BINANCE.COM. *What is Market Manipulation in Cryptocurrency?* [[online]]. Dostupné také z: <https://www.binance.com/en/blog/fiat/what-is-market-manipulation-in-cryptocurrency-421499824684902912>. [cit. 2022-02-13].
37. INFINOX. *What Are Time Frames in Trading?* [[online]]. Dostupné také z: <https://www.infinox.com/fsc/en/ix-intel/technical-analysis/what-is-time-frame-analysis>. [cit. 2022-02-13].

Seznam výpisu kódu

1	Ukázka navázání spojení přes websocket	42
2	Ukázka vytvoření topiců v Kafce	43
3	Ukázka konfigurace dockeru	43
4	Ukázka čtení topicu a paralelizace procesu	44
5	Ukázka zpracování po batchi (.foreachBatch)	45
6	Ukázka konfigurace sliding window	46
7	Ukázka použití sliding window	46
8	Ukázka zápisu celého Dataframe do Cassandra	47
9	Ukázka použití CQL query pro zápis	47
10	Ukázka konfigurace Fluentd	48
11	Ukázka použití Fluentd pro logování	49
12	Ukázka konfigurace Prometheus	49
13	Pokračování konfigurace Prometheus	50
14	Ukázka testování Kafka pipeline	54
15	Ukázka testování připojení do Cassandra	54

Seznam použitých zkratk

TA	Technical Analysis
P2P	Peer-to-peer
BTC	Bitcoin
ETH	Ethereum
MIT	Massachusetts Institute of Technology
POW	Proof of Work
TWh	Terawatthodiny
IT	Information Technology
BTCUSD	Bitcoin vs dolar
BTC/LTC	Bitcoin oproti Litecoinu
RM	Risk management
MA	Moving average
RRR	Risk/reward ratio
H&S	Head and shoulders
AWS	Amazon Web Services
MLLib	Machine learning library
HDFS	Hadoop Distributed File System
JVM	Java Virtual Machine

A. SEZNAM POUŽITÝCH ZKRATEK

HTTP Hyper text transfer protokol

OS Operation System

ELK Elastic search, Logstash, Kibana

API Application interface

IDE Integrated development environment

Uživatelská příručka

B.1 Spuštění aplikace

Aplikace je rozsáhlý a distribuovaný systém samostatně fungujících aplikací a nástrojů, které spolu komunikují přes vystavené API, nebo pomocí screpování dat z vystavených endpointů. Z tohoto důvodu je spuštění aplikace rozděleno do několika kroků.

- nainstalovat všechny potřebné nástroje a technologie
- nakonfigurovat jednotlivé nástroje pro komunikaci s ostatními
- spustit postupně všechny technologie
- poté spustit jednotlivé aplikace

Potřebné technologie, které je nutno mít nainstalované lokálně, nebo ve virtuálním kontejneru, jsou Apache Kafka a Zookeeper, Apache Cassandra, Grafana, Prometheus, Fluentd (Td-agent) a samozřejmě samotný virtualizační nástroj. Je možné, že bude potřeba nainstalovat i exportér, který zajistí přenos dat mezi Prometheus a dalšími systémy. V některých verzích může chybět.

Další druhy instalace jsou například stažení nástrojů z oficiálních stránek a nainstalování, případně rozbalení archivu do cílového adresáře. Homebrew je také možnost jak stáhnout tyto nástroje. Je to komplexní *utilita*, která dokáže stáhnout potřebné balíčky ze vzdálených repozitářů a spravovat jejich spustitelné *servisy*.

Když jsou všechny nástroje nainstalované, je nutno je nakonfigurovat. Ukázka konfiguračních souborů je k dispozici ve složce s projektem, stejně tak různé verze konfiguračního souboru `docker-compose.yaml`, který zajistí stáhnutí potřebných nástrojů. Poté už jen stačí spustit aplikace v jakémkoli IDE podporující Scala jazyk, nebo z příkazové řádky. Podrobnější instrukce jsou přiloženy v adresáři s projektem.

Obsah Git repozitáře

Odkaz na Git s prací: <https://gitlab.fit.cvut.cz/kubecpat/ni-diplomka>

readme.txt.....	stručný popis obsahu repozitáře
spark_stream_bot.....	adresář s aplikací
├── config.....	konfigurační soubory aplikace
├── data.....	metadata aplikace
├── docker_compose.....	konfigurační soubor dockeru
├── project.....	složka s metadaty editoru
├── README.md.....	popis aplikace v git adresáři
├── src.....	zdrojové kódy aplikace
├── target.....	metadata editoru
├── build.....	soubor s dependencemi pro sestavení programu
├── Dockerfile.....	konfigurační soubor dockeru
├── dockercompose.....	složka s parciálními docker soubory
text.....	adresář s textem práce
├── DP_Kubec_Patrik_2023.pdf.....	text práce ve formátu PDF
├── src.....	zdrojové kódy práce ve formátu L ^A T _E X